Surprisingly Simple Adapter Ensembling for Zero-Shot Cross-Lingual Sequence Tagging

Anonymous ACL submission

Abstract

Adapters are parameter-efficient modules added to pretrained Transformer models that facilitate cross-lingual transfer. Language adapters and task adapters can be separately trained and zero-shot transfer is enabled by pairing the language adapter in the target language with a task adapter trained on a high-resource language. However, there are many languages and dialects for which training language adapters would be difficult. In this work, we present a simple and efficient ensembling technique to transfer task knowledge to unseen target languages for which no language adapters exist. We compute a uniformly-weighted ensemble model over the top language adapters based on how well they perform on the test set of a high-resource language. We outperform the state-of-the-art model for this specific setting on named entity recognition (NER) and part-of-speech tagging (POS), across nine typologically diverse languages with relative performance improvements of up to 29% and 9% on NER and POS, respectively, on select target languages.

1 Introduction

Multilingual pretrained models have been established as a powerful first step towards cross-lingual NLP (Devlin et al., 2019; Conneau et al., 2020). A major appeal of these models is that they can bootstrap NLP tasks in very low-resource languages via zero-shot transfer (Wu and Dredze, 2019; Pires et al., 2019; Hsu et al., 2019). A dominant paradigm in zero-shot cross-lingual transfer is to finetune a multilingual model using task-specific data in a high-resource language before evaluating on the unseen target languages. *Adapter modules* (Rebuffi et al., 2017; Houlsby et al., 2019; Pfeiffer et al., 2020a,b, 2021) have recently emerged as another effective technique for zero-shot transfer. Adapters are new layers interspersed within the layers of the pretrained models. Only these new layers are fine-tuned while the weights of the original pretrained model are kept frozen, thus enabling efficient parameter sharing between tasks and languages with the help of task-specific and language-specific adapters.

Pfeiffer et al. (2020a) propose zero-shot transfer using adapters by stacking language-specific adapters (trained on unlabeled text) with task-specific adapters (trained on labeled data). This technique requires a language adapter for every test language which may not exist for a large fraction of the world's languages. Our main motivation is to improve zero-shot cross-lingual performance for such languages that do not have language adapters.

In recent work, Wang et al. (2021b) addressed this specific setting of zero-shot transfer to languages without any language adapters using a learnable weighted ensemble of related language adapters called Entropy Minimized Ensemble of Adapters (EMEA). Ensemble weights were learned for each test instance to minimize the entropy of the output distribution from the ensembled model. They found even simple ensembling with uniform weights to be effective on cross-lingual sequence tagging tasks and EMEA offered further improvements over vanilla ensembling. However, EMEA is costly at inference time due to the ensemble weight computations for each test instance.

In this work, we present a surprisingly simple and efficient ensembling strategy with no test-time computations that performs at par or outperforms EMEA on a diverse set of target languages. For a given task, the key idea is to evaluate all existing language adapters on a test set of a high-resource or related language, sort them in descending order of performance and pick the top few language adapters for our ensemble. This simple strategy performs surprisingly well. We also offer many supporting empirical analyses to further demonstrate the value of our ensembling techniques.

2 Our Adapter Ensembling Techniques

Our ensembling techniques are built on top of the MAD-X framework (Pfeiffer et al., 2020a,b) that we briefly describe below.

Adapters for Zero-Shot Transfer The MAD-X framework (Pfeiffer et al., 2020b) introduced language and task adapters as lightweight modules that are inserted within a pretrained multilingual model $\mathcal{M}$. MAD-X supports multiple tasks in multiple languages by passing the outputs of each layer of $\mathcal{M}$, denoted by h, through a language adapter $\mathcal{L}$ and a task adapter $\mathcal{T}$ to give $\mathcal{T}(\mathcal{L}(h))$. The resulting model is written as $\mathcal{T} \circ \mathcal{L} \circ \mathcal{M}$. For cross-lingual transfer from a source language L_{src} to a target language L_{tgt}, MAD-X adopts the following two-step approach. First, the models $\mathcal{L}_{\text{src}} \circ \mathcal{M}$ and $\mathcal{L}_{\text{tgt}} \circ \mathcal{M}$ are trained on unlabeled text in L_{src} and L_{tgt}, respectively, using the masked language modeling objective. Next, $\mathcal{T}$ is trained on labeled task data in L_{src} using the cascaded model $\mathcal{T} \circ \mathcal{L}_{\text{src}} \circ \mathcal{M}$. Finally, $\mathcal{T} \circ \mathcal{L}_{\text{tgt}} \circ \mathcal{M}$ can be used for zero-shot transfer to L_{tgt}.

