Comparison of mono and multilingual language models in sentiment classification

Baiqing Lyu
Boston University
baiqing@bu.edu

Ciaran Hikaru Ueda Fitzgerald
Boston University
cueda@bu.edu

Richard Chen
Boston University
richchen@bu.edu

Bowen Li
Boston University
bown@bu.edu

Tengzi Zhao
Boston University
tengzi@bu.edu

Shen Yan
Boston University
sy5nb@bu.edu

Abstract

Training multilingual models for sentiment analysis: by fine-tuning pre-trained multilingual language models: multiBERT, XLM-Roberta, and MT5 and their corresponding monolingual variant on the sentiment prediction task. We compare the performance of such multilingual models with (1) monolingual English model trained on English tweets and test on Arabic tweets translated to English (with pre-trained machine translation or Google Translate) and (2) multilingual model trained on English tweets and test on Arabic tweets. This is done to discover if multilingual models benefit from being trained on multilingual data.

Associated code and related documentation can be found here: `https://github.com/BaiqingL/CS505-Final`

1 BERT based models

For this section, we discuss the differences between the monolingual **B**idirectional **E**ncoder **R**epresentations from **T**ransformers (BERT) model and its multilingual multi-BERT model. We train the BERT model on English tweets and test with Arabic tweets translated to English, and train multiBERT on English tweets and test on Arabic tweets instead of the translated version.

We fine tuned both models for 4 epochs each. We chose to use the adamW optimizer implemented by PyTorch instead of the old huggingface implementation. This decision is made due to the fact that huggingface's optimizer has been deprecated. The chosen learning rate is 2×10^{-5} and the chosen epsilon is 1×10^{-8}. We chose this to mirror the training decisions of our previous homework assignments as that was the recommended parameters for these models. In addition, both `BERT` and `multiBERT` have been given the same hyper parameters as to ensure the only differentiating factor is the previous training and the fact that one model has been exposed to multilingual data while another one has not.

1.1 BERT Results

For the BERT model, the accuracy on English tweets translated from Arabic tweets after training on English tweets is 59.43%

Below is a figure of the training history, show-casing the training and validation loss of the fine tuning process for each epoch.

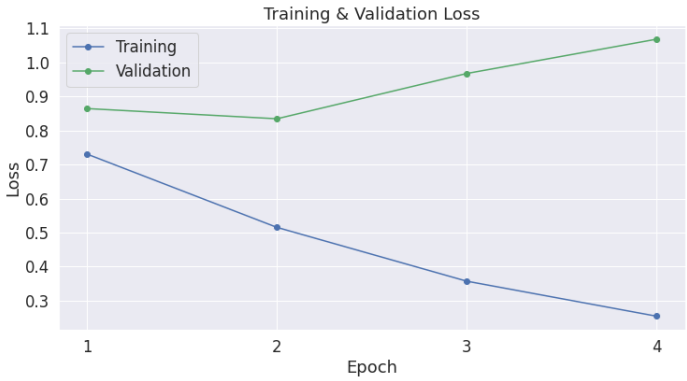

Figure 1: BERT training and validation loss.

1.2 MultiBERT Results

For MultiBERT, the zero-shot classification on Arabic tweets after training on English tweets is 50.82%. In addition to the project scope, we also tested this model on the same set of English tweets and found that the accuracy of this model is 58.03%.

Below is a figure of the training history, show-casing the training and validation loss of the fine tuning process for each epoch.

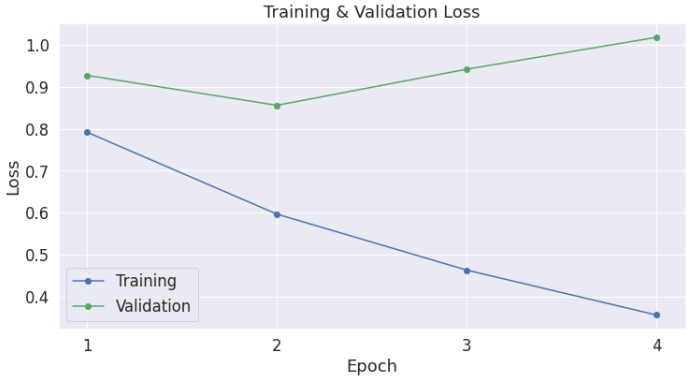

Figure 2: MultiBERT training and validation loss.

