Meta-Learning for Low-Resource Neural Machine Translation

Jiatao Gu*[†], Yong Wang*[†], Yun Chen[†], Kyunghyun Cho[‡] and Victor O.K. Li[†]

[†]The University of Hong Kong
[‡]New York University, CIFAR Azrieli Global Scholar
[†]{jiataogu, wangyong, vli}@eee.hku.hk
[†]yun.chencreek@gmail.com
[‡]kyunghyun.cho@nyu.edu

Abstract

In this paper, we propose to extend the recently introduced model-agnostic meta-learning algorithm (MAML, Finn et al., 2017) for low-resource neural machine translation (NMT). We frame low-resource translation as a meta-learning problem, and we learn to adapt to low-resource languages based on multilingual high-resource language tasks. We use the universal lexical representation (Gu et al., 2018b) to overcome the input-output mismatch across different languages. We evaluate the proposed meta-learning strategy using eighteen European languages (Bg, Cs, Da, De, El, Es, Et, Fr, Hu, It, Lt, Nl, Pl, Pt, Sk, Sl, Sv and Ru) as source tasks and five diverse languages (Ro, Lv, Fi, Tr and Ko) as target tasks. We show that the proposed approach significantly outperforms the multilingual, transfer learning based approach (Zoph et al., 2016) and enables us to train a competitive NMT system with only a fraction of training examples. For instance, the proposed approach can achieve as high as 22.04 BLEU on Romanian-English WMT'16 by seeing only 16,000 translated words ($\sim$ 600 parallel sentences).

1 Introduction

Despite the massive success brought by neural machine translation (NMT, Sutskever et al., 2014; Bahdanau et al., 2015; Vaswani et al., 2017), it has been noticed that the vanilla NMT often lags behind conventional machine translation systems, such as statistical phrase-based translation systems (PBMT, Koehn et al., 2003), for low-resource language pairs (see, e.g., Koehn and Knowles, 2017). In the past few years, various approaches have been proposed to address this issue. The first attempts at tackling this problem exploited the availability of monolingual corpora (Gulcehre et al., 2015; Sennrich et al., 2015; Zhang and Zong, 2016). It was later followed by approaches based on multilingual translation, in which the goal was to exploit knowledge from high-resource language pairs by training a single NMT system on a mix of high-resource and low-resource language pairs (Firat et al., 2016a,b; Lee et al., 2016; Johnson et al., 2016; Ha et al., 2016b). Its variant, transfer learning, was also proposed by Zoph et al. (2016), in which an NMT system is pretrained on a high-resource language pair before being fine-tuned on a target low-resource language pair.

In this paper, we follow up on these latest approaches based on multilingual NMT and propose a meta-learning algorithm for low-resource neural machine translation. We start by arguing that the recently proposed model-agnostic meta-learning algorithm (MAML, Finn et al., 2017) could be applied to low-resource machine translation by viewing language pairs as separate tasks. This view enables us to use MAML to find the initialization of model parameters that facilitate fast adaptation for a new language pair with a minimal amount of training examples (§3). Furthermore, the vanilla MAML however cannot handle tasks with mismatched input and output. We overcome this limitation by incorporating the universal lexical representation (Gu et al., 2018b) and adapting it for the meta-learning scenario (§3.3).

We extensively evaluate the effectiveness and generalizing ability of the proposed meta-learning algorithm on low-resource neural machine translation. We utilize 17 languages from Europarl and Russian from WMT as the source tasks and test the meta-learned parameter initialization against five target languages (Ro, Lv, Fi, Tr and Ko), in all cases translating to English. Our experiments using only up to 160k tokens in each of the target task reveal that the proposed meta-learning approach outperforms the multilingual translation

* Equal contribution.

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 3622–3631
Brussels, Belgium, October 31 - November 4, 2018.

approach across all the target language pairs, and the gap grows as the number of training examples decreases.

2 Background

Neural Machine Translation (NMT) Given a source sentence $X = \{x_1, ..., x_{T'}\}$, a neural machine translation model factors the distribution over possible output sentences $Y = \{y_1, ..., y_T\}$ into a chain of conditional probabilities with a left-to-right causal structure:

$$p(Y|X;\theta) = \prod_{t=1}^{T+1} p(y_t|y_{0:t-1}, x_{1:T'};\theta), \quad (1)$$

where special tokens y_0 ($\langle \mathrm{bos} \rangle$) and y_{T+1} ($\langle \mathrm{eos} \rangle$) are used to represent the beginning and the end of a target sentence. These conditional probabilities are parameterized using a neural network. Typically, an encoder-decoder architecture (Sutskever et al., 2014; Cho et al., 2014; Bahdanau et al., 2015) with a RNN-based decoder is used. More recently, architectures without any recurrent structures (Gehring et al., 2017; Vaswani et al., 2017) have been proposed and shown to speed up training while achieving state-of-the-art performance.

Low Resource Translation NMT is known to easily over-fit and result in an inferior performance when the training data is limited (Koehn and Knowles, 2017). In general, there are two ways for handling the problem of low resource translation: (1) utilizing the resource of unlabeled monolingual data, and (2) sharing the knowledge between low- and high-resource language pairs. Many research efforts have been spent on incorporating the monolingual corpora into machine translation, such as multi-task learning (Gulcehre et al., 2015; Zhang and Zong, 2016), back-translation (Sennrich et al., 2015), dual learning (He et al., 2016) and unsupervised machine translation with monolingual corpora only for both sides (Artetxe et al., 2017b; Lample et al., 2017; Yang et al., 2018).

