Leveraging High-Resource NMT Models to Translate Low-Resource Related Languages Through Adaptive Layers

Jardenna Mohazzab, Mehdi Oueslati, Venkat Mohit Sornapudi, Lea Tiyavorabun

Abstract

In this research we propose and evaluate methods of leveraging a *European Portuguese (eu-PT) to English* pre-trained model to translate the low-resource language pair *Brazilian Portuguese (br-PT) to English* while not degrading the performance on the high resource language pair. We investigate the performance tradeoff between two language pairs in a set of experiments. We find that the eu-PT and br-PT are substantially different and specialised models achieve the best performance on them. We demonstrate that adaptive layers are an effective tool to fight catastrophic forgetting and at the same time adapt to a new language variant.

1 Introduction

Neural machine translation (NMT) is the main approach for translating language pairs. However, State-Of-The-Art (SOTA) technologies like BERT and GPT3 are very data intensive. Therefore, these models are not suitable for a low-resource setting where little data is available. Low-resource settings are ill-defined and are generally taken to mean *less studied, resource scarce, less computerised, less privileged, less commonly taught, or low density*[MCH20]. Out of simplicity, they are often binned and very low-resource languages are defined as having parallel corpora smaller than 100k sentences [GGC+22], where SOTA language models such as BERT are trained with datasets larger than 3.3 billion words [DCLT18].

Currently there exist many low-resource language pairs in natural language processing (NLP). One of them is Brazilian Portuguese (br-PT) to English, with available corpora of around 5k sentence pairs. Thereby, it is hard to train a br-PT to English model on this small set of data.

However, since eu-PT and br-PT are similar we can leverage a pre-trained eu-PT to English model to translate from br-PT to English. There are several methods to approach this. The first being a simple zero-shot translation on a simple eu-PT to English Model. The second by using the Base Model and fully fine-tune it to br-PT. The third method is to freeze the Base Model and add adaptation layers and solely train the adaptation layers.

The benefits of the third approach are that there are fewer parameters to train, so training goes faster and is less computationally intensive. Secondly, the major benefit of the third method arises from the fact that since br-PT is so close to eu-PT, there are only a few layers we need to add to make it translate to br-PT. This is interesting because in this case we do not experience the phenomenon of catastrophic forgetting, because all the eu-PT weights are frozen and can be preserved, and all the knowledge for br-PT is in the plugin layer. This means the plugin can be used to translate to br-PT, but when the plugin is removed the eu-PT model is still intact. Consequently, we aim for a model that can translate both br-PT and eu-PT.

In this research we propose and evaluate several methods of leveraging a eu-PT to English pre-trained model to translate the low-resource language pair br-PT to English while not degrading the performance on the high resource language pair. Additionally, our architecture allows for good performance on br-PT and eu-PT, without a token indicating which dialect we are feeding the model.

2 Background

2.1 Catastrophic forgetting

Catastrophic forgetting occurs specifically when a neural network is trained sequentially on multiple tasks since the weights in the network that are important for task A are changed to meet the objectives of task B [Fre99]. Catastrophic forgetting in neural networks is a significant problem for

continual learning. In continual learning a model is learned for a large number of tasks sequentially without forgetting knowledge acquired from the preceding tasks, where the data in the previous tasks are not available anymore during training new instances [PHHL19]. The larger part of the current methods replay previous data during training, which violates the constraints of an optimal continual learning system [KPR+17]. There has been work that addresses the problem of catastrophic forgetting by introducing Relevance Mapping Networks (RMNs) [KPR+17]. The mappings reflect the relevance of the weights for the regarding task by assigning large weights to essential parameters where the RMN learns an optimised representational overlap that alleviates catastrophic forgetting [KPR+17].

2.2 Fine-tuning

Fine-tuning is a very popular way of applying or utilising transfer learning. Specifically, fine-tuning is a process that takes a model that has already been trained for one given task and then tunes the model to make it perform on a second similar task [TYNY19].

2.3 Adaptation layers

[HGJ+19] introduced adapters in 2019. Adapters serve the same purpose as fine-tuning but do it by stitching in layers to the main pre-trained model, and updating the weights of these new layers, whilst freezing the weight of the pre-trained model. On the other hand, in fine-tuning the pre-trained weights are also updated. This makes adapters more efficient, both in terms of time and storage, compared to fine-tuning. The benefits of adapters are in short, 1) task-specific layer-wise representation learning, 2) small, scalable, shareable, 3) reproducibility, 4) modularity of representations, and 5) non-interfering composition of information. In NLP we often use Multi-Task Learning (MTL) but MTL suffers from catastrophic forgetting and catastrophic inference. With adapters, we train the adapter for each task separately, overcoming both issues [PRP+20].