1.3 Comparison

We found that although `MultiBERT` has been exposed to more diverse data as compared to its monolingual base, it performs slightly worse even on the same testing data. The hypothesis is that since different languages has different ways of expression, it makes the model more complex and allows it to adapt to more languages at the cost of some accuracy.

In addition, it is important to note that the zero-shot classification model accuracy was over 50 percent for 3 classes, meaning that the model was definitely not randomly guessing. This shows there is some generalization of the multilingual model.

Similar studies found that both kinds of models achieve similar performance.[1] In our case, both models did have similar performance on the same testing data, however when the multilingual model was exposed to the pre-translated tweets, it did perform noticeably worse as to the monolingual model.

[1]Feijó, D.D., & Moreira, V.P. (2020). Mono vs Multilingual Transformer-based Models: a Comparison across Several Language Tasks. ArXiv, abs/2007.09757.

2 RoBERTa based models

The next pair of models are `RoBERTa` and `XLM-RoBERTa`. We used the same hyper-parameters and changed the batch size to 16 for `RoBERTa` and `XLM-RoBERTa` models. We trained each model for 4 epochs using the `AdamW` optimizer provided by HuggingFace.

2.1 RoBERTa Results

For the RoBERTa model, the accuracy on English tweets translated from Arabic tweets after training on English tweets is 61.22%

Below is a figure of the training history, show-casing the training and validation loss of the fine tuning process for each epoch.

Figure 3: RoBERTa training and validation loss.

2.2 XLM-RoBERTa Results

For XLM-RoBERTa, the zero-shot classification on Arabic tweets after training on English tweets is 54.39%. In addition to the project scope, we also tested this model on the same set of English tweets and found that the accuracy of this model is 59.14%.

Below is a figure of the training history, show-casing the training and validation loss of the fine tuning process for each epoch.

Figure 4: XLM-RoBERTa training and validation loss.

2.3 Comparison

We found that the accuracy of XLM-RoBERTa on the arabic tweets is lower than that of RoBERTa on the machine translated tweets. Also, each RoBERTa based model performed better than its corresponding BERT based model.

Since the RoBERTa based model is basically a better version of the BERT model, it makes sense that the comparison between the RoBERTa based models is similar to the comparison between the BERT models.

3 T5 based models

The final pair of models are `T5` and `mT5`. T5 stands for **T**ext-**t**o-**T**ext **T**ransfer **T**ransformer, which considers all NLP tasks as being text-to-text and is trained on the "Colossal Clean Crawled Corpus" (C4) dataset[2]. mT5 is a multilingual version of T5, which used mC4, an extension of C4 which covers 101 languages[3]. The difference that sets T5 apart from BERT-style models is that instead of labels, T5 outputs a text string as well. For our purposes, the inputs of the text-to-text task are tweets and the outputs are one word sentences of "positive", "neutral", or "negative" sentiment. This seamlessly adapts the T5 and mT5 models designed for text-to-text tasks to the task of sentiment classification.

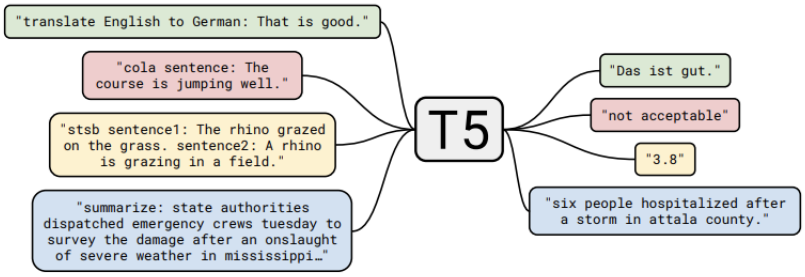

Figure 5: T5 treats every NLP task as a text-to-text task.[4]

Both models were trained on English tweets, with similar hyper-parameters. We trained both models for 2 epochs using the AdamW optimizer. We used a learning rate of 0.0003 and epsilon of 0.00000001 for the optimizer. The only difference was a batch size of 8 used for T5, and a batch size of 2 used for mT5. These hyper-parameters were used in a similar project which we adapted for this task[5]. We trained mT5 additionally with a learning rate of 0.00003 which improved the accuracy, but we will focus on the accuracy between the models with the same hyper-parameters for the sake of comparison.