For the second approach, prior researches have worked on methods to exploit the knowledge of auxiliary translations, or even auxiliary tasks. For instance, Cheng et al. (2016); Chen et al. (2017); Lee et al. (2017); Chen et al. (2018) investigate the use of a pivot to build a translation path between two languages even without any directed resource. The pivot can be a third language or even an image in multimodal domains. When pivots are not easy to obtain, Firat et al. (2016a); Lee et al. (2016); Johnson et al. (2016) have shown that the structure of NMT is suitable for multilingual machine translation. Gu et al. (2018b) also showed that such a multilingual NMT system could improve the performance of low resource translation by using a universal lexical representation to share embedding information across languages.

All the previous work for multilingual NMT assume the joint training of multiple high-resource languages naturally results in a universal space (for both the input representation and the model) which, however, is not necessarily true, especially for very low resource cases.

Meta Learning In the machine learning community, meta-learning, or learning-to-learn, has recently received interests. Meta-learning tries to solve the problem of "fast adaptation on new training data." One of the most successful applications of meta-learning has been on few-shot (or one-shot) learning (Lake et al., 2015), where a neural network is trained to readily learn to classify inputs based on only one or a few training examples. There are two categories of meta-learning:

1. learning a meta-policy for updating model parameters (see, e.g., Andrychowicz et al., 2016; Ha et al., 2016a; Mishra et al., 2017)

2. learning a good parameter initialization for fast adaptation (see, e.g., Finn et al., 2017; Vinyals et al., 2016; Snell et al., 2017).

In this paper, we propose to use a meta-learning algorithm for low-resource neural machine translation based on the second category. More specifically, we extend the idea of model-agnostic meta-learning (MAML, Finn et al., 2017) in the multilingual scenario.

3 Meta Learning for Low-Resource Neural Machine Translation

The underlying idea of MAML is to use a set of source tasks $\left\{\mathcal{T}^1, \ldots, \mathcal{T}^K\right\}$ to find the initialization of parameters θ^0 from which learning a target task $\mathcal{T}^0$ would require only a small number of training examples. In the context of machine translation, this amounts to using many high-resource language pairs to find good initial parameters and training a new translation model on a low-resource language starting from the found initial parame-

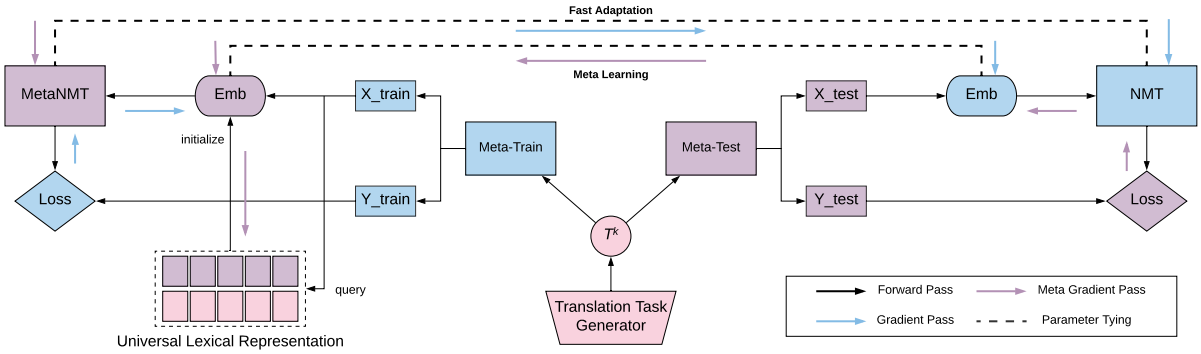


Figure 1: The graphical illustration of the training process of the proposed MetaNMT. For each episode, one task (language pair) is sampled for meta-learning. The boxes and arrows in blue are mainly involved in language-specific learning (§3.1), and those in purple in meta-learning (§3.2).

ters. This process can be understood as

$$\theta^* = \text{Learn}(\mathcal{T}^0; \text{MetaLearn}(\mathcal{T}^1, \ldots, \mathcal{T}^K)).$$

That is, we *meta-learn* the initialization from auxiliary tasks and continue to *learn* the target task. We refer the proposed meta-learning method for NMT to MetaNMT. See Fig. 1 for the overall illustration.

3.1 Learn: language-specific learning

Given any initial parameters θ^0 (which can be either random or meta-learned),

the prior distribution of the parameters of a desired NMT model can be defined as an isotropic Guassian:

$$\theta_i \sim \mathcal{N}(\theta_i^0, 1/\beta),$$

where $1/\beta$ is a variance. With this prior distribution, we formulate the language-specific learning process $\text{Learn}(D_\mathcal{T}; \theta^0)$ as maximizing the log-posterior of the model parameters given data $D_\mathcal{T}$:

$$\begin{aligned}\text{Learn}(D_\mathcal{T}; \theta^0) &= \arg\max_\theta \mathcal{L}^{D_\mathcal{T}}(\theta) \\ &= \arg\max_\theta \sum_{(X,Y)\in D_\mathcal{T}} \log p(Y|X,\theta) - \beta\|\theta - \theta^0\|^2,\end{aligned}$$

where we assume $p(X|\theta)$ to be uniform. The first term above corresponds to the maximum likelihood criterion often used for training a usual NMT system. The second term discourages the newly learned model from deviating too much from the initial parameters, alleviating the issue of overfitting when there is not enough training data. In practice, we solve the problem above by maximizing the first term with gradient-based optimization and early-stopping after only a few update steps.