Our goal is to adapt $\mathcal{M}$ to a new target language L_{new} that does not have a language adapter. Our ensembling techniques are all based on a simple averaging of outputs from a set of language adapters, $\mathcal{S} = \{\mathcal{L}_1, \ldots, \mathcal{L}_n\}$. That is, h of each layer in $\mathcal{M}$ is transformed as $\frac{1}{n}\sum_{i=1}^{n} \mathcal{L}_i(h)$. Our ensemble is fixed across all target languages and does not incur any test-time computations. Next, we discuss different strategies to choose $\mathcal{S}$.

ENSEMBLE-ALL. Wang et al. (2021b) advocate the use of languages that are perceived to be related to L_{new} for their ensembles. We argue this may not be an optimal strategy since it precludes the use of other (unrelated) language adapters that are well-trained and might potentially help L_{new}. Also, the presence of a task adapter trained on L_{src} in the model makes it unclear as to whether the chosen adapter languages should be similar to L_{src} or L_{new}. We first opt for the easiest choice of using an ensemble of all language adapters available on AdapterHub (Pfeiffer et al., 2020a). However, this is expensive in terms of memory and averages over a large number of adapters. The next two strategies aim at meaningfully reducing the size of $\mathcal{S}$.

EN-10. It is conceivable that there are certain high-performing language adapters that can be effective across all targets. In order to identify these "good" language adapters, for every available language adapter $\mathcal{L}_i$, we evaluate $\mathcal{T} \circ \mathcal{L}_i \circ \mathcal{M}$ on an English test set. We sort the adapters $\mathcal{L}_i$ in decreasing order of their performance and select the top K for our ensemble set $\mathcal{S}$. (We find $K = 10$ to be a good choice. More details are in Section 4.)

REL-10. Rather than evaluating on an English test set, evaluating on a language L_{rel} that is similar to L_{new} may be a better proxy for performance on L_{new}. Thus, we also select the top K language adapters for $\mathcal{S}$ based on their performance on a test set in L_{rel}. L_{rel} is identified a priori based on linguistic knowledge of the language and its relation to L_{new} (as was done in Wang et al. (2021b)).

3 Experimental Setup

Tasks and Datasets. We perform experiments on two tasks: Named entity recognition (NER) and Part-of-Speech tagging (POS). We use the WikiAnn dataset (Pan et al., 2017) for NER and Universal Treebank 2.0 (Nivre et al., 2018) for POS tagging. We report F1 scores averaged over 3 random seeds for all our experiments.

Model. We use the mBERT (Devlin et al., 2019) base model for all our experiments. We use pretrained language adapters from AdapterHub (Pfeiffer et al., 2020a). To train the task adapters and the EMEA ensembles, we use the hyperparameters specified in Wang et al. (2021b). Appendix C lists more implementation details.

Languages. We use the same three groups of languages listed in Wang et al. (2021b). Group 1 has Marathi (mr), Tamil (ta), Bengali (bn) and Bhojpuri (bho); Group 2 has Faroese (fo), Norwegian (no), Danish(da)3; and, Group 3 has Belarussian (be), Ukranian (uk) and Bulgarian (bg). Related languages for each group are Hindi (hi), Icelandic (is) and Russian (ru), and we also use Arabic(ar) and German (de) as additional adapters for the first and second groups, respectively. For our ensembles, we consider 45 pretrained language adapters available on AdapterHub (excluding Bengali and Bhojpuri that appear as target languages).

