Transfer Learning across Transformer Models on Multi Language Translation Task

Kevin Ngo, Prakhar Maini
W266: Natural Language Processing
UC Berkeley School of Information
{kngo, prakharmaini}@berkeley.edu

Abstract

*Recent work in translation [1, 2] has focused on creating large-scale multilingual transformer models capable of achieving state-of-the-art performance. These models attempt to leverage the shared structure of multiple languages to improve down-stream task performance in **low-resource languages**. In this work, we explore the potential of transfer learning between transformer models using two recently released multilingual transformer models (M2M-100 [1], T5 [3]) by utilizing the M2M-100's embeddings within the T5's architecture in order to expand T5's translation capabilities to low resource languages such as Estonian and Macedonian. Our experiments suggest that usage of T5's pretrained encoder & decoder weights improves model performance on previously untrained low-resource language translation even with small training dataset compared to random initialization of the weights. Augmenting the training with pre-learnt multilingual embeddings improves the performance further but this improvement is only seen with larger training set sizes.*

Keywords: Multilingual Machine Translation, Transformer, Transfer learning

1 Introduction

Neural network models have been very successful for a variety of Natural Language Processing (NLP) tasks including machine translation [6, 7]. These tasks often make use of transfer learning to quickly achieve strong results on the tasks without expensive and time-consuming training. However, most existing models [3] are trained on English centric tasks, limiting the use of the transfer learning framework significantly for translation tasks as roughly 80% of the world population doesn't speak English [4]. Recently, researchers have delivered multi-language transformer models (e.g. M2M-100 [1] and mT5 [2]) trained on massive multi-language corpuses. Previous translation models (e.g. BERT [5] and T5 [3]) have mostly focused on translation tasks between English and other high resource languages (i.e. French). The T5 model has shown promise in being applicable in multiple transfer learning tasks including translation. These translation tasks are limited to translation English to Russian, French, or Romanian. However, with the recent M2M-100 model [1], researchers were able to incorporate low resource languages within their model and the focus on non-English-Centric model brought gains of more than 10 BLEU when directly translating between non-English directions while performing competitively to the best single systems from WMT [1].

We explore the possibility of expanding T5's translation capabilities for low resource languages translation by utilizing the trained embeddings from the M2M-100 model. By using the tokenizer and embeddings from the M2M-100 model with the T5 architecture, we expect to transfer the learned multilingual translation context.

For our paper, we took a non English-centric approach to translation by selecting **English (en)** and two low resource languages – **Estonian (et)** and **Macedonian (mk)**. The motivation was to showcase the validity of transfer learning across the transformer models and understand the impact of utilizing learnt embeddings of low-resource language tokens in

multilingual translation. We used relatively small training sizes (30K, 60K and 120K sentences) in our experiments.

2 Overview

2.1 Background

In this section, we provide a brief review of the M2M-100 and T5 models which are at the center of our experiments in this paper. T5 is an encoder decoder type language model closely following Vaswani et al. 2017 [7] proposed transformer model architecture. T5's primary distinction is to treat every text processing problem as a "text-to-text" problem which allows to directly apply the same model, objective, training procedure and decoding process to every covered task and makes T5 a very flexible model able to perform multiple tasks simultaneously (including translation, summarization, question answering, classification etc.)

M2M-100 is a multi-language machine translation model that was trained on large scale many-to-many datasets of 100 languages. Multilingual translation models factorize computation when translating to many languages and share information between similar languages, which benefits low resource directions [8] and enables zero-shot translation [9]. The original transformer architecture has been designed for the bilingual case, where the target language is fixed. In the case of multilingual machine translation, the target language is not fixed. As a novelty, M2M-100 introduces a special language token in the encoder indicating the source language and, in the decoder, indicating the target language.

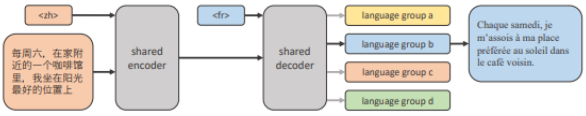

Figure 1: Architecture outline of M2M[1] model

In essence, T5 and M2M-100 share the core encoder-decoder architecture however M2M-100 tweaks the architecture by introducing the source and target language token and is more versatile due to being trained on an extensive corpus of 100 languages in multiple directions. The quantity of data required to do Many-to-Many translation increases quadratically with the number of languages, making neural networks with standard capacity underfit rapidly. M2M introduces many novelties to train larger models and considers a deterministic mixture -of-experts strategy to split the model parameters into non-overlapping groups of languages to reduce the need to densely update parameters and make training more parallelizable. This enabled direct translation between 100 languages without the need to pivot through English with comparable performance to bilingual models on WMT.