Thus, in the low-resource scenario, finding a good initialization θ^0 strongly correlates the final performance of the resulting model.

3.2 MetaLearn

We find the initialization θ^0 by repeatedly simulating low-resource translation scenarios using auxiliary, high-resource language pairs. Following Finn et al. (2017), we achieve this goal by defining the meta-objective function as

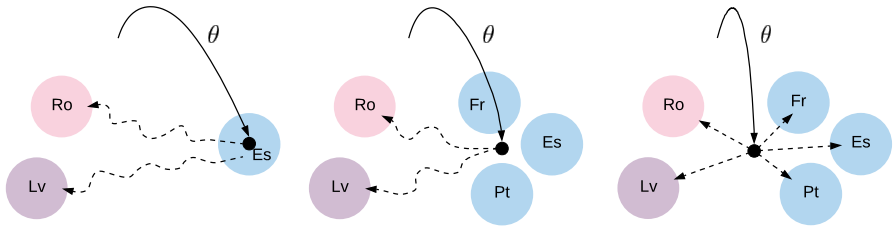
$$\begin{aligned}\mathcal{L}(\theta) =& \mathbb{E}_k \mathbb{E}_{D_{\mathcal{T}^k}, D'_{\mathcal{T}^k}} \\ & \left[\sum_{(X,Y)\in D'_{\mathcal{T}^k}} \log p(Y|X; \text{Learn}(D_{\mathcal{T}^k}; \theta)) \right],\end{aligned} \tag{2}$$

where $k \sim \mathcal{U}(\{1, \ldots, K\})$ refers to one meta-learning episode, and $D_\mathcal{T}$, $D'_\mathcal{T}$ follow the uniform distribution over $\mathcal{T}$'s data.

We maximize the meta-objective function using stochastic approximation (Robbins and Monro, 1951) with gradient descent. For each episode, we uniformly sample one source task at random, $\mathcal{T}^k$. We then sample two subsets of training examples independently from the chosen task, $D_{\mathcal{T}^k}$ and $D'_{\mathcal{T}^k}$. We use the former to *simulate* language-specific learning and the latter to *evaluate* its outcome. Assuming a single gradient step is taken only the with learning rate η, the simulation is:

$$\theta'_k = \text{Learn}(D_{\mathcal{T}^k}; \theta) = \theta - \eta \nabla_\theta \mathcal{L}^{D_{\mathcal{T}^k}}(\theta).$$

Once the simulation of learning is done, we evaluate the updated parameters θ'_k on $D'_{\mathcal{T}^k}$, The gradient computed from this evaluation, which we refer to as *meta-gradient*, is used to update the



(a) Transfer Learning (b) Multilingual Transfer Learning (c) Meta Learning

Figure 2: An intuitive illustration in which we use solid lines to represent the learning of initialization, and dashed lines to show the path of fine-tuning.

meta model θ. It is possible to aggregate multiple episodes of source tasks before updating θ:

$$\theta \leftarrow \theta - \eta' \sum_k \nabla_\theta \mathcal{L}^{D'_{\mathcal{T}^k}}(\theta'_k),$$

where η' is the meta learning rate.

Unlike a usual learning scenario, the resulting model θ^0 from this meta-learning procedure is not necessarily a good model on its own. It is however a good starting point for training a good model using only a few steps of learning. In the context of machine translation, this procedure can be understood as finding the initialization of a neural machine translation system that could quickly adapt to a new language pair by simulating such a fast adaptation scenario using many high-resource language pairs.

Meta-Gradient We use the following approximation property

$$H(x)v \approx \frac{\nabla(x + \nu v) - \nabla(x)}{\nu}$$

to approximate the meta-gradient:[1]

$$\begin{aligned}
&\nabla_\theta \mathcal{L}^{D'}(\theta') = \nabla_{\theta'} \mathcal{L}^{D'}(\theta') \nabla_\theta (\theta - \eta \nabla_\theta \mathcal{L}^D(\theta)) \\
&= \nabla_{\theta'} \mathcal{L}^{D'}(\theta') - \eta \nabla_{\theta'} \mathcal{L}^{D'}(\theta') H_\theta(\mathcal{L}^D(\theta)) \\
&\approx \nabla_{\theta'} \mathcal{L}^{D'}(\theta') - \frac{\eta}{\nu} \left[\nabla_\theta \mathcal{L}^D(\theta) \Big|_{\hat{\theta}} - \nabla_\theta \mathcal{L}^D(\theta) \Big|_{\theta} \right],
\end{aligned}$$

where ν is a small constant and

$$\hat{\theta} = \theta + \nu \nabla_{\theta'} \mathcal{L}^{D'}(\theta').$$

In practice, we find that it is also possible to ignore the second-order term, ending up with the following simplified update rule:

$$\nabla_\theta \mathcal{L}^{D'}(\theta') \approx \nabla_{\theta'} \mathcal{L}^{D'}(\theta'). \quad (3)$$

[1]We omit the subscript k for simplicity.

Related Work: Multilingual Transfer Learning The proposed MetaNMT differs from the existing framework of multilingual translation (Lee et al., 2016; Johnson et al., 2016; Gu et al., 2018b) or transfer learning (Zoph et al., 2016). The latter can be thought of as solving the following problem:

$$\max_\theta \mathcal{L}^{\text{multi}}(\theta) = \mathbb{E}_k \left[\sum_{(X,Y) \in D_k} \log p(Y|X;\theta) \right],$$

where D_k is the training set of the k-th task, or language pair. The target low-resource language pair could either be a part of joint training or be trained separately starting from the solution θ^0 found from solving the above problem.

