Improving Zero-Shot Performance in Pretrained Translation Models through Tied Representation Learning

Hamish Scott **Ian Wu** **Dhruv Kaul** **Daniel Shani**
University College London
{hamish.scott,ian.wu,dhruv.kaul,daniel.shani}.20@ucl.ac.uk

Abstract

Unsupervised pretraining has been shown to improve the performance of multilingual Neural Machine Translation (MNMT) in the supervised setting, but how it impacts performance in the traditionally-challenging zero-shot setting remains an open question. In this paper, we investigate the performance of the well-established pretrained model mBART50 in the zero-shot setting, and improve finetuning by introducing language invariance through *tied representation learning*, based on the work of Arivazhagan et al. (2019a). Our experiments in the low-resource regime show that vanilla mBART50 performs worse than pivoting in the zero-shot setting. Tied representations are able to improve performance by nearly 10 BLEU, surpassing the performance of pivoting while having a negligible impact on the supervised performance. By altering the tying strength, we are able improve this by a further 5 BLEU, surpassing the performance of supervised finetuning. Finally, we extend tied representation learning by applying it to selected encoder layers only, but find that this reduces both zero-shot and supervised performance.

1 Introduction

The recent success of bilingual Neural Machine Translation (NMT) (Bahdanau et al., 2016) (Sutskever et al., 2014) has driven significant interest in multilingual NMT (MNMT), where a single, neural model is trained to translate between many different language pairs (Firat et al., 2016). An important extension of multilingual translation is zero-shot translation, where models translate between language pairs that were not seen during training (Johnson et al., 2017). Zero-shot translation capabilities are desirable for at least two reasons. Firstly, they enable translation between languages where bitext is either low resource (where supervised translation tends to do poorly) or entirely unavailable, as is the case with rare languages or dead languages. Secondly, they reduce the computational burden required to build multilingual translation systems by allowing a proportion of translation directions to be inferred rather than trained.

Because neural models learn distributed representations of language (Lakew et al., 2018a), they should, in theory, facilitate generalisation to unseen language pairs, and we may therefore expect MNMT models to naturally possess some zero-shot capability. Unfortunately, current models perform significantly worse in the zero-shot than in the supervised setting (Johnson et al., 2017). One explanation is that vanilla MNMT models fail to learn *language invariant representations*, and thus an input sentence cannot be accurately mapped to the correct output sentence unless the particular language pair was seen during training (Arivazhagan et al., 2019a; Ben-David et al., 2007; Gu et al., 2019; Al-Shedivat and Parikh, 2019). The simplest solution to this problem involves building pivot models (Firat et al., 2016; Johnson et al., 2017), where the model performs two translation operations, using a common language (usually English) as a bridge. This method, however, suffers from compounding errors due to the double translation, and furthermore is inefficient and slow. This motivates the search for a more elegant multilingual translation method that is more suited to zero-shot translation.

One recent breakthrough in the field of machine translation has been the introduction of self-supervised pretrained models (Raffel et al., 2019; Edunov et al., 2019). These are typically generative models that are trained in an unsupervised manner, which can be finetuned for particular translation tasks (Lample and Conneau, 2019). The pretraining corpora are usually monolingual, which is often in abundance, even for rare or dead languages. One particular example of a pretrained translation model is mBART50 (Tang et al., 2020), which is a well-established transformer-based model that has been pretrained on monolingual corpora from 50 different languages. Finetuning mBART50 for both bilingual and supervised multilingual translation tasks has been shown to improve results by up to 20 BLEU in comparison with NMT models trained from scratch, achieving state-of-the-art results on many language pairs. Despite this, there appears to have been little work done to investigate the zero-shot capabilities of such models. With pretrained models fast becoming an integral part of the machine translation toolkit, due not only to their superior performance but also to their accessibility, we believe that important work must be done to assess and improve their zero-shot capabilities. Fortunately, there is good reason to believe that pretraining may improve zero-shot performance. Many authors attribute the success of pretrained MNMT to cross-lingual transfer, where learning underlying patterns in different languages helps when translating between them (Liu et al., 2020; Lample and Conneau, 2019; Gu et al., 2019). We may reasonably expect this advantage to apply to the zero-shot setting as well.

