Phylogeny-Inspired Adaptation of Multilingual Models to New Languages

Fahim Faisal, Antonios Anastasopoulos
Department of Computer Science, George Mason University
{ffaisal,antonis}@gmu.edu

Abstract

Large pretrained multilingual models, trained on dozens of languages, have delivered promising results due to cross-lingual learning capabilities on a variety of language tasks. Further adapting these models to specific languages, especially ones unseen during pre-training, is an important goal toward expanding the coverage of language technologies. In this study, we show how we can use language phylogenetic information to improve cross-lingual transfer leveraging closely related languages *in a structured, linguistically-informed manner*. We perform adapter-based training on languages from diverse language families (Germanic, Uralic, Tupian, Uto-Aztecan) and evaluate on both syntactic and semantic tasks, obtaining more than 20% relative performance improvements over strong commonly used baselines, especially on languages unseen during pre-training. [1]

1 Introduction

Language models have now become the standard for building state-of-the-art Natural Language Processing (NLP) systems. Beyond monolingual models, large-scale multilingual models covering more than 100 languages are now available, such as XLM-R by Conneau et al. (2020) and mBERT by Devlin et al. (2019), achieving competitive performance across languages from a variety of families and using various scripts.

Still, most of the 6500+ spoken languages in the world (Hammarström, 2016) are not covered –remaining unseen– by those models. Even languages with millions of native speakers like Lingala (with 15-20 million speakers in central Africa, mostly D.R. Congo) or Bambara (spoken by around 5 million people in Mali and neighboring countries) are not covered by any available language models at the time of writing.

A recent line of work (see §2) has shown that these large multilingual language models (MLMs) can be finetuned on individual languages to further improve performance. Even better, they can be even adapted to languages *unseen* during the pretraining stage.[2]

This work focuses on using adapters, a popular framework for such adaptation that has been proven successful for zero-shot and few-shot cross-lingual transfer. In particular, we significantly improve the adapter framework by drawing inspiration from a simple insight: that the adapters of related languages would likely need to perform the same function, and thus adapters could be trained leveraging multiple related languages. We impose a phylogenetically-inspired tree hierarchy for parameter-sharing between adapters and show empirically that our approach leads to large improvements with experiments on three NLP tasks on several language families.

2 Background

Adapting Large-Scale Models to Low-Resource Languages Multilingual language models (MLMs) can be used directly on unseen languages, or they can also be adapted using unsupervised methods. For example, Han and Eisenstein (2019) successfully used continued training with masked language modeling on unlabeled data to adapt an English BERT model to Early Modern English for sequence labeling. More recently, Muller et al. (2021) employed the same strategy (enhanced with transliteration to handle languages with different scripts) to adapt models for several unseen-during-pretraining languages.

Adapter Units Instead of fine-tuning the whole model, a more promising approach for adaptation uses dedicated units (*adapter units*) that are in-

[1] Code and data are publicly available: https://github.com/ffaisal93/adapt_lang_phylogeny

[2] The potential of such approaches is conditioned on the language's script and data availability, of course.

Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 434–452
November 20–23, 2022.

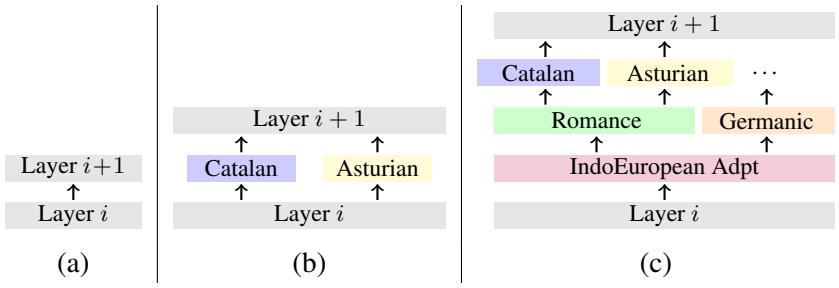

Figure 1: Incorporating phylogeny into neural models with adapters: starting with an unadapted model (a), current practice uses language-specific adapters between layers (b). We instead impose a phylogeny-informed tree hierarchy over adapters as in (c).

jected between the layers of the pre-trained model (see example in Figure 1.b) and can be trained on a new language, domain, or task (Vilar, 2018; Houlsby et al., 2019a; Pfeiffer et al., 2020a,c). There are two advantages in fine-tuning only these adapter components. Since they consist of only a small number of parameters, they can be adequately trained with a small number of training examples. In addition, as the pre-trained model remains invariant, they render *catastrophic forgetting* (French, 1999; Kirkpatrick et al., 2017) a non-issue.

Nevertheless, the application of these adapters has so far followed a simple, straight-forward protocol: insert the adapters, and train them individually for a new task or language. In our work, we investigate how we can improve this process, by incorporating additional linguistic information. The core idea is to incorporate phylogenetic information in the adapters' organization.

3 Phylogeny-Inspired Adaptation

Motivation Intuitively, given the similarities between two related lects (e.g. Catalan and Asturian), one should exploit that relationship to inform the adapters of both languages.

Thankfully, prior linguistic studies provide exactly the information we need in the form of phylogeny trees. Relationships between languages are typically represented as tree or network diagrams. In the phylogenetic trees we will use, languages are grouped based on their similarities; an internal node may (but not necessarily) correspond to a hypothesized linguistic ancestor. While often a phylogenetic network is more appropriate than a tree (e.g. in cases of borrowing, or when two languages influence each other in a bidirectional manner), in this work we will focus on trees as a first step towards phylogeny-inspired adaptation.

Implementation In a standard setting of adapting a language model from a source language to another target language, the typical approach (*e.g.* Pfeiffer et al., 2020c) is to have source and target specific language adapters, trained separately on unlabeled monolingual text with the masked language modeling (MLM) objective (Devlin et al., 2019). Then, one can train a task adapter on source language task data, stacking it on top of the source language adapter. At evaluation time, the source language adapter is replaced with the target language one.

As example, shown in Figure 1, consider three languages: Spanish, Catalan, and Asturian. To adapt a model for e.g. Named Entity Recognition (NER), the standard practice trains Spanish, Catalan, and Asturian language adapters separately: `L:Spanish`, `L:Catalan`, and `L:Asturian`. Using a language with labeled NER data (e.g. Spanish) then trains a task adapter `T:Spanish` using a stack of adapters `[L:Spanish, T:Spanish]`. At inference time we can then use a stack with the appropriate language adapter to perform the task in that language e.g., stack `[L:Asturian, T:Spanish]`.

Our approach follows the same principles, but adapters for multiple languages/genera/families are organized in a hierarchy following phylogenetic information and trained jointly. To continue with our running example, consider that all three languages belong to the Romance language group of the Indo-European family. We hence train five language type adapters jointly: `F:IndoEuro`, `G:Romance`, `L:Spanish`, `L:Catalan`, and `L:Asturian` which are stacked following the hierarchy depicted in Figure 1(c). So, examples from all IndoEuropean languages in our training mix are used to train the `F:IndoEuro` adapter, `G:Romance` is only trained on Romance languages data (if we have e.g. English or Danish in our mix, these data

are directed through a G:Germanic adapter), and we also have language-dedicated adapters. We ensure that each training batch includes data from a single language; so, for an Asturian batch we train the following stack of adapters: [F:IndoEuro, G:Romance, L:Asturian]. At inference time, we also add the task adapter, trained as before on a language with labeled data, on top of our language-hierarchy adapters.

4 Experimental Setup

Tasks We experiment on three NLP tasks:

1. Dependency Parsing (DEP),
2. POS tagging (POS), and
3. Natural Language Inference (NLI).

For (1) and (2), we evaluate on 31 languages from Universal Dependencies v2.9 (Zeman et al., 2021). For (3), we use 4 indigenous low-resource languages from AmericasNLI (Ebrahimi et al., 2021), an extension of XNLI (Conneau et al., 2018). The choice of tasks and datasets is to ensure broad language coverage and especially to ensure we can study language families with only partial representation in the MLM pre-training stage.