2.2 Datasets

Facebook launched the Large-Scale Multilingual MT Task [10] and the Flores-101 dataset at the Workshop for Machine Translation (WMT) focusing on large -scale translation tasks associated with low resource languages with the goal of fostering progress in the field of low-resource Multilingual Machine Translation. The Flores-101 dataset contains translations between 101 languages and 10,100 different directions. We decided to use a subset of the dataset provided from this task by selecting two low resource languages (et and mk) and pairing them with English for a total of 3 languages and 6 directions. The size of the training dataset varies significantly across language pairs. Macedonian (mk) has significantly less training data common with English (en) and Estonian (et) making it a good reference candidate for checking model performance for low resource languages.

Source / Target	EN	ET	MK
EN	0	35.71	2.72
ET	35.71	0	3.07
MK	2.72	3.07	0

Table 1: Number of training sentences in Millions

Table 1 summarizes data available for our translation task and we see that Macedonian (mk) has relatively less sentence pairs available with both et and en. We eventually ended up using only a small fraction of available data for training (20K in each direction maximum) to speed-up the training process.

2.3 Data Processing

When exploring the Flores-101 dataset, we noticed that some translations were incorrect. We filtered the dataset with the primary intention of keeping the majority of correct translations while reducing the amount of incorrect translations. We prioritized a large dataset with a few errors over a small dataset with all high quality translations because it's already difficult getting large corpus for low-resource languages. We hypothesize that having the model learn a couple incorrect translations would be better than not having the model learn from the same amount of correct translations because of the robustness of the transformer model architecture [12]. Any translation pairs where either text in the pair were less than 3 words and the difference in size of these two texts was greater than a specific constant were filtered out (English and Estonian had minimum size of 5, while the other two pairs had minimum size of 6). We created new tasks similar to the original T5 [3] by adding new prefixes to the dataset "[source language] to [target language]: ".

Source Sentence	Target Reference
Dan and Naphtalim, Gad and Asher.	4 (et)
Delete	@action:inmenu Edit (mk)
*3	*3 is a number. You…do not need to translate this (mk)

Table 2: Examples of Incorrect translation from the Flores-101

2.4 Approach

We started our exploration with two core hypotheses in mind:

1. Embeddings trained with massive multilingual corpuses preserve semantic information that would help with transfer learning on any down-stream task.
2. Given the versatility of the T5 model, using pre-trained weights of the encoder and decoder should enhance transfer learning for an entirely new task (e.g. Translating English to Estonian)

We extracted the tokenizer and the embeddings from the pre-trained model provided by Facebook [10] and used the T5 pre-trained model from HuggingFace transformer library in {full, partial and no} reinitialization settings.

2.5 Evaluation Method

Out of the available training data, we created a random sample of 997 sentence triplets as the validation set resulting in 5982 sentence pairs and 1012 sentence pairs as the test set resulting in 6072 sentence pairs. We used BLEU [11] score for each translation direction in the test set as a measure of translation quality.

3 Model

3.1 Models Architecture

We created 4 models with the same architecture but varying weights to validate our hypothesis about transfer learning across transformer models. Each model has an $d_e = 512$, $d_{FF} = 2048$, 8-headed attention, and 6 layers each in the encoder and decoder. We modified the decoder start token to be <s> instead of using T5's decoder start token, <pad>, or M2M-100's decoder start token, </s> and <target _lang> (see Appendix A for the effect of the decoder start token on BLEU score).

1. **Random Embeddings and Random Weights** (**RE-RW**): We randomly initialize the T5 weights and embedding layer. This was our **baseline**.
2. **Random Embeddings and T5 Weights** (**RE-TW**): We randomly initialize the embedding layer but maintain the T5 wei- ghts.
3. **M2M-100 Embeddings and Random Weights** (**ME-RW**): We randomly initialize the T5 weights but maintain the M2M-100 embedding layer.
4. **M2M-100 Embeddings and T5 Weights** (**ME-TW**): We maintain the T5 weights and the M2M-100 embedding layer.