In this paper, we investigate the zero-shot capabilities of finetuned mBART50 models, and propose a method to improve them. Building on the work of Arivazhagan et al. (2019a), we introduce an auxiliary loss function during the finetuning process that penalises distance between encoded representations of the same sentence in different languages, resulting in what we call *tied representations*. Our finetuned models are therefore trained to map sentences in different languages with the same meaning to the same *language invariant* representation, in addition to being trained to translate between bitexts. We also explore the effects of altering the tying strength, as well as that of selectively applying tied representations to only parts of the encoder, in an attempt to create both language invariant and language aware layers. Finally, we focus our work on low resource language pairs, where zero-shot translation is typically the most relevant (Fan et al., 2020; Lakew et al., 2018b). We make the following contributions:

1. We discover that, much like vanilla MNMT models, finetuned multilingual mBART50 models also demonstrate poor zero-shot performance in low resource regimes.

2. We introduce tied representation learning for mBART50, which improves zero-shot translation performance in low resource regimes by up to 15 BLEU, while having negligible impact on performance in the supervised directions. Not only do these results significantly improve upon those achieved by the corresponding pivot models, they in fact *surpass those achieved through supervised training*.

3. We find that applying tied representation learning to all encoder layers yields the best zero-shot performance, with selective application reducing zero-shot performance and negatively impacting supervised performance.

2 Related work

MNMT and Zero-Shot Translation MNMT was pioneered by Dong et al. (2015), where a shared GRU encoder with language specific decoders is used for one-to-many translation. Firat et al. (2016) propose an attention-based model that requires training both language specific encoders and decoders. In contrast, Johnson et al. (2017) and Ha et al. (2016) develop models featuring universal encoders and decoders, with improvements to both scalability and performance. Johnson et al. (2017) also consider zero-shot translation, and demonstrate that vanilla MNMT models perform poorly, with their zero-shot Spanish-to-Portuguese model achieving 6 BLEU fewer than their pivot model.

Some work has been done on identifying the factors that contribute to MNMT's poor zero-shot performance. Both Arivazhagan et al. (2019a) and Gu et al. (2019) find that vanilla MNMT systems often translate to the wrong target language during zero-shot inference. The former attribute this to representation alignment failure, while the latter attribute this to the model learning "spurious correlations" between target languages. In both cases, it appears the key problem arises from the inability of MNMT models to disentangle semantic meaning from language, which results in them learning language-specific representations. Recent attempts to improve zero-shot performance in MNMT include Zhang et al. (2020), which proposes using deeper network architectures and language-specific network components, and Fan et al. (2020), which features translation models trained on a 100-language, non-English-centric data-mined dataset.

Language Agnostic Representations The idea of coordinating representations has been thoroughly explored in multitask learning (Ben-David et al., 2007; Wang et al., 2016), and has been applied to improve machine translation. Gu et al. (2018) develop a model consisting of shared encoders controlled by a mixture of experts to improve supervised translation performance, with both components designed to enforce alignment of representations across languages. Al-Shedivat and Parikh (2019) enforce alignment by training their model to not only translate between pairs of sentences but also to agree on translations into a third language. Arivazhagan et al. (2019a) enforce alignment by introducing auxiliary losses that penalise differences in encoder representations of semantically identical sentences. They demonstrate that multilingual models trained in this manner are able to achieve zero-shot performance on-par with pivoting. Our work differs from theirs in three main respects. Firstly, they train translation models from scratch, whereas we finetune ours from unsupervised pretrained models. Secondly, they focus their work on high-resource language pairs (En-De 4.5M, En-Fr 39M) whereas we focus on low resource pairs (En-Az 5K, En-Tr 200K, En-Kk 3K), where zero-shot translation is both more relevant and more challenging (Lakew et al., 2018b). Finally, we extend the methodology by exploring selective application of the loss to only parts of the encoder.

Pretrained Language Models for Translation Language model pretraining has seen widespread adoption across NLP in recent years (Devlin et al., 2018; Le et al., 2019; Raffel et al., 2019; Lewis et al., 2019). In the machine translation field, many earlier works focused on pretraining only parts of the model. Lample and Conneau (2019), for example, pretrain only the encoder, while Edunov et al. (2019) pretrain only the decoder. Artetxe et al. (2017) employ unsupervised pretraining to the entire transformer model, but focus only on English, French and German. Building on this, Conneau et al. (2019) pretrain the general cross-lingual model XLM-R for use across a variety of NLP tasks, including translation, while Liu et al. (2020) develop mBART25, a denoising pretrained model designed specifically for multilingual translation. Tang et al. (2020) extend mBART25 to mBART50 by pretraining on 50 languages rather than 25. Both mBART50 and mBART25 significantly improve upon multilingual models trained from scratch, although both papers focus only on supervised rather than on zero-shot translation. One study of zero-shot translation in pretrained models is Ji et al. (2019), who pretrain their own transformer models and propose a novel method to improve its zero-shot capabilities. We focus instead on studying performance in accessible, powerful and well-established models that are already widely used.

