

# Soft Layer Selection with Meta-Learning for Zero-Shot Cross-Lingual Transfer

Weijia Xu\*<sup>†</sup> Batool Haider<sup>‡</sup> Jason Krone<sup>‡</sup> Saab Mansour<sup>‡</sup>

<sup>†</sup>Department of Computer Science, University of Maryland

<sup>‡</sup>Amazon AI

weijia@cs.umd.edu, {bhaider, kronej, saabm}@amazon.com

## Abstract

Multilingual pre-trained contextual embedding models (Devlin et al., 2019) have achieved impressive performance on zero-shot cross-lingual transfer tasks. Finding the most effective strategy to fine-tune these models on high-resource languages so that it transfers well to the zero-shot languages is a non-trivial task. In this paper, we propose a novel meta-optimizer to soft-select which layers of the pre-trained model to freeze during fine-tuning. We train the meta-optimizer by simulating the zero-shot transfer scenario. Results on cross-lingual natural language inference show that our approach improves over the simple fine-tuning baseline and X-MAML (Nooralahzadeh et al., 2020).

## 1 Introduction

Despite the impressive performance of neural models on a wide variety of NLP tasks, these models are extremely data hungry – training them requires a large amount of annotated data. As collecting such amounts of data for every language of interest is extremely expensive, cross-lingual transfer that aims to transfer the task knowledge from high-resource (*source*) languages for which annotated data are more readily available to low-resource (*target*) languages becomes a promising direction. Cross-lingual transfer approaches using cross-lingual resources such as machine translation (MT) systems (Wan, 2009; Conneau et al., 2018) or bilingual dictionaries (Prettenhofer and Stein, 2010) have effectively reduced the amount of annotated data required to obtain reasonable performance on the target language. However, such cross-lingual resources are often limited for low-resource languages.

Recent advances in cross-lingual contextual embedding models have reduced the need for cross-

lingual supervision (Devlin et al., 2019; Lample and Conneau, 2019). Wu and Dredze (2019) show that multilingual BERT (mBERT) (Devlin et al., 2019), a contextual embedding model pre-trained on the concatenated Wikipedia data from 104 languages without cross-lingual alignment, does surprisingly well on zero-shot cross-lingual transfer tasks, where they fine-tune the model on the annotated data from the source languages and evaluate on the target language. Wu and Dredze (2019) propose to freeze the bottom layers of mBERT during fine-tuning to improve the cross-lingual performance over the simple fine-tune-all-parameters strategy, as different layers of mBERT captures different linguistic information (Jawahar et al., 2019).

Selecting which layers to freeze for a downstream task is a non-trivial problem. In this paper, we propose a novel meta-learning algorithm for soft layer selection. Our meta-learning algorithm learns layer-wise update rate by simulating the zero-shot transfer scenario – at each round, we randomly split the source languages into a held-out language and the rest as training languages, fine-tune the model on the training languages, and update the meta-parameters based on the model performance on the held-out language. We build the meta-optimizer on top of a standard optimizer and learnable update rates, so that it generalizes well to large numbers of updates. Our method uses much less meta-parameters than the X-MAML approach (Nooralahzadeh et al., 2020) adapted from model-agnostic meta-learning (MAML) (Finn et al., 2017) to zero-shot cross-lingual transfer.

Experiments on zero-shot cross-lingual natural language inference show that our approach outperforms both the simple fine-tuning baseline and the X-MAML algorithm and that our approach brings larger gains when transferring from multiple source languages. Ablation study shows that both

\*Work done while interning at Amazon AI.

the layer-wise update rate and cross-lingual meta-training are key to the success of our approach.

## 2 Meta-Learning for Zero-Shot Cross-lingual Transfer

The idea of transfer learning is to improve the performance on the target task  $\mathcal{T}^0$  by learning from a set of related source tasks  $\{\mathcal{T}^1, \mathcal{T}^2, \dots, \mathcal{T}^K\}$ . In the context of cross-lingual transfer, we treat different languages as separate tasks, and our goal is to transfer the task knowledge from the source languages to the target language. In contrast to the transfer learning case where the inputs of the source and target tasks are from the same language, in cross-lingual transfer learning we need to handle inputs from different languages with different vocabularies and syntactic structures. To handle the issue, we use the pre-trained multilingual BERT (Devlin et al., 2019), a language model encoder trained on the concatenation of monolingual corpora from 104 languages.