The major difference between the proposed MetaNMT and these multilingual transfer approaches is that the latter do not consider how learning happens with the target, low-resource language pair. The former explicitly incorporates the learning process within the framework by simulating it repeatedly in Eq. (2). As we will see later in the experiments, this results in a substantial gap in the final performance on the low-resource task.

Illustration In Fig. 2, we contrast transfer learning, multilingual learning and meta-learning using three source language pairs (Fr-En, Es-En and Pt-En) and two target pairs (Ro-En and Lv-En). Transfer learning trains an NMT system specifically for a source language pair (Es-En) and finetunes the system for each target language pair (Ro-En, Lv-En). Multilingual learning often trains a single NMT system that can handle many different language pairs (Fr-En, Pt-En, Es-En), which may or may not include the target pairs (Ro-En, Lv-En). If not, it finetunes the system for each target pair, similarly to transfer learning. Both of these however aim at directly solving the source tasks. On the other hand, meta-learning trains the NMT system to be *useful for fine-tuning* on various tasks including the source and target tasks. This is done by repeatedly simulating the learning process on

low-resource languages using many high-resource language pairs (Fr-En, Pt-En, Es-En).

3.3 Unified Lexical Representation

I/O mismatch across language pairs One major challenge that limits applying meta-learning for low resource machine translation is that the approach outlined above assumes the input and output spaces are shared across all the source and target tasks. This, however, does not apply to machine translation in general due to the vocabulary mismatch across different languages. In multilingual translation, this issue has been tackled by using a vocabulary of sub-words (Sennrich et al., 2015) or characters (Lee et al., 2016) shared across multiple languages. This surface-level sharing is however limited, as it cannot be applied to languages exhibiting distinct orthography (e.g., Indo-Euroepan languages vs. Korean.)

Universal Lexical Representation (ULR) We tackle this issue by dynamically building a vocabulary specific to each language using a key-value memory network (Miller et al., 2016; Gulcehre et al., 2018), as was done successfully for low-resource machine translation recently by Gu et al. (2018b). We start with multilingual word embedding matrices $\epsilon^k_{\text{query}} \in \mathbb{R}^{|V_k| \times d}$ pretrained on large monolingual corpora, where V_k is the vocabulary of the k-th language. These embedding vectors can be obtained with small dictionaries of seed word pairs (Artetxe et al., 2017a; Smith et al., 2017) or in a fully unsupervised manner (Zhang et al., 2017; Conneau et al., 2018). We take one of these languages k' to build universal lexical representation consisting of a universal embedding matrix $\epsilon_u \in \mathbb{R}^{M \times d}$ and a corresponding key matrix $\epsilon_{\text{key}} \in \mathbb{R}^{M \times d}$, where $M < |V'_k|$. Both $\epsilon^k_{\text{query}}$ and ϵ_{key} are fixed during meta-learning. We then compute the language-specific embedding of token x from the language k as the convex sum of the universal embedding vectors by

$$\epsilon^0[x] = \sum_{i=1}^{M} \alpha_i \epsilon_u[i],$$

where $\alpha_i \propto \exp\left\{-\frac{1}{\tau}\epsilon_{\text{key}}[i]^\top A \epsilon^k_{\text{query}}[x]\right\}$ and τ is set to 0.05. This approach allows us to handle languages with different vocabularies using a fixed number of shared parameters (ϵ_u, ϵ_{key} and A.)

Learning of ULR It is not desirable to update the universal embedding matrix ϵ_u when fine-tuning on a small corpus which contains a limited set of unique tokens in the target language, as it could adversely influence the other tokens' embedding vectors. We thus estimate the change to each embedding vector induced by language-specific learning by a separate parameter $\Delta\epsilon^k[x]$:

$$\epsilon^k[x] = \epsilon^0[x] + \Delta\epsilon^k[x].$$

During language-specific learning, the ULR $\epsilon^0[x]$ is held constant, while only $\Delta\epsilon^k[x]$ is updated, starting from an all-zero vector. On the other hand, we hold $\Delta\epsilon^k[x]$'s constant while updating ϵ_u and A during the meta-learning stage.

	# of sents.	# of En tokens	Dev	Test
Ro-En	0.61 M	16.66 M	–	31.76
Lv-En	4.46 M	67.24 M	20.24	15.15
Fi-En	2.63 M	64.50 M	17.38	20.20
Tr-En	0.21 M	5.58 M	15.45	13.74
Ko-En	0.09 M	2.33 M	6.88	5.97

Table 1: Statistics of full datasets of the target language pairs. BLEU scores on the dev and test sets are reported from a supervised Transformer model with the same architecture.