3 Methodology

In this section we give an overview of multilingual finetuning and mBART50 and then introduce the method of tied representation learning

3.1 Multilingual Finetuning

The standard supervised training objective for MNMT is to maximise the log-likelihood $\mathcal{L}$ of a model parameterised by θ, given a dataset $\mathcal{D}$ of parallel sentences in a set of languages (Dabre et al., 2020):

$$\mathcal{L}(\theta) = \sum_{(x,y)\in\mathcal{D}} \log p(y|x;\theta).$$

Here, y is a target sentence and x is an input sentence. In this work we consider finetuning a model where θ is initialised by pretraining on a separate task and then updated in an online fashion by maximising

$$E_{(x,y)\sim P}\left[p(y|x;\theta)\right]$$

via online gradient descent for a given number of steps. Here P is a distribution over the data (for example uniform over a given dataset). We use temperature sampling (Arivazhagan et al., 2019b) over the size of the bitext dataset for each language pair. That is, the probability of sampling the language pair l_1, l_2 is

$$\frac{|\mathcal{D}_{l_1,l_2}|^T}{\sum_{i\neq j}|\mathcal{D}_{l_i,l_j}|^T}$$

where $\mathcal{D}_{l_i,l_j}$ is the dataset of parallel sentences in languages l_i and l_j and $T > 0$ is a temperature parameter.

3.2 BART and mBART50

In all of our experiments we use mBART50 as our pretrained model. mBART is a multilingual denoising autoencoder trained on monolingual corpora in 50 languages using the BART pretraining scheme (Lewis et al., 2019). It is trained to reconstruct texts that have been corrupted by masking phrases and permuting sentences. That is given K datasets, $\mathcal{D} = \{\mathcal{D}_1, \ldots, \mathcal{D}_K\}$, where each dataset $\mathcal{D}_i$ contains monolingual documents in language i, mBART aims to to maximise the log-likelihood

$$\mathcal{L}(\theta) = \sum_{\mathcal{D}_i \in \mathcal{D}} \sum_{x \in \mathcal{D}_i} \log p(x|g(x); \theta)$$

where g is a noising function and θ represents the model parameters.

The architecture of mBART is a single Transformer (Vaswani et al., 2017) with 12 encoder layers, 12 decoder layers, an inner model dimension of 1024 and 16 attention heads (corresponding to approximately 680 million parameters). Compared to (Vaswani et al., 2017) mBART has an additional layer normalisation layer on top of the encoder and decoder. For full details of the mBART model and BART pretraining scheme we refer the reader to Tang et al. (2020), Liu et al. (2020) and Lewis et al. (2019).

3.3 Tied Representation Learning

As is common in NMT, the models we consider decompose into an encoder $f_\phi(\cdot)$ and a decoder $g_\psi(\cdot)$, where the prediction for the next token in a partially translated sentence $y_{1:n}$ given input x is given by

$$p(\hat{y}_{n+1}|y_{1:n}, x, \theta) = g_\phi(f_\psi(x), y_{1:n})$$

with learnable parameters $\theta = \{\phi, \psi\}$.

As in Arivazhagan et al. (2019a) we consider encoding both x and y and then computing a differential similarity measure, $\kappa(\cdot, \cdot)$, between their encoded representations $f_\phi(x)$ and $f_\phi(y)$. During training we update the parameters of our model according to

$$E_{(x,y)\sim P}\left[\nabla_\theta\left[-\log p(y|x;\theta) + \lambda\kappa(f_\phi(x), f_\phi(y))\right]\right].$$

Here $\lambda > 0$ is a hyperparameter that controls the relative strength of the supervised loss and auxiliary (representation similarity) loss, and which we study in more detail in Section 4.3. We refer to this as the *tying strength* throughout the paper. We provide an illustration of our model in Figure 1.