3.2 Model training

We trained each of the 4 models above (i.e. RE-RW, RE-TW, ME-RW and ME-TW) with three different training set sizes (30K, 60K and 120K sentence pairs)

in order to understand the evolution of performance with increase in training size. The training set is a balanced dataset composed of the same amount of training pair in all 6 directions. We optimized each model using Adam with a learning rate of 0.003 for 30 epochs through convergence with early stop. To train faster, we used a max length of 50 tokens for input and target sentences. In validation and test dataset, most sentences were short in size which made this an efficient choice to speed-up training on a single GPU. However, we recognize the impact on translation quality and performance as our models will not be able to translate longer sentences properly.

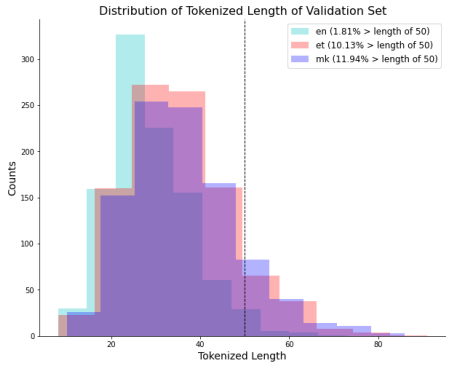

Figure 2: Distribution of token lengths across language pairs (Validation set)

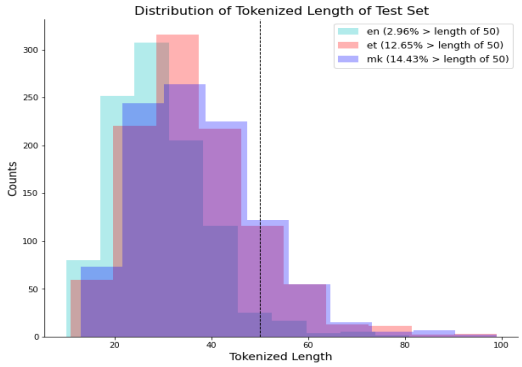

Figure 3: Distribution of token lengths across language pairs (Test set)

As evident from figure 2 and 3, in the test set, about 97% of English, 88% of the Estonian and 85% of the Macedonian sentences were unaffected by the 50 token truncation across validation and test sets. Given the high coverage, and the speed-up offered by this choice, we incorporated it in our experiments.

4 Results

4.1 Discussion of results

Our experiment has four model settings (RE-RW, RE-TW, ME-RW and ME-TW) and each setting is trained on three sizes of training datasets (i.e. 30K, 60K and 120K sentence pairs).

Figure 4: Model performance across training set sizes

Figure 4 summarises the results which indicate that:

1. Randomizing T5 weights results in poor tranlation when trained with a small training dataset. The BLEU score for RE-RW and ME-RW models on the test set remains close to 0 even as the training set size increases.
2. Maintaining pre-trained T5 weights has a large impact on test set performance. We observe significant increase in test set BLEU score even with a 30K sentence training set (i.e. 5K sentence pairs each direction). As we increase the training data size for our fine-tuning task, the test set performance continues to improve.
3. Multilingual embeddings from M2M-100 help improve the performance of our model on untrained translation. However, at the training data size of 30K sentence pairs, the test set BLEU score remains close to 0. As we increase the training data size, we see a large improvement in the test set BLEU scores and this model starts to outperform the RE- TW model. The importance of the multilingual embedding from M2M-100 can be seen as the difference between the

ME-TW and RE-TW model when the training set size is greater than 30K.

Overall, we achieved the best performance with the ME-TW model at the 120K training set size (avg. test set BLEU score across 6 directions = 3.5). This validates the hypothesis that T5 model is capable of transfer learning by utilizing the embeddings from different transformer models and this procedure is expected to produce better results (given large training data size) compared to fine tuning vanilla T5.

