Group 11: Transfer Learning and Model Agnostic Meta Learning (MAML) For NLP

1 Aman Singh 2 Boppana Tej Kiran 3 Naga Durga Krishna Mohan Eaty
4 Shivam Aggarwal 5 Aman Aryan
20111010, 20111070, 20111037, 20111058, 20111009
{amansingh20, tejkiranb20, kmohan20, shivama20, ryn20}@iitk.ac.in
Indian Institute of Technology Kanpur (IIT Kanpur)

Abstract

Today the success of Deep learning models in various domains is well known, but one problem associated with these models is they require a massive amount of data. In NLP, most of the data available is in the English language, which makes the task of directly applying deep learning to other low-resource languages very challenging. This paper explored two popular approaches, Transfer Learning (TL) and Model-Agnostic Meta-Learning (MAML), to deal with this problem. These approaches try to share the knowledge from the high-resource domains to the low-resource domains. We will explore the various techniques proposed in both methods and then propose a mini-project to implement some of these techniques in native Indian Languages.

1 Introduction

One main application of machine learning is Natural Language Processing. We can broadly divide NLP into two tasks - Processing and Generation. Processing is the ability of computers to process natural text, and Generation is predicting a word token given a set for previous tokens. Initially, both these tasks were achieved by rule-based systems[1]. Later recurrent neural architectures such as LSTM and GRU and feedforward networks like Transformers are used. These models were able to serve the purpose, but these models need huge compute resources and are data hungry. Lot of time and money can be saved if we could use a trained model directly by tweaking it a bit. That is what Transfer Learning (TL) does. Similar to the analogy of neural networks to the neurons in the brain, we can draw an analogy to transfer learning as "how humans try to learn from different activities and apply that inference to new tasks." In machine learning terminology, TL can be explained as given a source domain D_S and source task T_S as well as a target domain D_T and target task T_T. The objective of transfer learning (TL) is to enable us learn the model for target task T_T in domain D_T with the information gained from D_S and T_S. The advantages that TL brings are, it allows for faster convergence on the target task and the requirement of supervised annotated data is much less compared to training the model from scratch[1]. The motive for trying TL in NLP are results from computer vision and the scarce availability of annotated text for many languages.

Transfer learning can be divided into **Transductive** and **Inductive** transfer learning. In Transductive TL, $T_S = T_T$ but $D_S \neq D_T$. Domain Adaptation and Cross-Lingual learning comes under this category. In Inductive TL, $T_S \neq T_T$ and $D_S = D_T$ or $D_S \neq D_T$. Inductive TL can be further divided in two types Multi-task TL and Sequential TL. Most of the work done in TL is on Cross-Lingual TL and Sequential TL. Cross-Lingual TL is about adapting to a different language in target domain and Sequential TL involves learning multiple tasks in sequential fashion. Fine-tuning, Zero-shot learning, Feature-based learning and Adapter modules come under Sequential TL.

The other popular approach for these tasks is Meta-Learning. Model Agnostic Meta-learning (MAML)[2] is one such optimization-based Meta-learning technique. Meta-Learning is about "Learning how to learn" and model-agnostic means it is compatible with any model based on the Gradient Descent

algorithm. The main objective of the MAML is to train the parameters of the model such that we can get good results with very few examples in a short time for any new task.

For MAML, an entire task T_i is equivalent to a single training example. This task T_i is sampled from a distribution of similar tasks $p(T)$. Training a model using MAML happens in two phases. The first phase is task-specific learning, and the second phase is meta-optimization across tasks.

The training procedure of MAML is as follows: First, we initialize model parameter θ randomly, then for each epoch sample K tasks $T_1, \ldots ...T_K$ from the task distribution $p(T)$. For each sampled task, train a copy of the model on training data D_i^{train} using the gradient descent algorithm.

$$\textbf{Task-specific learning: } \theta_i = \theta - \alpha \nabla_\theta L_i(\theta, D_i^{train}) \ \forall \ i \in [1, k]$$

In the meta-optimization phase, optimize the original parameter θ using the adapted parameters from task-specific learning after **s** gradient steps. In this phase, use the test data for each sampled task D_i^{test}).

$$\textbf{Meta-optimization phase: } \theta = \theta - \beta \nabla_\theta \sum_{I=1}^{K} L(\theta_i^s, D_i^{test})$$

Here α and β are the gradient-based update learning rates. This trained model can be used on any new task sampled from the same distribution $p(T)$. The primary motivation behind MAML is to establish a framework that can be used across various architectures and problem settings. MAML can be applied to classification, regression, policy gradient-based reinforcement learning with very slight changes. The primary application of MAML is in the field of zero-shot, one-shot, and few-shot learning.