Language Families We study dependency parsing and POS-tagging on languages from the Germanic, Uralic and Tupian families.[3] For NLI, we work with languages from Uto-aztecan and Tupian families. See Appendix Table 7 for the complete list of languages we use to train family, group and language adapters.

Pretraining Corpora For language adapter training we collect corpora from a variety of sources. See Appendix A for the complete list of our data sources. As we experiment with a large number of low-resource and endangered languages, the number of sentences per language ranges from 3000 sentences to 1 million (i.e. the high resource ones). Following previous work, we experiment with upsampling for the low-resource languages in our mix, to reduce data sparsity and to ensure they are adequately modeled.

[3]To be accurate, the Germanic languages are a branch (genus) of the Indo-European family, not a distinct language family themselves.

Family	Genus	Tasks
Germanic	East Germanic, West Germanic	POS, DEP
Uralic	Finnic, Hungarian, Permic, Mordvinic, Sami	POS, DEP
Tupian	Tupari, Tupi-Guarani, Munduruku	NLI, POS, DEP
Uto-Aztecan	Tepiman, Corachol, Yaqui, Aztecan, Tarahumaran	NLI

Table 1: Language families and genera we study.

Adapter Training For jointly training phylogeny-inspired adapters, we select training data from the language families/group presented in Table 1. Irrespective of task and setting, we train standard adapter architectures (Üstün et al., 2020) leveraging the AdapterHub.ml (Pfeiffer et al., 2020b) framework.

We train the task adapter by stacking it on top of the hierarchical language adapters. We follow the cross-lingual transfer setting of Pfeiffer et al. (2020c) where we select a high-resource language for task training: we use English for transfer for all families except Uralic, for which we switch to Estonian. In terms of base model choice, we use MBERT for DEP, POS and XLM-R for NLI.[4] For dependency parsing we train using the objective of Glavaš and Vulić (2021), which is a modified variant of the standard deep biaffine attention dependency parser (Dozat and Manning, 2017). For all other tasks, we use simple classification heads as in previous literature.

Baselines and Model Variations We evaluate two common baselines for cross-lingual transfer:

1. [T]: Using only the task adapter trained on some high-resource language; and
2. [LT]: Using the stack of target language and task adapter.

We will denote our phylogeny inspired adapted models as [FGLT]: jointly trained [Family, Group, Target Language] stack and task adapter. We also perform analyses and ablations without some parts of the task: for instance, [FT] and [FGT]

[4]Results with both models for all tasks are available in Appendix: B.

denote stacks using only family (and genus) and task adapters without language-specific ones.

5 Results

General Observations We present our experimental results covering all three tasks in Table 2, showing average performance for the baselines and our proposed method. We further split the results for languages seen and not seen by mBERT during pretraining. Compared to the [T] and [LT] baselines, we observe substantial performance improvements in 10 out of 12 task-family specific settings using [FGLT]. A visualization of all three task results with a breakdown per language is also available in Figure 2.

Looking at Figure 2, it is quite apparent how phylogeny inspired adaptation uplifts the performance of low-resource languages, especially the ones unseen during pretraining. For example, we evaluate dependency parsing on 3 such Germanic languages (Faroese, Gothic and Swiss German). All 3 languages benefit from the proposed adaptation approach with maximum 16.46% improvement over the best performing baseline for Gothic (see Table 8).

This positive drift of performance becomes more obvious for Uralic languages. Here, 8 out of 11 languages are extremely low-resource ones and unseen during pretraining. We obtain improvements over baseline in 7 out of these 8. We further observe similar trends in POS-Tagging for both Germanic and Uralic languages irrespective of the choice of base language model (see Appendix Tables 8—11).

The other language families we focus on are Tupian, Uto-Aztecan, comprised of indigenous and very low-resource languages (Ebrahimi et al., 2021). In case of Tupian languages on DEP-Parsing and POS-Tagging, we observe model adaptation does not result in improvement over baselines on mBERT. However, when we use XLM-R with model adaptation, average performance improves all around for these two tasks. In addition, for NLI, which is a task requiring higher semantic capabilities, we conduct experiments on four languages from Uto-Aztecan and Tupian families. As before, the combination of XLM-R with phylogenetic adaptation outperforms all other settings.

Among the baselines, the task-adapter-only baseline [T] performs better in Germanic and Tupian DEP-Parsing compared to the [LT] baseline. This points out the known problems with negative interference (Wang et al., 2019, 2020, *inter alia*). On the contrary, token classification tasks like POS-Tagging gets significant benefits from using the [LT] baseline. Compared to these, [FGLT] leads to consistent performance improvements. Even though our method does not uplift the result for Tupian DEP-Parsing and POS-Tagging, it is worth noting that it does not hurt either, unlike e.g. [T] which hurts in DEP-Parsing (-0.3 points compared to -5.1 points). Last, outperforming the average baseline of four indigenous American languages (Ebrahimi et al., 2021), points out the effective adaptation capabilities of phylogeny-based adaptation. See Appendix B for detailed language specific results.

True Zero-Shot Adaptation For a large number of extremely low-resource languages not seen during the pre-training of current language models, there may be no easily obtainable textual data to even perform MLM training to train a language-specific adapter. We explore such a scenario and investigate whether the language-family adaptors can be used instead of language-specific ones.

We simulate this scenario in two settings. First for 3 Uralic languages: Skolt Sami (sms), Moksha (mdf) and Karelian (krl). We discard their data from the training set and train other adapters jointly as before. During evaluation, we just use a high-resource language adapter (L:Estonian) instead of the missing language adapters. In addition, we explore this scenario in 4 Tupian languages: Akuntsu (aqz), Makuráp (mpu), Tupinambá (tpn) and Kaapor (urb) where we actually do not have any available training data (except (urb). So we replace the language adapter with a higher-resource one (L:Guajajára).

Results are presented in Table 3. Looking at the rows with phylogenically inspired adaptation [FGLT], we see 1.82% improvement on average for Tupian languages over the best performing baseline ([T]). Except Makuráp (mpu), all other 3 Tupian languages benefit from using our family adapters. Perhaps the most important result is the one on Tupinambá (tpn) which gets drastically impacted when using only baseline language adapter [LT](-13.16% from [T]) but performs much better with [FGLT](+9.21% over [T]).

For Uralic languages, even our model ablations (shown in Table 3) perform better than the baselines: these are [FT] and [FGT] where we get rid of the language adapter part and just draw in-