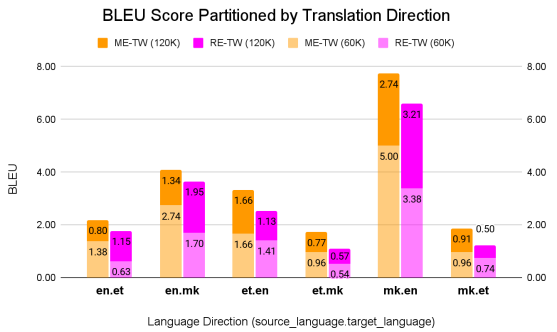

Figure 5: Comparison of BLEU gains from 60K training examples to 120K training examples

Figure 5 provides a view into the impact of training data size on the improvement in model performance across all directions for the top 2 models (ME-TW and RE-TW). We see that:

1. Across all translation directions, final BLEU score is higher for the ME-TW model with 120K sentence training size. Even at lower training size (i.e. 60K), performance of ME-TW is superior to RE-TW.
2. The RE-TW model shows larger gain in BLEU score averaged across all directions (108%) compared with ME-TW model (73%) when dataset size doubles. This observation informs us to conduct the experiment with larger training data size to understand the steady state performance difference between the two models
3. For low resource language pairs (et & mk), ME-TW performance is significantly better (53% avg.) than the RE-TW model for both 60K and 120K training data sizes. It shows that low-resource language translation can be improved via transfer learning.
4. There is a large gap between en → mk and mk → en performance. We explore the reasons in Appendix B.

4.2 Discussion of Errors.

Using a few examples from en-mk pairs, we try to understand the translation errors.

Source (mk): Постојат многу нешта кои треба да ги земете во предвид пред и кога патувате некаде.
Google translate ref: There are many things to consider before and when you travel somewhere.
Target (en): There are many things you have to take into consideration before and when you travel somewhere.

ME-TW translation:

- **10K training (BLEU 38.7)**: There are many things that you **need** to take into consideration before you **to me**.
- **20K training (BLEU 64.5)**: There are many things you **need** to take into consideration before and when you **do**.

Observations:

1. Google translate reference[1] is quite close to the target sentence, however, google translate version is more succinct.
2. Both 10K and 20K models are confusing "have to" with "need to". From the perspective of translation quality, this doesn't seem like an error.
3. Model translations are first person whereas the reference translation is in third person. This is a unique artifact of the training set.
4. Both models found it hard to translate the end of sentence (i.e. "travel somewhere"). The models are not able to capture the travel context which is presented at the very end of the source sentence ("патувате некаде").

Source (en): *He produced over 1,000 stamps for Sweden and 28 other countries.*
Google translate ref: Тој произведе над 1000 марки за Шведска и 28 други земји.
Target (mk): *Изработил преку 1000 поштенски марки за Шведска и 28 други земји.*

[1] Google Translate references are taken from translate.google.com as of July 25, 2021

ME-TW translation:

- **10K training (BLEU 34.5)**: **Toj** произлезе од 1000 возач за Sweden и 28 други земји.
- **Google translate ref**: It came from 1000 drivers for Sweden and 28 other countries.
- **20K training (BLEU 53.3)**: **Toj** произведу -ва околу 1000 бодови за Шведска и 28 други земји.
- **Google translate ref**: It produces about 1000 points for Sweden and 28 other countries.

Observations:

1. The target macedonian sentence seems to be missing the reference for subject pronoun "He". Interestingly, both models are able to pick this up and correct for it ("Toj").
2. Both models have problems translating the word "stamp" in the source sentence. It is a possible artifact of limited exposure to the word due small training size. The 20K sentence model was able to pick-up macedonian translation of the word "Sweden".
3. Finally, it is interesting to note that the 10K model was able to identify the correct tense of the verb ("came") however the verb itself was incorrect. In the case of the 20K model, the verb was correct, however the tense used was wrong. It seems that model training was happening in the right direction and more training time and data might have helped the model get the correct translation.

4.3 Limitations

It's important to note the key limitations to our experiments when thinking about the generalizability of our work:

1. **Max Cap on token length**: Looking at the sentence size in our validation and test set, we decided to cap the max token size to 50. This limits the generalizability of our exploration for longer sentences.
2. **Limited training set size**: We conducted our experiments with a relatively small amount of paired data with a total of 120K sentence pairs covering 6 directions. Even in our largest training set, the translation model was trained on only 20K sentences for each direction which is small compared to massive training sets used in recent models [1, 2]. Hence, our results don't showcase the steady state performance that can be achieved by the respective models with more training time and ample training data.
3. **Limited model exploration**: We explored the T5 model as the only target architecture for understanding the efficiency of transfer learning across different model types. The reason for selecting T5 is that T5 was trained to do translation tasks without any further training required. It is possible that our results are dependent on the choice of base model architecture. More work needs to be done in order to ensure that our insights are generalizable across different model architectures. We plan to test our hypotheses using MBart and MBart-50 which have a similar encoder-decoder architecture as T5 and are specifically designed to perform multilingual translation.
4. **Focus on translation task**: All of our experiments were focused on the single downstream task of translation. More work needs to be done in order to ensure that our insights are generalizable across different downstream tasks such as Summarization, Classification, and Question and Answering.

