International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 14, No.1 (Sep-2023)

http://dx.doi.org/10.12785/ijcds/140182

Finetuning Deep Learning Model for Review Rating Prediction

Asmae BENTALEB[1] **and Jaafar ABOUCHABAKA**[1]

[1]*Laboratory of Research in Informatics, Faculty of Science, Ibn Tofail University, Kénitra, Morocco*

Received 16 May. 2022, Revised 9 Aug. 2023, Accepted 9 Sep. 2023, Published 22 Sep. 2023

Abstract: : It has been proven that the fact of reviewing websites has an enormous impact on the shopping behavior of customer. Usually, the system generates a star rating out of 5 for each online review based on the text input by the user. This is called the review rating prediction problem where the rating star is predicted for a given product or service based on the review text. This issue has become widely known and discussed in the field of deep learning and Natural Language Processing. Researchers have developed this domain interestingly especially with the emergence of the concept of transfer learning. XLNet is considered among the main pretrained models available through the transformers library. It is used for text classification and can be fine-tuned on downstream tasks. This article presents a study of the literature concerning these concepts. Later on, it presented a fine-tuning approach of the XLNet algorithm through two main phases based on the concept of transfer learning. To prove the effectiveness of this approach, experiments were done on the Yelp Dataset. Noticeably, the classification task using the new model has achieved an interesting result of 77%. It outperformed the XLNet's SOTA accuracy that was 70%.

Keywords: Transfer Learning, XLNet, Pretrained Model, Deep Learning

1. Introduction

Nowadays, publishing online reviews for services or products has become very crucial. This is because of several reasons; customers usually, before purchasing a service, they tend to check the reviews filled by previous clients, to decide about whether it is a good one or not [1]. Also, reviews are checked by business owners as well since they give them an idea about the feedback of their clients, and hence, the success or failure of their businesses. These reviews are mainly represented via stars thanks to several machine and deep learning mechanisms. Automatic review rating that refers to multi class text classification is considered to be one of the main tasks of Natural Language Processing. It serves at rating customers' reviews within a range of five classes from 1 to 5. In fact, deep neural networks have been used widely in achieving NLP tasks including multi class text classification. They have resulted in good performance. However, since deep neural networks contain millions of parameters and features and require a large amount of data to converge, training them from scratch to achieve each NLP task is quite expensive and time consuming. This is because they can take days and weeks to converge to a model that is performing and stable. Here comes the need for a solution that would improve the effectiveness of such tasks and facilitate their achievements. The research in the field of text classification under NLP has been aligned with the introduction of the concept of transfer learning. This concept borrows the knowledge acquired from a relevant source domain to a target domain in order to perform a prediction task [2]. This technique has contributed to reducing the cost and complexity of training deep neural networks for every NLP tasks. BERT is one of the pretraining techniques introduced recently in 2018. It refers to Bidirectional Encoder Representations from Transformers (BERT) [3]. Thanks to this approach, many state of the art models (SOTA) have been created for several NLP tasks including text classification, question answering, natural language inference and others. Still, BERT has some limitations that have been overcome thanks to XLNet [4]. It is considered to be the latest pretrained language model. It is a generalized autoregressive pretraining approach that allows bidirectional context learning and examining all the possible permutations of the factorization order. XLNet has borrowed many concepts from the Transformer-XL architecture [5]. As for the main contributions of this article, they can be summarized in the following points:

- Tuning the XLNet Model through phase 1 using common standards found in the literature.
- Tuning the XLNet Model through phase 2 by altering its architecture thanks to using transfer learning, adding a fully connected layer, freezing the other layers' parameters and finetuning some of them to improve the accuracy.
- Achieving new SOTA results with significant improvements over the previous approaches. Achieving 72% accuracy and then 77% in phase 2.
- Comparing the performance of XLNet with some of the classical machine learning algorithms and deep learning transformers.

E-mail address: asmae.bentaleb@uit.ac.ma, jaafar.abouchabaka@uit.ac.ma

2. Related Works

Throughout the literature, there have been several approaches and methodologies to tackle the review rating prediction problem for the specific dataset Yelp. One of the proposed solutions of the Yelp Dataset Challenge was to predict business stars based on the users' text review. For feature generation, it used three methods, whereas, for the learning phase, four machine learning algorithms were trained and tested including the Decision Tree Regression, the Support Vector Regression and the Linear Regression. The results of experimentation on the Yelp Dataset have shown that Linear Regression had the best performance compared to the other configurations [6]. Another proposition handled the issue of predicting the ratings of restaurants' reviews and treated it like a five class classification. It was mainly based on testing several feature extraction and supervised methods in order to build 16 systems for prediction. At last, the performance of each system is analyzed in order to conclude the configuration of feature extraction and machine learning algorithm that generates the best results. This system has been designed to generate star ratings based on comments written by end users in the form of text to evaluate Yelp [7].