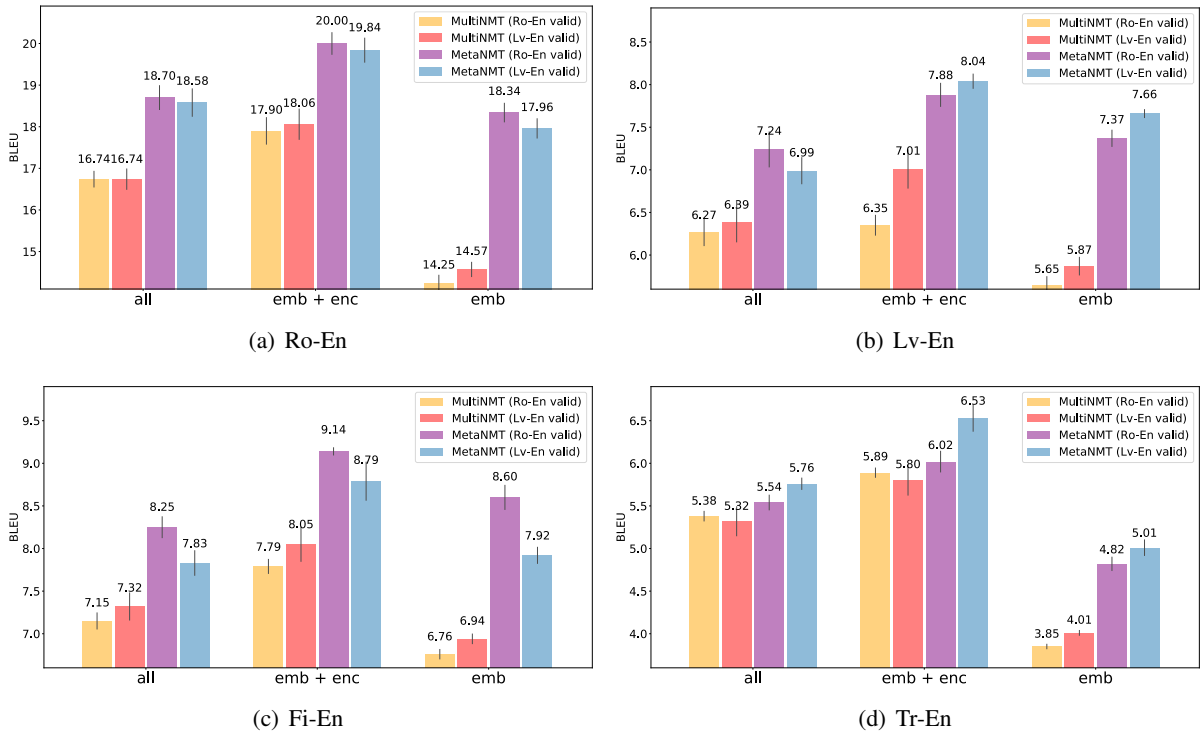
4 Experimental Settings

4.1 Dataset

Target Tasks We show the effectiveness of the proposed meta-learning method for low resource NMT with extremely limited training examples on five diverse target languages: Romanian (Ro) from WMT'16,[2] Latvian (Lv), Finnish (Fi), Turkish (Tr) from WMT'17,[3] and Korean (Ko) from Korean Parallel Dataset.[4] We use the officially provided train, dev and test splits for all these languages. The statistics of these languages are presented in Table 1. We simulate the low-resource translation scenarios by randomly sub-sampling the training set with different sizes.

Source Tasks We use the following languages from Europarl[5]: Bulgarian (Bg), Czech (Cs), Danish (Da), German (De), Greek (El), Spanish (Es), Estonian (Et), French (Fr), Hungarian (Hu), Italian (It), Lithuanian (Lt), Dutch (Nl), Polish (Pl), Portuguese (Pt), Slovak (Sk), Slovene (Sl) and

[2] http://www.statmt.org/wmt16/translation-task.html
[3] http://www.statmt.org/wmt17/translation-task.html
[4] https://sites.google.com/site/koreanparalleldata/
[5] http://www.statmt.org/europarl/



(a) Ro-En (b) Lv-En

(c) Fi-En (d) Tr-En

Figure 3: BLEU scores reported on test sets for {Ro, Lv, Fi, Tr} to En, where each model is first learned from 6 source tasks (Es, Fr, It, Pt, De, Ru) and then fine-tuned on randomly sampled training sets with around 16,000 English tokens per run. The error bars show the standard deviation calculated from 5 runs.

Swedish (Sv), in addition to Russian (Ru)[6] to learn the intilization for fine-tuning. In our experiments, different combinations of source tasks are explored to see the effects from the source tasks.

Validation We pick either Ro-En or Lv-En as a validation set for meta-learning and test the generalization capability on the remaining target tasks. This allows us to study the strict form of meta-learning, in which target tasks are unknown during both training and model selection.

Preprocessing and ULR Initialization As described in §3.3, we initialize the query embedding vectors $\epsilon^k_{\text{query}}$ of all the languages. For each language, we use the monolingual corpora built from Wikipedia[7] and the parallel corpus. The concatenated corpus is first tokenized and segmented using byte-pair encoding (BPE, Sennrich et al., 2016), resulting in $40,000$ subwords for each language. We then estimate word vectors using fastText (Bojanowski et al., 2016) and align them across all the languages in an unsupervised way using MUSE (Conneau et al., 2018) to get multilingual word vectors. We use the multilingual word vectors of the 20,000 most frequent words in English to form the universal embedding matrix ϵ_u.

4.2 Model and Learning

Model We utilize the recently proposed Transformer (Vaswani et al., 2017) as an underlying NMT system. We implement Transformer in this paper based on (Gu et al., 2018a)[8] and modify it to use the universal lexical representation from §3.3. We use the default set of hyperparameters ($d_{\text{model}} = d_{\text{hidden}} = 512$, $n_{\text{layer}} = 6$, $n_{\text{head}} = 8$, $n_{\text{batch}} = 4000$, $t_{\text{warmup}} = 16000$) for all the language pairs and across all the experimental settings. We refer the readers to (Vaswani et al., 2017; Gu et al., 2018a) for the details of the model. However, since the proposed meta-learning method is model-agnostic, it can be easily extended to any other NMT architectures, e.g. RNN-based sequence-to-sequence models with attention (Bahdanau et al., 2015).

