Meeting the Needs of Low-Resource Languages: The Value of Automatic Alignments via Pretrained Models

Abteen Ebrahimi[♢] **Arya D. McCarthy**[▽] **Arturo Oncevay**[♡]
Luis Chiruzzo[△] **John E. Ortega**[Ω] **Gustavo A. Giménez-Lugo**[♣]
Rolando Coto-Solano[ϕ] **Katharina Kann**[♢]
[♢]University of Colorado Boulder [▽]Johns Hopkins University [♡]University of Edinburgh
[△]Universidad de la República, Uruguay [Ω]Northeastern University
[♣]Universidade Tecnológica Federal do Paraná [ϕ]Dartmouth College

Abstract

Large multilingual models have inspired a new class of word alignment methods, which work well for the model's pretraining languages. However, the languages most in need of automatic alignment are low-resource and, thus, not typically included in the pretraining data. In this work, we ask: *How do modern aligners perform on unseen languages, and are they better than traditional methods?* We contribute gold-standard alignments for Bribri–Spanish, Guarani–Spanish, Quechua–Spanish, and Shipibo-Konibo–Spanish. With these, we evaluate state-of-the-art aligners with and without model adaptation to the target language. Finally, we also evaluate the resulting alignments extrinsically through two downstream tasks: named entity recognition and part-of-speech tagging. We find that although transformer-based methods generally outperform traditional models, the two classes of approach remain competitive with each other.

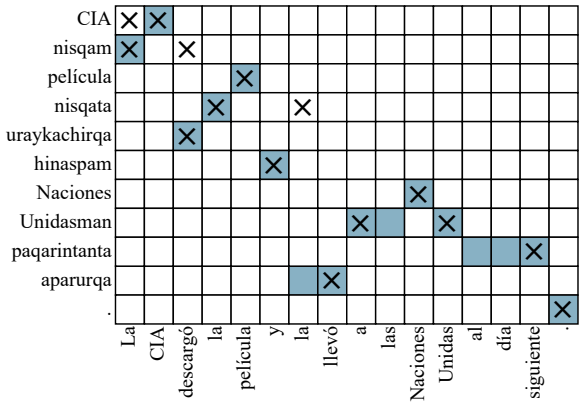


Figure 1: A word alignment between Quechua and Spanish (shaded), as well as mBERT+TLM's predicted alignment (marked by ×'s). FastAlign and Giza++ cannot take advantage of surface features of proper names and borrowings. We evaluate alignments intrinsically via AER and extrinsically with POS-tagging and NER models learned on annotations projected across alignments from Spanish.

1 Introduction

Word alignment is a valuable tool for extending the coverage of natural language processing (NLP) applications to low-resource languages through, e.g., statistical machine translation (SMT; Koehn and Knowles, 2017; Duh et al., 2020) or annotation projection (Yarowsky et al., 2001; Smith and Smith, 2004; Nicolai et al., 2020; Eskander et al., 2020). The traditional approach for generating alignments has been with statistical methods such as Giza++ (Och and Ney, 2003) and FastAlign (Dyer et al., 2013), which provide strong alignment quality while remaining quick and lightweight to run. Recently, new methods have been proposed which extract alignments from massive *pretrained multilingual models*, and outperform these longstanding methods (Dou and Neubig, 2021).

However, results on other NLP tasks, such as part-of-speech (POS) tagging and named-entity recognition (NER), have shown that, while pretrained models generally work well out-of-the-box for high-resource languages, performance is far lower for low-resource ones, particularly those which are unseen during pretraining (Pires et al., 2019; Wu and Dredze, 2020; Muller et al., 2021; Lee et al., 2022). Models can be adapted (Gururangan et al., 2020; Chau et al., 2020) to improve performance, but this comes with a large computational cost. Given these two considerations, for *unseen* low resource languages it remains unclear (1) whether modern neural approaches based on adapted pretrained models generate higher-quality alignments than traditional approaches and (2) if so, whether the quality difference is severe enough to justify the additional computational cost.

We investigate this by collecting gold-standard alignments for Bribri, Guarani, Quechua, and

Our code and data can be found at https://github.com/abteen/alignment.

Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pages 3912–3926
May 2-6, 2023

Shipibo-Konibo. These languages are low-resource and unrepresented in the pretraining data of popular models—a relevant real-world scenario. In addition to intrinsically evaluating alignment quality, we measure the downstream utility of each method for training POS-tagging and NER models by annotation projection.

We find traditional and neural methods to be competitive, but pretrained models result in slightly lower alignment error rates and stronger downstream task performance, even for initially unseen languages. Through further analysis, we also find that adaptation may be a more reliable approach given minimally available resources. Taken together, these results indicate that alignment from multilingual models can indeed be a valuable tool for low-resource languages, but traditional approaches continue to be a strong option and should still be considered for practical applications.