In this report we discuss about the Evolution of these techniques**(Section 2)**, there applications**(Section 3)**, Proposed future work**(Section4)**

2 Evolution of Transfer Learning and MAML

In this section, we discuss some of the popular techniques or models in transfer learning and MAML for NLP. These techniques mostly inspire many ideas that are coming up nowadays in some way or other.

2.1 Transfer Learning

ELMo (Embeddings from Language Models): In tasks like machine translation and question answering, we found that instead of training embedding layers from scratch, the use of pre-trained embedding like Word2Vec, GLoVe embedding improves the performance by many folds. But these were not able to model capture context. For this, Matthew E. Peters et al.[3], proposed **ELMo**, which is a feature-based TL technique, it models complex characteristics of words and along with various contexts. It learns the embedding by training a bidirectional Language Model (biLM) and taking representations from the internal states. By intuition, we know that lower layers model low-level things like syntax and semantics, and higher levels tend to grasp the linguistic context information. Embedding is created by training the biLM (bidirectional language model), giving context-independent word token representations as inputs. For usage in downstream tasks, a single vector representation is created, which is a linear combination of all hidden state representations.

UMLFiT (Universal Language Model Fine-tuning): Prior to UMLFiT, while the deep learning models achieved SOTA results in NLP, they need to be still trained from scratch. There were various attempts to crack the Transfer Learning recipe in NLP to match the results that Transfer Learning produced in Computer Vision. However, they were mostly unsuccessful. There were no standard fine tuning techniques. Various ad-hoc fine-tuning techniques borrowed from Computer Vision were applied for TL in NLP. Howard et al.[4] claims that it was not the idea of fine-tuning an LM that was incorrect, but the approach to fine-tuning was not correct. They proposed fine-tuning techniques for classification tasks in NLP which are, (1) The discriminative fine-tuning (Discr.) and slanted triangular learning rates (STLR) for fine-tuning the LM and (2) Gradual Unfreezing for fine-tuning Target task classifier. The discriminative fine-tuning uses different learning rates for each layer, In STLR, the learning rate is increased linearly to a certain extent and decreased linearly until convergence.

GPT (Generative Pre-Training): GPT aims at drawing out higher-level semantics rather than just word-level semantics, which was the case with the previous models. The model has two phases of

training - unsupervised fine-tuning and supervised fine-tuning. Till then, LSTMs are used for language modeling. Alec Radford et al.[5] proposed Transformer-decoder based language model. GPT uses a common language modeling objective for pre-training. In fine-tuning, a linear output layer is added based on the fine-tuning task. The weights of the linear output layer and the transformer model parameters are fine-tuned during fine-tuning, using a standard supervised prediction objective (the probability of output given inputs). GPT gave good results with improvements in GLUE score by 5.5% and in MultiNLI text entailment task by 1.5% and 5.7% in question and answering. Later, GPT-2 and GPT-3 were developed, which uses the same base idea, and these became the current state of the art.

BERT (Bidirectional Encoder Representations from Transformers): It is a language representation model used to model languages and transfer them to other tasks. Jacob Devlin et al.[6] proposed BERT, which is inspired by Transformer. In BERT, we use transformer encoders instead of decoders as in GPT. Novelty in BERT is a bidirectional language representation that models context in both directions, providing better performance. The model is first pre-trained on an unsupervised corpus using two tasks (Masked Language Model and Next Sentence Prediction) and fine-tuned on a supervised task. The main problem that comes with bi-directionality is that the token sees itself during training. In a Masked Language model, we mask certain tokens, and the task of the model is to predict these tokens. For sentence prediction, we use sentence pairs, wherein 50% of pairs are such that sentence B follows sentence A and the rest are any two random sentences in the corpus. The model's task is given a sentence pair predicting whether sentence B follows A or not. The advantage that BERT gives is a unified model with minimal changes for downstream tasks. Some experimental numbers for BERT are an improvement of 7.7% in GLUE score and 4.6% improvement in MultiNLI accuracy, and 1.5% improvement in SQuAD question and answering task.