Baselines. We reproduce the following baselines from Wang et al. (2021b)[1] :1. EN: English language adapter. 2. RELATED: Single related language adapter. 3. ENSEMBLE-REL: Ensemble of

[1]We observe very high variance in F1s across random seeds for certain languages. This leads to the difference with reported numbers in Wang et al. (2021b), although the overall trends remain the same. E.g., our ta scores are much worse for NER and far better for POS compared to Wang et al. (2021b).

Table 1: Averaged F1 scores for POS tagging and NER. Best scores for each target language are highlighted in bold.

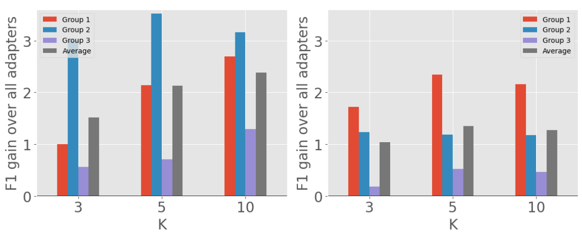
Task	Method	MR	BN	TA	FO	NO	DA	BE	UK	BG	AVG
NER	En	44.6	51.7	22.9	61.9	72.7	79.5	60.3	57.5	68.2	57.7
	Related	45.6	41.5	18.6	59.8	69.8	72.9	61.1	52.9	66.7	54.3
	Ensemble-Rel	51.7	51.5	28.7	63.3	73.9	79.6	65.5	58.2	71.1	60.4
	EMEA-1	53.0	56.2	30.1	64.9	74.0	80.1	66.6	59.6	72.1	61.8
	EMEA-10	**54.2**	57.4	31.2	**65.1**	**74.1**	**80.5**	67.1	**60.6**	73.1	62.6
	Ensemble-All	49.9	59.9	37.2	55.9	72.6	78.2	67.0	57.1	72.3	61.1
	En-10	49.8	62.4	38.9	62.5	73.6	79.2	67.0	57.3	73.3	62.7
	Rel-10	51.8	**63.0**	**40.3**	62.8	73.8	79.5	**67.7**	58.9	**73.6**	**63.5**

Task	Method	MR	BHO	TA	FO	NO	DA	BE	UK	BG	AVG
POS	En	62.7	39.6	61.6	73.7	84.7	87.8	80.1	81.4	84.7	72.9
	Related	53.9	**46.6**	56.4	73.5	77.4	82.9	76.1	76.5	80.5	69.3
	Ensemble-Rel	64.0	45.6	61.8	75.2	84.0	88.1	81.2	81.4	84.7	74.0
	EMEA-1	64.4	45.7	62.4	**75.3**	83.9	88.1	81.1	81.3	84.7	74.1
	EMEA-10	65.2	45.4	63.1	75.2	84.1	88.2	81.4	81.4	84.9	74.3
	Ensemble-All	64.8	43.5	67.7	72.6	84.2	88.1	81.9	81.8	84.9	74.4
	En-10	**68.6**	45.0	**68.5**	74.3	84.8	88.1	82.1	82.1	85.2	75.4
	Rel-10	67.9	46.3	68.2	**75.3**	**84.9**	**88.3**	**82.4**	**82.2**	**85.4**	**75.7**

an English adapter, a related language adapter and additional adapters (as listed in Wang et al. (2021b), if available). 4. EMEA-1/EMEA-10: One or ten steps of test-time entropy minimization applied to the ensemble in Ensemble-Rel.

4 Results

Our main results are listed in Table 1. En-10 is consistently better than EMEA-10 on POS tagging for most of the target languages, with the highest improvement obtained for ta. Rel-10 further improves over En-10 with small but consistent improvements on POS tagging. (We note an advantage of En-10 in that it is entirely agnostic of the target language, unlike Rel-10 that requires a related language.) For the NER task, the Indian language group of mr, bn and ta is most benefited overall by Rel-10 compared to EMEA-10 and F1 scores on most of the other target languages using Rel-10 are comparable to that obtained using EMEA-10.



(a) Named Entity Recognition (b) Part of Speech Tagging

Figure 1: Improvement over Ensemble-All using different ensemble sizes K with Rel-K.