2 Related Work

Alignment Word alignment is a long studied task, with origins in the IBM models for statistical machine translation (Brown et al., 1993), which are the basis of Giza++ (Och and Ney, 2003) and FastAlign (Dyer et al., 2013). As these approaches can only generate one-to-many alignments, models are trained in both forward and reverse directions (reversing the role of source and target), and final alignments are created via symmetrization heuristics (Och and Ney, 2000; Koehn et al., 2005); other approaches explicitly symmetrize during training (Matusov et al., 2004; Liang et al., 2006).[1] While these models rely on only position and word identity information, subword information can be integrated without requiring costly inference (Berg-Kirkpatrick et al., 2010), leading to better parameter estimation for rare words. Alignments can also be extracted from neural translation models (Chen et al., 2020; Zenkel et al., 2020).

Word alignment also enables annotation projection (Yarowsky and Ngai, 2001; Yarowsky et al., 2001) which can offer strong performance, particularly for low-resource languages (Buys and Botha, 2016; Ortega and Pillaipakkamnatt, 2018; Nicolai and Yarowsky, 2019; Nicolai et al., 2020; Eskander et al., 2020).

Multilingual Transformer Models Pretrained multilingual models (Devlin et al., 2019; Conneau and Lample, 2019; Conneau et al., 2020; Xue et al., 2021) have become the de facto standard approach for cross-lingual transfer. In general, these models are an extension of their monolingual variants, created by including data from many languages in their pretraining. They rely on a subword vocabulary (Kudo and Richardson, 2018) which jointly spans all of the pretraining languages. Models are pretrained using a masked language modeling (MLM) objective and a translation language modeling (TLM; Conneau and Lample, 2019) objective that uses parallel sentences. Outside of continued pretraining (Gururangan et al., 2020), models can be adapted using Adapters (Pfeiffer et al., 2020) or through vocabulary adaptation (Wang et al., 2020; Hong et al., 2021). Word alignment methods which depend on these models have also been proposed (Jalili Sabet et al., 2020; Nagata et al., 2020); we focus on AWESoME align (Dou and Neubig, 2021) because it outperforms other unsupervised methods.

3 Experiment 1: Intrinsic Evaluation

3.1 Experimental Setup

Languages We focus on four Indigenous languages spoken in the Americas for our experiments. **Bribri (bzd)** is a tonal language in the Chibchan family spoken by approximately 7000 people in Costa Rica. **Guarani (gn)** is a polysynthetic language in the Tupi–Guarani family spoken by around 6 million people across South America. **Quechua (quy)** is a family of Indigenous languages—from which we study Quechua Chanka—spoken across the Peruvian Andes by over 6 million people, and **Shipibo-Konibo (shp)** is a language spoken by around 30,000 people in Peru, Bolivia, and Brazil (Cardenas and Zeman, 2018). The latter three languages are agglutinative.

Training Data For training, we use the parallel data between Spanish and our languages described by Mager et al. (2021).[2] We note that there is a distinct difference in the amount of unlabeled data available within the four languages: Guarani and Quechua have considerably more data available. These two languages also have monolingual

[1] The poor estimation of rare words' translation parameters also motivates symmetrization; without this, rarely observed words become *garbage collector words* (Moore, 2004).

[2] Although parallel, digitized Bibles exist for over 1600 languages (McCarthy et al., 2020), groups may object to annotating the Bible for historical or cultural reasons.

text available in Wikipedia, which we extract using WikiExtractor (Attardi, 2015). The exact number of parallel and monolingual sentences for all languages is shown in Table B.1.

Evaluation Data To create gold standard alignments for evaluation, we sample multi-way parallel examples from AmericasNLI (Ebrahimi et al., 2022), allowing for multi-parallel alignments (Xia and Yarowsky, 2017) across all languages. Samples for the development and test sets are taken from their respective splits in the AmericasNLI dataset. Development examples were collected first, manually checked, and corrected. Examples with misalignments in punctuation, numbers, or named entities were not used. After a period of development with this data, the test set of 50 examples was collected and manually verified. Annotations were collected using JHU's open-source Turkle platform.[3] We ask annotators to only mark *sure* alignments. Additional discussion on data collection and the test set can be found in §6.

Metrics We evaluate automatic alignments via alignment error rate (AER; Och and Ney, 2000). Because we only collect sure alignments, this is equivalent to the balanced F-measure (Fraser and Marcu, 2007). We give additional metrics in Table C.3.