XLNET: BERT predicts the masked tokens independently, so it doesn't learn how they influence one-another. XLNet[7] is an auto-regressive language model which outputs the joint probability of a sequence of tokens based on the transformer architecture with recurrence. Its training objective calculates the probability of a word token conditioned on all permutations of word tokens in a sentence, as opposed to just those to the left or just those to the right of the target token. Attention is divided into two streams (1) content stream encodes both the context and information from the context. (2) Query stream has access to the contextual information and the position but not the content. To reduce the optimization difficulty, XLNet only predicts the last tokens in a factorization order.

ALBERT: Bert is very intensive in computation both for training and inference. ALBERT[8] is a Transformer architecture based on BERT. It uses a transformer encoder with GELU nonlinearities. The three major contributions are made over design choice of BERT. (1)Factorized embedding parameterization: project into a lower-dimensional embedding space, and then project it to the hidden space by decomposing the large vocabulary embedding matrix into two small matrices. Embedding parameters reduced to $O(V \times E + E \times H)$. (2)Cross-layer parameter sharing: ALBERT shares all parameters across all layers. This prevents the parameters from growing with depth of the network. (3) Inter-sentence coherence loss: a sentence-order prediction (SOP) loss is used, which avoids topic prediction and instead focuses on modelling inter-sentence coherence.

mBERT, XLM: The pre-trained models like ELMo, BERT, GPT have shown capabilities of transferring knowledge learned to specific NLP tasks with limited or no training data. But they can't handle tasks when training and test instances are in different languages. mBERT, XLM are popular attempts made to address this problem to build cross lingual language models that are pre-trained on multi-lingual data sets. mBERT was simultaneously pre-trained on Wikipedia data for 104 languages with impressive performance for zero shot cross lingual transfer NLI task without any cross-lingual supervision. XLM further improves mBERT. It added a supervised learning task unlike in mBERT where there is no provision to pre-train on a supervised task. It is released with the paper [9], which proposes a pre-training strategy in which the model is trained on a supervised task (TLM: Translation LM) coupled with one of the two unsupervised tasks (CLM: Causal LM, MLM: Masked LM). XLM also uses transformer architecture. It obtained a new SOTA of 38.5 BLEU on WMT'16 Romanian-English for NMT.

2.2 MAML

Transfer learning uses representation from the source task on the target task. The learned parameters favor the source task on which the model is pre-trained. **Meta-Learning** is a recent approach in which the model is trained in such a way that it can generalize even on an unseen task. Based on the method of applying, one can broadly categorize Meta-learning into three categories:

Metric-based: This approach aims to learn a metric or similarity between inputs. The learned metric is used to predict the result based on some comparison function. Early work in this category included Siamese Network [10], in which two same networks are joined by an energy function which is L1 distance combined with sigmoid activation. After this, concept of Matching Networks [11] was proposed, which uses cosine distance to measure similarity between input and a set of output classes.It make use of attention and memory components in network architecture. Prototypical Network [12] computes a prototype vector for each class which is the mean of embedding vectors corresponding to class samples. Euclidean distance is used as a metric function to compute similarity. Relation Network [13] uses a relation module(neural network) to learn the similarity between inputs. The metric-based learning does not generalise well when a new task is not related to the previous task.

Model-based: [14] This approach focus on obtaining a model that can be adapted fast to new tasks. Popular models in this approach are Memory Augmented Neural Networks (MANN) and Meta-Nets.

Optimization-based: The optimization-based meta-learning tries to find general initial parameters, which can be fine-tuned on any target task very rapidly using just a few examples. LSTM meta-learner uses LSTM based network to learn optimization for another neural network. This model was very restricted about its architecture and applications (i.e., classification only). Model-Agnostic Meta-Learning (MAML) overcomes the issue of restriction by proposing an algorithm which can be applied to any gradient descent based model. Now we will explore the variants of the MAML algorithm.

MAML: MAML is an easy to implement yet robust algorithm which can capture the learning process during meta-optimization. On the Omniglot dataset, for the 5-way 1-shot and 5-shot classification task, MAML reported an accuracy score of 98.7 ± 0.4% and 99.9 ± 0.1%, respectively, which was better than all previous meta-learning models.