3 Model Architectures

To investigate how adaptive layers can improve low-resource translation, we designed an adaptive architecture, a baseline, and fine-tuned the baseline. We started by forking the official tutorial for Transformer-driven translations on Tensorflow: 'Neural machine translation with a Transformer and Keras' [neu].

3.1 Tokenizer

We used the tokenizer from [neu]. During training, the model solely sees eu-PT data. Because eu-PT and br-PT are similar the same tokenizer can be used. In general eu-PT and br-PT are mutually intelligible. While there are subtle variations in grammar, lexical preferences, and spelling between the two standards, it is their pronunciations that differ the most [sim].

3.2 Base Model

For the structure of the Base Model we implement the original transformer with an encoder and a decoder. However, with fewer layers, to keep the training time reasonable, we refer you to [neu]. More specifically, we reduce the number of layers of the encoder and decoder from eight to four. Furthermore, we reduced the layer size for the Feed-forward Neural Networks in the encoder and decoder from (512, 1024) to (128,512). We trained the model on the *eu-PT - English* dataset for 20 Epochs.

Since the target sequences are padded, it is important to apply a padding mask when calculating the loss. The loss function that is used during training is cross-entropy loss. In addition, the Adam optimizer is used with a custom learning rate scheduler according to the formula in the original Transformer paper [VSP+17], which can be found in equation 1.

$$lrate = d_{model}^{-0.5} * min(step\_num^{-0.5}, step\_num \cdot warmup\_steps^{-1.5}) \quad (1)$$

3.3 Fine-tuned Model

For the Fine-tuned Model, we used the pretrained Base Model, set all parameters to trainable, and continued training the model. The model was trained on the *br-PT* dataset in order to adapt to the very close dialect. We trained the model for 20 Epochs.

3.4 Adaptive Model

For the Adaptive Model, we again used the pretrained Base Model. We use the pretrained encoder and decoder. Then we insert two Dense layers, our adaptation layers, and add the pretrained final layer.

The encoder, decoder, and final layer are frozen, so in the 20 Epochs we trained, only the adaptation layers were trained on the *br-PT* Dataset.

4 Experiments and Evaluation

We aim for English as a target language. Because we have very little br-PT data, we have a much better chance of adopting a model where we do not change the target language. That is, then at least the target does not need to learn the output of a completely new language.

4.1 Data

We used a br-PT dataset named: Pira [data] and a eu-PT dataset that contains ted talks [datc] merged with a eu-PT dataset from taboeta [datb].

The vocabulary size of the parallel dataset is 54970 for English and 58546 for br-PT. The amount of sentences for both languages are 8668 where the average sentence length is 13.01 and 13.86. In figure 2, 3, 4, and 5 in the Appendix you can see the sequence length distribution for all the languages and datasets. For the dataset on English to eu-PT the vocabulary size is 179622 and 1802647 respectively. The number of sequences is 268452 for both and the average sentence length is 6.69 vs 6.71.

4.2 Experimental setup

In the first experiment we take the eu-PT to English model and perform zero-shot translation to br-PT. In the second experiment we first perform a couple of training passes on the br-PT data whereafter we evaluate the performance on the task of translating br-PT to English. In the third experiment we freeze the model and add adaptation layers and solely train the adaptation layers.

4.3 Evaluation

During evaluation we will use the BLEU scores [PRWZ02] which is a method for automatic evaluation of machine translation. We compute and compare the BLEU scores of the Baseline Model, the Bleu scores of Adaptive Model, and the Fine-tuned Model on European eu-PT and br-PT. The results can be found in Table 2. For a more qualitative evaluation, we look at a few translated sentences Table 1, as well as the attention weights of the Base Model Figure 1.

5 Results and Discussion

Firstly, to make sure that our Base Model works, we take a close look at the performance in quantitative and qualitative manner. We see that the the BLEU scores for eu-PT are quite higher than the BLEU score for br-PT 20.00 vs. 10.98. This is a good indicator that our model is performing well on the original language and it is expected that the zero-shot performance on another language is significantly worse. Furthermore, we take a look at the attention weights, to see if the attention heads weigh words in sentences reasonably. This is indeed the case, as can be seen in Figure 1. 'book' and 'livro' are consistently getting high attention weights. As well as 'primero' and 'the first' or 'first' or 'this is the first', which are all reasonable. Lastly, we take a look at the qualitative analysis of the translation of which an example can be found in table 1. Here we can see that the translation works very well, only with the fill word 'so' prepended to the ground-truth. This is also a very interesting result, as one word added, even if it is a fill word like in this scenario not changing the meaning, it reduces the BLEU score of this sentence quite a bit. In conclusion we can say, that our Base Model is working properly.