3.2 Models

Traditional Aligners We use Giza++ (Och and Ney, 2003) and FastAlign (Dyer et al., 2013) as our traditional aligners. Giza++ is based on IBM Models 1–5 (Brown et al., 1993). FastAlign (Dyer et al., 2013) is a re-parameterization of IBM Model 2. We use the implementation and hyperparameters of Zenkel et al. (2020), which relies on MGiza++ (Gao and Vogel, 2008) and the standard FastAlign package. Both approaches run on CPUs, and their training time ranges between 6 seconds to 3 minutes for FastAlign, and 43 seconds to 22 minutes for Giza++. We use the union of the forward and reverse alignments, as this symmetrization heuristic offers the best result for all languages on the development set. We show the performance of other heuristics in Table C.2.

Neural Aligners AWESoME (Dou and Neubig, 2021) identifies alignment links by considering cosine similarities between hidden layer representations of tokens in a neural encoder. We consider two such encoders: mBERT (Devlin et al., 2019) and XLM-R (Liu et al., 2019), and we use the default AWESoME configuration to extract alignments. We give layer-by-layer alignment performance in Figure C.1.

[3] https://github.com/hltcoe/turkle

Model	Method	BZD	GN	QUY	SHP	AVG.
AWESoME (mBERT)	BL	70.03	63.13	67.02	60.41	65.15
	+MLM-T	68.95	49.68	46.59	58.17	55.85
	+MLM-ST	70.63	50.25	42.52	58.66	55.52
	+TLM	58.43	**43.10**	**36.96**	**52.34**	**47.71**
AWESoME (XLM-R)	BL	80.15	73.11	75.24	69.21	74.43
	+MLM-T	76.89	65.44	53.65	65.16	65.29
	+MLM-ST	77.53	64.55	52.90	66.56	65.39
	+TLM	74.90	58.84	43.25	63.48	60.12
FastAlign	Union	**51.40**	43.52	54.06	54.67	50.91
Giza++	Union	55.61	49.92	66.01	60.84	58.10
mBERT	+MLM-WT	-	***40.00***	46.00	-	43.00
XLM-R	+MLM-WT	-	52.27	48.83	-	50.55

Table 1: AER, in percentages, for each language and method. The best overall result for each language is bolded, while the best model within each method is underlined. We separate results which use Wikipedia, as they are not directly comparable.

Model Adaptation We experiment with three adaptation schemes based on continued pretraining (+TLM, +MLM-T, and +MLM-ST) which rely on unlabeled data and further train the model using MLM (Gururangan et al., 2020) before alignments are extracted. We focus on these objectives as they have been used by prior work for general model adaptation, and they work well in situations with limited resources (Ebrahimi and Kann, 2021). As we have access to bitext between Spanish and the target languages, for the +TLM scheme each example is the concatenation of a Spanish sentence with its translation. For +MLM-**T** we adapt using solely the **t**arget side of the available data, and for +MLM-**ST** we adapt on both the **s**ource and **t**arget; however, this data is treated as monolingual data and not explicitly aligned. +MLM-WT denotes target language adaptation which includes Wikipedia data. The duration of adaptation depends on the GPU and method used; it ranges from around 6 minutes for Bribri to 4 hours for Quechua. We provide additional training details in Appendix A.

3.3 Results

Traditional vs. Neural Aligners We present results in Table 1. The best traditional method is FastAlign, and the best neural approach is with mBERT+TLM. Comparing the two, we see that

the lowest error rate is achieved with the neural approach for all languages except for Bribri, where FastAlign offers 7.03% absolute improvement. Of the other three languages, the performance for two is close: the difference in performance for Guarani is only 0.42% and 2.33% for Shipibo-Konibo. For Quechua, +TLM improves over FastAlign by 17.10%.

Comparing Adaptation Strategies With mBERT, +MLM-T improves performance over the non-adapted baseline by 9.30% on average, with +MLM-ST increasing this gain to 9.63% and +TLM offering the highest improvement of 17.44%, consistent with prior work on *seen* languages (Dou and Neubig, 2021). Per language, the largest and smallest gains are for Quechua (30.06%) and for Shipibo-Konibo (8.07%); intuitively, gains from adaptation are proportional to the size of the adaptation data. For XLM-R, we again see relative gains from adaptation, with +TLM offering the highest performance increase.

Additional Monolingual Data Neural approaches can easily benefit from additional monolingual data. Adding Wikipedia data results in the highest performance for Guarani, outperforming the previous best approach by 3.1%. In contrast, while the additional data for Quechua does help relative to +MLM-T, it does not outperform +TLM. This difference in performance may be due to the relative sizes of the additional data; the Guarani Wikipedia has 1.3× as many tokens as the target-side parallel data, while the Quechua Wikipedia only has 0.5× as many.