The most widely used approach to zero-shot cross-lingual transfer using multilingual BERT is to fine-tune the BERT model  $\theta$  on the source language tasks  $\mathcal{T}^{1\dots K}$  with training objective  $\mathcal{L}$

$$\theta^* = \text{Learn}(\mathcal{L}, \mathcal{T}^1, \dots, \mathcal{T}^K; \theta)$$

and then evaluate the fine-tuned model  $\theta^*$  on the target language task  $\mathcal{T}^0$ . The gap between training and testing can lead to sub-optimal performance on the target language.

To address the issue, we propose to train a meta-optimizer  $f_\phi$  for fine-tuning so that the fine-tuned model generalizes better to unseen languages. We train the meta-optimizer by

$$\phi^* = \text{Learn}(\mathcal{L}, \mathcal{T}^k; \text{MetaLearn}(\mathcal{L}, \mathcal{T}^{1\dots K} \setminus \mathcal{T}^k; \phi))$$

where  $\mathcal{T}^k$  is a ‘‘surprise’’ language randomly selected from the source language tasks  $\mathcal{T}^{1\dots K}$ .

### 2.1 Meta-Optimizer

Our meta-optimizer consists of a standard optimizer as the base optimizer and a set of meta-parameters to control the layer-wise update rates. An update step is formulated as:

$$\begin{aligned} \theta^t &= \theta^{t-1} - \lambda \odot \Delta\theta^t \\ \Delta\theta^t &= f_{opt}(\mathbf{g}^1, \dots, \mathbf{g}^t) \end{aligned} \quad (1)$$

where  $\theta^t$  represent the parameters of the learner model at time step  $t$ , and  $\Delta\theta^t$  is the update vector produced by the base optimizer  $f_{opt}$  given the

---

### Algorithm 1: Meta-Training

---

**Input:** Training data  $\{\mathcal{D}_1, \dots, \mathcal{D}_K\}$  in the source languages, learner model  $M$  with parameters  $\theta$ , and meta-optimizer with base optimizer  $f_{opt}$  and meta-parameters  $\phi$ .

**Output:** Meta-optimizer with parameters  $\phi$ .

```

1  $s \leftarrow 1$ 
2 Randomly initialize  $\phi^0$ .
3 repeat  $N$  times
4    $t \leftarrow 1$ 
5   Initialize  $\theta^0$  with mBERT and random
   values for the classification layer.
6   Randomly select a test language  $k$  to form
   the test data  $\mathcal{D}_{test} = \mathcal{D}_k$ .
7    $\mathcal{D}_{train} \leftarrow \{\mathcal{D}_1, \dots, \mathcal{D}_K\} \setminus \mathcal{D}_{test}$ 
8   repeat  $L$  times
9      $\mathbf{X}^t, \mathbf{Y}^t \leftarrow$  random batch from  $\mathcal{D}_{train}$ 
10     $\mathcal{L}^t \leftarrow \mathcal{L}(M(\mathbf{X}^t; \theta^{t-1}), \mathbf{Y}^t)$ 
11     $\mathbf{g}^{1\dots t} \leftarrow [\mathbf{g}^{1\dots t-1}, \nabla_{\theta^{t-1}} \mathcal{L}^t]$ 
12     $\Delta\theta^t \leftarrow f_{opt}(\mathbf{g}^1, \dots, \mathbf{g}^t)$ 
13     $\theta^t \leftarrow \theta^{t-1} - \sigma(\phi^{s-1}) \odot \Delta\theta^t$ 
14     $t \leftarrow t + 1$ 
15  end
16   $\mathbf{X}, \mathbf{Y} \leftarrow \mathcal{D}_{test}$ 
17   $\mathcal{L}_{test} \leftarrow \mathcal{L}(M(\mathbf{X}; \theta^t), \mathbf{Y})$ 
18   $\phi^s \leftarrow \text{Update}(\phi^{s-1}, \nabla_{\phi^{s-1}} \mathcal{L}_{test})$ 
19   $s \leftarrow s + 1$ 
20 end
```

---

gradients  $\{\mathbf{g}^i = \nabla_{\theta^{i-1}} \mathcal{L}^i\}_{i=1}^t$  at the current and previous steps. The function  $f_{opt}$  is defined by the optimization algorithm and its hyper-parameters. For example, a typical gradient descent algorithm uses  $f_{opt} = \alpha \mathbf{g}^t$  where  $\alpha$  represents the learning rate. A standard optimization algorithm will update the model parameters by:

$$\theta^t = \theta^{t-1} - f_{opt}(\mathbf{g}^1, \dots, \mathbf{g}^t) \quad (2)$$