Another approach was about combining the content based method with the collaborative filtering one (network prediction models). Experiments on the Yelp dataset revealed that combining these two approaches leads to better performance [8]. Another novel approach for predicting users' reviews included both user and product context. It started first by modelling the contextual information of reviews related to users. Then, modelling reviews' contextual information concerning products. Experiments and tests on Yelp along with other datasets have shown better performance of the proposed method compared to the previous ones found in the literature [9]. A newer methodology to solve the review rating prediction problematic suggested a framework that integrated information about user and product along with external memory. It started by generating representations of user and product, then built user and product specific documents. Experiments and tests on Yelp and other datasets have shown a good performance of the proposed model. Combining product and user memory has led to a better mechanism to learning user and product representations and predicting ratings for reviews [10]. The following solution analyzed two ways of predicting reviews rating for restaurants in Yelp. Mainly, the sentiment analysis and opinion mining model to achieve the classification of text reviews. Experiments and testing have used the Yelp dataset that is rich of text reviews. They were based on comparing the results of the algorithm of machine learning "Naive Bayes" and the "CLSTM: convolution Long Short Term Memory" which is an algorithm of deep learning with word2vec and "Glove: Global Vector". The results have shown that the CLSTM is the best in terms of performance for the classification of reviews [11].

A newer approach of tackling the review rating prediction problem was about deep learning "BI GRU model", and was composed of two main milestones. The first one consisted of the prediction of polarity while the second one performed the prediction of review rating using the results of the first phase. Results of the experiment performed on the Yelp Dataset showed a noticeable enhancement in the performance of the rating prediction [12]. The following proposition handled the review star rating prediction problem especially for the Yelp dataset: It started first by building a balanced dataset since the original one was unbalanced. For the learning phase, four machine learning algorithms were used including: Naive Bayes, Logistic Regression, Random Forest [13] and Linear Support Vector Machine [14] and four transformer based models were used also: BERT[3], DistilBERT[15] RoBERTa [16], and XLNet [4]. These models were compared and transformers based models showed a better accuracy of 70%. In the coming sections, the paper will present a methodology to improve the performance of the pretrained language model XLNet since it is considered to be the best one used for the task of text classification. The purpose of this article is to finetune this algorithm so that it outperforms the SOTA's accuracy of XLNet [17]. The following sections define the basic concepts that will be used in the proposed architecture, including transfer learning and the definition of the algorithm XLNet.

3. Transfer Learning

Transfer learning is an approach in the domain of artificial intelligence and machine learning that serves at grabbing the knowledge acquired from a given task (NLP one) to a destination task as shown in Figure 1 [2] Going back to the

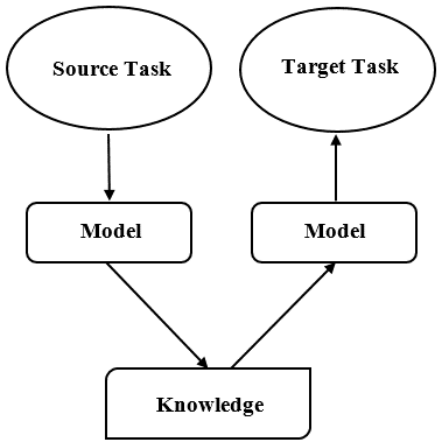

Figure 1. Illustration of the transfer learning Concept

literature, transfer learning has been introduced at first in the year of 1993 [18]. Throughout the years, this technique has become crucial in building new artificial intelligence and deep learning solutions. This is through building pretrained models and training them on large datasets like Wikipedia. Then, making them available through deep learning libraries

like TensorFlow, PyTorch and Hugging Face. For NLP tasks like text classification, there have been many pretrained models like Open AI GPT Series, ELMo Variations and BERTs (BERT, RoBERTa, DistilBERT, and XLNet). This latter is the focus of this paper. It will be explained further in the coming sections.