Varying the ensemble size. Figure 1 shows the gain in averaged F1 scores for the three language groups over Ensemble-All, for three different values of K. Considering the overall average F1 scores, $K = 10$ is the best setting for NER and $K = 5$ and $K = 10$ are comparable for POS. Given these trends, we set $K = 10$ for all subsequent experiments.

Changing the task adapter. We verify whether our ensembling technique helps if we had access to a task adapter trained on a related language (rather than English). Table 2 shows F1 scores for POS of group 1 languages using a Hindi task adapter. Hi Top 10 clearly outperforms the other two ensembling techniques based on average F1 scores.

Evaluating different ensembling techniques. In order to disentangle the importance of ensembling from the importance of choosing source language adapters, we examine how performance varies using different ensembling techniques in Table 3. Ensemble-Rand-10 uses 10 randomly chosen language adapters and Ensemble-LV-10 picks the top 10 language adapters based on simi-

Table 2: F1 scores for POS tagging using a Hindi task adapter and different ensembling techniques.

Method	MR	BHO	TA	AVG
En Task + hi top 10	68.1	46.6	68.3	61.0
Hi Task + en,hi,ar	63.7	**53.8**	67.9	61.8
Hi Task + en top 10	66.9	52.7	70.4	63.3
Hi Task + hi top 10	**68.5**	52.9	**71.1**	**64.2**

Table 3: F1 scores for POS and NER tasks using different ensembling techniques.

TASK	METHOD	MR	BN	TA	FO	NO	DA	BE	UK	BG	AVG
NER	ENSEMBLE-ALL	49.9	59.9	37.2	55.9	72.6	78.2	67.0	57.1	72.3	61.1
	ENSEMBLE-RAND-10 10	47.7	56.9	35.9	57.3	72.1	77.7	66.2	57.3	71.5	59.9
	ENSEMBLE-LV-10	50.2	57.3	38.0	58.6	**74.0**	79.0	67.4	57.8	72.6	61.7
	EN-10	49.8	62.4	38.9	62.5	73.6	79.2	67.0	57.3	73.3	62.7
	REL-10	**51.8**	**63.0**	**40.3**	**62.8**	73.8	**79.5**	**67.7**	**58.9**	**73.6**	**63.5**

TASK	METHOD	MR	BHO	TA	FO	NO	DA	BE	UK	BG	AVG
POS	ENSEMBLE-ALL	64.8	43.5	67.7	72.6	84.2	88.1	81.9	81.8	84.9	74.4
	ENSEMBLE-RAND-10 10	64.5	43.5	66.5	72.9	83.9	88.2	81.8	81.6	85.0	74.2
	ENSEMBLE-LV-10	67.4	45.2	67.9	73.6	84.1	88.1	82.1	82.0	85.0	75.0
	EN-10	**68.6**	45.0	**68.5**	74.3	84.8	88.1	82.1	82.1	85.2	75.4
	REL-10	67.9	**46.3**	68.2	**75.3**	**84.9**	**88.3**	**82.4**	**82.2**	**85.4**	**75.6**

larity between geographical vectors corresponding to the target and source languages (Littell et al., 2017). We observe that our proposed ensembling techniques outperform the others on (almost) all target languages for both POS and NER.

5 Related Work

Pfeiffer et al. (2020a,b) introduces the MAD-X framework for NLP tasks and creates a repository of pretrained language and task adapters that enable cross-lingual transfer. In this work, we focus on zero-shot transfer to target languages for which even language adapters do not exist. Wang et al. (2021b) focuses on the very same setting and serves as our main comparison. They draw inspiration from test-time adaptation techniques (Wang et al., 2021a) and ensemble over language adapters at test time using learned ensemble weights for each test instance. These test time computations significantly add to the inference cost. In contrast, our simple ensembling techniques do not require costly test-time computations and yield superior performance on both POS and NER tasks. Our work adds to the existing literature on factors that impact or limit zero-shot transfer (Lin et al., 2019; Lauscher et al., 2020; Turc et al., 2021).