[6] A subsample of approximately 2M pairs from WMT'17.

[7] We use the most recent Wikipedia dump (2018.5) from https://dumps.wikimedia.org/backup-index.html.

[8] https://github.com/salesforce/nonauto-nmt

Meta-Train	Ro-En		Lv-En		Fi-En		Tr-En		Ko-En	
	zero	finetune	zero	finetune	zero	finetune	zero	finetune	zero	finetune
–		$00.00 \pm .00$		$0.00 \pm .00$		$0.00 \pm .00$		$0.00 \pm .00$		$0.00 \pm .00$
Es	9.20	$15.71 \pm .22$	2.23	$4.65 \pm .12$	2.73	$5.55 \pm .08$	1.56	$4.14 \pm .03$	0.63	$1.40 \pm .09$
Es Fr	12.35	$17.46 \pm .41$	2.86	$5.05 \pm .04$	3.71	$6.08 \pm .01$	2.17	$4.56 \pm .20$	0.61	$1.70 \pm .14$
Es Fr It Pt	13.88	$18.54 \pm .19$	3.88	$5.63 \pm .11$	4.93	$6.80 \pm .04$	2.49	$4.82 \pm .10$	0.82	$1.90 \pm .07$
De Ru	10.60	$16.05 \pm .31$	5.15	$7.19 \pm .17$	6.62	$7.98 \pm .22$	3.20	$6.02 \pm .11$	1.19	$2.16 \pm .09$
Es Fr It Pt De Ru	15.93	$20.00 \pm .27$	6.33	$7.88 \pm .14$	7.89	$9.14 \pm .05$	3.72	$6.02 \pm .13$	1.28	$2.44 \pm .11$
All	18.12	$\mathbf{22.04 \pm .23}$	9.58	$\mathbf{10.44 \pm .17}$	11.39	$\mathbf{12.63 \pm .22}$	5.34	$\mathbf{8.97 \pm .08}$	1.96	$\mathbf{3.97 \pm .10}$
Full Supervised	31.76		15.15		20.20		13.74		5.97	

Table 2: BLEU Scores w.r.t. the source task set for all five target tasks.

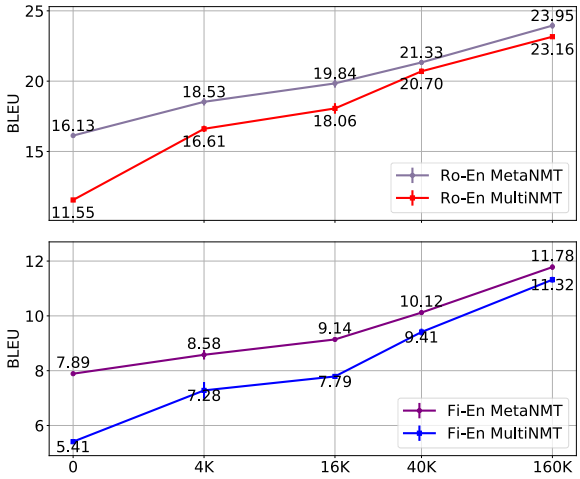


Figure 4: BLEU Scores w.r.t. the size of the target task's training set.

Learning We meta-learn using various sets of source languages to investigate the effect of source task choice. For each episode, by default, we use a single gradient step of language-specific learning with Adam (Kingma and Ba, 2014) per computing the meta-gradient, which is computed by the first-order approximation in Eq. (3).

For each target task, we sample training examples to form a low-resource task. We build tasks of 4k, 16k, 40k and 160k English tokens for each language. We randomly sample the training set five times for each experiment and report the average score and its standard deviation. Each fine-tuning is done on a training set, early-stopped on a validation set and evaluated on a test set. In default without notation, datasets of 16k tokens are used.

Fine-tuning Strategies The transformer consists of three modules; embedding, encoder and decoder. We update all three modules during meta-learning, but during fine-tuning, we can selectively tune only a subset of these modules. Following (Zoph et al., 2016), we consider three fine-tuning strategies; (1) fine-tuning all the modules (all), (2) fine-tuning the embedding and encoder, but freezing the parameters of the decoder (emb+enc) and (3) fine-tuning the embedding only (emb).

5 Results

vs. Multilingual Transfer Learning We meta-learn the initial models on all the source tasks using either Ro-En or Lv-En as a validation task. We also train the initial models to be multilingual translation systems. We fine-tune them using the four target tasks (Ro-En, Lv-En, Fi-En and Tr-En; 16k tokens each) and compare the proposed meta-learning strategy and the multilingual, transfer learning strategy. As presented in Fig. 3, the proposed learning approach significantly outperforms the multilingual, transfer learning strategy across all the target tasks regardless of which target task was used for early stopping. We also notice that the emb+enc strategy is most effective for both meta-learning and transfer learning approaches. With the proposed meta-learning and emb+enc fine-tuning, the final NMT systems trained using only a fraction of all available training examples achieve 2/3 (Ro-En) and 1/2 (Lv-En, Fi-En and Tr-En) of the BLEU score achieved by the models trained with full training sets.

vs. Statistical Machine Translation We also test the same Ro-En datasets with $16,000$ target tokens using the default setting of Phrase-based MT (Moses) with the dev set for adjusting the parameters and the test set for calculating the final performance. We obtain $4.79(\pm 0.234)$ BLEU point, which is higher than the standard NMT performance (0 BLEU). It is however still lower than both the multi-NMT and meta-NMT.