Our meta-optimizer is different in that we perform gated update using parametric update rates  $\lambda$ , which is computed by  $\lambda = \sigma(\phi)$ , where  $\phi$  represents the meta-parameters of the meta-optimizer  $f_\phi$ . The sigmoid function ensures that the update rates are within the range  $[0, 1]$ . Different from Andrychowicz et al. (2016) in which the optimizer parameters are shared across all coordi-

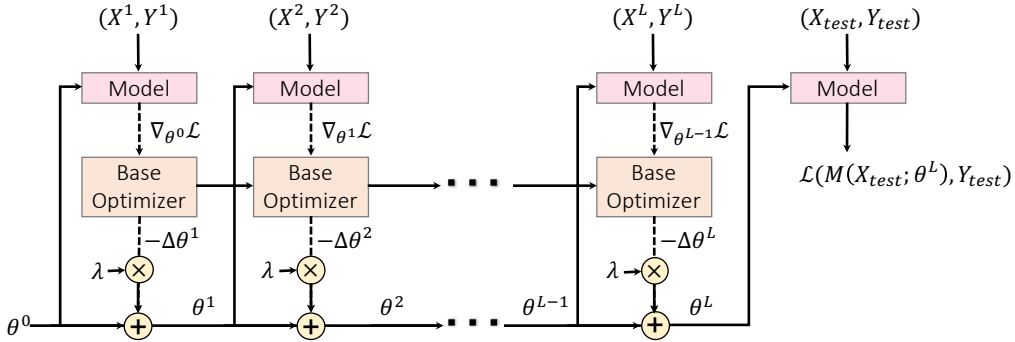


Figure 1: Computational graph for the forward pass of the meta-optimizer. Each batch  $(\mathbf{X}^t, \mathbf{Y}^t)$  is from the training data  $\mathcal{D}_{train}$ , and  $(\mathbf{X}_{test}, \mathbf{Y}_{test})$  denotes the entire test set. The meta-learner is comprised of a base optimizer that takes the history and current step gradients as inputs and suggests an update  $\Delta\theta^t$ , and the meta-parameters that control the layer-wise update rates  $\lambda$  for the learner model  $\theta$ . The dashed arrows indicate that we do not back-propagate the gradients through that step when updating the meta-parameters.

nates of the model, our meta-optimizer learns different update rates for different model layers. This is based on the findings that different layers of the BERT encoder capture different linguistic information, with syntactic features in middle layers and semantic information in higher layers (Jawahar et al., 2019). And thus, different layers may generalize differently across languages.

Figure 1 illustrates the computational graph for the forward pass when training the meta-optimizer. Note that as the losses  $\mathcal{L}^t$  and gradients  $\nabla_{\theta^{t-1}} \mathcal{L}^t$  are dependent on the parameters of the meta-optimizer, computing the gradients along the dashed edges would normally require taking second derivatives, which is computationally expensive. Following Andrychowicz et al. (2016), we drop the gradients along the dashed edges and only compute gradients along the solid edges.

## 2.2 Meta-Training

A good meta-optimizer will, given the training data in the source languages and the training objective, suggest an update rule for the learner model so that it performs well on the target language. Thus, we would like the training condition to match that of the test time. However, in zero-shot transfer we assume no access to the target language data, so we need to simulate the test scenario using only the training data on the source languages.

As shown in Algorithm 1, at each episode in the outer loop, we randomly choose a test language  $k$  to construct the test data  $\mathcal{D}_{test} = \mathcal{D}_k$  and use the remaining data as the training data  $\mathcal{D}_{train}$ .

Then, we re-initialize the parameters of the learner model and start the training simulation. At each training step, we first use the base optimizer  $f_{opt}$  to compute the update vector  $\Delta\theta^t$  based on the current and history gradients  $\mathbf{g}^{1..t}$ . We then perform the gated update using the meta-optimizer  $\phi^{s-1}$  with Eq. (1). The resulting model  $\theta^t$  can be viewed as the output of a forward pass of the meta-optimizer. After every  $L$  iterations of model update, we compute the gradient of the loss on the test data  $\mathcal{D}_{test}$  with respect to the old meta parameters  $\phi^{s-1}$  and make an update to the meta parameters. Our meta-learning algorithm is different from X-MAML (Nooralahzadeh et al., 2020) in that 1) X-MAML is designed mainly for few-shot transfer while our algorithm is designated for zero-shot transfer, and 2) our algorithm uses much less meta-parameters than X-MAML as it only requires training the update rate for each layer while in X-MAML we meta-learn the initial parameters of the entire model.