4. XLNET

As far as the natural language processing is concerned, the unsupervised learning has been used widely and showed interesting results [19]. Such approaches train neural networks using large data and finetune them on tasks for downstream. Several unsupervised pretraining objectives have been used, notably the autoregressive language modelling and autoencoding for language modelling. Concerning the autoregressive language modelling (AR), it is used to train models to encode in a context that is unidirectional (forward or backward). Such approach is not efficient in providing models of deep bidirectional contexts. Nevertheless, the understanding of downstream language needs bidirectional information of context [20]. For the autoencoding language modelling, it works through reconstructing the original inputs by eliminating the corrupted inputs, thanks to the use of a [MASK] instead of the corrupted data. BERT [3] is considered as an autoencoder language model. This method has its advantages, since the [Mask] is used during the pretraining phase, and given that such symbols are not present in the data used for finetuning, this results in a discrepancy in finetuning pretrained data. This approach assumes the independency among the predicted masked tokens given the unmasked ones. Given the advantages and disadvantages of pretraining objectives mentioned previously, XLNET [4] has been proposed as a generalized autoregressive method which has taken the advantages of AR and AE language modelling and has avoided their shortcomings. It does not use a fixed factorization order like in the AR pretraining models. It uses all the possible permutations of factorization order. Consequently, the position's context can be established by tokens from both left and right. This allows each position to use contextual information from all sides which makes the pretraining phase here bidirectional. Unlike the BERT model, XLNET does not reconstruct the corrupted data, therefore, it is not subject to the discrepancy that happens at finetuning time. Also, it eliminates the independence among tokens assumed in BERT. In case of having tasks including long texts, and in order to enhance the performance, XLNET has incorporated the mechanism of segment recurrence and the aspect of Transformer-XL [5] which is the relative encoding scheme in the phase of pretraining. As far as the transformer XL architecture is concerned, since the factorization order used to achieve permutations is ambiguous, it is required to reparameterize the transformer XL in order to remove ambiguity. Finally, XLNet has been introduced in two forms, the base one and the large one. This paper's contribution will be based mainly on the base one due to the resources' limitations and costs. It is made of 12-layer, 768-hidden, 12-heads and 110M parameters.

5. THE PROPOSED APPROACH

A. Problem Statement

As discussed in the previous section, there have been several methods in the literature handling the reviews' rating issue or the multi class text classification. Since nowadays most of the complicated NLP tasks are resolved using deep learning and especially neural networks, finetuning them has become a challenging task. This is due to many reasons:
- The number of hyperparameters is noticeably high.
- It is computationally very expensive to train the deep neural network from scratch for every NLP task since it might take days and weeks to converge.

B. The Proposed Finetuning Approach for XLNET

This paper introduces a new model based on XLNet that would enhance the performance of the multi class classification for reviews. This is thanks to using the concept of transfer learning supported by the transformers library. Consequently, the XLNet pretrained model is available online to be downloaded and finetuned on downstream tasks. Figure 2 explains the architecture of this new model which is based on loading the XLNet and using the knowledge it has already acquired while pretraining and adding a fully connected layer on the top to improve the accuracy of reviews' classification.
The proposed architecture is made of the following modules:
- The preprocessing of the texts to be classified: balancing the data set amongst the reviews' categories. Splitting the data into training, validation and testing datasets.
- The tokenization phase: before feeding texts to the XLNET model, it has to be tokenized into tokens and small units.
- The loading of the pretrained model of XLNET available and already trained on huge datasets. This is called transfer learning, which is about transferring the knowledge that the model has learnt from other data sources and applying them on the target data source (Yelp Dataset for this article's case). This approach helps in overcoming the issues related to low accuracy resulted from training the model in small datasets. This is generally due to the limitations of computational resources. This concept helps in using already learned patterns from a large model like XLNET on different datasets.
- Adding a fully connected layer whose outputs' size corresponds to the targeted number of classes.
- Concerning the training of the proposed model's layers, the ones imported from the pretrained model are freezed and not touched during the training except for few ones to accommodate the implementation's environment whereas the output layer is trained from scratch. The output layer is trained using a higher learning rate which is almost 10 times the one used in the pretrained model.