Impact of Validation Tasks Similarly to training any other neural network, meta-learning still requires early-stopping to avoid overfitting to a

specific set of source tasks. In doing so, we observe that the choice of a validation task has non-negligible impact on the final performance. For instance, as shown in Fig. 3, Fi-En benefits more when Ro-En is used for validation, while the opposite happens with Tr-En. The relationship between the task similarity and the impact of a validation task must be investigated further in the future.

Training Set Size We vary the size of the target task's training set and compare the proposed meta-learning strategy and multilingual, transfer learning strategy. We use the emb+enc fine-tuning on Ro-En and Fi-En. Fig. 4 demonstrates that the meta-learning approach is more robust to the drop in the size of the target task's training set. The gap between the meta-learning and transfer learning grows as the size shrinks, confirming the effectiveness of the proposed approach on extremely low-resource language pairs.

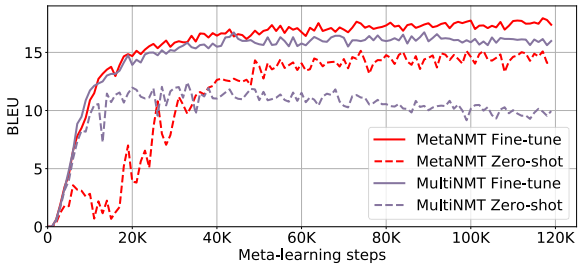


Figure 5: The learning curves of BLEU scores on the validation task (Ro-En).

Impact of Source Tasks In Table 2, we present the results on all five target tasks obtained while varying the source task set. We first see that it is always beneficial to use more source tasks. Although the impact of adding more source tasks varies from one language to another, there is up to 2× improvement going from one source task to 18 source tasks (Lv-En, Fi-En, Tr-En and Ko-En). The same trend can be observed even without any fine-tuning (i.e., unsupervised translation, (Lample et al., 2017; Artetxe et al., 2017b)). In addition, the choice of source languages has different implications for different target languages. For instance, Ro-En benefits more from {Es, Fr, It, Pt} than from {De, Ru}, while the opposite effect is observed with all the other target tasks.

Training Curves The benefit of meta-learning over multilingual translation is clearly demonstrated when we look at the training curves in Fig. 5. With the multilingual, transfer learning approach, we observe that training rapidly saturates and eventually degrades, as the model overfits to the source tasks. MetaNMT on the other hand continues to improve and never degrades, as the meta-objective ensures that the model is adequate for fine-tuning on target tasks rather than for solving the source tasks.

Sample Translations We present some sample translations from the tested models in Table 3. Inspecting these examples provides the insight into the proposed meta-learning algorithm. For instance, we observe that the meta-learned model without any fine-tuning produces a word-by-word translation in the first example (Tr-En), which is due to the successful use of the universal lexcial representation and the meta-learned initialization. The system however cannot reorder tokens from Turkish to English, as it has not seen any training example of Tr-En. After seeing around 600 sentence pairs (16K English tokens), the model rapidly learns to correctly reorder tokens to form a better translation. A similar phenomenon is observed in the Ko-En example. These cases could be found across different language pairs.

6 Conclusion

In this paper, we proposed a meta-learning algorithm for low-resource neural machine translation that exploits the availability of high-resource languages pairs. We based the proposed algorithm on the recently proposed model-agnostic meta-learning and adapted it to work with multiple languages that do not share a common vocabulary using the technique of universal lexcal representation, resulting in MetaNMT. Our extensive evaluation, using 18 high-resource source tasks and 5 low-resource target tasks, has shown that the proposed MetaNMT significantly outperforms the existing approach of multilingual, transfer learning in low-resource neural machine translation across all the language pairs considered.

The proposed approach opens new opportunities for neural machine translation. First, it is a principled framework for incorporating various extra sources of data, such as source- and target-side monolingual corpora. Second, it is a generic framework that can easily accommodate existing and future neural machine translation systems.

Source (Tr)	google mülteciler için 11 milyon dolar toplamak üzere bağış eşleştirme kampanyasını başlattı .
Target	google launches donation-matching campaign to raise $ 11 million for refugees .
Meta-0	google refugee fund for usd 11 million has launched a campaign for donation .
Meta-16k	google has launched a campaign to collect $ 11 million for refugees .
Source (Ko)	이번에 체포되어 기소된 사람들 중에는 퇴역한 군 고위관리 , 언론인 , 정치인 , 경제인 등이 포함됐다
Target	among the suspects are retired military officials , journalists , politicians , businessmen and others .
Meta-0	last year , convicted people , among other people , of a high-ranking army of journalists in economic and economic policies , were included .
Meta-16k	the arrested persons were included in the charge , including the military officials , journalists , politicians and economists .

Table 3: Sample translations for Tr-En and Ko-En highlight the impact of fine-tuning which results in syntactically better formed translations. We highlight tokens of interest in terms of reordering.

Acknowledgement

This research was supported in part by the Facebook Low Resource Neural Machine Translation Award. This work was also partly supported by Samsung Advanced Institute of Technology (Next Generation Deep Learning: from pattern recognition to AI) and Samsung Electronics (Improving Deep Learning using Latent Structure). KC thanks support by eBay, TenCent, NVIDIA and CIFAR.