4 Experiment 2: Extrinsic Evaluation

We further compare aligner performance extrinsically by evaluating downstream task performance when using a projected training set. We consider two tasks: NER and POS tagging.

4.1 Experimental Setup

Data Due to the limited availability and quality of evaluation datasets, we focus on Guarani for this experiment. We use the test set provided by Rahimi et al. (2019) for NER and Universal Dependencies (Nivre et al., 2020) for POS. For experiments where we finetune directly on English or Spanish, we use the provided training data.

Model	Train Source	POS	NER
mBERT	en	10.36	46.64
	es	19.82	49.18
+TLM	es	36.94	49.62
+MLM-T	es	34.69	**55.25**
+MLM-ST	es	33.78	52.34
mBERT	mBERT	31.53	47.54
	+MLM-T	38.29	47.97
	+MLM-ST	**42.34**	49.80
	+TLM	40.99	49.80
	FastAlign	37.84	46.55
	Giza++	39.19	48.33

Table 2: POS tagging (accuracy) and NER results (F1) for Guarani. *Model* denotes if baseline or adapted mBERT is used. *Train Source* defines the training data used for finetuning; language codes indicate training on original data, while alignment methods denote how a projected training set was created.

Annotation Projection To create the projected training sets, we first annotate the (unlabeled) Spanish parallel data with Stanza (Qi et al., 2020) and generate bidirectional alignments using each method. We then project the tags from Spanish to Guarani using type and token constraints as described by Buys and Botha (2016).

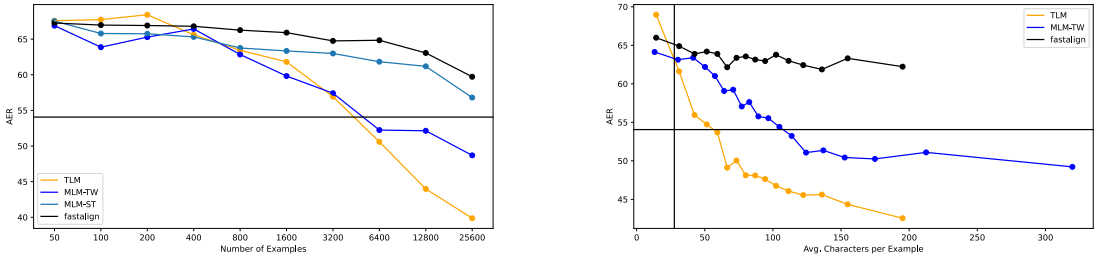
Models For baseline performance, we finetune mBERT on the provided English and Spanish training sets for each task. Additionally, we also finetune adapted versions of mBERT on Spanish training data – English is omitted as performance is worse and adaptation data is in Spanish. Finally, we evaluate performance when finetuning mBERT on the training sets created through projection.

4.2 Results

We present results for both tasks in Table 2.

POS For POS tagging, the baseline zero-shot performance is extremely poor, and we see a minimum increase of 11.71% accuracy when using any projection method. Giza++ outperforms FastAlign, as well as projection with +MLM-T, however the best performance is achieved with +MLM-ST, with +TLM offering the second best result. While the ordering of methods changes, the best performance is still achieved with the neural approaches, consistent with the results of Experiment 1.

NER For NER, baseline performance is high: inspecting the data shows that many entities have English or Spanish names, and as multilingual models already have knowledge of these two languages,



(a) Subset Analysis

(b) Length Analysis

Figure 2: Plots for data analysis. In Figure 2b, a vertical line denotes the average example length for Bribri.

standard aligners with projection may not effectively leverage surface word-form clues. However, they remain a valuable indication of alignment quality. Among the projection-based approaches, we find that using Giza++ again outperforms +MLM-T and FastAlign but falls short of +MLM-ST and +TLM.

Overall, considering what both downstream tasks indicate regarding alignment quality, neural models adapted using Spanish and target-language data—either sentence-aligned or unaligned—consistently outperform traditional methods.

5 Analysis

As data for low-resource languages often varies considerably in both amount and length, we consider two additional analysis experiments which control for these factors. We focus solely on Quechua, as it has the most parallel data available. Results are presented in Figure 2 with numerical results in Tables C.4 and C.5.

Subset Analysis For this analysis, we ask how the performance of neural alignment depends on the amount of data and with how much data it surpasses traditional approaches. We subsample the adaptation data, and use this to extract alignments using both FastAlign and AWESoME. Results for this experiment can be seen in Figure 2a. For reference, we also plot the AER obtained when using FastAlign on all the available training data as an upper bound for the performance of the traditional approaches. In the smallest extreme, all methods are roughly equivalent. However, as the number of examples increases, adaptation using +TLM and +MLM-WT improves at a faster rate than other approaches: with only 6400 sentence pairs, these approaches overtake the best expected performance of FastAlign.