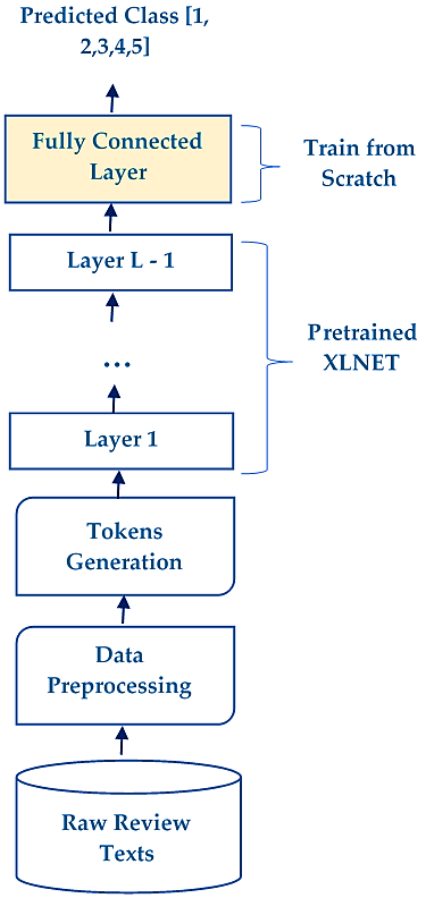

Figure 2. New XLNet Model Architecture for Reviews Rating Prediction

6. Experiments

A. Setup

Setting up the experiment has gone through the following phases:
- Selecting the best cloud platform to run the experiment. Several ones were tried and the choice was a machine on the cloud that could grant GPU instances in the concerned area.
- Creating the GPU instance. The used operating system was Ubuntu 20.04 to be able to handle the simpletransformers library. Two GPUs have been chosen (Tesla 100) and the instance was deployed using the necessary configuration (cuda).
- Installing Jupyter Notebook and uploading the json files (review and business).
- The following python libraries were used during the experimentation:pandas, numpy, hyperopt, seaborn, pytorch, defaultdict and simpletransformers.

B. Overview of The Yelp Dataset

For the sake of achieving this experimentation, the Yelp Dataset [21] offered by the Yelp Organization, has been used. It is one of the online platforms. It allows end user to publish their reviews concerning the different services and categories. Within the Yelp open dataset, there have been several research works on machine learning mechanisms that help in predicting ratings for restaurants based on clients' reviews. These mechanisms are considered to be crucial and fundamental for the Yelp business. First, it helps in predicting ratings from reviews automatically, also, it can filter reviews to prevent malicious competition from other businesses. The Yelp Review Rating Prediction task can be performed in several ways mainly: sentiment analysis and 5- star rating classification. These data is assembled in the Yelp Open Dataset. It contains six json files including: business, review, tip, user, photo data and check-in. The focus of this article will be mainly on the review and business tables. They are composed of 8,635,403 reviews and 160,585 businesses in 8 metropolitan areas [17]. The choice of the review and business tables has been made because the review data is considered to be the most diverse and interesting to analyze. The review json object is made of the following attributes: the business ID, user ID, stars (integer values between and including 1 and 5), review text, date and votes. The business json object is represented by the following attributes (the business ID, its name, location, stars, review count, opening hours). The business file contains data relevant to several categories including: travel, restaurants, hotels and shopping. The users' reviews may differ according to the category they are related to. This is why it is so crucial to build the review rating prediction for each category separately. Therefore, in this work, the focus will be only on the restaurant category [7]. After applying a filter on the business file to select only the ones related to restaurants, the result was: 63944. Then, after the extraction of the reviews related to the category of business from the review json file, the result was 5,055,992 reviews [17]. For the context of this article, due to resources' limitations, reviews' size has been reduced. Later, after grouping the reviews per value of classification, it was found that most of the reviews are grouped with values (4 and 5) as shown in Table 1.

TABLE I. Number of reviews per stars

Star	Number of Reviews
1	212615
2	152945
3	212755
4	414077
5	684351

For evaluating the results of the experiments, several metrics were used:
- Accuracy: it refers to the measurement of the ratio of correct predictions over the total number of instances evaluated, and is calculated through Equation (1) [22].
where:
tp: true positive

tn: true negative
fp:false positive
fn: false negative

$$\text{Accuracy} = (tp + tn) / (tp+fp+tn+fn) \quad (1)$$

- Precision: it is called the rate of true positive. It is the portion of relevant retrieved items out of all retrieved items. The higher the value is the better [23]:

$$\text{Precision} = tp/(tp + fp) \quad (2)$$

- Recall: it is the rate of true positive which is the portion of retrieved items that are relevant out of all relevant items. The higher the value is the better [24]:

$$\text{Recall} = tp/(tp + fn) \quad (3)$$

- F1 Score: it combines the two previous metrics. This is because it is the harmonic mean of the precision and recall. It is expressed in the following equation [25]:

$$\text{F1 Score} = tp/(tp + 0.5(fp+fn)) \quad (4)$$

During the experimentation, confusion matrix was used. It englobes precision recall, accuracy and f1 score. This is to give an idea about the performance of the classification model.