References

Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W Hoffman, David Pfau, Tom Schaul, and Nando de Freitas. 2016. Learning to learn by gradient descent by gradient descent. In *Advances in Neural Information Processing Systems*, pages 3981–3989.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2017a. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 451–462.

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2017b. Unsupervised neural machine translation. *arXiv preprint arXiv:1710.11041*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Yun Chen, Yang Liu, Yong Cheng, and Victor OK Li. 2017. A teacher-student framework for zero-resource neural machine translation. *arXiv preprint arXiv:1705.00753*.

Yun Chen, Yang Liu, and Victor OK Li. 2018. Zero-resource neural machine translation with multi-agent communication game. *arXiv preprint arXiv:1802.03116*.

Yong Cheng, Yang Liu, Qian Yang, Maosong Sun, and Wei Xu. 2016. Neural machine translation with pivot languages. *arXiv preprint arXiv:1611.04928*.

Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder–Decoder approaches. In *Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. *International Conference on Learning Representations*.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. *arXiv preprint arXiv:1703.03400*.

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

Orhan Firat, Baskaran Sankaran, Yaser Al-Onaizan, Fatos T Yarman Vural, and Kyunghyun Cho. 2016b. Zero-resource translation with multi-lingual neural machine translation. In *EMNLP*.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann Dauphin. 2017. Convolutional sequence to sequence learning. *arXiv preprint arXiv:1705.03122*.

Jiatao Gu, James Bradbury, Caiming Xiong, Victor O. K. Li, and Richard Socher. 2018a. Non-autoregressive neural machine translation. *ICLR*.

Jiatao Gu, Hany Hassan, Jacob Devlin, and Victor OK Li. 2018b. Universal neural machine translation for extremely low resource languages. *arXiv preprint arXiv:1802.05368*.

Caglar Gulcehre, Sarath Chandar, Kyunghyun Cho, and Yoshua Bengio. 2018. Dynamic neural turing machine with continuous and discrete addressing schemes. *Neural computation*, 30(4):857–884.

Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loic Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2015. On using monolingual corpora in neural machine translation. *arXiv preprint arXiv:1503.03535*.

David Ha, Andrew Dai, and Quoc V Le. 2016a. Hypernetworks. *arXiv preprint arXiv:1609.09106*.

Thanh-Le Ha, Jan Niehues, and Alexander Waibel. 2016b. Toward multilingual neural machine translation with universal encoder and decoder. *arXiv preprint arXiv:1611.04798*.

Di He, Yingce Xia, Tao Qin, Liwei Wang, Nenghai Yu, Tieyan Liu, and Wei-Ying Ma. 2016. Dual learning for machine translation. In *Advances in Neural Information Processing Systems*, pages 820–828.

Melvin Johnson, Mike Schuster, Quoc V Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, et al. 2016. Google's multilingual neural machine translation system: enabling zero-shot translation. *arXiv preprint arXiv:1611.04558*.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. *arXiv preprint arXiv:1706.03872*.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 48–54. Association for Computational Linguistics.

Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. 2015. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338.

Guillaume Lample, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2017. Unsupervised machine translation using monolingual corpora only. *arXiv preprint arXiv:1711.00043*.

Jason Lee, Kyunghyun Cho, and Thomas Hofmann. 2016. Fully character-level neural machine translation without explicit segmentation. *arXiv preprint arXiv:1610.03017*.

Jason Lee, Kyunghyun Cho, Jason Weston, and Douwe Kiela. 2017. Emergent translation in multi-agent communication. *arXiv preprint arXiv:1710.06922*.

Alexander Miller, Adam Fisch, Jesse Dodge, Amir-Hossein Karimi, Antoine Bordes, and Jason Weston. 2016. Key-value memory networks for directly reading documents. *arXiv preprint arXiv:1606.03126*.

Nikhil Mishra, Mostafa Rohaninejad, Xi Chen, and Pieter Abbeel. 2017. Meta-learning with temporal convolutions. *arXiv preprint arXiv:1707.03141*.

Herbert Robbins and Sutton Monro. 1951. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Improving neural machine translation models with monolingual data. *arXiv preprint arXiv:1511.06709*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Edinburgh neural machine translation systems for wmt 16. *arXiv preprint arXiv:1606.02891*.

Samuel L Smith, David HP Turban, Steven Hamblin, and Nils Y Hammerla. 2017. Offline bilingual word vectors, orthogonal transformations and the inverted softmax. *arXiv preprint arXiv:1702.03859*.

Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, pages 4080–4090.

Ilya Sutskever, Oriol Vinyals, and Quôc Lê. 2014. Sequence to sequence learning with neural networks. In *NIPS*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Oriol Vinyals, Charles Blundell, Tim Lillicrap, Daan Wierstra, et al. 2016. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*, pages 3630–3638.

Zhen Yang, Wei Chen, Feng Wang, and Bo Xu. 2018. Unsupervised neural machine translation with weight sharing. *arXiv preprint arXiv:1804.09057*.

Jiajun Zhang and Chengqing Zong. 2016. Exploiting source-side monolingual data in neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1535–1545.

Meng Zhang, Yang Liu, Huanbo Luan, and Maosong Sun. 2017. Earth mover's distance minimization for unsupervised bilingual lexicon induction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1934–1945. Association for Computational Linguistics.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer learning for low-resource neural machine translation. *arXiv preprint arXiv:1604.02201*.