There are many potential choices for the form of κ, which are partially explored in Arivazhagan et al. (2019a). They consider an adversarial auxiliary loss where κ is a neural network with trainable parameters, and compare this with a simple functional form for κ. They find that there is no benefit gained from using an adversarial objective and that it made training less stable and more sensitive to the choice of hyperparameters. Therefore in our work we consider a purely functional form for κ. Given the encoded representations, $\mathbf{x}^{enc}_{1:m}$ and $\mathbf{y}^{enc}_{1:n}$ of sequences $x_{1:m}$ and $y_{1:n}$ we compute the cosine distance between $\mathbf{x}^{enc}_{1:m}$ and $\mathbf{y}^{enc}_{1:n}$ after max-pooling over the time dimension. That is

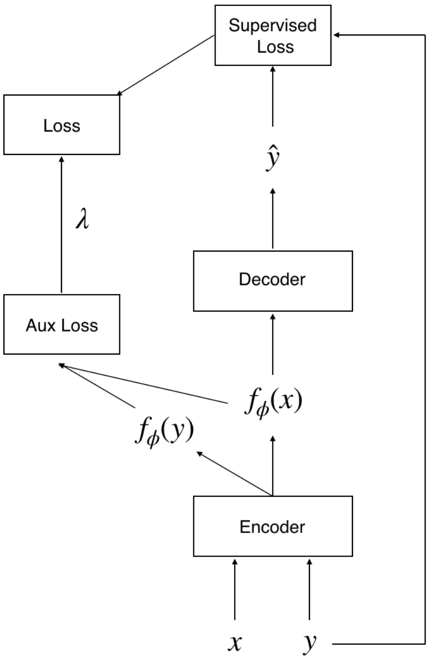

Figure 1: Illustration of tied representation learning. x and y represent an input/target sentence pair and are mapped to encoder representations $f_\phi(x)$ and $f_\phi(y)$ respectively. The auxiliary loss is calculated from these representations. The input representation $f_\phi(x)$ is then transformed by the decoder, yielding prediction $\hat{y}$, from which the supervised loss if computed. The combined loss is computed as the sum of the supervised loss and the auxiliary loss, with hyperparameter λ controlling their relative strengths

$$\kappa(\mathbf{x}^{enc}_{1:m}, \mathbf{y}^{enc}_{1:n}) = 1 - \cos\left(\max_{1\leq i\leq m} \mathbf{x}^{enc}_i, \max_{i\leq j\leq n} \mathbf{y}^{enc}_j\right).$$

The auxiliary loss will therefore increase as the learned representations of x and y differ. As a result, our models learns to *tie* these representations, and because x and y differ only in language and not in semantics, these representations can be said to demonstrate *language invariance*.

4 Experiments

4.1 Experimental Setup

Dataset We conduct all of our experiments on the TED multilingual dataset (Qi et al., 2018)

- a 60 language dataset of parallel translations of TED talk transcripts. The full training split contains approximately 250K parallel sentences, although not all languages are available for all sentences. We focus most of our work on translating between English (En), Turkish (Tr) and Azerbaijani (Az), and treat the Az ↔ Tr directions as the zero-shot directions when required. En ↔ Tr is a low/medium resource pair with about 200K training examples, while En ↔ Az and Tr ↔ Az are low resource pairs, with fewer than 5K training examples. However, because Az and Tr are both Turkic languages and are linguistically quite similar, we expect cross-lingual transfer to be reasonably effective. The exact sizes of training, validation and test splits for the language pairs used in our experiments are given in Appendix A.

Preprocessing We adopt the same preprocessing procedure as (Liu et al., 2020) and use their publicly available tokenizer via the HuggingFace API (Wolf et al., 2020). This uses a sentence-piece model (Kudo and Richardson, 2018) with a shared vocabulary across languages of around 250K subword tokens. Source and target sequences are prepended by a language token indicating the language of each sequence.

Finetuning We finetune both multilingual and bilingual models. We initialise all models with the pretrained weights (also via the HuggingFace API) and train for 40K steps with teacher forcing using 0.3 dropout and a batch size of 5. We use the Adam optimizer (Kingma and Ba, 2017) with 2500 warm-up steps and a maximum learning rate of 3e-5. For bilingual translation we train on the relevant bitext data for the source and target languages while for multilingual translation we sample a source and target language at each step from the chosen subset of languages (Section 3.1). We sample languages with a temperature parameter of 0.7.

Decoding At test time we initialise the decoder with the language token for the desired target language and use beam-search decoding with beam size 5 and length penalty 1.0. All results are reported in BLEU scores (Papineni et al., 2002).

4.2 Zero-shot Translation with Tied Representation Learning

This section explores the zero-shot translation performance of finetuned MNMT models with and without tied representation learning. We finetune bilingual (BL), vanilla multilingual (ML) and tied representation multilingual (TRL) models on supervised (BL, ML, TRL) and zero-shot (ML, TRL) translation tasks.

4.2.1 Models

Bilingual Models (BL) We finetune separate mBART50 models on each translation direction, generating six bilingual models that serve as baselines. We also use the bilingual models to build our pivot baseline. For Tr → Az pivoting, we first translate the source sentence into English using the Tr → En bilingual model, and then translate the intermediate English sentence into the target sentence using the En → Az bilingual model, and likewise for the reverse direction.