## 3 Experiments

We evaluate our meta-learning approach on natural language inference. Natural Language Inference (NLI) can be cast into a sequence pair classification problem where, given a premise and a hypothesis sentence, the model needs to predict whether the premise entails the hypothesis, contradicts it, or neither (neutral). We use the Multi-Genre Natural Language Inference Corpus (Williams et al., 2018), which consists of 433k English sentence pairs labeled with textual entailment information, and the XNLI dataset (Conneau

	fr	es	de	ar	ur	bg	sw	th	tr	vi	zh	ru	el	hi	avg
Devlin et al. (2019)	–	74.30	70.50	62.10	58.35	–	–	–	–	–	63.80	–	–	–	–
Wu and Dredze (2019)	74.60	74.90	72.00	66.10	58.60	69.80	49.40	55.70	62.00	71.90	70.40	69.80	67.90	61.20	66.02
Nooralahzadeh et al. (2020)	74.42	75.07	71.83	66.05	61.51	69.45	49.76	55.39	61.20	71.82	71.11	70.19	67.95	62.20	66.28
Aux. language	el	el	el	el	el	el	el	el	el	el	ur	ur	ur	ur	
Fine-tuning baseline	75.42	75.77	72.57	67.22	61.08	70.23	<b>51.70</b>	<b>51.03</b>	<b>64.26</b>	71.61	<b>72.52</b>	69.97	69.16	55.40	66.28
Meta-Optimizer	<b>75.78</b>	<b>75.87</b>	<b>73.15</b>	<b>67.34</b>	<b>62.00</b>	<b>70.47</b>	51.22	50.54	63.96	<b>72.06</b>	72.32	<b>70.20</b>	<b>69.34</b>	<b>55.88</b>	<b>66.44</b>
Aux. language: el + ur															
Fine-tuning baseline	74.87	75.78	72.27	66.96	62.73	70.16	50.21	48.20	63.86	71.61	71.97	70.24	69.64	56.04	66.04
Meta-Optimizer	<b>75.53</b>	<b>75.93</b>	<b>72.68</b>	<b>67.04</b>	<b>63.33</b>	<b>70.88</b>	<b>51.51</b>	<b>49.89</b>	<b>64.33</b>	<b>72.06</b>	<b>72.36</b>	<b>70.32</b>	<b>70.38</b>	<b>56.29</b>	<b>66.61</b>

Table 1: Accuracy of our approach compared with baselines on the XNLI dataset (averaged over five runs). We compare our approach (*Meta-Optimizer*) with our fine-tuning baseline with one or two auxiliary languages, the fine-tuning results in Devlin et al. (2019), the highest scores (with a selected subset of layers fixed during fine-tuning) in Wu and Dredze (2019), the best zero-shot results using X-MAML (Nooralahzadeh et al., 2020) with one auxiliary language. We boldface the highest scores within each auxiliary language setting.

et al., 2018), which has 2.5k development and 5k test sentence pairs in 15 languages including English (en), French (fr), Spanish (es), German (de), Greek (el), Bulgarian (bg), Russian (ru), Turkish (tr), Arabic (ar), Vietnamese (vi), Thai (th), Chinese (zh), Hindi (hi), Swahili (sw), and Urdu (ur). We use this dataset to evaluate the effectiveness of our meta-learning algorithm when transferring from English and one or more low-resource auxiliary languages to the target language.