6 Discussion and Conclusion

We identify a *core set* of common language adapters appearing in the top-10 lists of en, hi, is and ru. Figure 2 visually displays the languages that appear in all four lists; nine of the seventeen languages appear in three or more lists. We conjecture that, along with the related language, it is important to ensemble over this core set of language adapters. These adapters perform well across target languages, regardless of how they relate to the target, owing to various reasons such as size and diversity of data used to train the language adapters (Lin et al., 2019). (Appendix A elaborates on an experiment using a core set.)

The main limitation of EMEA is its slow inference speed. REL-10 is significantly faster then EMEA: With a batch size of 1, REL-10 processes 26.3 examples/second, as opposed to just 6.67 and 0.86 examples/second by EMEA-1 and EMEA-10, respectively. Further, Wang et al. (2021b) observed that the performance of EMEA-10 decays with increasing batch size, while REL-10 has no such limitation. With a batch size of 32, REL-10 processes as many as 110 examples per second. (Appendix B shows how EMEA-1 and EMEA-10 could be used with the ensembles identified by REL-10 to further improve performance.).

Future Work. While we present a simple ensembling technique, we do not yet have a clear understanding of why the “core set" of language adapters performs well on most target languages. This knowledge would help in training more high-performing language adapters. We leave this important question for future work.

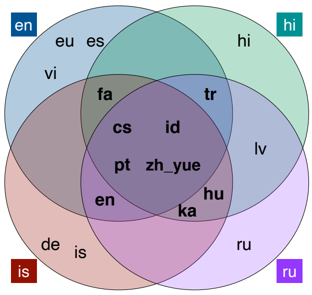


Figure 2: Visualization of the top ten language adaptors for $L_{\text{rel}} \in \{\text{en,hi,is,ru}\}$. Note the significant overlap in language adapters across the four choices of L_{rel}.

References

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. 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.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.

Tsung-Yuan Hsu, Chi-Liang Liu, and Hung-yi Lee. 2019. Zero-shot reading comprehension by cross-lingual transfer learning with multi-lingual language representation model. 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 5933–5940, Hong Kong, China. Association for Computational Linguistics.

Anne Lauscher, Vinit Ravishankar, Ivan Vulić, and Goran Glavaš. 2020. From zero to hero: On the limitations of zero-shot language transfer with multilingual Transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4483–4499, Online. Association for Computational Linguistics.

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.

Patrick Littell, David R. Mortensen, Ke Lin, Katherine Kairis, Carlisle Turner, and Lori Levin. 2017. URIEL and lang2vec: Representing languages as typological, geographical, and phylogenetic vectors. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 8–14, Valencia, Spain. Association for Computational Linguistics.

Joakim Nivre, Rogier Blokland, Niko Partanen, and Michael Rießler. 2018. Universal dependencies 2.2.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1958, Vancouver, Canada. Association for Computational Linguistics.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2021. AdapterFusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 487–503, Online. Association for Computational Linguistics.

Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulić, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. 2020a. AdapterHub: A Framework for Adapting Transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020): Systems Demonstrations*, pages 46–54, Online. Association for Computational Linguistics.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020b. MAD-X: An Adapter-Based Framework for Multi-Task Cross-Lingual Transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7654–7673, Online. Association for Computational Linguistics.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Sylvestre-Alvise Rebuffi, Hakan Bilen, and Andrea Vedaldi. 2017. Learning multiple visual domains with residual adapters. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Iulia Turc, Kenton Lee, Jacob Eisenstein, Ming-Wei Chang, and Kristina Toutanova. 2021. Revisiting the primacy of english in zero-shot cross-lingual transfer. *arXiv preprint arXiv:2106.16171*.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. 2021a. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*.

Xinyi Wang, Yulia Tsvetkov, Sebastian Ruder, and Graham Neubig. 2021b. Efficient test time adapter ensembling for low-resource language varieties. In

Findings of the Association for Computational Linguistics: EMNLP 2021, pages 730–737, Punta Cana, Dominican Republic. Association for Computational Linguistics.

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.