Vanilla Multilingual Models (ML) We finetune an mBART50 model on Tr ↔ En and Az ↔ En sentence pairs, and infer the Tr ↔ Az translations during test time using zero-shot translation.

Tied Representation Multilingual Model (TRL) We finetune an mBART50 model on Tr ↔ En and Az ↔ En sentence pairs, and infer the Tr ↔ Az translations during test time using zero-shot translation. We apply tied representation learning (Section 3.3) to every layer in the encoder during finetuning, and set the tying strength to 1.0. We denote this specific model by TRL-1.0.

Note that we do not compare our results to NMT models trained from scratch. This comparison is made in Tang et al. (2020), and given that we follow very similar experimental protocol to them, we refer readers to their work.

4.2.2 Results

As shown in Table 1, the zero-shot performance of vanilla multilingual models is poor, nearly 10 BLEU below the supervised bilingual baselines and 5 BLEU below the pivot model, thus demonstrating that finetuning on its own does not give rise to zero-shot capability in MNMT models. The use of tied representation learning improves zero-shot performance by up to nearly 10 BLEU, surpassing pivot model performance by over 3 BLEU, although it

Direction	BL	ML	TRL-1.0	Pivot
En → Tr	21.9	24.6	23.6	-
Tr → En	32.0	32.0	32.5	-
En → Az	12.6	14.1	13.6	-
Az → En	19.3	24.6	23.3	-
Tr → Az	16.8	**5.6**	**15.0**	11.5
Az → Tr	15.1	**6.2**	**12.1**	10.6

Table 1: Bilingual (BL), Vanilla Multilingual (ML), Tied Representation Multilingual (TRL-1.0) and bilingual pivot translation test results. Zero-shot results are shown in **bold**. Tied representation learning with tying strength 1.0 improves zero-shot performance by up to nearly 10 BLEU, surpassing pivot model performance by over 3 BLEU.

nonetheless achieves a worse result than the supervised bilingual baseline by 2-3 BLEU. It is also noticeable that utilising tied representation learning in this way has negligible impact on translations in the supervised directions, thus demonstrating that our approach can be adopted with little cost. Finally, we note that the multilingual vanilla model generally outperforms the bilingual baseline in the supervised directions. This was also observed in (Tang et al., 2020) and (Liu et al., 2020), both of whom attribute this to multilingual pretraining, which they claim encourages cross-lingual transfer.

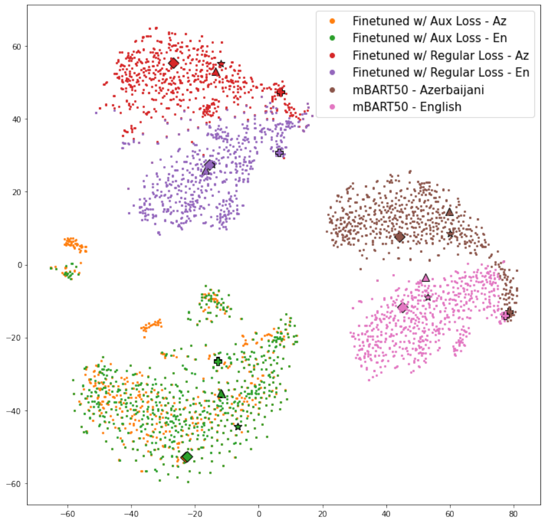

Figure 2: t-SNE plots of encodings of English-Azerbaijani sentences for 3 different models: base mBART50 and mBART50 finetuned with/without auxiliary loss. Each point corresponds to the representation of a sentence in either Az or En. We highlight four example sentence pairs using different symbols. Notice that equivalent sentences are mapped to the same locations for the tied representation model, while for the other models they are far apart.

In Figure 2, we demonstrate the effect of the tied representation loss by visualising the encoded representations of pairs of sentences in English and Azerbaijani. We extract encoder representations for different sentences, apply max-pooling over the time dimension and then use t-SNE (van der Maaten and Hinton, 2008) to perform dimensionality reduction, producing two-dimensional plots of the embeddings. For the base mBART50 and vanilla finetuned models, the representations of the two languages can be easily separated. For the tied representation model, the low-dimensional projections of the representations for the same sentences in different languages fall on top of each other. This demonstrates that our auxiliary loss function leads to more language-invariant representations of sentences, resulting in semantic alignment.