First-order MAML (fo-MAML):[2] MAML meta-optimization step took a gradient of loss w.r.t the initial parameters whereas fo-MAML takes the gradient w.r.t. adapted parameters. This eliminates the computation of Hessian. On Mini-Imagenet, fo-MAML performed equivalent to MAML. The accuracy score obtained by MAML and fo-MAML on (1-shot, 5-shot) are (48.70 ± 1.84%, 63.11 ± 0.92%) and (48.07 ± 1.75% , 63.15 ± 0.91%) respectively.

MAML++ [15]: This is an improvement over the existing MAML. The proposed algorithm uses the Multi-Step loss to deal with instability in MAML, Derivative order Annealing for poor generalization performance, Cosine Annealing for outer loop learning rate, and some other techniques. On Mini-Imagenet for 5-way (1-shot, 5-shot), MAML++ reported an accuracy score of (52.15+0.26%, 68.32+0.44%) and established a new state-of-the-art at that time.

Bayesian-MAML [16]: This paper deals with the issue that MAML tends to overfit in some cases because, in MAML, they try to estimate all the models with a single optimal model. This does not take into account the uncertainty and may result in poor performance in some cases. Thus, a probabilistic variant of MAML is proposed, which is fast and robust to new tasks.

X-MAML:[17] This variant of MAML uses a single high-resource language (h) and a set of low-resource language (L). The model is pre-trained on h and this pre-trained model is meta-trained on some auxiliary language A from L like the original MAML. The intuition behind it was there is only a single high-resource language, English, and the rest of the languages can be used as low-resource tasks. For zero-shot cross-lingual NLI, X-MAML obtained an average accuracy score for all language tasks of 68.98% while Multi-BERT obtained 65.33%, but Multi-Bert outperformed X-MAML in few-shot benchmarks. On MLQA dataset, X-MAML was able to improve the average accuracy score obtained by XLM, $XLM - R_{base}$, $XLM - R_{Large}$.

Tree-MAML:[18] This algorithm tries to exploit the hierarchical task relationships. The intuition is that sharing knowledge across unrelated tasks might hurt performance. The Tree MAML adapts the model using a hierarchical tree structure. In the tree, the top layer contains all tasks, and according to similarity, tasks are clustered down the hierarchy. On the XNLI dataset, the average score by fixed tree MAML is 73.76%. Here the models are trained on all the tasks except the target task. The authors also presented a Learned Tree-MAML (Here, they dynamically learn the tree using the

clustering algorithm), but it fails to perform as good as fixed Tree-MAML.

Meta-Curriculum MAML:[19] In this variant, a curriculum (using sentence level divergence scoring) is created where the easy tasks are fed to the model first, and then the difficulty of tasks gradually increases. The intuition is to learn a similar curriculum for each domain to avoid overfitting to a single domain. So the first initial curricula enhance overall representation learning while the last one focuses on domain-specific knowledge. Meta curriculum performed better than the Meta-MT [20] (NMT using MAML) and performed better on the unseen domain. This gave the average BLEU score of 31.034 while the Meta-MT gave an average BLEU score of 30.465. Meta-Curriculum MAML was able to outperform other models on totally unseen data of COVID-19.

Attentive Task agnostic meta-learning [21]: In ATAML, unlike normal MAML, all parameters are split into two sets, shared parameters and task-specific parameters. During the inner loop, only task-specific parameters are updated. This algorithm was specifically designed for few-shot text classification, and it outperforms MAML significantly on 1-shot learning. On 5-way (1-shot,5-shot) on miniRCV1 dataset, ATAML gave an accuracy of (54.05%, 72.79%) while MAML gave an accuracy of (47.09%, 72.65%).

3 Applications of Transfer learning and MAML in NLP

In this section, we will discuss about some of the attempts made by researchers to apply transfer learning and model agnostic meta learning mainly in cross lingual domain.

One way of doing Cross-lingual transfer learning in NMT is we first pre-train a model on a high-resource parent language and then fine-tune on low-resource child language. This generally requires shared vocabulary between languages, but many languages do not share vocabularies. Yunsu Kim et al.[22] proposed to use cross-word embedding instead of vocabulary. In this, the child language embedding is learned by skip-grams method, and the parent language embedding is taken from pre-trained NMT. Then a mapping between these two is learned by minimizing the L2 norm by inducing artificial noises. We can use this idea if we have a pre-trained English-Hindi NMT and we transfer it to some English-Tamil translation.