A Ensembling over a core set

To investigate the idea of a core set of language adapters, we introduce a new method, ENSEMBLE-CORE. We select adapters that perform well consistently across all 4 source languages: en,hi,is,ru. We first normalize the F1 scores in each ranked list to lie between 0 and 1 such that the best adapter gets a score of 1 and the worst gets a score of 0. We then add the normalized scores from each source language for a given adapter, and rank the adapters in decreasing order of cumulative score. In our experiments, we use an ensemble of the top 9 adapters from this list (fixed across target groups), and include the related language as the tenth adapter for each group. From Table 4, the F1 scores using the above-mentioned core set of language adapters are very comparable to those obtained using REL-10.

B EMEA with the ensembles identified by REL-10

Table 5 shows the results with learning ensemble weights using EMEA-1 and EMEA-10 on the ensemble of adapters chosen by REL-10. We choose K=10 for both POS and NER based on the results shown in Fig. 1. We find that the F1 scores using EMEA with REL-10 are marginally better than REL-10 alone.

C Implementation Details

All the experiments were run on an NVIDIA 11Gb GeForce GTX 1080 Ti. The NER task adapter was trained for 100 epochs and the POS adapter was trained for 50 epochs. In both cases, we use a learning rate of 1e-4 and an effective batch size of 32. We choose the best model checkpoint based on performance on a dev set. For EMEA, we use a learning rate of $\gamma = 10$. These are the same hyperparameters specified by (Wang et al., 2021b). We use the code shared by Wang et al. (2021b)[2] to reproduce all the baseline numbers.

[2] https://github.com/cindyxinyiwang/emea

Table 4: Comparison of ENSEMBLE-CORE with REL-10

TASK	METHOD	MR	BN	TA	FO	NO	DA	BE	UK	BG	AVG
NER	REL-10	51.8	63.0	**40.3**	**62.8**	73.8	**79.5**	**67.7**	**58.9**	**73.6**	**63.5**
	ENSEMBLE-CORE	**51.9**	**63.6**	39.6	61.7	**74.0**	79.4	67.4	58.2	73.4	63.2
TASK	METHOD	MR	BHO	TA	FO	NO	DA	BE	UK	BG	AVG
POS	REL-10	67.9	46.3	68.2	**75.3**	**84.9**	**88.3**	**82.4**	**82.2**	**85.4**	75.6
	ENSEMBLE-CORE	**68.5**	**46.9**	**68.5**	75.2	84.9	88.2	82.3	82.1	85.3	**75.8**

Table 5: Performance of EMEA-1 and EMEA-10 when used in conjunction with REL-10.

TASK	METHOD	MR	BN	TA	FO	NO	DA	BE	UK	BG	AVG
NER	EMEA-1	53.0	56.2	30.1	64.9	74.0	80.1	66.6	59.6	72.1	61.8
	EMEA-10	**54.2**	57.4	31.2	**65.1**	**74.1**	**80.5**	67.1	60.6	73.1	62.6
	REL-10	51.8	63.0	40.3	62.8	73.8	79.5	67.7	58.9	73.6	63.5
	REL-10 + EMEA-1	51.9	65.0	40.8	63.6	74.0	79.9	68.3	60.1	73.9	64.2
	REL-10 + EMEA-10	52.6	**66.1**	**41.5**	63.9	74.0	80.3	**68.9**	**61.9**	**74.6**	**64.9**
TASK	METHOD	MR	BHO	TA	FO	NO	DA	BE	UK	BG	AVG
POS	EMEA-1	64.4	45.7	62.4	75.3	83.9	88.1	81.1	81.3	84.7	74.1
	EMEA-10	65.2	45.4	63.1	75.2	84.1	88.2	81.4	81.4	84.9	74.3
	REL-10	67.9	46.3	68.2	75.3	**84.9**	88.3	82.4	**82.2**	**85.4**	75.6
	REL-10 + EMEA-1	68.0	**46.5**	68.0	75.2	84.8	**88.4**	82.4	**82.2**	**85.4**	75.7
	REL-10 + EMEA-10	**69.2**	46.1	**68.6**	**75.4**	84.7	88.3	**82.5**	**82.2**	**85.4**	**75.8**