Sentiment Analysis

Igor Mamorski (ID. 326962784), Nadav Talmon (ID. 203663950)

Submitted as final project report for the NLP course, Reichman University, 2023

1 Introduction

In this paper, we present our efforts to fine-tune the powerful BERT (Bidirectional Encoder Representations from Transformers) model for sentiment analysis and explore its ability to detect sentiments in texts translated from other languages using machine translation techniques.
This project aims to understand how well BERT's sentiment analysis generalizes across languages. By fine-tuning on English data and testing on translated reviews, we seek to evaluate the model's cross-lingual adaptability and potential challenges in handling linguistic variations.
We are aware of the existence of pre-trained models that can achieve relatively high accuracy on single or multilingual sentiment analysis tasks. However, our main goal in this work was to assess the performance of the basic BERT model in handling reviews translated from languages other than English. Additionally, we are comparing its performance against that of the pre-trained multilingual BERT model.
This report presents our methodology, data collection, and results, shedding light on BERT's performance in sentiment analysis across languages.

1.1 Related Works

This project uses the pre-trained Bert model proposed by Google researchers Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova [1].
Furthermore, this project draws inspiration from the techniques described in work titled "How to Fine-Tune BERT for Text Classification?" by Chi Sun, Xipeng Qiu, Yige Xu, Xuanjing Huang et al. [5]. The paper provided valuable insights into the fine-tuning process of BERT for text classification, which has been instrumental in our endeavor to tailor BERT for sentiment analysis.

2 Solution

2.1 General approach

Our general approach is to understand the key features in a written review with respect to its sentiment and apply it as a preprocessing step to enhance the performance of our data set in the downstream task of sentiment analysis. We achieve this by fine-tuning a pretrained Language Model (LLM).
We explore two main approaches:

1. A model trained exclusively on reviews in English (fig. 1).
2. A model that is trained on reviews in multiple chosen languages (fig. 3).

During inference, the English model receives a translated review from another translation model, allowing it to process and analyze reviews in English regardless of the original language. In contrast, the multi-language model directly processes the review in its original form without any translation (fig. 2 and 3). By utilizing these two distinct approaches, we endeavor to evaluate their respective strengths and potential impact on sentiment analysis results. The English model relies on translated inputs to comprehend non-English reviews, while the multi-language model inherently handles the linguistic diversity present in the dataset.
Furthermore, we adopt two primary approaches for labeling the sentiment in our project:

1. Five labels - Negative, Somewhat Negative, Neutral, Somewhat Positive, Positive
2. Two labels - Negative, Positive

During the preprocessing of datasets, we take into consideration the manner in which humans deduce the sentiment of a review. Observably, the sentiment tends to be predominantly situated at the beginning or the end(or even both) of a review, while the more detailed aspects related to the product, movie, or other subjects are often found in the middle.
While our ideal scenario would involve processing the entire review comprehensively, this approach proves to be computationally expensive and practically infeasible due to limited resources and the presence of very long reviews.
Hence, we adopt an alternative preprocessing methodology. We concatenate the first N tokens with the last M tokens, thereby forming a new review that maintains the same sentiment. This approach allows us to strike a balance between computational efficiency and sentiment representation.
During this process, we came across a noteworthy paper titled "How to Fine-Tune BERT for Text Classification?" [5], which delves into a similar experimental approach. The insights from this paper have been influential in guiding our own methodology.