5 Conclusion

In this paper, we explored the potential of transfer learning between transformer models (M2M-100 and T5) and its effectiveness in expanding the translation capabilities of the existing models (i.e. T5) for previously untrained low-resource languages. We demonstrated that pre-trained encoder-decoder weights from T5 are able to generalize over untrained low-resource languages and to improve performance compared to random initialization of weights even with limited training data. Augmenting the training with pre-learnt multilingual embeddings from M2M -100 resulted in an additional boost in performance at larger training set sizes. Multilingual embeddings significantly improved performance (+53% avg. BLEU score) on the low resource language pair (et-mk). With as little as 20K sentence pairs, the model is

able to extract nuances of the languages. We release all code and pre-trained datasets to facilitate future work and plan to test the generalizability of our hypothesis using MBart and MBart-50.

References

[1] Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, Naman Goyal, Tom Birch, Vitaliy Liptchinsky, Sergey Edunov, Edouard Grave, Michael Auli and Armand Joulin. Beyond English centric Multilingual Machine translation. arXiv preprint arXiv:2010.11125, 2020.

[2] Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant and Colin Raffel. mT5: A Massively Multilingual Pre-trained Text-to-text Transformer. arXiv preprint arXiv:2010.11934, 2021.

[3] 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. Journal of Machine Learning Research, 21(140):1–67, 2020.

[4] David Crystal. Two thousand million? English today, 24(1):3–6, 2008.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics, 2019.

[6] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate.CoRR, abs/1409.0473, 2014.

[7] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Advances in Neural Information Processing Systems, volume 30, pages 5998–6008, 2017.

[8] Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Dmitry Lepikhin, Melvin Johnson, Maxim Krikun, Mia Xu Chen, Yuan Cao, George Foster, Colin Cherry, et al. Massively multilingual neural machine translation in the wild: Findings and challenges. arXiv preprint arXiv:1907.05019, 2019.

[9] Jiatao Gu, Yong Wang, Kyunghyun Cho, and Victor OK Li. Improved zero-shot neural machine translation via ignoring spurious correlations. arXiv preprint arXiv:1906.01181, 2019.

[10] EMNLP 2021: Sixth Conference on Machine Translation (WMT21)

[11] 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, 2002.

[12] Yu-Lun Hsieh, Minhao Cheng, Da-Cheng Juan, Wei Wei, Wen-Lian Hsu and Cho-Jui Hsieh. On the Robustness of Self-Attentive Models. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics:1520–1529, 2019.

Appendix

A. Effect of decoder start token

We started our exploration by setting the decoder start token as </s> replicating M2M-100's architecture choice. We noticed that our model had difficulty converging, which became more apparent when trained with larger datasets. Model converged properly after changing the decoder start token to either <pad> or <s>. Setting the decoder start token to <s> is more appropriate for our dataset since the validation and test dataset contained whole sentence translations rather than single word translations. Figure 6 indicates that there is a slight increase in BLEU score when using <s> instead of <pad> as the decoder start token, however this can be potentially attributed to the randomness of initialization and training.

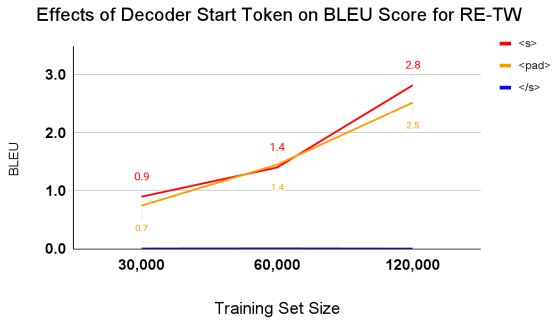

Figure 6: Comparison of decoder start token on BLEU score for RE-TW model

B. Differential performance between paired directions (EN <> MK)

Figure 5 suggests a large gap in BLEU score between English to Macedonian translation (BLEU=4.08) and Macedonian to English translation (BLEU=7.74). In order to understand this gap deeper, we analysed three hypotheses:

- **Difference in translation quality:** Looking at the section 4.2 example 2 (translation from English to Macedonian), we see that the Google translation reference is superior as it captures the pronoun "Toj" alongside many other tokens that are shared by our model outputs, however, missing in the target translation. We used google translate results as the target sentences (test set) and tested our model outputs against that. As figure 7 shows below, althou- gh we see a huge increase in the Test BLEU scores, it doesn't answer our question of differential perform- ance as there still appears to be a significant delta.