C. Features Extraction

On their website, Yelp has given the possibility to end users to write their comments freely. The field's nature is a text field. Hence, for a given review, users might include many punctuation marks, capital letters, some special characters or useless words to describe their opinions towards Yelp. Therefore, to make the review meaningful, it shall be preprocessed and cleansed at first. There comes the role of the concept of Feature Extraction thanks to the use of Python libraries [7]. Vectorizers from the scikit learn library were used for that purpose, they included according to [17]:
- The conversion of every character to lower case.
- The deletion of stop words (used in English).
- The use of bigrams and unigrams.
- The use of the vectorizers count and Tf-idf whith both, there integer and binary versions.
- Assigning the value 5 to the min document frequency.
According to these results, the vectorizer that performed the best is the Tf-idf(Integer). Table 2 compares the performance of each vectorizer in terms of accuracy and f1 score.

TABLE II. Evaluation Matrices for the vectorizers used on the Validation Set

Vectorizer	Accuracy	F1 Score
Count(Integer)	0.6321	0.6369
Count(Binary)	0.6285	0.6293
TfIdf (Integer)	0.6387	0.6431
Tf-Idf(Binary)	0.6358	0.6420

D. Tokenisation for the Transformers' Algorithms

Concerning the XLNET classifier, it is designed to handle text in a certain format. Its inputs data has to be tokenized in subwords with certain criteria of size. As to perform such operation, the following XLNet Tokenizer has been used from the transformers' library. This step has been implemented using the following steps: - Tokenizing the input text into ids.
- Denoting the end of sentences through adding the adequate special characters at the end.
- Truncating or padding characters to respect the fixed sequence length (128 to suit the resources available in the case of this work).
- In order to stop the model from focusing on padding tokens, attention masks were created to guide the model to tokens ids where it should apply attention.
- Appending the created attention masks along with the tokenized inputs to the dataframe. After completing this step, the created inputs ids and attention masks are converted to torch tensors since it is the datatype required by the XLNet model. Later on, torch DataLoader are created as iterators of data since they help in saving memory while training. This is because unlike normal loops, while interating, DataLoaders do not load the whole dataset to memory.

E. Implementation Details

1) Experiment 1: Testing Text Classification Using Classical Machine Learning Algorithm

After performing the following milestones:
- The yelp dataset description and presentation using python and scikit learn library.
- The splitting of the data to build the training, testing and validation dataset that will be used in training models in a balanced way since the original one was imbalanced.
- The pre-processing of the data to remove invalid elements and feature extraction to build matrices. Several methods were used and compared. The yelp dataset has been trained and tested on a set of the most known machine learning algorithms in order to compare them later with the performance of XLNet and prove its efficiency. The models used in the experiment are : Naive Bayes [5], Logistic Regression [8], Random Forest [12], Linear SVM [11]. The accuracy metric was used in the evaluation. The table below depicts the results of this experiment.

TABLE III. Comparison of Machine Learning Algorithms

Model	Accuracy	F1 Score	Training time
Naive Bayes	0.6150	0.6222	00:00:05
Logistic Regression	0.6407	0.6454	00:12:42
Random Forest	0.5954	0.5870	00:58:42
Linear SVM	0.6199	0.6043	00:00:35

2) Experiment 2: Testing Text Classification Using Transformer based models

Transformer based models were used as well: BERT (base, uncased), BERT(base, cased), BERT (large, cased), DistilBERT (base, uncased), DistilBERT(base, cased), RoBERTa (base) XLNet (base, cased) and were compared used the evaluation matrices (see Table 4 in the next page):

3) Experiement 3: First Phase of Hyperparameter Optimisation of XlNet Based On Some Literature Standards

Since the base form of XLNet has a large number of parameters, during this experiment, the focus will be mainly on (maximum sequence length, train batch size, number of training epochs and learning rate). After searching in the literature, the following recommendations were found concerning the chosen hyperparmeters:

- Train batch size: According to [26], the recommended batch size used for finetuning models in case of text classification is 32.
- Number of train epochs: according to [3], the recommended number of epochs for finetuning is 2,3 and 4 and the choice depends on the availability of resources at execution time.
- Learning rate: according to [3], this parameter is recommended to be in the range of 2e-5, 3e-5 and 5e5.
- Max sequence length: as far as the classification task is concerned, it is recommended to set this parameter to 256 [26].