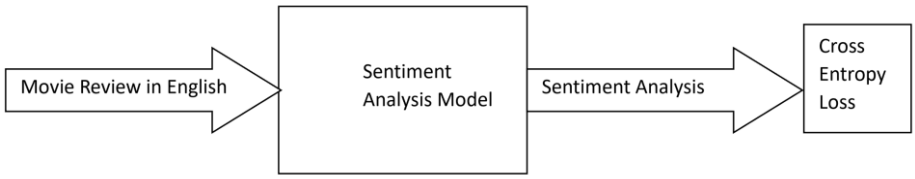

Figure 1: Training approach of English only model

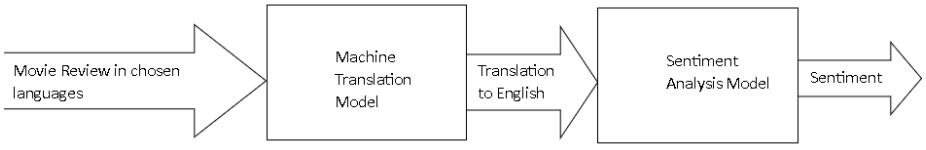

Figure 2: Inference Approach of English only model

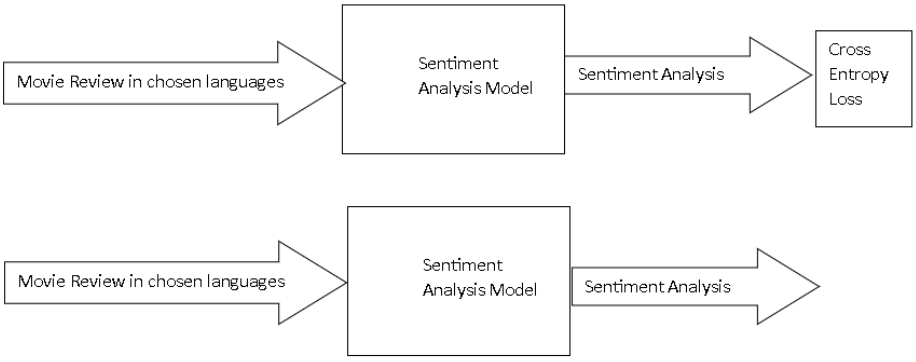

Figure 3: Training and Inference Multilingual Model

2.2 Design

To conduct our experiments, we utilize the open-source amazon_multi_review [2] and IMDB [3] datasets, which are readily available on HuggingFace. During our investigation, we explore various pretrained models, including BERT, XLNET. Bert was explored in its cased and uncased versions.

Due to the IMDB dataset consisting of English-only reviews, which did not align with the primary goal of our project, we directed our focus mainly on the Amazon dataset. To handle the translation of its non-English reviews in the test set, we employed a machine translation solution (further described below) during the inference stage. This approach allowed us to effectively work with non-English reviews, which were essential for our project's objectives.

Throughout this project, our main computing resources consisted of the Google Colab and Kaggle platforms, which offered free GPU usage, effectively alleviating the computational burden. Additionally, we utilized a home PC equipped with an NVIDIA GeForce GTX 1660 TI for certain tasks.

The dataset contains reviews in English, Japanese, German, French, Chinese, and Spanish. For each language, there are 200,000, 5,000, and 5,000 reviews in the training, development, and test sets, respectively. The maximum number of reviews per reviewer is 20, and the maximum number of reviews per product is 20. All reviews are truncated after 2,000 characters, and all reviews are at least 20 characters long. For labeling, each review is associated with a star rating ranging from 1 to 5, indicating the sentiment or rating given by the reviewer.

Our experimental process began by fine-tuning the BERT-cased and uncased models on the entire Amazon reviews dataset. However, due to resource constraints, the training duration for the complete dataset extended to approximately 22 hours. This extended training time posed challenges in terms of computational resources and time management. To address these challenges, we decided to adopt alternative strategies to enhance the training process. One such approach was reducing the dataset size. As a first step, we removed Chinese and Japanese from the dataset. Then, we choose a subset of examples from the dataset. This reduction allowed us to work with a more manageable subset of data without compromising the overall quality of information for our analysis. This additional measure further streamlined the data and contributed to shorter training times.

By combining these two data reduction steps, we significantly decreased the size of the dataset, resulting in a much more efficient training process. As a result, the training duration was notably reduced to approximately 2 hours, enabling us to expedite the experimentation phase while still obtaining meaningful and reliable results.

As the subsequent step, we employed the "head+tail" approach for preprocessing the reviews, which was previously described in [5]. In this approach, each review is tokenized, then, the first N head tokens and the last M tail tokens are concatenated to form the final token sequence. The paper suggested using $N = 128$ and $M = 382$. We also conducted experiments with different head and tail lengths, such as $N = M = 64; N = 510, M = 0; N = 0, M = 510$, the

paper's approach surprisingly did not always achieve the best results.
For training our models, we employed the AdamW optimizer with a learning rate of $1e-3$ and utilized 10000 warmup steps. These settings allowed us to effectively fine-tune our models and achieve promising results in our sentiment analysis task.
After the initial cycles of training, the multiclass analysis results did not meet our expectations. In order to shift our focus towards comparing multilingual and translated test results, we made a decision to convert the dataset into a binary set.
To achieve this, we performed the following transformations:

1. We dropped reviews labeled with 3 to improve results. Neutral responses sometimes caused ambiguity, leading to lower accuracy in sentiment analysis. By eliminating the neutral category, we aimed to enhance performance and clarity in our sentiment classification task.

2. We merged the labels 1 and 2 into the negative category.

3. We merged the labels 3 and 4 into the positive category.

By applying these transformations, we successfully transformed the original multiclass sentiment labels into a binary sentiment categorization, which enabled us to concentrate on comparing the performance of multilingual and translated test results in a more straightforward manner.
Despite overcoming the challenges posed by limited computing resources, we encountered certain limitations at various stages of the project. Particularly, efficient translation of the datasets emerged as a significant challenge. Initially, we opted to use the Google Translate API as our primary approach. However, this method proved to have several disadvantages, including prolonged translation times and severe restrictions on the free tier, which hindered our progress. These limitations compelled us to seek alternative solutions to overcome the translation bottleneck and continue with our research effectively.
As an alternative approach, we explored the T5 model [4] for translation purposes. While the T5 model demonstrated effectiveness in translating sentences, it faced the limitation of truncating the reviews to a degree that rendered them unsuitable for our specific requirements.
After multiple attempts over several days, we eventually found a free Python library that efficiently utilized the Google Translate API. Since our specific need was to translate the test set, which had already been shortened by removing Japanese and Chinese languages, this approach proved more suitable. With this method, we successfully obtained the translated test set within a reasonable time frame of approximately 4 hours, enabling us to proceed with our evaluations.