Another task in Cross-Lingual setting is Question Generation. Vishwajeet Kumar et al.[23] proposed the idea in which a Transformer based sequence to sequence model with two encoders and decoders, one for each language, is used. In the proposed model, four layer encoders and decoders are used in which the first two are trained for a specific language, and the other two layers shared between languages to enable shared latent space between languages. In pre-training, we use two techniques, denoising auto-encoding and back-translation, for updating the parameters. For fine-tuning, a supervised question and answering dataset is used to fine-tune encoders and decoders separately. This model is experimented with to transfer knowledge from English to Hindi.

Further attempts in cross-lingual domain was made by Haoyang Huang et al.[24]. They developed a universal Language Encoder called Unicoder by Pre-training with Multiple Cross-lingual Tasks. This model is an improvisation to XLM, in which, pre-training is done on a single cross-lingual task with parallel corpora. The idea is more cross lingual tasks can further improve the performance of XLM like models for multi-lingual tasks. So, in addition to MLM & TLM used in XLM, the Unicoder is also pre-trained on three new cross-lingual pre-training tasks. Which are cross-lingual word recovery, cross-lingual paraphrase classification and cross-lingual masked language model. In addition to these pre-training tasks, a multi language fine tuning technique was proposed. In this, fine-tuning is done on source language training data and data translated from target language into source language. Shijie Wu et al.[25] explored the broader cross-lingual potential of mBERT as a zero-shot language transfer model on 5 NLP tasks covering a total of 39 languages from various language families: NLI, document classification, NER, POS tagging, and dependency parsing. They have investigated the most effective strategy for utilizing mBERT and tried to determine to what extent mBERT generalizes away from language-specific features. They observed that it outperforms the models that use cross-lingual embeddings, which typically have more cross-lingual supervision and that language-specific information is preserved in all layers of mBERT and sharing subwords help cross-lingual transfer between languages.

J. Gu et al. [26] proposed to use MAML for low resource NMT. They framed the low-resource translation as a meta-learning problem and generated initial parameters for low resource language

by meta training on high resource language. To avoid mismatch across language pairs, they used Universal-Lexicon Representation(ULR) method. As the underlying NMT, they used Transformer with slight modification to use the ULR.

For the Domain Adaptation task in NMT, META-MT is an approach that uses the first order MAML algorithm for training the NMT systems. It trains the model to adapt to many target domains using as few in-domain data as possible. Meta Curriculum Learning is a recent approach to domain adaptation task. It tries to improve translation on unseen domains too. F. Nooralahzadeh et al. [17] used XMAML for zero-shot cross-lingual NLI and Question Answering. They used BERT pretrained on English as the base model for the zero-shot cross-lingual NLI and XLM model for QA. Using XMAML for cross-lingual transfer can also yield language-agnostic meta parameters.

4 Future Directions

On surveying TL and MAML, we observed that not much cross-lingual work has been done for Indian Languages. Inspired by the effectiveness of MAML in zero-shot scenarios in cross-lingual tasks, we decided to apply MAML for cross-lingual NLI on Indian Languages. For this task, we require context hypothesis pairs in multiple Indian Languages, which are not available. So, we have downloaded Hindi-NLI-data from Midas-Research [27] [1] . To prepare the hypothesis context pairs for other languages, we will use Google AutoML. For each language, we will sample the N tasks with 5K sentences in each task. We plan to use pre-trained BERT for meta-learning on training tasks and perform zero-shot and few-shot on target languages. The training tasks and the target task will be disjoint. We have formulated the problem statement uptill now.

5 Conclusion

In this survey, we explored different TL and MAML techniques applied in NLP. We conclude that similar to computer vision TL and MAML improved the performance on NLP tasks.The main advantage that transfer learning and MAML give, in general, is the amount of fine-tuning that is required. In the cross-lingual domain, techniques mainly focus on learning representations in a shared latent space instead of vocabularies. The main observation in MAML is that all the applications use the first-order MAML, which is computationally efficient. MAML generally performs better than TL for zero-shot scenarios as it learns better generalization, whereas, in the presence of abundant data, TL outperforms MAML.Using the learnings, we proposed an implementation technique for a cross-lingual language inference task on Indian languages.