### 3.1 Model and Training Configurations

Our model is based on the multilingual BERT (mBERT) (Devlin et al., 2019) implemented in GluonNLP (Guo et al., 2020). As in previous work (Devlin et al., 2019; Wu and Dredze, 2019), we tokenize the input sentences using WordPiece, concatenate them, feed the sequence to BERT, and use the hidden representation of the first token ( $[CLS]$ ) for classification. The final output is computed by applying a linear projection and a softmax layer to the hidden representation. We use a dropout rate of 0.1 on the final encoder layer and fix the embedding layer during fine-tuning. Following Nooralahzadeh et al. (2020), we fine-tune mBERT by 1) fine-tune mBERT on the English data for one epoch to get initial model parameters, and 2) continue fine-tuning the model on the other source languages for two epochs. We compare using the standard optimizer (fine-tuning baseline) and our meta-optimizer for Step 2. We use Adam optimizer (Kingma and Ba, 2015) with a learning rate of  $2 \times 10^{-5}$ ,  $\beta_1 = 0.9$ , and  $\beta_2 = 0.999$  as the standard optimizer and base optimizer in our

meta-optimizer. To train our meta-optimizer, we use Adam with a learning rate of 0.05 for  $N = 10$  epochs with  $L = 15$  training batches per iteration (Algorithm 1). Different from Nooralahzadeh et al. (2020) who select the auxiliary languages for each target language that lead to the best transfer results, we simulate a more realistic scenario where only a limited set of auxiliary languages is available. We choose two distant auxiliary languages – Greek (Hellenic branch of the Indo-European language family) and Urdu (Indo-Aryan branch of the Indo-European language family) – and evaluate the transfer performance on the other languages.

### 3.2 Main Results

As shown in Table 1, we compare our meta-learning approach with the fine-tuning baseline and the zero-shot transfer results reported in prior work that uses mBERT. Our approach outperforms the fine-tuning methods in Devlin et al. (2019) by 1.6–8.5%. Compared with the best fine-tuning method in Wu and Dredze (2019) which freezes a selected subset of mBERT layers during fine-tuning, our approach achieves +0.4% higher accuracy on average. We compare our approach with a strong fine-tuning baseline which achieves competitive accuracy scores to the best X-MAML results (Nooralahzadeh et al., 2020) using a single auxiliary language, even though we limit our choice of the auxiliary language to Greek and Urdu, while Nooralahzadeh et al. (2020) select the best auxiliary language among all languages except for the target one. Overall, our approach outperforms the strong fine-tuning

	fr	es	de	ar	ur	bg	sw	th	tr	vi	zh	ru	el	hi	avg
Meta-Optim	<b>75.53</b>	<b>75.93</b>	<b>72.68</b>	<b>67.04</b>	<b>63.33</b>	<b>70.88</b>	<b>51.51</b>	<b>49.89</b>	<b>64.33</b>	<b>72.06</b>	<b>72.36</b>	<b>70.32</b>	<b>70.38</b>	<b>56.29</b>	<b>66.61</b>
No layer-wise update	73.45	73.90	70.73	65.19	60.31	69.10	50.87	46.47	62.74	70.42	70.24	68.85	68.17	53.50	64.57
No cross-lingual meta-train	73.66	74.84	71.54	66.15	61.16	69.33	50.89	48.43	63.16	71.57	70.53	69.14	67.93	55.07	65.24

Table 2: Ablation results on the XNLI dataset using Greek and Urdu as the auxiliary languages (averaged over five runs). Results show that ablating the layer-wise update rate or cross-lingual meta-training degrades accuracy on all target languages.

baseline on 10 out of 14 languages and by +0.2% accuracy on average.

Our approach brings larger gains when using two auxiliary languages – it outperforms the fine-tuning baseline on all languages and improves the average accuracy by +0.6%. This suggests that our meta-learning approach is more effective when transferring from multiple source languages.<sup>1</sup>

### 3.3 Ablation Study

Our approach is different from [Andrychowicz et al. \(2016\)](#) in that 1) it adopts layer-wise update rates while the meta-parameters are shared across all model parameters in [Andrychowicz et al. \(2016\)](#), and 2) it trains the meta-parameters in a cross-lingual setting while [Andrychowicz et al. \(2016\)](#) is designated to few-shot learning. We conduct ablation experiments on XNLI using Greek and Urdu as the auxiliary languages to understand how they contribute to the model performance.

**Impact of Layer-Wise Update Rate** We compare our approach with its variant that replaces the layer-wise update rate with one update rate for all layers. Table 2 shows that our approach significantly outperforms this variant on all target languages with an average margin of 2.0%. This suggests that layer-wise update rate contributes greatly to the effectiveness of our approach.

**Impact of Cross-Lingual Meta-Training** We measure the impact of cross-lingual meta-training by replacing the cross-lingual meta-training in our approach with a joint training of the layer-wise update rate and model parameters. As shown in Table 2, ablating the cross-lingual meta-training

degrades accuracy significantly on all target languages by 1.4% on average, which shows that our cross-lingual meta-training strategy is beneficial.