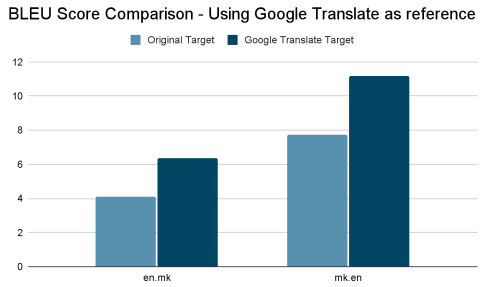

Figure 7: BLEU Score difference on using Google Translated for Macedonian Target Sentences

- **Difference in coverage between train & test**: As our training and test datasets are very small in size (20K and 1K sentences respectively for each direct- ion), there is a chance that a big portion of tokens in test sets are not present in training. This will make the model unable to translate well on the test set. As we see in figure 8, we see a substantial difference in the tokens (unigram and bigram) that are exposed only in the test set but not in training for English and Estonian compared to Macedonian. We think that this differential in bigram coverage is a contributing factor in why the performance in one direction (en→mk) is substantially different from the other (mk→en) direction even though the directions have been trained using the same examples.

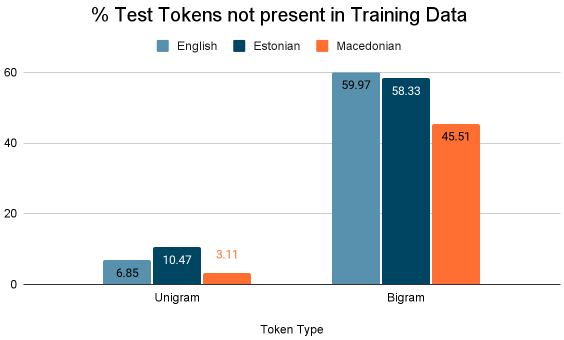

Figure 8: Difference in unigram and bigram coverage across languages in Training and test sets

- **Default effectiveness of pre-trained embeddings**: As we extracted and used the M2M-100 embedding weights in our model, we expect our model to roughly mirror the performance of the pretrained M2M-100 model on the test set which should be more noticeable when trained with a smaller dataset. Looking at Figure 9, we can see that mk→en has the best performance and a large performance difference compared to the other directions in both our model and the M2M-100 model. Our models' learning depends on the effectiveness of the pretrained embedding weights and the coverage between training and test sets. Macedonian has good coverage, while English and Estonian do not as seen in Figure 7. Seen in Table 3, the BLEU for en→et and mk→en is low compared to the other directions. This poor performance is reflected in our model's performance for these two directions regardless of the coverage difference of MK and EN. Looking at et→mk, the M2M-100's BLEU

score is decent, but since ET had poor coverage, our model achieved a poor BLEU score. For en→mk and et→en, M2M-100 had decent BLEU score with et→en slightly outperforming en→mk. However, for our model, en→mk slightly outperformed et→en, because ET had a worse coverage, not seeing more unigrams and bigrams, than EN.

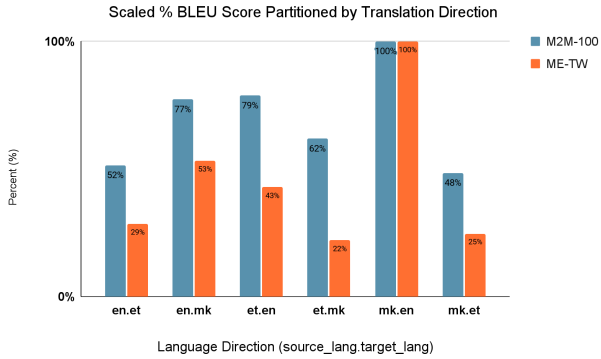

Figure 9: BLEU score of each direction (scaled by dividing each direction's BLEU by the max BLEU of the same mode).

Direction	M2M-100 BLEU
EN → ET	13.4
EN → MK	20.1
ET → EN	20.5
ET → MK	16.1
MK → EN	26.0
MK → EN	12.6

Table 3: BLEU score of M2M-100 on the test dataset