6 Individual Contributions

Name	Contribution in Survey	Project Implementation
Aman Singh	Xlnet, Albert, Types of TL, QA, Survey on TL, NLI, Text classification	Data Generation and Pre-Processing
Tej Kiran	ULMFiT, RoBERTa, XLM, TL in low resource multi-lingual domain	BERT for NLI task
Krishna Mohan	BERT, GPT, ELMo, Cross-Lingual TL techniques, Survey on TL	BERT for NLI task
Shivam Aggarwal	MAML++, Bayesian MAML, X-MAML, ATAML, few shot MAML techniques	Meta-training and testing
Aman Aryan	MAML, XMAML, TreeMAML, MetaCurriculum Learning, MAML in Few Shot NMT, Domain Adaptation, Cross-Lingual Tasks.	Meta-training and testing

[1] https://github.com/midas-research/hindi-nli-data

References

[1] A. Malte and P. Ratadiya, "Evolution of transfer learning in natural language processing," *arXiv preprint arXiv:1910.07370*, 2019.

[2] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *International Conference on Machine Learning*, pp. 1126–1135, PMLR, 2017.

[3] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," *arXiv preprint arXiv:1802.05365*, 2018.

[4] J. Howard and S. Ruder, "Universal language model fine-tuning for text classification," *arXiv preprint arXiv:1801.06146*, 2018.

[5] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," 2018.

[6] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[7] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, and Q. V. Le, "Xlnet: Generalized autoregressive pretraining for language understanding," *arXiv preprint arXiv:1906.08237*, 2019.

[8] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," *arXiv preprint arXiv:1909.11942*, 2019.

[9] G. Lample and A. Conneau, "Cross-lingual language model pretraining," *arXiv preprint arXiv:1901.07291*, 2019.

[10] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *ICML deep learning workshop*, vol. 2, Lille, 2015.

[11] O. Vinyals, C. Blundell, T. Lillicrap, K. Kavukcuoglu, and D. Wierstra, "Matching networks for one shot learning," *arXiv preprint arXiv:1606.04080*, 2016.

[12] J. Snell, K. Swersky, and R. S. Zemel, "Prototypical networks for few-shot learning," *arXiv preprint arXiv:1703.05175*, 2017.

[13] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1199–1208, 2018.

[14] W. Yin, "Meta-learning for few-shot natural language processing: A survey," *arXiv preprint arXiv:2007.09604*, 2020.

[15] A. Antoniou, H. Edwards, and A. Storkey, "How to train your maml," *arXiv preprint arXiv:1810.09502*, 2018.

[16] J. Yoon, T. Kim, O. Dia, S. Kim, Y. Bengio, and S. Ahn, "Bayesian model-agnostic meta-learning," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 7343–7353, 2018.

[17] F. Nooralahzadeh, G. Bekoulis, J. Bjerva, and I. Augenstein, "Zero-shot cross-lingual transfer with meta learning," *arXiv preprint arXiv:2003.02739*, 2020.

[18] J. R. Garcia, F. Freddi, F.-T. Liao, J. McGowan, T. Nieradzik, D.-s. Shiu, Y. Tian, and A. Bernacchia, "Meta-learning with maml on trees," *arXiv preprint arXiv:2103.04691*, 2021.

[19] R. Zhan, X. Liu, D. F. Wong, and L. S. Chao, "Meta-curriculum learning for domain adaptation in neural machine translation," *arXiv preprint arXiv:2103.02262*, 2021.

[20] A. Sharaf, H. Hassan, and H. Daumé III, "Meta-learning for few-shot nmt adaptation," *arXiv preprint arXiv:2004.02745*, 2020.

[21] X. Jiang, M. Havaei, G. Chartrand, H. Chouaib, T. Vincent, A. Jesson, N. Chapados, and S. Matwin, "On the importance of attention in meta-learning for few-shot text classification," *arXiv preprint arXiv:1806.00852*, 2018.

[22] Y. Kim, Y. Gao, and H. Ney, "Effective cross-lingual transfer of neural machine translation models without shared vocabularies," *arXiv preprint arXiv:1905.05475*, 2019.

[23] V. Kumar, N. Joshi, A. Mukherjee, G. Ramakrishnan, and P. Jyothi, "Cross-lingual training for automatic question generation," *arXiv preprint arXiv:1906.02525*, 2019.

[24] H. Huang, Y. Liang, N. Duan, M. Gong, L. Shou, D. Jiang, and M. Zhou, "Unicoder: A universal language encoder by pre-training with multiple cross-lingual tasks," *arXiv preprint arXiv:1909.00964*, 2019.