Looking at the Fine-tuned Model, we can see that the baseline model after fine-tuning performs very well on br-PT 30.5, which tops the performance of our Base Model on eu-PT. This makes sense, as the initial training on eu-PT was already done and since the two languages are semantically very close the additional fine-tuning bumped the performance significantly. Very interesting is, however, that the performance of the Fine-tuned Model on eu-PT dropped significantly at the same time, with a BLEU score of 4.53. This can be attributed to catastrophic forgetting, which is very common for continual learning.

Looking at the performance of the Adaptive Model, we see that the BLEU score of br-PT increased from 10.98 to 13.16, so by two BLEU points, which is a good improvement. This improvement is worse than for the Fine-tuned Model (30.50 vs 13.16). This is as expected since the first is a mono lingual and the second being a multilingual model now. In specific, for the Fine-tuned Model all weights are updated which is a significantly higher number of parameters than in the model with adaptation layers where the weights of the initial model are frozen. At the same time the

Input (EU Port.)	este é um problema que temos que resolver.
Prediction	so this is a problem we have to solve.
Ground truth	this is a problem we have to solve .

Table 1: Base Model Translation after training on eu-PT corpus, translating a eu-PT sentence to English.

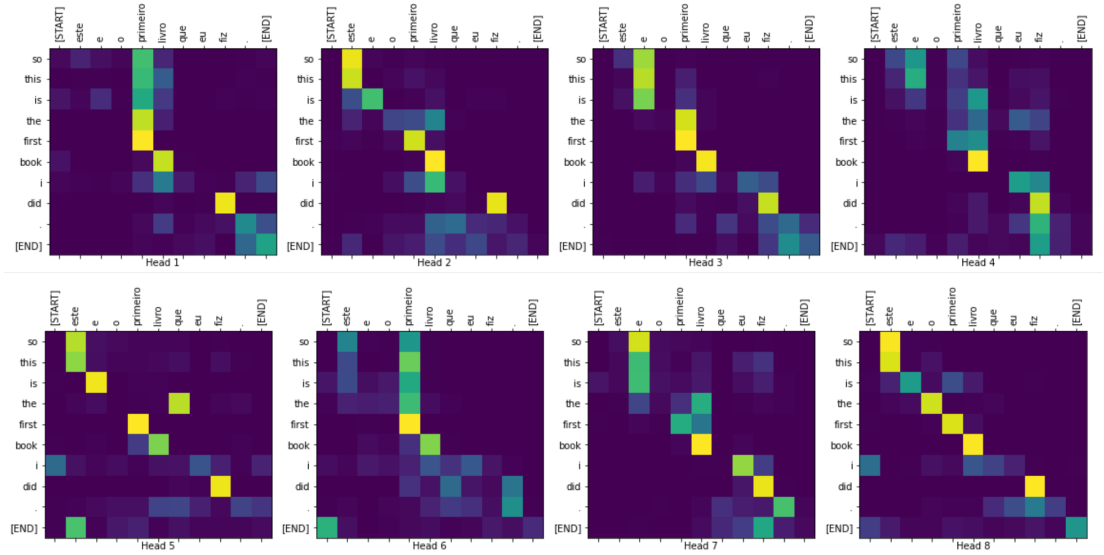

Figure 1: Visualised attention weights for Base Model eu-PT to English

performance on eu-PT decreases, just like in the Fine-tuned Model. However, not as significantly. It drops from 20.00 BLEU points in the Base Model to 13.87 in the Adaptive Model vs. four BLEU points for the Fine-tuned Model. This shows, that adaptive layers help with reducing catastrophic forgetting, as the original model is frozen and the learned distribution is not completely overridden, as it is often the case in catastrophic forgetting while fine-tuning. Additionally, our model has the perk, of being always able to recover the performance of the Base Model, as its parameters are frozen. This can be done by adding a second out put the model, that skips the adaptive layers or by only loading the pretrained model.

To conclude, for the Adaptive Model the BLEU scores of eu-PT and br-PT are similar (13.87 vs 13.16). It is good to see a fair distribution in BLEU score because it is a multilingual model now. Importantly, it is a multilingual model, that does not input a token indicating whether br-PT or eu-PT is the input language.

6 Conclusion and Outlook

We could see that after fine-tuning, catastrophic forgetting occurred on the eu-PT while for the Adaptive model this was not so much the case. Additionally, we noted that the BLEU scores for the model with adaptive layers were fairly distributed among languages which is in line with the notion of it being a multilingual model now. It is important to point out, that the experiment was done in a very low-resource setting and that our multilingual Adaptive Model does not take any input token, that describes which language we are translating from. This is not usually the case and is helpful in the case, that there is no information as to which language is the input. For further research, it would be interesting to add two embedding layers to the Adaptive Model, initialise them with the Base Model Embeddings and let them adapt in the training process. We expect it to lead to an increased performance, as right now the embedding layer did not change, which might be sensible since it is a translation from a different language.

	eu-PT	br-PT
Base Model	20.00	10.98
Fine-tuned Model	4.53	30.50
Adaptive Model	13.87	13.16

Table 2: BLEU scores of translating eu-PT and br-PT to English for several approaches.