## 4 Related Work

### 4.1 Cross-lingual Transfer Learning

The idea of cross-lingual transfer is to use the annotated data in the source languages to improve the task performance on the target language with minimal or even zero target labeled data (aka zero-shot). There is a large body of work on using external cross-lingual resources such as bilingual word dictionaries ([Prettenhofer and Stein, 2010](#); [Schuster et al., 2019b](#); [Liu et al., 2020a](#)), MT systems ([Wan, 2009](#)), or parallel corpora ([Eriguchi et al., 2018](#); [Yu et al., 2018](#); [Singla et al., 2018](#); [Conneau et al., 2018](#)) to bridge the gap between the source and target languages. Recent advances in unsupervised cross-lingual representations have paved the road for transfer learning without cross-lingual resources ([Yang et al., 2017](#); [Chen et al., 2018](#); [Schuster et al., 2019a](#)). Our work builds on [Mulcaire et al. \(2019\)](#); [Lample and Conneau \(2019\)](#); [Pires et al. \(2019\)](#) who show that language models trained on monolingual text from multiple languages provide powerful multilingual representations that generalize across languages. Recent work has shown that more advanced techniques such as freezing the model’s bottom layers ([Wu and Dredze, 2019](#)) or continual learning ([Liu et al., 2020b](#)) can further boost the cross-lingual performance on downstream tasks. In this paper, we explore meta-learning to softly select the layers to freeze during fine-tuning.

### 4.2 Meta Learning

A typical meta-learning algorithm consists of two loops of training: 1) an *inner loop* where the learner model is trained, and 2) an *outer loop* where, given a meta-objective, we optimize a set of meta-parameters which controls aspects of the learning process in the inner loop. The

<sup>1</sup>Using two auxiliary languages improves over one auxiliary language the most on lower-resource languages in mBERT pre-training (such as Turkish and Hindi), but does not bring gains or even hurts on high-resource languages (such as French and German). This is consistent with the findings in prior work that the choice of the auxiliary languages is crucial in cross-lingual transfer ([Lin et al., 2019](#)). We leave further investigation on its impact on our meta-learning approach for future work.

goal is to find the optimal meta-parameters such that the inner loop performs well on the meta-objective. Existing meta-learning approaches differ in the choice of meta-parameters to be optimized and the meta-objective. Depending on the choice of meta-parameters, existing work can be divided into four categories: (a) neural architecture search (Stanley and Miikkulainen, 2002; Zoph and Le, 2016; Baker et al., 2016; Real et al., 2017; Zoph et al., 2018); (b) metric-based (Koch et al., 2015; Vinyals et al., 2016); (c) model-agnostic (MAML) (Finn et al., 2017; Ravi and Larochelle, 2016); (d) model-based (learning update rules) (Schmidhuber, 1987; Hochreiter et al., 2001; Maclaurin et al., 2015; Li and Malik, 2017).

In this paper, we focus on model-based meta-learning for zero-shot cross-lingual transfer. Early work introduces a type of networks that can update their own weights (Schmidhuber, 1987, 1992, 1993). More recently, Andrychowicz et al. (2016) propose to model gradient-based update rules using an RNN and optimize it with gradient descent. However, as Wichrowska et al. (2017) point out, the RNN-based meta-optimizers fail to make progress when run for large numbers of steps. They address the issue by incorporating features motivated by the standard optimizers into the meta-optimizer. We instead base our meta-optimizer on a standard optimizer like Adam so that it generalizes better to large-scale training.

Meta-learning has been previously applied to few-shot cross-lingual named entity recognition (Wu et al., 2019), low-resource machine translation (Gu et al., 2018), and improving cross-domain generalization for semantic parsing (Wang et al., 2021). For zero-shot cross-lingual transfer, Nooralahzadeh et al. (2020) introduce an optimization-based meta-learning algorithm called X-MAML which meta-learns the initial model parameters on supervised data from low-resource languages. By contrast, our meta-learning algorithm requires much less meta-parameters and is thus simpler than X-MAML. Bansal et al. (2020) show that MAML combined with meta-learning for learning rates improves few-shot learning. Different from their approach which learns layer-wise learning rates only for task-specific layers specified as a hyper-parameter as part of the MAML algorithm, our approach learns layer-wise learning rates for all layers, and we show the effectiveness of our approach

without being used with MAML on zero-shot cross-lingual transfer.