[25] S. Wu and M. Dredze, "Beto, bentz, becas: The surprising cross-lingual effectiveness of bert," *arXiv preprint arXiv:1904.09077*, 2019.

[26] J. Gu, Y. Wang, Y. Chen, K. Cho, and V. O. Li, "Meta-learning for low-resource neural machine translation," *arXiv preprint arXiv:1808.08437*, 2018.

[27] S. Uppal, V. Gupta, A. Swaminathan, H. Zhang, D. Mahata, R. Gosangi, R. R. Shah, and A. Stent, "Two-step classification using recasted data for low resource settings," in *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, (Suzhou, China), pp. 706–719, Association for Computational Linguistics, Dec. 2020.

[28] D. W. Otter, J. R. Medina, and J. K. Kalita, "A survey of the usages of deep learning for natural language processing," *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[29] Y. Peng, S. Yan, and Z. Lu, "Transfer learning in biomedical natural language processing: an evaluation of bert and elmo on ten benchmarking datasets," *arXiv preprint arXiv:1906.05474*, 2019.

[30] Z. Chi, L. Dong, F. Wei, W. Wang, X.-L. Mao, and H. Huang, "Cross-lingual natural language generation via pre-training," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, pp. 7570–7577, 2020.

[31] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[32] B. Zoph, D. Yuret, J. May, and K. Knight, "Transfer learning for low-resource neural machine translation," *arXiv preprint arXiv:1604.02201*, 2016.

[33] J.-K. Kim, Y.-B. Kim, R. Sarikaya, and E. Fosler-Lussier, "Cross-lingual transfer learning for pos tagging without cross-lingual resources," in *Proceedings of the 2017 conference on empirical methods in natural language processing*, pp. 2832–2838, 2017.

[34] Z. Alyafeai, M. S. AlShaibani, and I. Ahmad, "A survey on transfer learning in natural language processing," *arXiv preprint arXiv:2007.04239*, 2020.

[35] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *International conference on artificial neural networks*, pp. 270–279, Springer, 2018.

[36] S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning, "A large annotated corpus for learning natural language inference," *arXiv preprint arXiv:1508.05326*, 2015.

[37] Y.-A. Chung, H.-Y. Lee, and J. Glass, "Supervised and unsupervised transfer learning for question answering," *arXiv preprint arXiv:1711.05345*, 2017.

[38] Z. Shaheen, G. Wohlgenannt, and E. Filtz, "Large scale legal text classification using transformer models," *arXiv preprint arXiv:2010.12871*, 2020.

[39] R. Puri and B. Catanzaro, "Zero-shot text classification with generative language models," *arXiv preprint arXiv:1912.10165*, 2019.

[40] Y. Kim, P. Petrov, P. Petrushkov, S. Khadivi, and H. Ney, "Pivot-based transfer learning for neural machine translation between non-english languages," *arXiv preprint arXiv:1909.09524*, 2019.

[41] M. A. Hedderich, L. Lange, H. Adel, J. Strötgen, and D. Klakow, "A survey on recent approaches for natural language processing in low-resource scenarios," *arXiv preprint arXiv:2010.12309*, 2020.

[42] Z. Liu, R. Zhang, Y. Song, and M. Zhang, "When does maml work the best? an empirical study on model-agnostic meta-learning in nlp applications," *arXiv preprint arXiv:2005.11700*, 2020.

[43] S. Deng, N. Zhang, Z. Sun, J. Chen, and H. Chen, "When low resource nlp meets unsupervised language model: Meta-pretraining then meta-learning for few-shot text classification (student abstract)," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, pp. 13773–13774, 2020.

[44] T. Bansal, R. Jha, and A. McCallum, "Learning to few-shot learn across diverse natural language classification tasks," *arXiv preprint arXiv:1911.03863*, 2019.

[45] E. Triantafillou, T. Zhu, V. Dumoulin, P. Lamblin, U. Evci, K. Xu, R. Goroshin, C. Gelada, K. Swersky, P.-A. Manzagol, *et al.*, "Meta-dataset: A dataset of datasets for learning to learn from few examples," *arXiv preprint arXiv:1903.03096*, 2019.

[46] Z.-Y. Dou, K. Yu, and A. Anastasopoulos, "Investigating meta-learning algorithms for low-resource natural language understanding tasks," *arXiv preprint arXiv:1908.10423*, 2019.