As per the configuration recommendations collected from the literature, the following configuration has been used in the experimentation and adapted according to the resources available for training this task: 32 in the Batch size; 128 Sequence length; a learning rate of 1e-5 and 2 Epochs. Since the dataset's size is huge, and the whole model will be trained, a setting of 2 GPUs has been used to run the experiment as explained in the section "Setup". The finetuning of XLNet has resulted in an accuracy of 72% which beats the SOTA accuracy of multi class text classification on the Yelp data set "restaurants" performed in this article,using XLNet. Given that the accuracy achieved in the SOTA was 70% [17].

Validation Set
Accuracy: 0.720024

	precision	recall	f1-score	support
0	0.7858	0.8158	0.8005	31863
1	0.5406	0.5046	0.5220	22711
2	0.6169	0.5597	0.5869	32075
3	0.6315	0.5815	0.6055	61175
4	0.8072	0.8713	0.8380	102176
accuracy			0.7200	250000
macro avg	0.6764	0.6666	0.6706	250000
weighted avg	0.7128	0.7200	0.7154	250000

Figure 3. Results of XLNet Hyperparameter Optimization

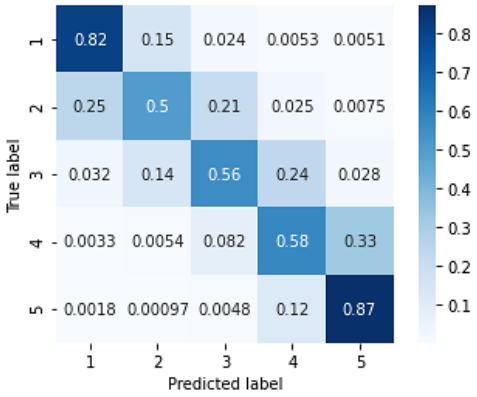

Figure 4. Evaluation Matrix

4) Experiment 4: Second Phase of XLNet Finetuning Based on The Proposed Architecture

In order to outperform the SOTA's performance of XLNet and enhance the results of the first experiment, the proposed model of this article has been tested on the Yelp dataset also. In this experiment, since it is based on the concept of transfer learning. The pretrained XLNet model has been loaded from the transformers library via the XLNetModel class:

```
class XLNetForMultiLabelSequenceClassification(torch.nn.Module):

  def __init__(self, num_labels=5,dropout=0.1):
    super(XLNetForMultiLabelSequenceClassification, self).__init__()
    self.num_labels = num_labels
    self.dropout = dropout
    self.xlnet = XLNetModel.from_pretrained('xlnet-base-cased')
    self.classifier = torch.nn.Linear(768, num_labels)
```

Figure 5. Loading of XLNet Classifier

While training, the set of parameters of this model has been frozen to decrease the complexity of the training with a low learning rate. Few parameters like the batch size have been changed to 32 as well as the max sequence length to 128 to fit the characteristics of this experiment. The output layer has been added on the top of the model as mentioned in the proposed model architecture. It is a fully connected layer made of neurons that would enhance the quality of reviews' classifications. it has been created with having five classes since in this experiments' case, there are five classes to be predicted. It was created using the torch library and the loss function BCEWithLogitsLoss. It was trained from scratch using a higher learning rate as to acquire new knowledge from the new dataset which is Yelp. In this experiment, the number of epochs has been increased to 3 and learning rate of the output layer to 2e-5. As shown in figure 6, the achieved accuracy was 77%.