References

[data] Pirá: A Bilingual Portuguese-English Dataset for Question-Answering about the Ocean, the Brazilian coast, and climate change. https://github.com/C4AI/Pira. Accessed: 2022-09-19.

[datb] Taboeta dataset. https://tatoeba.org/en/downloads. Accessed: 2022-09-19.

[datc] tensorflow dataset: When and Why are Pre-trained Word Embeddings Useful for Neural Machine Translation? . https://github.com/neulab/word-embeddings-for-nmt. Accessed: 2022-09-19.

[DCLT18] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2018.

[Fre99] Robert M French. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, 3(4):128–135, 1999.

[GGC+22] Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. The Flores-101 Evaluation Benchmark for Low-Resource and Multilingual Machine Translation. *Transactions of the Association for Computational Linguistics*, 10:522–538, 05 2022.

[HGJ+19] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR, 09–15 Jun 2019.

[KPR+17] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

[MCH20] Alexandre Magueresse, Vincent Carles, and Evan Heetderks. Low-resource languages: A review of past work and future challenges. 2020.

[neu] Neural machine translation with a Transformer and Keras. https://www.tensorflow.org/text/tutorials/transformer. Accessed: 2022-09-19.

[PHHL19] Dongmin Park, Seokil Hong, Bohyung Han, and Kyoung Mu Lee. Continual learning by asymmetric loss approximation with single-side overestimation. *CoRR*, abs/1908.02984, 2019.

[PRP+20] Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulic, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. Adapterhub: A framework for adapting transformers. *CoRR*, abs/2007.07779, 2020.

[PRWZ02] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics.

[sim] Similarity of brazilian and portuguese. https://www.portuguesepedia.com/european-vs-brazilian-portuguese/. Accessed: 2022-09-19.

[TYNY19] Edna Chebet Too, Li Yujian, Sam Njuki, and Liu Yingchun. A comparative study of fine-tuning deep learning models for plant disease identification. *Computers and Electronics in Agriculture*, 161:272–279, 2019.

[VSP+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.

7 Appendix

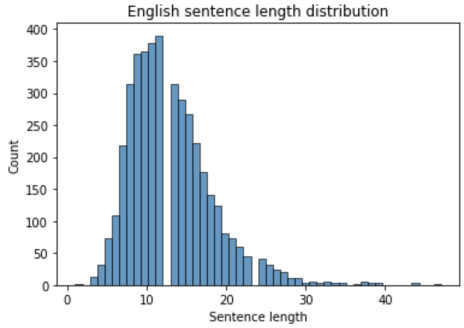

Figure 2: English sequence length distribution br-PT dataset

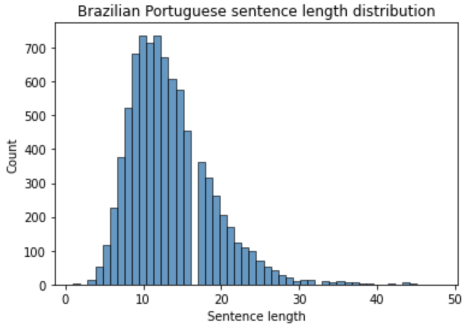

Figure 3: br-PT sequence length distribution br-PT dataset

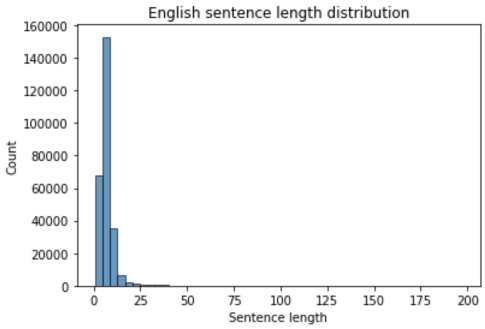

Figure 4: English sequence length distribution eu-PT dataset

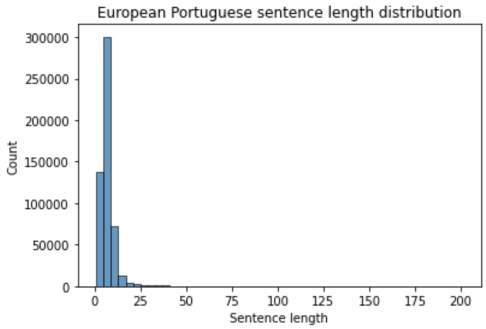

Figure 5: eu-PT sequence length distribution eu-PT dataset