## 5 Conclusion

We propose a novel meta-optimizer that learns to soft-select which layers to freeze when fine-tuning a pretrained language model (mBERT) for zero-shot cross-lingual transfer. Our meta-optimizer learns the update rate for each layer by simulating the zero-shot transfer scenario where the model fine-tuned on the source languages is tested on an unseen language. Experiments show that our approach outperforms the simple fine-tuning baseline and the X-MAML algorithm on cross-lingual natural language inference.

## References

- Marcin Andrychowicz, Misha Denil, Sergio Gómez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando de Freitas. 2016. [Learning to learn by gradient descent by gradient descent](#). In *Advances in Neural Information Processing Systems*, pages 3981–3989. Curran Associates, Inc.
- Bowen Baker, Otkrist Gupta, Nikhil Naik, and Ramesh Raskar. 2016. Designing neural network architectures using reinforcement learning. *arXiv preprint arXiv:1611.02167*.
- Trapit Bansal, Rishikesh Jha, and Andrew McCallum. 2020. [Learning to few-shot learn across diverse natural language classification tasks](#). In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5108–5123, Barcelona, Spain (Online). International Committee on Computational Linguistics.
- Xilun Chen, Yu Sun, Ben Athiwaratkun, Claire Cardie, and Kilian Weinberger. 2018. [Adversarial deep averaging networks for cross-lingual sentiment classification](#). *Transactions of the Association for Computational Linguistics*, 6:557–570.
- Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. [XNLI: Evaluating cross-lingual sentence representations](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485, Brussels, Belgium. Association for Computational Linguistics.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [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.
- Akiko Eriguchi, Melvin Johnson, Orhan Firat, Hideto Kazawa, and Wolfgang Macherey. 2018. [Zero-shot cross-lingual classification using multilingual neural machine translation](#). *CoRR*, abs/1809.04686.
- Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1126–1135. JMLR. org.
- Jiatao Gu, Yong Wang, Yun Chen, Victor O. K. Li, and Kyunghyun Cho. 2018. [Meta-learning for low-resource neural machine translation](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3622–3631, Brussels, Belgium. Association for Computational Linguistics.
- Jian Guo, He He, Tong He, Leonard Lausen, Mu Li, Haibin Lin, Xingjian Shi, Chenguang Wang, Junyuan Xie, Sheng Zha, Aston Zhang, Hang Zhang, Zhi Zhang, Zhongyue Zhang, Shuai Zheng, and Yi Zhu. 2020. [Gluoncv and gluonnlp: Deep learning in computer vision and natural language processing](#). *Journal of Machine Learning Research*, 21(23):1–7.
- Sepp Hochreiter, A Steven Younger, and Peter R Conwell. 2001. Learning to learn using gradient descent. In *International Conference on Artificial Neural Networks*, pages 87–94. Springer.
- Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. [What does BERT learn about the structure of language?](#) In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657. Association for Computational Linguistics.
- Diederick P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.
- Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. 2015. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2. Lille.
- Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *Advances in Neural Information Processing Systems (NeurIPS)*.
- Ke Li and Jitendra Malik. 2017. Learning to optimize. In *International Conference on Learning Representations*.
- Yu-Hsiang Lin, Chian-Yu Chen, Jean Lee, Zirui Li, Yuyan Zhang, Mengzhou Xia, Shruti Rijhwani, Junxian He, Zhisong Zhang, Xuezhe Ma, Antonios Anastasopoulos, Patrick Littell, and Graham Neubig. 2019. [Choosing transfer languages for cross-lingual learning](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3125–3135, Florence, Italy. Association for Computational Linguistics.
- Zihan Liu, Genta Indra Winata, Zhaojiang Lin, Peng Xu, and Pascale Fung. 2020a. [Attention-informed mixed-language training for zero-shot cross-lingual task-oriented dialogue systems](#). *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8433–8440.
- Zihan Liu, Genta Indra Winata, Andrea Madotto, and Pascale Fung. 2020b. [Exploring fine-tuning techniques for pre-trained cross-lingual models via continual learning](#).
- Dougal Maclaurin, David Duvenaud, and Ryan Adams. 2015. Gradient-based hyperparameter optimization through reversible learning. In *International Conference on Machine Learning*, pages 2113–2122.
- Phoebe Mulcaire, Jungo Kasai, and Noah A. Smith. 2019. [Polyglot contextual representations improve crosslingual transfer](#). 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 3912–3918. Association for Computational Linguistics.
- Farhad Nooralahzadeh, Giannis Bekoulis, Johannes Bjerva, and Isabelle Augenstein. 2020. [Zero-shot cross-lingual transfer with meta learning](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4547–4562, Online. Association for Computational Linguistics.
- Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. [How multilingual is multilingual bert?](#) *CoRR*, abs/1906.01502.
- Peter Prettenhofer and Benno Stein. 2010. [Cross-language text classification using structural correspondence learning](#). In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1118–1127. Association for Computational Linguistics.
- Sachin Ravi and Hugo Larochelle. 2016. Optimization as a model for few-shot learning. In *International Conference on Learning Representations*.
- Esteban Real, Sherry Moore, Andrew Selle, Saurabh Saxena, Yutaka Leon Suematsu, Jie Tan, Quoc V Le, and Alexey Kurakin. 2017. Large-scale evolution of image classifiers. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2902–2911. JMLR. org.
- Jürgen Schmidhuber. 1987. *Evolutionary principles in self-referential learning, or on learning how to learn: the meta-meta-... hook*. Ph.D. thesis, Technische Universität München.