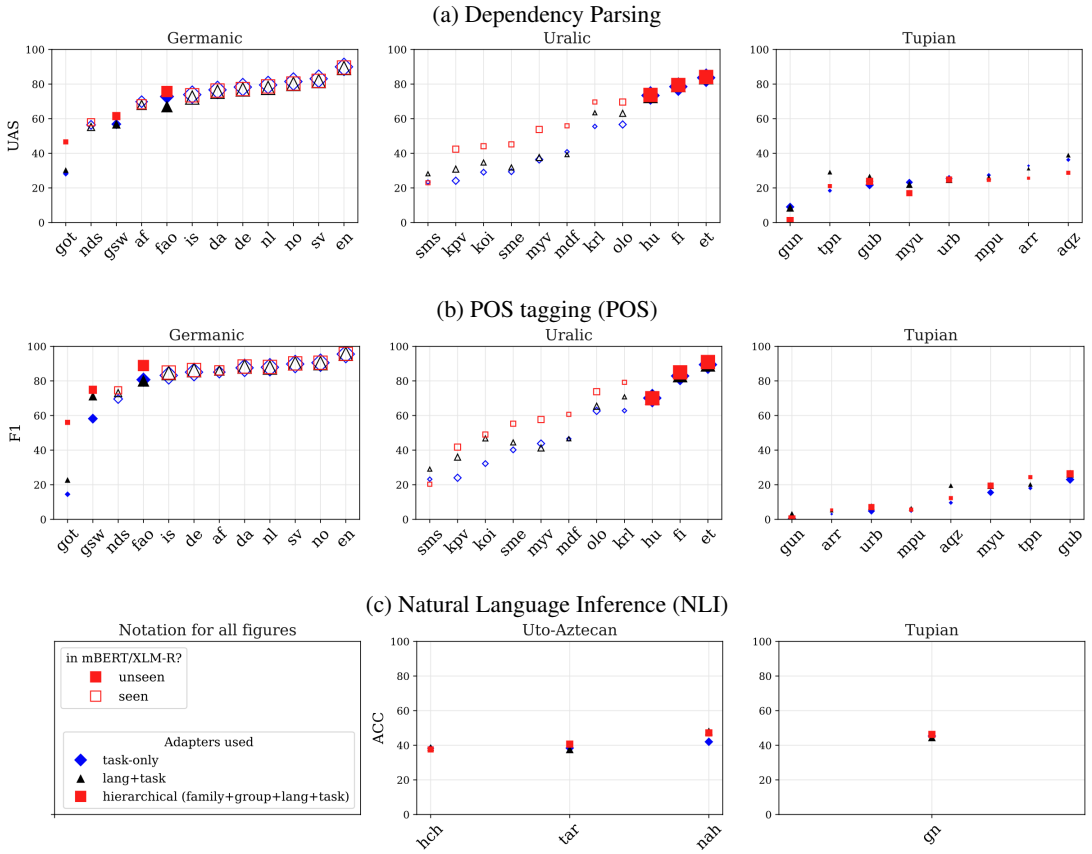

Figure 2: Visualizing three different task results across languages (marker size relative to MLM training data size). In most cases, and especially in languages unseen during pre-training, our hierarchical phylogeny-inspired adapters outperform the baselines.

Task (metric):	**Dep-Parsing (UAS)**			**POS-Tagging (F1-score)**			**NLI (Acc.)**	
Language-Family	Germanic	Uralic	Tupian	Germanic	Uralic	Tupian	Uto-Aztecan	Tupian
Language-Count (Unseen, Total)	(3,12)	(8,11)	(8,8)	(3,12)	(8,11)	(8,8)	(3,3)	(1,1)
Baselines								
BASE-LM+ [T]	52.5 (70.6)	36.9 (48.3)	**24.1**	51.1 (77.3)	41.9 (52.5)	9.9	39.6	45.3
BASE-LM+ [LT]	50.8 (69.2)	41.1 (51.4)	19.0	57.9 (79.6)	47.5 (56.7)	**13.2**	41.3	44.4
Phylogenically inspired								
BASE-LM+ [FGLT]	**60.1** (**72.3**)	**50.5** (**58.3**)	23.8	**73.3** (**83.7**)	**54.7** (**62.2**)	12.6	**41.8**	**46.3**

Table 2: Average results per language family across different tasks. We report averages both for languages unseen during pretraining, and for all languages in the mix (the latter in parentheses). Base language model (BASE-LM) is mBERT for Dep-Parsing, POS-Tagging and XLM-R for NLI. We use the following language for task adapter training: English for Germanic, Tupian and Uto-Aztecan and Estonian for Uralic.

ference from family and genre adapters. Specifically, [FGT] shows consistent improvement for all 3 Uralic languages, even though the model never observed the target language texts during neither base model pretraining nor adapter training.

6 Further Discussion

We perform additional ablation studies where we show that our proposed approach provides sustainable performance in constrained settings with reduced parameter counts. In addition, we explore data up-sampling for low-resource languages in language families with large data imbalances across the language members. This simple approach points towards the further improvement scope with limited data availability. Detailed analysis of both these experiments are presented below.

Parameter Reduction Stacking multiple adapters instead of a single language adapter

Uralic (language adpater: est)					
Model	Training	sms	mdf	krl	avg
Baselines					
MBERT+	[T] (est)	23.37	40.89	55.53	39.93
MBERT+	[LT] (est)	23.82	41.08	53.68	39.53
Phylogenically inspired					
MBERT+	[FGLT] (est)	23.74	**42.01**	53.98	39.91
Ablations					
MBERT+	[FT] (est)	**25.81**	39.37	57.18	40.78
MBERT+	[FGT] (est)	24.48	41.35	**58.99**	**41.60**

Tupian (language adpater: gub)						
Model	Training	aqz	mpu	tpn	urb	avg
Baselines						
MBERT+	[T] (eng)	27.50	**23.97**	22.37	24.59	24.61
MBERT+	[LT] (eng)	22.50	17.81	9.21	25.41	18.73
Phylogenically inspired						
MBERT+	[FGLT] (eng)	**27.50**	19.86	**31.58**	**26.78**	**26.43**
Ablations						
MBERT+	[FT] (eng)	21.25	17.81	14.47	17.76	17.82
MBERT+	[FGT] (eng)	22.50	17.12	19.74	22.13	20.37

Table 3: Dependency parsing with extremely low-resource languages in the absence of language specific adapters (true zero-resource scenario).

comes with extra parameter cost.[5] To assess whether we can integrate phylogenetic information while keeping the adapter parameter counts limited, we perform parameter reduction using a constant factor. For example, consider a single language adapter [L] which has down/upword projections with L:Proj×Layer parameters leading to a parameter count of 2×48×768. Instead we can use a dimension reduced by a factor of 3 and add two extra adapters ([FGL]) without increasing the parameter count 2×(F:Proj+G:Proj+L:Proj)×FGL:Output; to be accurate: 2×(16+16+16)×768. Contrast these with our solution without this constant factor parameter reduction, which will add 2×(48+48+48)×768 parameters to be learned.

The results, tested on Uralic languages for the dependency parsing task, are reported in Table 4. Importantly, we observe consistent performance improvement in [FGLT] over baseline [LT] irrespective of the parameter count. Among these two selections, the [FGLT] one with constrained parameter count (885312) comes with a 1.29% performance trade off which still outperforms the baseline by 4 points on average. Further looking into each individual language result, we find an interesting trend in Skolt Sami (sme). This is the only language where performance drops in constrained [FGLT] compared to the baseline which then drops further when we move to the upscaled [FGLT]. Likewise, we observe performance improvement in any language using sustained model elevates further in upscaled model.

Deep vs Wide Adapters Our FGLT setting makes two important changes to the baseline LT one. First, it stacks 3 language-related adapters as opposed to a single one. Second, it shares some of these adapters between languages. An important question is whether the performance improvements are due to stacking (making the model *deeper*) or due to the parameter sharing between languages. To answer this question, we perform another ablation where we replace the 2×(F:Proj+G:Proj+L:Proj)×FGL:Output setting with 2×(L:Proj+L:Proj+L:Proj)×LLL:Output. Essentially, we create a stack of 3 language-specific adapters.

We will first contrast the baseline [LT] (which has a single *wide adapter*) to this deeper version [LLLT]. We keep the parameter count equal between the two using the same parameter reduction as in the previous paragraph. We find that the [LLLT] setting does indeed improve performance, but only for high-resource languages, even exceeding the upscaled phylogenetic setting [FGLT] (see Table 4). For 7 out of 8 low-resource languages unseen by mBERT, however, the performance degrades in [LLLT] compared to [LT]. Hence, we conclude that deeper stacks of adapters are better than a single wide adapter, but without the adapter parameter sharing this only benefits high-resource languages.

We want to further focus on this second point about parameter sharing: in Table 4, compare rows [LLLT] and [FGLT] under the reduced parameter count. For *all* unseen languages, [FGLT] yields significant improvements, leading to almost 5 UAS points higher on average.