Length Analysis Aligner performance may not only be affected by the total number of examples available, but also by the length of these examples. This is doubly relevant for low-resource languages, as resources may be limited to sources which do not contain long (or even complete) sentences. To see how the performance of each method may vary when faced with examples of different lengths, we sort the unlabeled data by the number of characters, and partition the examples in groups of 7508, the total number of examples available for Bribri. We choose this amount as it is representative how much data may be available for other low-resource languages. As before, the expected upper bound FastAlign performance is denoted. For the shortest group, all methods are similar; however, AWESoME alignments improve with longer sequences, with +TLM showing the quickest decrease in error rate. We attribute the improved AER when adapting using longer sequences to the increased number of tokens available for adaptation. For Quechua, the performance of AWESoME align is sensitive to both the number of examples and sequence length. In contrast, FastAlign only shows a small improvement as example length increases.

6 Conclusion

In this work, we have investigated the performance of modern word aligners versus classical approaches for languages *unseen* to pretrained models. While classical methods remain competitive, the lowest AER on average is achieved by modern neural approaches. However, using these models comes with a larger computational cost. Therefore, the trade-off between training requirements and overall performance must be considered. If access to computing resources is limited or training time is a factor, classical approaches remain a viable approach which should not be discounted.

Ethics and Limitations

Ethics Statement

When collecting data in an Indigenous language, it becomes vital that the process does not exploit any member of the community or commodify the language (Schwartz, 2022). Further, it is important that members of the community benefit from the dataset. While the creation of a word alignment dataset will not directly impact community members, we believe that it can contribute to the development of tools, such as translation systems, that can be directly beneficial, and that increasing the visibility of these languages within the research community will further spur the creation of useful systems. Our annotations were created by either co-authors of the paper or by native speakers of the languages, who were compensated at a rate chosen with the minimum hourly salary in their respective countries taken into account.

Limitations

Test Set Size One limitation of our work is the size of the evaluation set used for our main results. This arises from the general difficulty in collecting annotations and data for low-resource, and particularly Indigenous languages. The size of the test set was chosen to balance the trade-off between the cost of annotation collection and experimental validity. Fortunately, for the task of word alignment the main metric used to summarize performance—alignment error rate—does not depend directly on the number of examples in the evaluation set, but the total number of alignments, of which there are a sufficiently high number in our evaluation set. However, even when only considering the number of examples, our test set is still within the same order of magnitude as other widely used word alignment evaluation sets, such as the Romanian–English test set which consists of 248 examples (Mihalcea and Pedersen, 2003), and the English–Inuktitut and English–Hindi test sets which have 75 and 90 examples each, respectively (Martin et al., 2005).

We run a small experiment to gain insight into how much precision is lost when using a test set of size 50, versus 248, which we choose as this is the size of the widely used Romanian–English test set mentioned above. We take 100 independent samples without replacement from the Romanian-English test set, each of size 50, and evaluate the performance of FastAlign and AWESoME align. For FastAlign, we use the training data defined by Mihalcea and Pedersen (2003), and for Awesome, we use mBERT with no additional finetuning. The distributions of AER are shown in Figure A.1, with summary statistics in Table A.1. We can see that the standard deviation of both distributions is relatively low, around 2%. At the extremes, we see a difference of -4.70% and $+4.90\%$, and -4.28% and $+6.4\%$ for FastAlign and AWESoME align respectively, between the min/max values of our distribution as compared to the whole set AER. Considering these points, we believe that the size of our evaluation set does not invalidate our experimental results and main conclusions; however, we note that additional care must be taken when comparing specific models whose performances are close together, particularly when this performance is low or close to random.

Test Set Domain Other limitations of our work arise from the sources of data used. Annotations were done using sentences sampled from AmericasNLI, which itself is a translation of XNLI. As such, any errors from the original XNLI dataset, which may have propagated through translation, will persist in our dataset as well (annotators were given the option to modify target language sentences to correct any errors). Furthermore, due to translation, the sentences may not be directly representative of a natural utterance which would be spoken by members of the communities.

Language Selection The languages we highlight in this work are true low-resource languages, and present challenges commonly faced by other low-resource languages. Namely, these languages have a relatively small amount of easily available and clean unlabeled data, are typically unseen from most released pretrained models, and are morphologically different from typically used source languages. However, one feature of these languages which may inflate aligner performance is the language script: all of our target languages share the same script with the two source languages which we use. This may lead to higher occurrences of shared words or entities, making alignment easier. As such, our results may not generalize fully to other low-resource languages which have a different script from the source languages, or which may have a script which is unseen to the underlying pretrained model.