3.1 T5 Results

For the T5 model, the accuracy on English tweets translated from Arabic tweets was 62.17%.

3.2 mT5 Results

For the mT5 model, the accuracy on Arabic tweets (zero-shot classification) was 25.00%.

The accuracy of the version trained with a lower learning rate was 36.92%.

3.3 Comparison

The T5 model had the best accuracy of any model that we tested, proving its robustness for many NLP tasks. However, the mT5 model performed extremely poorly. It turned out that the mT5 model learned

[2]Raffel et al., "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer." 2

[3]Xue et al., "MT5." 2

[5]Patil "exploring-T5"

to predict positive sentiment for most tweets, sometimes predict neutral and would never predict negative sentiment. We believe this may be caused by a bias in the training dataset, in combination with the challenge of zero-shot classification. The training dataset contained relatively few negative sentiment tweets, so the model may have over fit to predict positive and negative sentiments. Another possibility is that the low accuracy for mT5 is due to the small batch size 2, which is very close to online learning and is not widely used in similar projects. The mT5 paper reported results on sentence-pair classification, structured prediction and question answering tasks, but nothing explicitly on sentiment classification so we do not have a baseline to compare to[6].

4 Conclusion

For all three models, we have largely observed the same trend. All monolingual variants of a certain model structure performed somewhat better with the machine translated text than the multilingual model with the origin language text.

One possibility is that multilingual models achieve their ability to adapt on multiple languages at the sacrifice of accuracy. This would show a trade off of why not to use multilingual language models for every task possible and the use case for monolingual models in language inference.

There are many reasons this trade off could exist. One reason may be that languages have different nuances of expression, and one model may be insufficient to capture them all with the same degree of accuracy as a specialized monolingual model. Monolingual tokenizers has also been observed to provide better performance by replacing out multilingual tokenizers in their models during a monolingual downstream task. [7]

It is also noteworthy that the monolingual models were able to be used because there was access to a machine translation system. In the specific case of sentiment analysis, the vocabulary used in a sentence is generally a stronger determinant of overall sentiment than the structural or grammatical features of the sentence. It just so happens that although machine translation may struggle with accurately translating the structural and grammatical features, it can still translate the vocabularies fairly accurately, thus making the method effective specifically for sentiment analysis. This could be a major contributor towards the effectiveness of the monolingual models that leveraged machine translation over the multilingual ones that did not. Further research can be done on comparing monolingual and multilingual models' performances on more complex tasks involving natural language.

[6] Xue et al., 4

[7] Rust et al., "How Good is Your Tokenizer? On the Monolingual Performance of Multilingual Language Models."

References

[1] Feijó, D.D., & Moreira, V.P. (2020). Mono vs Multilingual Transformer-based Models: a Comparison across Several Language Tasks. ArXiv, abs/2007.09757.

[2] Patil, Suraj. (May 2020). "exploring-T5" https://github.com/patil-suraj/exploring-T5/blob/master/t5_fine_tuning.ipynb

[3] Raffel, Colin, 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," 2019. https://doi.org/10.48550/ARXIV.1910.10683.

[4] Rust, Phillip, Jonas Pfeiffer, Ivan Vulic, Sebastian Ruder and Iryna Gurevych. "How Good is Your Tokenizer? On the Monolingual Performance of Multilingual Language Models." ACL (2021).

[5] Xue, Linting, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. "MT5: A Massively Multilingual Pre-Trained Text-to-Text Transformer," 2020. https://doi.org/10.48550/ARXIV.2010.11934.