Effect of Upsampling For most of the Uralic, Germanic and all of the Tupian and Uto-Aztecan low-resource languages, we had very little amount of training data available. As a result, this limited data availability creates within-family data imbalance, especially for Germanic and Uralic languages. To address this issue, we perform a simple data upsampling on all low resource languages from these two families. Here, the upsampling factor is inversely proportional to the per-language token count. A language with very low word count is

[5] We note, though, that this additional cost is still a very small fraction of the overall model's parameter count.

Uralic (DEP-Parsing)													
		MBERT-SEEN			MBERT-UNSEEN								
Model	Training	est	fin	hun	koi	kpv	krl	mdf	myv	olo	sme	sms	avg
Adapter Parameter count: constrained (885312)													
MBERT+	[LT] (est)	84.05	79.08	73.00	32.30	26.85	53.52	37.52	35.08	54.30	26.23	25.89	47.98
MBERT+	[LLLT] (est)	86.01	79.51	74.47	32.30	27.71	49.23	37.39	33.34	51.21	25.73	20.56	47.04
MBERT+	[FGLT] (est)	83.23	78.48	72.63	37.43	32.21	64.06	44.12	39.79	64.78	30.75	24.26	51.98
Adapter Parameter count: Upscaled (2655936 or, 3×885312)													
MBERT+	[FGLT] (est)	84.20	79.59	73.10	38.14	35.55	65.77	44.52	42.77	67.94	31.62	22.78	53.27

Table 4: Effect of parameter reduction in dependency parsing (Metric: UAS) on Uralic languages.

Model	Training	sme	koi	fin*	myv	olo	mdf	hun*	sms	kpv	est*	krl	avg
Original datasize:		10k	10k	1M	29k	19k	5k	1M	3k	13k	1M	5k	
MBERT+	[FGLT] (et)	31.62	38.14	79.59	42.77	67.94	44.52	73.10	22.78	35.55	84.20	65.77	53.27
Upsampled:		100k	60k	1M	87k	116k	28k	1M	29k	40k	1M	36k	
MBERT+	[FGLT] (et)	45.16	44.10	79.45	53.77	69.62	55.88	73.73	23.00	42.40	84.10	69.65	58.26

Table 5: Dependency parsing result (UAS) upsampling datasize (* columns are the high-resourced ones and not up-sampled, the presented datasize is approximate sentence count per language)

Model	Training	fao	kpv	urb	avg
DEP (task adpater: eng)					
Baselines					
MBERT+	[T]	72.80	24.15	24.59	40.51
MBERT+	[LT]	66.93	30.87	25.41	41.07
Phylogenically inspired					
MBERT+	[FGLT]	**75.70**	**42.40**	**26.78**	**48.29**
Random Tree					
MBERT+	[FGLT]	66.19	28.53	24.04	39.59
POS (task adpater: eng)					
Baselines					
MBERT+	[T]	80.70	24.02	4.79	36.50
MBERT+	[LT]	79.93	35.96	7.13	41.01
Phylogenically inspired					
MBERT+	[FGLT]	**88.88**	**41.74**	7.10	**45.91**
Random Tree					
MBERT+	[FGLT]	86.66	35.96	**13.66**	45.43

Table 6: Adapters arranged following a phylogenetically-inspired tree perform significantly better than ones following random counterfactual tree. Parameter sharing between similar languages leads to significantly better results for the unseen languages in both tasks.

sampled in large numbers compared to the ones with higher word count.

We use the upsampled dataset for all the dependency parsing and POS tagging experiments we perform on these two language families (Appendix Table 2, 8, 9, 10, 11). The positive upsampling effect is obvious when we compare the dependency parsing results on Uralic upsampled dataset with the one with original datasize in Table 5. Note that we do not upsample the 3 high resource ones: Estonian (et), Finnish (fi), and Hungarian (hu) and experiment on the other languages, where we can make a number of interesting observations.

First, though the original sentence count is same (10k) for North Sami (sme) and Komi Permyak (koi) the upsampled size is different for these two languages: 100k and 60k respectively. The reason behind this difference is, we perform word-count based upsampling and the average sentence length turns out to be less for koi thus assigned with a low sampling factor. Hence, the one with higher upsampled sentence count (sme) results in large performance improvement of 13.54 points, while it was the one with second lowest score in the non-upsampled setting. Secondly, we observe performance improvements for all low-resource languages. It would be interesting to explore the resource dependent performance variation that could be attributed to data sampling choices. For now, we keep this open for future studies.

On the other hand, we cannot clearly claim that extremely low-resource languages always benefit from upsampling. For example, Skolt Sami (sms) is the one with lowest data availability (3k) and lowest original score (22.78). Upsampling more than 9x times results in only 0.22% improvement. We suspect that data quality might play an important role here, considering that we had to scrape the few data available online for sms (wan), whereas the corpus we use for sme was collected by Goldhahn et al. (2012) following standard approaches and with NLP applications in mind.

Random vs Phylogenetic Tree One key hypothesis of ours is that language family tree information is beneficial for modeling low-resource languages.

To further solidify this claim, we compare adapters based on a linguistically-informed tree (like the one we have been using in all previous experiments) to adapters based on a counterfactual (hypothetical) language tree. We construct a random language family hierarchy and train the adapter stacks jointly like before instead of using the phylogenetically informed ones. We make the random tree structure typologically diverse while keeping one low-resource language from either Germanic (Faroese), Uralic (Komi Zyrian) or Tupian (Kaapor) present in each newly defined genus (see Table 15 in Appendix D for the random family tree structure). In Table 6, we report results in Dependency parsing and POS tagging tasks for these 3 languages under each of these settings. The results for dependency parsing are to a large extent conclusive: the adapters following the random tree perform worse than the baselines, while the phylogenetically-inspired ones are significantly better. The random-tree adapters do indeed outperform the baselines for POS tagging, but again for 2 of the 3 low-resource languages fall short compared to the phylogenetically-inspired ones. Curiously, for Kaapor, this random-tree model outperforms all other models, but all of them are still extremely bad (with only an accuracy of 13% in the best case); nevertheless, we will further investigate this result in future work.

Indo-European Family Tree Going beyond our original setup, we conduct one additional experiment where we do joint-training on the whole Indo-European language family as shown in Figure 1. The only difference is that essentially, by adding a *root* adapter `R` we have a stack of four jointly trained adapters `[RFGL] (R:IndoEuro)` instead of just three (i.e. `[FGL]`). Interestingly, the performance on the dependency parsing tasks gets negatively impacted for almost all languages (see Table 14). We hypothesize that this is due to the inherent diversity of the Indo-European family. Despite sharing a common ancestor (Proto-Indo-European), the IE family groups that we work with here (Germanic, Romance, Slavic, Celtic, Greek, Indo-Aryan) are too typologically different from each other, and forcing them to share a common root negates the gains of the group-specific adapters. We plan to investigate this further in future work.

7 Related Work

Continuous effort is being put to improve cross-lingual transfer across languages as well as making language models capable enough to go beyond high resource domains. Recently, Wang et al. (2022), proposed an approach to combine lexicons with monolingual/parallel data for pretraining. It expands the modeling capability to thousands more languages largely including under-represented languages with limited to zero corpus availability. It is now proven that, pretraining on closely related languages yields better result for zero-shot transfer (Pires et al., 2019) and continued pretraining on a larger number of languages leads to further improvement (Fujinuma et al., 2022). However, training on some specific languages can still hurt the performance of other languages (Conneau et al., 2020). As a result, it is crucial to prevent negative inference while keeping the performance equitable and robust across languages (Wang et al., 2019, 2020).

To make the performance robust across languages, it is important to identify how much linguistic information is currently in place inside these big multilingual models. Recent studies have done investigation on this hypothesis by probing language models for linguistic typology (Choenni and Shutova, 2022; Stańczak et al., 2022) as well as phylogheny (Rama et al., 2020). These studies have measured phylogenetic distance and typological similarity across languages so that we can make informed cross-lingual transfer. In line with these findings, (Zhao et al., 2021) has done experiments to remove the language specific information by stackable vector operations which further improve the cross-lingual representation. One recent study (Foroutan et al., 2022) dives further into identifying language-neutral and language-specific subspace inside the representation space of multilingual models and now it is proven that the shared representation space is the one helping to perform effective cross-lingual transfer.