Acknowledgements

We would like to thank Roque Helmer Luna-Montoya (Academia Mayor de la Lengua Quechua in Cuzco, Perú) and Richard Castro Mamani (Universidad Nacional de San Antonio Abad and Hinantin Software) for their help in annotating and verifying the Quechua–Spanish alignments. We would also like to thank Liz Karen Chavez Sanchez for annotating the Shipibo-Konibo–Spanish alignments. A.D.M. is supported by an Amazon Fellowship and a Frederick Jelinek Fellowship.

References

Giuseppe Attardi. 2015. Wikiextractor. https://github.com/attardi/wikiextractor.

Taylor Berg-Kirkpatrick, Alexandre Bouchard-Côté, John DeNero, and Dan Klein. 2010. Painless unsupervised learning with features. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 582–590, Los Angeles, California. Association for Computational Linguistics.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

Jan Buys and Jan A. Botha. 2016. Cross-lingual morphological tagging for low-resource languages. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1954–1964, Berlin, Germany. Association for Computational Linguistics.

Ronald Cardenas and Daniel Zeman. 2018. A morphological analyzer for Shipibo-konibo. In *Proceedings of the Fifteenth Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 131–139, Brussels, Belgium. Association for Computational Linguistics.

Ethan C. Chau, Lucy H. Lin, and Noah A. Smith. 2020. Parsing with multilingual BERT, a small corpus, and a small treebank. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1324–1334, Online. Association for Computational Linguistics.

Yun Chen, Yang Liu, Guanhua Chen, Xin Jiang, and Qun Liu. 2020. Accurate word alignment induction from neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 566–576, Online. Association for Computational Linguistics.

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 and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

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.

Zi-Yi Dou and Graham Neubig. 2021. Word alignment by fine-tuning embeddings on parallel corpora. In *Conference of the European Chapter of the Association for Computational Linguistics (EACL)*.

Kevin Duh, Paul McNamee, Matt Post, and Brian Thompson. 2020. Benchmarking neural and statistical machine translation on low-resource African languages. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2667–2675, Marseille, France. European Language Resources Association.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia. Association for Computational Linguistics.

Abteen Ebrahimi and Katharina Kann. 2021. How to adapt your pretrained multilingual model to 1600 languages. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4555–4567, Online. Association for Computational Linguistics.

Abteen Ebrahimi, Manuel Mager, Arturo Oncevay, Vishrav Chaudhary, Luis Chiruzzo, Angela Fan, John Ortega, Ricardo Ramos, Annette Rios, Ivan Vladimir Meza Ruiz, Gustavo Giménez-Lugo, Elisabeth Mager, Graham Neubig, Alexis Palmer, Rolando Coto-Solano, Thang Vu, and Katharina Kann. 2022. AmericasNLI: Evaluating zero-shot natural language understanding of pretrained multilingual models in truly low-resource languages. In

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 6279–6299, Dublin, Ireland. Association for Computational Linguistics.

Ramy Eskander, Smaranda Muresan, and Michael Collins. 2020. Unsupervised cross-lingual part-of-speech tagging for truly low-resource scenarios. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4820–4831, Online. Association for Computational Linguistics.

Alexander Fraser and Daniel Marcu. 2007. Squibs and discussions: Measuring word alignment quality for statistical machine translation. *Computational Linguistics*, 33(3):293–303.

Qin Gao and Stephan Vogel. 2008. Parallel implementations of word alignment tool. In *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*, pages 49–57, Columbus, Ohio. Association for Computational Linguistics.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Jimin Hong, TaeHee Kim, Hyesu Lim, and Jaegul Choo. 2021. AVocaDo: Strategy for adapting vocabulary to downstream domain. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4692–4700, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Masoud Jalili Sabet, Philipp Dufter, François Yvon, and Hinrich Schütze. 2020. SimAlign: High quality word alignments without parallel training data using static and contextualized embeddings. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1627–1643, Online. Association for Computational Linguistics.

Philipp Koehn, Amittai Axelrod, Alexandra Birch Mayne, Chris Callison-Burch, Miles Osborne, and David Talbot. 2005. Edinburgh system description for the 2005 IWSLT speech translation evaluation. In *Proceedings of the Second International Workshop on Spoken Language Translation*, Pittsburgh, Pennsylvania, USA.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39, Vancouver. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

En-Shiun Lee, Sarubi Thillainathan, Shravan Nayak, Surangika Ranathunga, David Adelani, Ruisi Su, and Arya McCarthy. 2022. Pre-trained multilingual sequence-to-sequence models: A hope for low-resource language translation? In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 58–67, Dublin, Ireland. Association for Computational Linguistics.