Finally, we note that the auxiliary loss converges to a small number during training, which further suggests that our model has learned tied representations. See Appendix B for more details.

4.3 Tying Strength

This section studies the effect of increasing or decreasing the tying strength λ (Section 3.3). We finetune four additional tied representation multilingual models, with tying strengths of 0.1, 0.5, 5.0 and 10.0. We train these models in the exact same manner as we trained the TRL-1.0 model from Section 4.2.1.

4.3.1 Results

We provide the results of this experiment in Table 2. Firstly, increasing tying strength up to 5.0 improves zero-shot performance, with the TRL-5.0 model achieving nearly 20 BLEU on zero-shot Tr → Az, *surpassing* the performance of the supervised bilingual model by nearly 3 BLEU. It is also clearly possible to over-tie, with TRL-10.0 yielding worse zero-shot results than TRL-5.0. Secondly, decreasing tying strength tends to hurt zero-shot performance, with TRL-0.1 yielding comparable results to the vanilla model. Lastly, we note that over- or under-tying reduces performance in the supervised directions. This is expected in the strong-tying setting, where the model focuses too much on language invariance and neglects translation accuracy, but is somewhat surprising in the weak-tying setting, where we may have expected regression to the vanilla model. This suggests that an absence of language invariance may be preferable to the partial presence of it, and that the benefits to translation

Direction	TRL-0.1	TRL-0.5	TRL-1.0	TRL-5.0	TRL-10.0
En → Tr	20.3	23.0	23.6	23.0	20.1
Tr → En	25.6	29.1	32.5	31.2	27.0
En → Az	11.6	14.6	13.6	14.5	7.8
Az → En	18.9	21.6	23.3	24.5	20.1
Tr → Az	**4.3**	**10.2**	**15.0**	**19.7**	**16.6**
Az → Tr	**5.5**	**6.5**	**12.1**	**13.2**	**8.2**

Table 2: Supervised and zero-shot translation results for different tying strengths. Zero-shot results are shown in **bold**. Stronger tying tends to improve zero-shot results, although over-tying is possible. Tying that is either too weak or too strong may also affect translation performance in the supervised directions.

only manifest when this invariance is properly enforced.

4.4 Selective Application of Tied Representation Learning

In this section, we freeze the parameters of certain layers of the encoder during backpropagation and optimisation with respect to the auxiliary loss, and only train those layers to minimise the supervised translation loss. This should, ideally, separate encoder layers into language invariant and language aware layers, where the former are trained to minimise both the auxiliary and the supervised losses, and the latter are trained to minimise only the supervised loss. We organise the layers such that language invariant layers are always located deeper inside the encoder, in order to ensure that the final encoder representation remains language invariant.

The key idea behind the selective application of tied representation learning is to allow earlier encoder layers to maintain greater flexibility in their learned representations. This occurs because we have removed the constraint that these representations be language invariant, and instead they can be optimised solely to improve translation quality. Because later layers remain language invariant, our model still nonetheless learns a language invariant representation following the encoding phase, but we hope that this representation may be better suited to translation as a result of the earlier layers maintaining their language-aware state.

We explore applying tied representation learning to only the final 2, 4, 6, 8 and 10 layers of the encoder. We use a tying strength of 1.0 for all five experiments, and, asides from freezing the appropriate layers during backpropagation and optimisation, we train these models in the exact same manner as we trained the TRL-1.0 model from Section 4.2.1. We label our models with the convention n-m, where n represent the number of non-tied encoder layers, and m the number of tied encoder layers.

4.4.1 Results

As shown in Table 3, the performance of the 2-10 and 4-8 layer splits is marginally worse compared to the completely-tied model, while the remaining layer combinations did significantly worse across all language pairs. The most surprising result is the 10-2 model, which appears to be entirely unable to generalise in the zero-shot setting, performing even worse than the vanilla multilingual model.

We conjecture that models with half or more of the encoder layers un-tied perform unsuccessfully because too few parameters receive a language invariant signal. This, in turn, means that the representations of encoded language pairs are allowed to diverge as inputs propagate. In other words, placing the auxiliary loss on an insufficient subset of the encoder network creates too severe a bottleneck for the language-aware representations to then become invariant. This results in the encoder learning weaker representations, making multilingual translation more difficult. Overall, we conclude that selective tying at the encoder layer level of abstraction is not superior to using complete tying across the entire encoder network. However, we believe that future work may explore this idea further and could consider tying specific lower-level components of the encoder network, such as the attention mechanism or individual attention heads.