As opposed to the standard fine-tuning of large-scale language models, a more focused trend is to perform efficient parameter selection thus reducing the overall computation cost and carbon footprints (Houlsby et al., 2019b). Adapters are such highly customized light-weight neural network layers on top of base models. Because of this higher

flexibility, there are studies already in place looking into the adapter-level optimization according to the nature of data and network layers (Moosavi et al., 2022). In addition, using language specific units in a modular fashion in the pre-training stage was shown to be beneficial in recent work (Pfeiffer et al., 2022).

8 Limitations and Future Work

While we already incorporated task evaluation on a diverse set of language families ranging from extremely low resourced Uralic ones to indigenous AmericasNLI (Ebrahimi et al., 2021) languages, our experiments are still limited in terms of typological diversity. In future, we want to further extend the typological diversity of languages we use. At the same time, we would like to democratize the full force of language genetical properties in steps beyond just finetuning thus making the resource scarce languages more accessible.

9 Conclusion

In this work, we present an adapter-based approach to leverage language phylogenetic information for better cross-lingual adaptation. Our experiments on a diverse set of tasks and languages show significant performance improvements over commonly-used strong baselines. Even better, we show that under the exact same adapter parameter count settings, using smaller adapters but forcing adapter sharing between genetically related languages improves performance on true zero-resource scenarios. These improvements are particularly stark for languages unseen in the pre-training stage of large multilingual language models, providing a direct path towards better adaptation and language coverage for language technologies.

Acknowledgements

This work is generously supported by NSF Award IIS-2125466 and by a Google Award for Inclusion Research.

References

Bible in finno-ugric languages. Online resource.

Gothic bible. Online resource.

Language page of scripture earth. Online resource.

Public domain komi-zyrian data. Online resource.

Wanca website. Online resource.

Tatyana Boyko, Nina Zaitseva, Natalia Krizhanovskaya, Andrew Krizhanovsky, Irina Novak, Nataliya Pellinen, and Aleksandra Rodionova. 2022. The open corpus of the veps and karelian languages: Overview and applications. *KnE Social Sciences*, 7(3):29–40.

William Bright and David Brambila. 1976. Diccionario raramuri-castellano (tarahumar). 57:975.

Luis Chiruzzo, Pedro Amarilla, Adolfo Ríos, and Gustavo Giménez Lugo. 2020. Development of a Guarani - Spanish parallel corpus. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2629–2633, Marseille, France. European Language Resources Association.

Rochelle Choenni and Ekaterina Shutova. 2022. Investigating Language Relationships in Multilingual Sentence Encoders Through the Lens of Linguistic Typology. *Computational Linguistics*, pages 1–38.

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.

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.

Timothy Dozat and Christopher D. Manning. 2017. Deep biaffine attention for neural dependency parsing. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Abteen Ebrahimi, Manuel Mager, Arturo Oncevay, Vishrav Chaudhary, Luis Chiruzzo, Angela Fan, John Ortega, Ricardo Ramos, Annette Rios, Ivan Vladimir, Gustavo A. Giménez-Lugo, Elisabeth Mager, Graham Neubig, Alexis Palmer, Rolando A. Coto Solano,

Ngoc Thang Vu, and Katharina Kann. 2021. Americasnli: Evaluating zero-shot natural language understanding of pretrained multilingual models in truly low-resource languages. *CoRR*, abs/2104.08726.

Negar Foroutan, Mohammadreza Banaei, Remi Lebret, Antoine Bosselut, and Karl Aberer. 2022. Discovering language-neutral sub-networks in multilingual language models.

Robert M. French. 1999. Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 3(4):128–135.

Yoshinari Fujinuma, Jordan Boyd-Graber, and Katharina Kann. 2022. Match the script, adapt if multilingual: Analyzing the effect of multilingual pretraining on cross-lingual transferability. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1500–1512, Dublin, Ireland. Association for Computational Linguistics.

Goran Glavaš and Ivan Vulić. 2021. Is supervised syntactic parsing beneficial for language understanding tasks? an empirical investigation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3090–3104, Online. Association for Computational Linguistics.

Dirk Goldhahn, Thomas Eckart, and Uwe Quasthoff. 2012. Building large monolingual dictionaries at the Leipzig corpora collection: From 100 to 200 languages. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 759–765, Istanbul, Turkey. European Language Resources Association (ELRA).

Ximena Gutierrez-Vasques, Gerardo Sierra, and Isaac Hernandez Pompa. 2016. Axolotl: a web accessible parallel corpus for Spanish-Nahuatl. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 4210–4214, Portorož, Slovenia. European Language Resources Association (ELRA).

Harald Hammarström. 2016. Linguistic diversity and language evolution. *Journal of Language Evolution*, 1(1):19–29.

Xiaochuang Han and Jacob Eisenstein. 2019. Unsupervised domain adaptation of contextualized embeddings for sequence labeling. 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 4238–4248, Hong Kong, China. Association for Computational Linguistics.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019a. Parameter-efficient transfer learning for NLP. arXiv:1902.00751.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019b. Parameter-efficient transfer learning for nlp.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526.

Manuel Mager, Dionico Gonzalez, and Ivan Meza. 2017. Probabilistic finite-state morphological segmenter for wixarika (huichol).

Nafise Sadat Moosavi, Quentin Delfosse, Kristian Kersting, and Iryna Gurevych. 2022. Adaptable adapters. arXiv:2205.01549.

Benjamin Muller, Antonios Anastasopoulos, Benoît Sagot, and Djamé Seddah. 2021. When being unseen from mBERT is just the beginning: Handling new languages with multilingual language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 448–462, Online. Association for Computational Linguistics.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous pipelines for processing huge corpora on medium to low resource infrastructures. Proceedings of the Workshop on Challenges in the Management of Large Corpora (CMLC-7) 2019. Cardiff, 22nd July 2019, pages 9 – 16, Mannheim. Leibniz-Institut für Deutsche Sprache.

Jonas Pfeiffer, Naman Goyal, Xi Victoria Lin, Xian Li, James Cross, Sebastian Riedel, and Mikel Artetxe. 2022. Lifting the curse of multilinguality by pretraining modular transformers. arXiv:2205.06266.

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: System Demonstrations*, pages 46–54, Online. Association for Computational Linguistics.

Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulić, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. 2020b. Adapterhub: A framework for adapting transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 46–54.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020c. 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.

Taraka Rama, Lisa Beinborn, and Steffen Eger. 2020. Probing multilingual BERT for genetic and typological signals. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1214–1228, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Rueter. 2018. Rueter/open-erme-erzya: Open erme erzya. Online resource.

Janine Siewert, Yves Scherrer, Martijn Wieling, and Jörg Tiedemann. 2020. LSDC - a comprehensive dataset for low Saxon dialect classification. In *Proceedings of the 7th Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 25–35, Barcelona, Spain (Online). International Committee on Computational Linguistics (ICCL).

Karolina Stańczak, Edoardo Ponti, Lucas Torroba Hennigen, Ryan Cotterell, and Isabelle Augenstein. 2022. Same neurons, different languages: Probing morphosyntax in multilingual pre-trained models.

Ahmet Üstün, Arianna Bisazza, Gosse Bouma, and Gertjan van Noord. 2020. UDapter: Language adaptation for truly Universal Dependency parsing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2302–2315, Online. Association for Computational Linguistics.

David Vilar. 2018. Learning hidden unit contribution for adapting neural machine translation models. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 500–505, New Orleans, Louisiana. Association for Computational Linguistics.