Percy Liang, Ben Taskar, and Dan Klein. 2006. Alignment by agreement. In *Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, HLT-NAACL '06, page 104–111, USA. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692.

Manuel Mager, Arturo Oncevay, Abteen Ebrahimi, John Ortega, Annette Rios, Angela Fan, Ximena Gutierrez-Vasques, Luis Chiruzzo, Gustavo Giménez-Lugo, Ricardo Ramos, Ivan Vladimir Meza Ruiz, Rolando Coto-Solano, Alexis Palmer, Elisabeth Mager-Hois, Vishrav Chaudhary, Graham Neubig, Ngoc Thang Vu, and Katharina Kann. 2021. Findings of the AmericasNLP 2021 shared task on open machine translation for indigenous languages of the Americas. In *Proceedings of the First Workshop on Natural Language Processing for Indigenous Languages of the Americas*, pages 202–217, Online. Association for Computational Linguistics.

Joel Martin, Rada Mihalcea, and Ted Pedersen. 2005. Word alignment for languages with scarce resources. In *Proceedings of the ACL Workshop on Building and Using Parallel Texts*, pages 65–74, Ann Arbor, Michigan. Association for Computational Linguistics.

Evgeny Matusov, Richard Zens, and Hermann Ney. 2004. Symmetric word alignments for statistical machine translation. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 219–225, Geneva, Switzerland. COLING.

Arya D. McCarthy, Rachel Wicks, Dylan Lewis, Aaron Mueller, Winston Wu, Oliver Adams, Garrett Nicolai, Matt Post, and David Yarowsky. 2020. The Johns Hopkins University Bible corpus: 1600+ tongues for typological exploration. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2884–2892, Marseille, France. European Language Resources Association.

Rada Mihalcea and Ted Pedersen. 2003. An evaluation exercise for word alignment. In *Proceedings of the HLT-NAACL 2003 Workshop on Building and Using Parallel Texts: Data Driven Machine Translation and Beyond*, pages 1–10.

Robert C. Moore. 2004. Improving IBM word alignment model 1. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 518–525, Barcelona, Spain.

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.

Masaaki Nagata, Katsuki Chousa, and Masaaki Nishino. 2020. A supervised word alignment method based on cross-language span prediction using multilingual BERT. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 555–565, Online. Association for Computational Linguistics.

Garrett Nicolai, Dylan Lewis, Arya D. McCarthy, Aaron Mueller, Winston Wu, and David Yarowsky. 2020. Fine-grained morphosyntactic analysis and generation tools for more than one thousand languages. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3963–3972, Marseille, France. European Language Resources Association.

Garrett Nicolai and David Yarowsky. 2019. Learning morphosyntactic analyzers from the Bible via iterative annotation projection across 26 languages. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1765–1774, Florence, Italy. Association for Computational Linguistics.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal Dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4034–4043, Marseille, France. European Language Resources Association.

Franz Josef Och and Hermann Ney. 2000. Improved statistical alignment models. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, pages 440–447, Hong Kong. Association for Computational Linguistics.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

John Ortega and Krishnan Pillaipakkamnatt. 2018. Using morphemes from agglutinative languages like Quechua and Finnish to aid in low-resource translation. In *Proceedings of the AMTA 2018 Workshop on Technologies for MT of Low Resource Languages (LoResMT 2018)*, pages 1–11, Boston, MA. Association for Machine Translation in the Americas.

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

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.

Afshin Rahimi, Yuan Li, and Trevor Cohn. 2019. Massively multilingual transfer for NER. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 151–164, Florence, Italy. Association for Computational Linguistics.

Lane Schwartz. 2022. Primum Non Nocere: Before working with Indigenous data, the ACL must confront ongoing colonialism. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 724–731, Dublin, Ireland. Association for Computational Linguistics.

David A. Smith and Noah A. Smith. 2004. Bilingual parsing with factored estimation: Using English to parse Korean. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 49–56, Barcelona, Spain. Association for Computational Linguistics.

Zihan Wang, Karthikeyan K, Stephen Mayhew, and Dan Roth. 2020. Extending multilingual BERT to low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2649–2656, Online. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame,

Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Shijie Wu and Mark Dredze. 2020. Are all languages created equal in multilingual BERT? In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 120–130, Online. Association for Computational Linguistics.

Patrick Xia and David Yarowsky. 2017. Deriving consensus for multi-parallel corpora: an English Bible study. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 448–453, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

David Yarowsky and Grace Ngai. 2001. Inducing multilingual POS taggers and NP bracketers via robust projection across aligned corpora. In *Second Meeting of the North American Chapter of the Association for Computational Linguistics*.