- Jürgen Schmidhuber. 1992. Learning to control fast-weight memories: An alternative to dynamic recurrent networks. *Neural Computation*, 4(1):131–139.
- Jürgen Schmidhuber. 1993. A neural network that embeds its own meta-levels. In *IEEE International Conference on Neural Networks*, pages 407–412. IEEE.
- Sebastian Schuster, Sonal Gupta, Rushin Shah, and Mike Lewis. 2019a. [Cross-lingual transfer learning for multilingual task oriented dialog](#). 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 3795–3805. Association for Computational Linguistics.
- Tal Schuster, Ori Ram, Regina Barzilay, and Amir Globerson. 2019b. [Cross-lingual alignment of contextual word embeddings, with applications to zero-shot dependency parsing](#). 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 1599–1613. Association for Computational Linguistics.
- Karan Singla, Dogan Can, and Shrikanth Narayanan. 2018. [A multi-task approach to learning multilingual representations](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 214–220. Association for Computational Linguistics.
- Kenneth O Stanley and Risto Miikkulainen. 2002. Evolving neural networks through augmenting topologies. *Evolutionary computation*, 10(2):99–127.
- Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. 2016. Matching networks for one shot learning. In *Advances in neural information processing systems*, pages 3630–3638.
- Xiaojun Wan. 2009. [Co-training for cross-lingual sentiment classification](#). In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 235–243. Association for Computational Linguistics.
- Bailin Wang, Mirella Lapata, and Ivan Titov. 2021. [Meta-learning for domain generalization in semantic parsing](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 366–379. Online. Association for Computational Linguistics.
- Olga Wichrowska, Niru Maheswaranathan, Matthew W Hoffman, Sergio Gómez Colmenarejo, Misha Denil, Nando Freitas, and Jascha Sohl-Dickstein. 2017. Learned optimizers that scale and generalize. In *International Conference on Machine Learning*, pages 3751–3760.
- Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. [A broad-coverage challenge corpus for sentence understanding through inference](#). 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 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.
- Qianhui Wu, Zijia Lin, Guoxin Wang, Hui Chen, Börje F Karlsson, Biqing Huang, and Chin-Yew Lin. 2019. Enhanced meta-learning for cross-lingual named entity recognition with minimal resources. *arXiv preprint arXiv:1911.06161*.
- Shijie Wu and Mark Dredze. 2019. [Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 833–844, Hong Kong, China. Association for Computational Linguistics.
- Zhilin Yang, Ruslan Salakhutdinov, and William W. Cohen. 2017. [Transfer learning for sequence tagging with hierarchical recurrent networks](#). In *International Conference on Learning Representations*.
- Katherine Yu, Haoran Li, and Barlas Oguz. 2018. [Multilingual seq2seq training with similarity loss for cross-lingual document classification](#). In *Proceedings of The Third Workshop on Representation Learning for NLP*, pages 175–179. Association for Computational Linguistics.
- Barret Zoph and Quoc V Le. 2016. Neural architecture search with reinforcement learning. *arXiv preprint arXiv:1611.01578*.
- Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V Le. 2018. Learning transferable architectures for scalable image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8697–8710.