Xinyi Wang, Sebastian Ruder, and Graham Neubig. 2022. Expanding pretrained models to thousands more languages via lexicon-based adaptation.

Zirui Wang, Zihang Dai, Barnabás Póczos, and Jaime Carbonell. 2019. Characterizing and avoiding negative transfer. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11285–11294.

Zirui Wang, Zachary C. Lipton, and Yulia Tsvetkov. 2020. On negative interference in multilingual models: Findings and a meta-learning treatment. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4438–4450, Online. Association for Computational Linguistics.

Daniel Zeman, Joakim Nivre, Mitchell Abrams, Elia Ackermann, Noëmi Aepli, Hamid Aghaei, Željko Agić, Amir Ahmadi, Lars Ahrenberg, Ajede, and et al. 2021. Universal dependencies 2.9. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Wei Zhao, Steffen Eger, Johannes Bjerva, and Isabelle Augenstein. 2021. Inducing language-agnostic multilingual representations. In *Proceedings of *SEM 2021: The Tenth Joint Conference on Lexical and Computational Semantics*, pages 229–240, Online. Association for Computational Linguistics.

A Dataset

Detailed data source with statistics are presented in table 7.

B Language Specific Task Results

Detailed language specific task results are presented in table 8, 9, 10, 11, 12 and 13.

Dependency Parsing For dependency parsing, we perform experiments on Germanic, Uralic and Tupian languages. We observe, phylogeny based joint training performs better for 10 out of 11 Germanic and Uralic languages unseen by mbert. In addition all of the Tupian ones are unseen by mbert and joint training performs better than the language based adapter baseline `[LT]`. Similar trend is visible in case of Germanic high resource languages where using the language based adapter baseline `[LT]` hurts the overall performance. Though, joint training does not cross the performance threshold of just using the task adapter baseline `[T]` in case of majority high resource ones, it doesn't do negative interference like language adapter based baseline either. At the same time, the performance improvement for unseen low resource languages are significant while using joint training. Thus phylogeny based joint training keeps a performance balance across languages with diverse data availability.

POS Tagging For POS tagging task, we select the same language and settings like before we used in dependency parsing. In POS tagging, the language adapter does not make negative interference like it made in case of dependency parsing. However, using phylogny based joint training still performs better than all the baseline in majority Germanic and Uralic languages. In case of Tupian languages, we see improvement using phylogeny based adaptation in 4 out of 8 languages.

NLI Our NLI results are presented in table 12 and 13. In addition, we reprot the zero-shot baseline results from (Ebrahimi et al., 2021) where the pretrained language model was continually trained on monolingual task language before training on downstream english task data. In our adaptation settings, we follow the `[FGLT]` combinations. Our approach does better for low resource ones (i.e.) while joint training results in optimal performance.

C Dependency Parsing on Indo-European Family

The dependency parsing results comprising Indo-European family branches are presented in table 14.

D Random Family Tree

In our random family tree construction, we select 9 languages from 9 different language family branches. We group these languages into 3 genus while keeping one language in each genus from either Germanic, Tupian or Uralic language family on which we report our experimental result. The tree structer is presented in table 15.

Family	Genus	Language	ISO 639-3	Size	Source
Germanic	North	Danish	dan	1M	OSCAR (Ortiz Suárez et al., 2019)
	North	Faroese	fao	300K	(Goldhahn et al., 2012)
	North	Icelandic	isl	1M	OSCAR (Ortiz Suárez et al., 2019)
	North	Norwegian	nor	1M	OSCAR (Ortiz Suárez et al., 2019)
	North	Swedish	swe	1M	OSCAR (Ortiz Suárez et al., 2019)
	West	Afrikaans	afr	120K	OSCAR (Ortiz Suárez et al., 2019)
	West	German	deu	1M	OSCAR (Ortiz Suárez et al., 2019)
	West	English	eng	1M	OSCAR (Ortiz Suárez et al., 2019)
	West	Gothic	got	4.4K	Bible (wul)
	West	Low Saxon	nds	95.5K	(Siewert et al., 2020)
	West	Dutch	nld	1M	OSCAR (Ortiz Suárez et al., 2019)
	West	Swiss German	gsw	100K	(Goldhahn et al., 2012)
Tupian	Munduruku	Munduruku	myu	8.7K	Bible (spl)
	Tupi Guaraní	Guaraní	grn	26K	(Chiruzzo et al., 2020)
	Tupi Guaraní	Simba Guaraní	gnw	6.7K	Bible (spl)
	Tupi Guaraní	Guajajára	gub	33.9K	Bible (spl)
	Tupi Guaraní	Mbya Guaraní	gun	50.5K	Bible (spl)
	Tupi Guaraní	Kaapor	urb	9.3K	Bible (spl)
	Tupari	Akuntsu	aqz	-	-
	Tupari	Makuráp	mpu	-	-
	Tupi-Guarani	Tupinambá	tpn	-	-
Uralic	Finnic	Estonian	est	1M	OSCAR (Ortiz Suárez et al., 2019)
	Finnic	Finnish	fin	1M	OSCAR (Ortiz Suárez et al., 2019)
	Finnic	Karelian	krl	5K	Bible (krl)
	Finnic	Livvi	olo	19K	(Boyko et al., 2022)
	Hungarian	Hungarian	hun	1M	OSCAR (Ortiz Suárez et al., 2019)
	Mordvinic	Moksha	mdf	5K	Bible (krl)
	Mordvinic	Erzya	myv	29K	(Rueter, 2018)
	Permic	Komi Permyak	koi	10K	(Goldhahn et al., 2012)
	Permic	Komi Zyrian	kpv	13K	(kpv)
	Sami	North Sami	sme	10K	(Goldhahn et al., 2012)
	Sami	Skolt Sami	sms	3K	(wan)
Uto-Aztecan	Aztecan	Nahuatl	nah	16K	(Gutierrez-Vasques et al., 2016)
	Corachol	Cora	crn	10.1K	Bible (spl)
	Corachol	Huichol	hch	8.9K	(Mager et al., 2017)
	Tarahumaran	Rarámuri	tar	14.7K	(Bright and Brambila, 1976)
	Tepiman	Northern Tepehuan	ntp	6.5K	Bible (spl)
	Tepiman	O'odham	ood	6.5K	Bible (spl)
	Tepiman	Southern Tepehuan	stp	7K	Bible (spl)
	Yaqui	Mayo	mfy	7K	Bible (spl)
	Yaqui	Yaqui	yaq	6.5K	Bible (spl)

Table 7: Dataset statistics and sources of the language datasets we work with.

Germanic

Model Training	MBERT-SEEN afr	dan	deu	eng	isl	nds	nld	nor	swe	MBERT-UNSEEN fao	got	gsw	avg
Baselines													
MBERT+ [T] (eng)	**69.83**	**76.65**	**78.27**	**89.95**	**73.90**	56.86	**79.49**	**81.47**	**83.09**	72.80	28.20	56.43	70.58
MBERT+ [LT] (eng)	67.97	75.56	76.89	89.28	72.22	56.65	77.79	80.07	81.72	66.93	30.15	55.23	69.20
Phylogenically inspired													
MBERT+ [FGLT] (eng)	68.34	76.26	77.13	89.56	73.51	61.50	78.64	80.30	81.87	**75.70**	**46.61**	**57.94**	**72.28**
Ablations													
MBERT+ [LT] (eng)	63.41	69.39	71.22	79.97	63.77	56.51	72.11	72.03	75.03	64.85	38.69	50.32	64.78
MBERT+ [FLT] (eng)	68.26	76.10	77.47	89.38	73.10	**62.40**	78.52	80.39	82.12	75.05	46.02	57.81	72.22