4.5 Tied Representation Learning for Extreme-Low Resource Translation

Our model has been developed thus far with zero-shot translation in mind. However, in many real-world settings, we may have access to very small bitext datasets. Translating these datasets is a challenge similar to zero-shot translation. In this section, we investigate whether the techniques we have

Direction	ALL	2-10	4-8	6-6	8-4	10-2
En → Tr	23.6	23.5	14.5	20.2	20.8	21.4
Tr → En	32.5	30.5	20	24.9	21.9	22.7
En → Az	13.6	13.6	9.3	5.2	10.7	0.001
Az → En	24.3	21.3	16.4	15.7	17.8	15.7
Tr → Az	**15**	**9.7**	**10.6**	**7.5**	**6.5**	**0.001**
Az → Tr	**12.1**	**8.0**	**9.1**	**2.0**	**5.7**	**0.001**

Table 3: Supervised and zero-shot translation results for selective representation tying. Zero-shot results are shown in **bold**. The ALL column denotes the standard setting where all encoder layers are tied. 2-10 denotes the model where tying is applied to only the last 10 layers, and likewise for the other columns. Selective application of tied representation learning appears to reduce performance in both the supervised and zero-shot settings

Direction	ML	TRL-1.0
En → Az	10.7	11.6
Az → En	8.6	9.2
En → Kk	5.4	4.9
Kk → En	12.4	12.6
Az → Kk	0.1	8.1
Kk → Az	8.7	9.8

Table 4: Vanilla Multilingual (ML) and Tied-Representation (TRL-1.0) models in an extreme-low resource supervised setting. Note the improvements in the Az ↔ Kk directions.

developed in this paper are applicable to such a scenario.

We evaluate the performance of tied representation learning in extremely-low ($<$ 10k) resource settings. We consider the extreme-low resource triplet English-Azerbaijani-Kazakh (Kk). In addition to training on the En ↔ Az (5K) and En ↔ Kk (3K) directions, we also train on the Kk ↔ Az directions, where we only have access to 440 parallel sentences. We train our models for 15K rather than 40K steps, which we believe is sufficient given the small sizes of all our parallel bitexts.

4.5.1 Results on Extreme-Low Resource Translation

We train models with (TRL-1.0) and without (ML) tied representation learning, and display our results in Table 4. In this setting, the vanilla multilingual model performs poorly in the Kk ↔ Az directions due to the very limited number of training pairs present. Tied representation learning shows impressive gains in BLEU score, particularly in the Az → Kk direction. This highlights the increased performance of tied representation learning in an important real-world setting, which exemplifies the contributions of our methodology.

5 Conclusion

This paper investigates the performance of the pretrained language model mBART50 in the zero-shot setting, and describes a method to improve it. We show that vanilla finetuning yields poor zero-shot performance, but demonstrate that enforcing language invariance through tied representations leads to significant improvements, with negligible impact on supervised performance. We are able to surpass both pivoting and supervised pretraining in the low-resource regime, demonstrating that *zero-shot translation can be an alternative to supervised finetuning*. We also investigate layer-selective use of tied representations, and show that tying all layers in the encoder leads to the best results. Overall, we hope that our work may lead to improvements in real life translation systems, and inspire further research into zero-shot translation in pretrained models.

Our findings present many interesting avenues for future work. Firstly, we can investigate whether our results generalise to other pretrained language models, and by doing so gain greater insight into both zero-shot translation and unsupervised pretraining. Secondly, we can attempt to build large-scale many-to-many translation systems using our method, which may highlight the practical benefits and pitfalls associated with it. Finally, we may be able to improve upon tied representation learning by *directly* applying a separate auxiliary loss to each layer, rather than simply freezing the parameters of certain layers during backpropagation and optimisation (Section 4.4). This may correct for the bottleneck problem we described earlier by allowing us to directly control the level of language invariance at every layer.

References

Maruan Al-Shedivat and Ankur P. Parikh. 2019. Consistency by agreement in zero-shot neural machine translation. *CoRR*, abs/1904.02338.

Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Roee Aharoni, Melvin Johnson, and Wolfgang Macherey. 2019a. The missing ingredient in zero-shot neural machine translation.

Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Dmitry Lepikhin, Melvin Johnson, Maxim Krikun, Mia Xu Chen, Yuan Cao, George Foster, Colin Cherry, Wolfgang Macherey, Zhifeng Chen, and Yonghui Wu. 2019b. Massively multilingual neural machine translation in the wild: Findings and challenges.

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2017. Unsupervised neural machine translation. *CoRR*, abs/1710.11041.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2016. Neural machine translation by jointly learning to align and translate.

Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. 2007. Analysis of representations for domain adaptation. In *Advances in Neural Information Processing Systems*, volume 19. MIT Press.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116.

Raj Dabre, Chenhui Chu, and Anoop Kunchukuttan. 2020. A comprehensive survey of multilingual neural machine translation.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Daxiang Dong, Hua Wu, Wei He, Dianhai Yu, and Haifeng Wang. 2015. Multi-task learning for multiple language translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1723–1732, Beijing, China. Association for Computational Linguistics.

Sergey Edunov, Alexei Baevski, and Michael Auli. 2019. Pre-trained language model representations for language generation. 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 4052–4059, Minneapolis, Minnesota. Association for Computational Linguistics.

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. 2020. Beyond english-centric multilingual machine translation. *CoRR*, abs/2010.11125.

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016. Multi-way, multilingual neural machine translation with a shared attention mechanism.

Jiatao Gu, Hany Hassan, Jacob Devlin, and Victor O.K. Li. 2018. Universal neural machine translation for extremely low resource languages. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 344–354, New Orleans, Louisiana. Association for Computational Linguistics.

Jiatao Gu, Yong Wang, Kyunghyun Cho, and Victor O.K. Li. 2019. Improved zero-shot neural machine translation via ignoring spurious correlations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1258–1268, Florence, Italy. Association for Computational Linguistics.

Thanh-Le Ha, Jan Niehues, and Alexander H. Waibel. 2016. Toward multilingual neural machine translation with universal encoder and decoder. *CoRR*, abs/1611.04798.

Baijun Ji, Zhirui Zhang, Xiangyu Duan, Min Zhang, Boxing Chen, and Weihua Luo. 2019. Cross-lingual pre-training based transfer for zero-shot neural machine translation. *CoRR*, abs/1912.01214.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation.

Diederik P. Kingma and Jimmy Ba. 2017. Adam: A method for stochastic optimization.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing.

Surafel M. Lakew, Mauro Cettolo, and Marcello Federico. 2018a. A comparison of transformer and recurrent neural networks on multilingual neural machine translation.

Surafel Melaku Lakew, Quintino F. Lotito, Matteo Negri, Marco Turchi, and Marcello Federico. 2018b. Improving zero-shot translation of low-resource languages. *CoRR*, abs/1811.01389.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining.

Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoît Crabbé, Laurent Besacier, and Didier Schwab. 2019. Flaubert: Unsupervised language model pre-training for french. *CoRR*, abs/1912.05372.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. 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. Association for Computational Linguistics.

Ye Qi, Devendra Singh Sachan, Matthieu Felix, Sarguna Janani Padmanabhan, and Graham Neubig. 2018. When and why are pre-trained word embeddings useful for neural machine translation?

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR*, abs/1910.10683.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2020. Multilingual translation with extensible multilingual pretraining and finetuning.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.

Weiran Wang, Raman Arora, Karen Livescu, and Jeff A. Bilmes. 2016. On deep multi-view representation learning: Objectives and optimization. *CoRR*, abs/1602.01024.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. 2020. Improving massively multilingual neural machine translation and zero-shot translation.

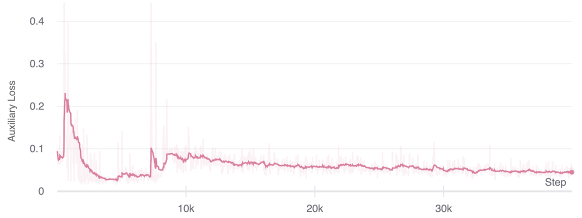

Figure 3: Auxiliary loss for the TRL-1.0 model during training

A Dataset Sizes

The languages used in our experiments are Azerbaijani (Az), English (En), Kazakh (Kk) and Turkish (Tr). The size of the bitext dataset for each language pair we use in the training, validation and test splits of the TED multilingual dataset are given below.

	Train	Val	Test
Az-En	5946	671	903
Az-Kk	442	127	214
Az-Tr	5585	490	811
En-Kk	3317	938	775
En-Tr	182471	4045	5029

Table 5: Number of parallel sentences for each language pair used in our experiments in each split of TED multilingual.

B Training Auxiliary Loss

A plot of the evolution of the auxiliary loss for the TRL-1.0 model is given in Figure 3. Although the loss is unstable for the first 10K training steps, it re-gains stability and smoothly decreases before converging to a small value. This demonstrates that the encoder representations learned are similar for semantically-identical sentences in different languages.

Our code is publicly available on GitHub: `https://github.com/IanYHWu/NLP_project.git`