3 Experimental results

Throughout our research, we conducted numerous training iterations using cased and uncased variations of BERT, along with the multilingual model. However, during each training process, we observed that the models tended to stop improving after just 1-2 epochs. As a result, the majority of our training sessions comprised only 1 epoch. Despite this constraint, we focused on optimizing our training approach to achieve the best possible results within these limitations.
When evaluating the English-only model approach, we had more models as options to use. We tried to train other models, such as XLNET as well. XLNET proved to be highly effective. Regardless of the chosen model, we tried training on the Amazon data set and evaluating both translated test set and the test set itself of Amazon as well as the IMDB test set and vice versa. We did this to test which dataset allows the model to generalize better and yield an overall higher accuracy.
In 1 presented results with the highest F1 score for each model while using binary classification, meaning that we're splitting the labels into two categories: "Negative", and "Positive".
Table columns description:

1. Model Name - Model used for fine-tuning.
2. Neutral:
    - Yes - when splitting the labels, neutral label 3 was merged with the positive labels.
    - No - label 3 was omitted.
3. Dataset - which dataset was used for fine-tuning.
4. Test Set - which test set was used for evaluation:
    - Test - The dataset's test set with its corresponding languages.
    - Translated - Spanish, German, and French reviews from the Amazon dataset translated to English.
5. F1 Score - Weighted average F1 score.
6. N - number of head tokens used for truncation
7. M - Number of tail tokens used for truncation.

Model Name	Neutral	Dataset	Test Set	F1 Score	N	M
XLNET Cased	-	IMDB	IMDB Test	0.94835	0	510
XLNET Cased	-	IMDB	Amazon Translated	0.8817	64	64
Bert Cased	-	IMDB	IMDB Test	0.91115	128	382
Bert Cased	-	IMDB	Amazon Translated	0.8555	510	0
Bert Uncased	-	IMDB	IMDB Test	0.93	128	382
Bert Uncased	-	IMDB	Amazon Translated	0.8779	0	510
XLNET Cased	Yes	Amazon	Amazon Test	0.87912	128	382
XLNET Cased	Yes	Amazon	Amazon Translated	0.86606	128	382
XLNET Cased	Yes	Amazon	IMDB Test	0.89125	0	510
Bert Cased	Yes	Amazon	Amazon Test	0.87574	0	510
Bert Cased	Yes	Amazon	Amazon Translated	0.86294	0	510
Bert Cased	Yes	Amazon	IMDB Test	0.84195	0	510
Bert Multilingual Cased	Yes	Amazon	Amazon Test	0.82722	128	382
Bert Multilingual Cased	Yes	Amazon	Amazon Translated	0.8556	128	382
Bert Multilingual Uncased	Yes	Amazon	Amazon Test	0.83428	0	510
Bert Multilingual Uncased	Yes	Amazon	Amazon Translated	0.8657	0	510
Bert Uncased	Yes	Amazon	Amazon Test	0.86838	128	382
Bert Uncased	Yes	Amazon	Amazon Translated	0.85336	128	382
Bert Uncased	Yes	Amazon	IMDB Test	0.86845	0	510
XLNET Cased	No	Amazon	Amazon Test	0.92495	510	0
XLNET Cased	No	Amazon	Amazon Translated	0.9216	64	64
XLNET Cased	No	Amazon	IMDB Test	0.91305	0	510
Bert Cased	No	Amazon	Amazon Test	0.918	128	382
Bert Cased	No	Amazon	Amazon Translated	0.9106	128	382
Bert Cased	No	Amazon	IMDB Test	0.85455	0	510
Bert Uncased	No	Amazon	Amazon Test	0.92125	64	64
Bert Uncased	No	Amazon	Amazon Translated	0.91275	64	64
Bert Uncased	No	Amazon	IMDB Test	0.8681	0	510
Bert Multilingual Cased	No	Amazon	Amazon Test	0.90665	64	64
Bert Multilingual Cased	No	Amazon	Amazon Translated	0.90805	64	64
Bert Multilingual Uncased	No	Amazon	Amazon Test	0.912	0	510
Bert Multilingual Uncased	No	Amazon	Amazon Translated	0.9094	0	510

Table 1: Training Results (Best N,M Foreach Model)

4 Discussion

In this project, we presented our methodology for sentiment detection in reviews composed in languages other than English. We accomplished this by leveraging models trained on English datasets. Our investigation demonstrated that these models could proficiently assess sentiment even when employed to analyze translated reviews from diverse languages. This revelation underscores the model's resilience and proficiency in performing cross-language sentiment analysis.
Interestingly, the XLNET model exhibited slightly superior performance compared to the Bert model in the English-only case.
Another interesting discovery is that models trained on the Amazon dataset were able to generalize better than on IMDB. That is, models trained on Amazon's dataset perform better on the IMDB dataset and translated set than model trained on IMDB dataset and tested on Amazon's translated set.

5 Code

Our code is provided in the following link: nlp-project.

References

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[2] Phillip Keung, Yichao Lu, György Szarvas, and Noah A. Smith. The multilingual amazon reviews corpus, 2020.

[3] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.

[4] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2020.

[5] Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. How to fine-tune bert for text classification?, 2020.