Uralic

Model Training	MBERT-SEEN est	fin	hun	MBERT-UNSEEN koi	kpv	krl	mdf	myv	olo	sme	sms	avg
Baselines												
MBERT+ [T] (est)	83.67	78.51	73.42	29.08	24.15	55.53	40.89	36.45	56.65	29.34	23.37	48.28
MBERT+ [LT] (est)	83.95	79.41	73.10	34.68	30.87	63.41	39.23	37.58	63.10	31.85	**28.18**	51.40
Phylogenically inspired												
MBERT+ [FGLT] (est)	**84.10**	**79.45**	73.73	**44.10**	**42.40**	**69.65**	**55.88**	**53.77**	**69.62**	**45.16**	23.00	**58.26**
Ablations												
MBERT+ [LT] (est)	75.68	71.45	66.97	36.83	32.51	60.60	41.28	39.57	62.70	33.12	23.89	49.51
MBERT+ [FLT] (est)	83.72	78.84	**73.78**	37.31	34.55	68.13	50.13	47.24	68.95	41.71	24.63	55.36

Tupian

Model Training	MBERT-UNSEEN aqz	arr	gub	gun	mpu	myu	tpn	urb	avg
Baselines									
MBERT+ [T] (eng)	**27.50**	**33.82**	26.07	9.11	**23.97**	**25.46**	22.37	24.59	**24.11**
MBERT+ [LT] (eng)	22.50	26.66	19.69	**11.55**	17.81	19.19	9.21	25.41	19.00
Phylogenically inspired									
MBERT+ [FGLT] (eng)	**27.50**	26.01	**28.46**	10.45	19.86	19.56	**31.58**	**26.78**	23.77
Ablations									
MBERT+ [LT] (eng)	21.25	24.20	23.78	10.30	15.75	23.62	18.42	26.50	20.48
MBERT+ [FLT] (eng)	25.00	26.45	26.66	9.86	17.12	20.30	19.74	22.68	20.97

Table 8: Dependency Parsing Task Results (base model: MBERT, metric: UAS).

Germanic

Model	Training	XLM-R-SEEN afr	dan	deu	eng	isl	nds	nld	nor	swe	XLM-R-UNSEEN fao	got	gsw	avg
Baselines														
XLM-R+	[T] (eng)	68.36	74.82	77.07	85.00	74.36	44.73	77.01	79.66	81.94	70.20	25.04	42.87	66.75
XLM-R+	[LT] (eng)	**69.78**	76.38	**78.54**	87.22	**76.12**	56.60	**78.70**	**81.43**	**83.46**	**74.17**	23.47	56.37	70.19
Phylogenically inspired														
XLM-R+	[FGLT] (eng)	69.74	**76.56**	78.00	**87.38**	75.80	**58.54**	78.68	81.33	83.31	73.47	**38.18**	**63.09**	**72.01**
Ablations														
XLM-R+	[LT] (eng)	67.67	73.73	75.52	83.65	73.30	53.16	76.33	78.65	80.86	68.68	32.45	55.40	68.28
XLM-R+	[FLT] (eng)	69.66	76.41	78.11	87.29	75.97	57.63	78.75	81.49	83.44	73.67	36.88	62.53	71.82

Uralic

Model	Training	XLM-R-SEEN est	fin	hun	XLM-R-UNSEEN koi	kpv	krl	mdf	myv	olo	sme	sms	avg
Baselines													
XLM-R+	[T] (est)	82.02	78.59	73.16	31.94	30.25	61.47	34.41	34.46	56.45	26.27	31.07	49.10
XLM-R+	[LT] (est)	**84.25**	**80.11**	**74.72**	33.37	31.31	65.03	33.62	31.91	58.47	25.72	28.25	49.71
Phylogenically inspired													
XLM-R+	[FGLT] (est)	83.39	79.40	73.61	**40.76**	**39.00**	**67.84**	**37.71**	**38.66**	**67.07**	**29.11**	31.21	**53.44**
Ablations													
XLM-R+	[LT] (est)	81.67	77.80	72.14	33.85	30.71	62.57	30.18	33.44	63.44	23.96	30.33	49.10
XLM-R+	[FLT] (est)	83.22	79.41	74.05	39.93	38.12	66.52	37.25	38.20	66.20	28.23	**31.73**	52.99

Tupian

Model	Training	XLM-R-UNSEEN aqz	arr	gub	gun	mpu	myu	tpn	urb	avg
Baselines										
XLM-R+	[T] (eng)	**33.75**	**29.47**	17.40	3.95	**24.66**	**30.63**	19.74	25.14	23.09
XLM-R+	[LT] (eng)	32.50	28.99	17.88	**3.96**	21.92	27.68	22.37	24.86	22.52
Phylogenically inspired										
XLM-R+	[FGLT] (eng)	27.50	28.52	**28.51**	3.84	23.29	28.41	25.00	**28.69**	**24.22**
Ablations										
XLM-R+	[LT] (eng)	27.50	29.25	19.40	3.38	21.23	26.57	**28.95**	19.40	21.96
XLM-R+	[FLT] (eng)	23.75	28.82	23.59	3.50	19.86	28.04	23.68	26.50	22.22

Table 9: Dependency Parsing Task Results (base model: XLM-R, metric: UAS).

Germanic

Model	Training	MBERT-SEEN afr	dan	deu	eng	isl	nds	nld	nor	swe	MBERT-UNSEEN fao	got	gsw	avg
Baselines														
MBERT+	[T] (eng)	85.08	87.55	85.04	95.50	83.18	69.53	87.88	90.49	89.74	80.70	14.50	58.18	77.28
MBERT+	[LT] (eng)	85.93	88.23	86.16	95.64	84.49	72.93	87.70	90.22	90.10	79.93	22.60	71.07	79.58
Phylogenically inspired														
MBERT+	[FGLT] (eng)	**86.09**	88.31	**86.27**	**95.66**	**84.83**	**74.54**	88.06	90.50	**90.10**	**88.88**	**56.03**	**74.86**	**83.68**
Ablations														
MBERT+	[LT] (eng)	85.03	87.40	84.68	94.23	82.89	71.82	86.37	88.18	88.61	82.31	47.23	70.25	80.75
MBERT+	[FLT] (eng)	86.08	**88.36**	86.08	95.62	84.45	73.86	**88.15**	**90.52**	89.95	88.15	55.47	73.65	83.36

Uralic

Model	Training	MBERT-SEEN est	fin	hun	MBERT-UNSEEN koi	kpv	krl	mdf	myv	olo	sme	sms	avg
Baselines													
MBERT+	[T] (est)	89.39	82.85	70.07	32.22	24.02	62.79	46.53	43.79	62.67	40.15	23.21	52.52
MBERT+	[LT] (est)	89.49	83.29	70.38	46.78	35.96	70.78	46.55	41.26	65.37	44.46	**29.03**	56.67
Phylogenically inspired													
MBERT+	[FGLT] (est)	**90.88**	**84.93**	69.98	**49.01**	**41.74**	**79.17**	**60.69**	**57.69**	**73.75**	**55.27**	20.32	**62.13**
Ablations													
MBERT+	[LT] (est)	87.12	82.21	68.67	39.83	34.73	72.90	50.58	45.83	67.80	49.13	25.44	56.75
MBERT+	[FLT] (est)	90.55	83.99	**70.45**	41.96	36.64	76.76	52.89	50.25	70.62	51.28	20.56	58.72

Tupian

Model	Training	MBERT-UNSEEN aqz	arr	gub	gun	mpu	myu	tpn	urb	avg
Baselines										
MBERT-R+	[T] (eng)	9.60	3.06	23.02	0.37	4.95	15.52	18.02	4.79	9.92
MBERT-R+	[LT] (eng)	**19.35**	4.88	26.21	**2.42**	**6.25**	19.33	20.00	7.13	**13.20**
Phylogenically inspired										
MBERT-R+	[FGLT] (eng)	12.28	**5.44**	26.32	0.23	5.62	19.49	**24.39**	7.10	12.61
Ablations										
MBERT-R+	[LT] (eng)	13.79	3.65	**26.92**	0.21	3.57	17.37	17.86	6.60	11.25
MBERT-R+	[FLT] (eng)	18.64	3.71	26.62	0.20	4.68	**21.01**	21.31	**7.43**	12.95

Table 10: Parts of Speech Task Results (base model: MBERT, metric: F1).