TABLE IV. Comparison of Deep Learning Algorithms

Model	Accuracy	F1 Score	Training time
Bert(base, uncased)	0.6911	0.6963	05:36:05
Bert(base, cased)	0.6971	0.7013	02:38:37
Bert(large, cased)	0.7004	0.7050	05:02:00
DistilBERT(base, uncased)	0.6847	0.6897	02:54:52
DistilBERT(base, cased)	0.6944	0.6985	01:21:47
RoBERTa(base)	0.7029	0.7080	05:32:44
XLNET	0.7044	0.7087	07:11:19

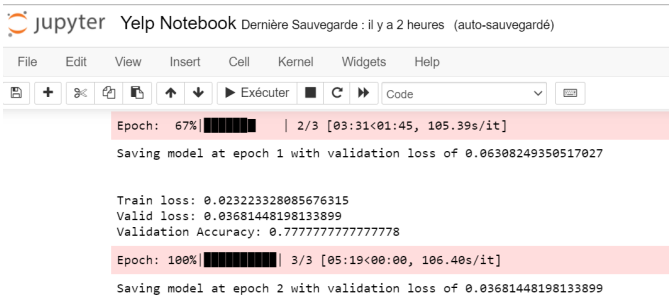

Figure 6. Results of the task of multi class classification of reviews on the validation set using the proposed model

The following figure shows the enhancement of the loss function during the epochs for the validation dataset:

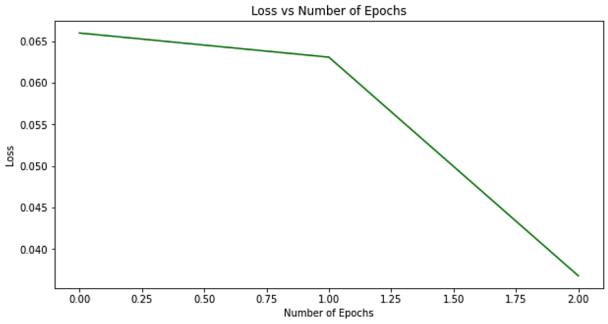

Figure 7. Plot of the progress of the loss function during the task of multi class classification on the validation set

7. Results and Discussion

During the implementation phase of this article, four different experiments have been conducted for two main reasons: first, proving the effectiveness of XLNet compared to the traditional machine learning learning algorithms, and transformers based ones. Second, to finetune this model so that it outperforms the acutal SOTA's performance in the task of multi-label classification.

As per the results of experiment 1 that has tested a set of the most known classical machine learning algorithms on the Yelp Dataset to perform the yelp reviews rating prediction, it has shown that the best algorithm which is Logistic Regression has achieved an accuracy of 64only. For experiment 2 it has been applied on a set of the most known transformer based deep learning algorithms like BERT, DistilBERT with XLNet. It showed that XLNet outperformed all of them with an accuracy of 70%.

Concerning experiment 3 where XLNet has been applied with some finetuning on hyperparameters like batch size and number of epochs and sequence length,the result was much better and reached 72% that bypassed the SOTA performance of XLNet on Yelp dataset which is 70%. This result proves the efficiency of XLNet on down-stream tasks like text classification. It proved its learning capability and the quality of its predictions. In the fourth experiment, the proposed architecture has been applied on the XLNet pretrained model. It has shown a good improvement of accuracy that has reached 77%. This shows the role of adding a fully connected layer on the top of XLNet and the usefulness of training it with a higher learning rate while keeping the knowledge that the model has already acquired while pretraining.

8. Conclusions and Future Work

Throughout this article, the review rating prediction problem has been tackled and more specifically finetuning one of the most known pretrained models used to achieve this NLP task. First, a state of art of the solutions, already proposed in this sense, has been established. This paper handled the issue of finetuning the XLNet algorithm. It has proposed an architecture that added a fully connected layer on the top with some specific configurations while training. Such proposition has shown through the experimentations a noticeable improvement of the XLNet on the Yelp Dataset compared to the SOTA one. Other experiments have been performed to show the effectiveness of XLNet compared to other classical machine learning algorithms. Future works could go towards applying the XLNet pretrained model on other interesting datasets to explore more their performance matrices and take advantages from the concept of transfer learning.

We believe that future works could achieve an even better result than ours, especially with more available and affordable resources that would allow running the proposed classification modem for more epochs in order to achieve a better performance. Another future work could be applying

the proposed solution to another famous and large dataset in order to better confirm its efficiency.

References

[1] J. Chambua, Z. Niu, A. Yousif, and J. Mbelwa, "Tensor factorization method based on review text semantic similarity for rating prediction," *Expert Systems with Applications*, vol. 114, pp. 629–638, 2018.

[2] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He, "A comprehensive survey on transfer learning," *Proceedings of the IEEE*, vol. PP, pp. 1–34, 07 2020.

[3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *NAACL*, 2019.