Germanic

Model	Training	XLM-R-SEEN afr	dan	deu	eng	isl	nds	nld	nor	swe	XLM-R-UNSEEN fao	got	gsw	avg
Baselines														
XLM-R+ [T]	(eng)	**87.27**	**89.14**	87.64	96.34	85.64	55.77	87.75	91.15	91.49	81.29	16.50	47.67	76.47
XLM-R+ [LT]	(eng)	87.25	89.05	87.53	96.36	85.55	70.21	87.73	91.12	91.35	87.16	15.41	66.37	79.59
Phylogenically inspired														
XLM-R+ [FGLT]	(eng)	86.98	88.94	**88.09**	**96.44**	85.62	**74.31**	**87.94**	91.11	91.35	**88.85**	**41.75**	**76.52**	**83.16**
Ablations														
XLM-R+ [LT]	(eng)	86.75	89.05	87.77	96.36	**85.80**	71.16	87.89	91.08	**91.52**	88.23	34.60	68.65	81.57
XLM-R+ [FLT]	(eng)	86.92	89.00	87.86	96.40	85.78	72.39	87.97	**91.17**	91.38	88.81	39.23	73.43	82.53

Uralic

Model	Training	XLM-R-SEEN est	fin	hun	XLM-R-UNSEEN koi	kpv	krl	mdf	myv	olo	sme	sms	avg
Baselines													
XLM-R+ [T]	(est)	96.61	**89.31**	83.98	47.30	38.39	70.39	43.15	44.21	64.99	37.74	34.84	59.17
XLM-R+ [LT]	(est)	96.64	89.30	83.61	46.97	39.57	74.55	41.89	43.95	65.86	36.58	33.32	59.29
Phylogenically inspired													
XLM-R+ [FGLT]	(est)	96.69	89.23	83.31	**56.93**	**47.37**	**81.41**	**47.88**	**49.40**	**73.71**	**46.68**	**35.79**	**64.40**
Ablations													
XLM-R+ [LT]	(est)	96.54	89.22	83.61	48.42	41.07	80.00	43.87	46.01	72.15	41.63	35.15	61.61
XLM-R+ [FLT]	(est)	**96.71**	89.21	**84.24**	50.38	42.94	80.70	44.88	46.29	72.71	42.05	35.96	62.37

Tupian

Model	Training	XLM-R-UNSEEN aqz	arr	gub	gun	mpu	myu	tpn	urb	avg
Baselines										
XLM-R-R+ [T]	(eng)	6.25	**5.92**	26.05	**5.13**	8.16	16.07	21.62	6.91	12.01
XLM-R-R+ [LT]	(eng)	6.96	4.80	27.16	2.67	6.10	20.96	26.79	6.56	12.75
Phylogenically inspired										
XLM-R-R+ [FGLT]	(eng)	11.86	4.89	**37.35**	4.35	7.27	**23.86**	23.53	**12.74**	**15.73**
Ablations										
XLM-R-R+ [LT]	(eng)	**15.83**	5.36	27.05	4.26	**9.85**	13.91	**26.67**	8.11	13.88
XLM-R-R+ [FLT]	(eng)	12.60	4.36	32.19	4.58	4.52	17.53	25.64	8.98	13.80

Table 11: Parts of Speech Task Results (base model: XLM-R, metric: F1).

Model Training	grn	hch	nah	tar	avg
Baselines					
MBERT+ [T] (eng)	33.60	33.20	33.60	33.33	33.43
MBERT+ [LT] (eng)	34.40	33.20	33.60	**33.73**	33.73
Phylogenically inspired					
MBERT+ [FGLT] (eng)	**36.13**	33.47	**33.88**	33.33	**34.20**
Ablations					
MBERT+ [LT] (eng)	33.33	33.33	33.20	33.07	33.23
MBERT+ [FLT] (eng)	**33.73**	33.73	33.47	33.33	33.57

Table 12: NLI Task Results on AmericasNLI (Ebrahimi et al., 2021) languages (base model: MBERT, metric: ACC).

Model Training	grn	hch	nah	tar	avg
Baselines					
XLM-R+ [T] (eng)	45.33	38.27	42.01	38.40	41.00
XLM-R+ [LT] (eng)	44.40	**38.53**	**47.83**	37.47	42.06
Phylogenically inspired					
XLM-R+ [FGLT] (eng)	46.27	37.60	47.15	**40.67**	**42.92**
Ablations					
XLM-R+ [LT] (eng)	46.27	37.20	44.17	40.27	41.98
XLM-R+ [FLT] (eng)	**47.87**	38.27	45.66	38.27	42.52
zero shot w/ mlm baseline:					
XLM-R+mlm (eng)	52.44	37.25	46.21	39.82	43.93

Table 13: NLI Task Results on AmericasNLI (Ebrahimi et al., 2021) languages (base model: XLM-R, metric: ACC).

Celtic

Model Training	bre	wel	gle	gla	glv	avg
MBERT+ [FGLT] (gle)	23.48	23.17	27.60	20.60	13.84	21.74
MBERT+ [RFGLT] (gle)	17.63	21.32	28.40	17.92	9.08	18.87

Germanic

Model Training	afr	dan	deu	eng	fao	got	gsw	isl	nds	nld	nor	swe	avg
MBERT+ [FGLT] (eng)	69.18	76.51	77.79	90.34	76.86	48.28	65.30	73.25	54.88	78.86	81.20	82.59	72.92
MBERT+ [RFGLT] (eng)	63.79	70.82	70.75	84.52	65.79	41.63	53.81	66.55	49.59	70.98	73.99	76.07	65.69

Indic

Model Training	bho	ben	hin	mar	san	urd	xnr	avg
MBERT+ [FGLT] (mar)	16.61	54.69	19.55	58.25	23.67	14.72	32.42	31.42
MBERT+ [RFGLT] (mar)	18.50	31.25	18.55	49.76	17.42	10.61	30.63	25.24

Iranian

Model Training	fas	kmr	avg
MBERT+ [FGLT] (fas)	91.07	41.64	66.35
MBERT+ [RFGLT] (fas)	86.02	36.95	61.49

Romance

Model Training	cat	spa	fre	fro	glg	ita	lig	nap	por	rum	avg
MBERT+ [FGLT] (spa)	90.63	92.44	84.25	58.09	74.74	82.24	68.61	70.0	86.05	82.84	78.99
MBERT+ [RFGLT] (spa)	80.50	82.04	72.94	42.40	68.76	71.60	58.98	50.0	73.48	68.79	66.95

Slavic

Model Training	bel	bul	chu	ces	hrv	orv	pol	qpm	rus	slk	slv	srp	avg
MBERT+ [FGLT] (rus)	77.28	79.98	32.25	78.35	79.17	62.26	80.39	62.57	77.83	82.07	81.48	80.31	72.83
MBERT+ [RFGLT] (rus)	68.77	69.54	28.54	67.72	68.69	55.96	68.59	49.13	65.93	69.05	71.39	72.08	62.95

Table 14: Dependency Parsing Task Results on Indo-European language family (base model: MBERT, metric: UAS).

Family	Genus	Language (Original Family)	ISO 639-3
Random	R1	Bulgarian (Slavic)	bul
	R1	Irish (Celtic)	gle
	R1	Kaapor (Tupian)	urb
	R2	Basque (Language Isolate)	baq
	R2	Komi Zyrian (Uralic)	kpv
	R2	Telugu (Dravidian)	tel
	R3	Faroese (Germanic)	fao
	R3	Hebrew (Semitic)	heb
	R3	Hindi (Indic)	hin

Table 15: Random Language Family construction.