[4] Z. Yang, Z. Dai, Y. Yang, J. G. Carbonell, R. Salakhutdinov, and Q. V. Le, "Xlnet: Generalized autoregressive pretraining for language understanding," in *NeurIPS*, 2019.

[5] Z. Dai, Z. Yang, Y. Yang, J. Carbonell, Q. Le, and R. Salakhutdinov, "Transformer-xl: Attentive language models beyond a fixed-length context," 01 2019, pp. 2978–2988.

[6] M. Fan and M. Khademi, "Predicting a business star in yelp from its reviews text alone," 01 2014.

[7] N. Asghar, "Yelp dataset challenge: Review rating prediction," *ArXiv*, vol. abs/1605.05362, 2016.

[8] L. Perez, "Predicting yelp star reviews based on network structure with deep learning," *ArXiv*, vol. abs/1712.04350, 2017.

[9] B. Wang, S. Xiong, Y. Huang, and X. Li, "Review rating prediction based on user context and product context," *Applied Sciences*, vol. 8, p. 1849, 10 2018.

[10] Z. Yuan, F. Wu, J. Liu, C. Wu, Y. Huang, and X. Xie, "Neural review rating prediction with user and product memory," *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 2019.

[11] A. Jungsher, M. Suleman, and M. Alim, "Robust review rating prediction model based on machine and deep learning: Yelp dataset," 07 2020.

[12] B. H. Ahmed and A. S. Ghabayen, "Review rating prediction framework using deep learning," *Journal of Ambient Intelligence and Humanized Computing*, Mar. 2020.

[13] B. Ait Hammou, A. Ait Lahcen, and S. Mouline, "An effective distributed predictive model with matrix factorization and random forest for big data recommendation systems," *Expert Systems with Applications*, vol. 137, pp. 253–265, 2019.

[14] T. Hastie, R. Tibshirani, J. Friedman, and J. Franklin, "The elements of statistical learning: Data mining, inference, and prediction," *Math. Intell.*, vol. 27, pp. 83–85, 11 2004.

[15] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter," *CoRR*, vol. abs/1910.01108, 2019.

[16] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *ArXiv*, vol. abs/1907.11692, 2019.

[17] Z. Liu, "Yelp review rating prediction: Machine learning and deep learning models," *ArXiv*, vol. abs/2012.06690, 2020.

[18] L. Y. Pratt, "Discriminability-based transfer between neural networks," in *Advances in Neural Information Processing Systems*, S. Hanson, J. Cowan, and C. Giles, Eds., vol. 5. Morgan-Kaufmann, 1992.

[19] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, Jun. 2018, pp. 2227–2237.

[20] A. M. Dai and Q. V. Le, "Semi-supervised sequence learning," in *Advances in Neural Information Processing Systems*, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, Eds., vol. 28. Curran Associates, Inc., 2015.

[21] yelp, "Yelp dataset," 2021.

[22] T. Silveira, M. Zhang, X. Lin, Y. Liu, and S. Ma, "How good your recommender system is? a survey on evaluations in recommendation," *International Journal of Machine Learning and Cybernetics*, vol. 10, 05 2019.

[23] J. Katzman, U. Shaham, A. Cloninger, J. Bates, T. Jiang, and Y. Kluger, "Deepsurv: Personalized treatment recommender system using a cox proportional hazards deep neural network," *BMC Medical Research Methodology*, vol. 18, 02 2018.

[24] Z. Huang, C. Yu, J. Ni, H. Liu, C. Zeng, and Y. Tang, "An efficient hybrid recommendation model with deep neural networks," *IEEE Access*, vol. 7, pp. 137 900–137 912, 2019.

[25] H. Dalianis, *Evaluation Metrics and Evaluation*, 05 2018, pp. 45–53.

[26] F. Carvalho and C. Castro, "Empirical analysis on the state of transfer learning for small data text classification tasks using contextual embeddings," in *Anais do 14 Congresso Brasileiro de Inteligência Computacional*, B. J. T. Fernandes and A. {Pereira Júnior}, Eds. ABRICOM, 2019, pp. 1–7.

Professor Jaafar Abouchabaka A permanent professor at the faculty of science in Ibn Tofail University. He is specialized in the field of computer science, especially data mining and artificial intelligence

Dr. Asmae Bentaleb A graduate student of Al Akhawayn University majored in the field of software engineering with a minor of business administration. She obtained her masters degree in data mining and warehousing. She got her phd from Ibn Tofail University in the field of big data and artificial intelligence.