David Yarowsky, Grace Ngai, and Richard Wicentowski. 2001. Inducing multilingual text analysis tools via robust projection across aligned corpora. In *Proceedings of the First International Conference on Human Language Technology Research*.

Thomas Zenkel, Joern Wuebker, and John DeNero. 2020. End-to-end neural word alignment outperforms GIZA++. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1605–1617, Online. Association for Computational Linguistics.

A Training Details and Hyperparameters

We compare two data loading strategies for adaptation: a naïve approach where each example in the dataset represents an example in the loaded training examples, and a packing strategy following the FULL-SENTENCES approach of Liu et al. (2019). We use the hyperparameters described by Ebrahimi et al. (2022) – a learning rate of 2e-5, batch size of 32, and warmup ratio of 1% – however due to the different loading strategy we tune the total amount of training time. We experiment with 40 and 80 epochs of training, using the alignment development set to select the final hyperparameters. For both MLM-T and MLM-ST we find that packing sequences yields better results, however for +TLM we use the naïve strategy to preserve sentence alignment. We use packing by default for Wikipedia data, due to the length of extracted documents. For all adaptation methods we find that training for 80 epochs is best, except for +MLM-ST, which we train for 40. We train with 1 Nvidia A100 or 2 V100 GPUs. Due to the computational cost associated with pretraining, we only conduct one model run for each language and method. We pretrain our models using Huggingface (Wolf et al., 2020).

Training Time As mentioned in Section 3.2, for adaptation the training duration depends on the GPU and method used, with times ranging from around 6 minutes for Bribri to 4 hours for Quechua. For the statistical approaches, both run solely on CPUs, and their training time ranges between 6 seconds to 3 minutes for FastAlign, and 43 seconds to 22 minutes for Giza++. However, GPU availability is not always certain – to roughly compare training times given a more restricted setting, we run our adaptation experiments without access to any GPUs, and compute an estimate for the total training time using only CPUs as approximately 2 weeks.

	Whole Set AER	Avg. AER	AER Std.	Min AER	25%	50%	75%	Max AER
FastAlign	35.00	35.09	1.94	30.30	33.70	35.00	36.23	39.90
Awesome	28.23	28.26	2.04	23.95	26.66	28.12	29.71	34.63

Table A.1: Summary statistics for subsample AER distribution.

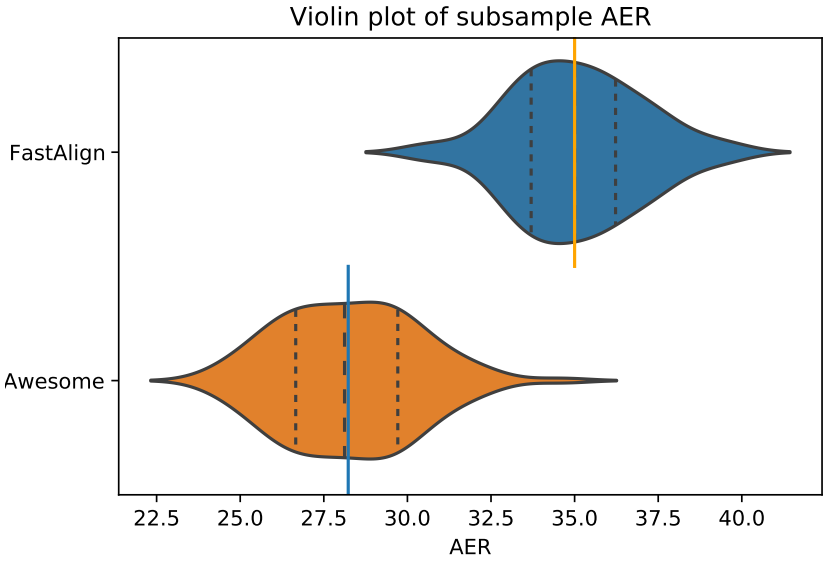


Figure A.1: Distribution of AER when using FastAlign and AWESoME align to evaluate subsets of size 50 taken from a complete evaluation set of size 248. Quartiles are displayed using dashed lines, while inverted colors represent the AER calculated when evaluating on the complete set.

B Dataset Features

Feature	BZD	GN	QUY	SHP
Number of examples (Parallel)	7,508	26,032	121,064	14,592
Number of examples (Wiki)	-	4721	22610	-
Number of tokens - MLM-T	123,992	1,104,645	3,912,582	179,451
Number of tokens - TLM	194,798	2,006,996	6,697,771	328,427
Number of tokens - Wiki	-	1,460,240	2,023,297	-
Number of dev examples	50	48	45	46
Number of test examples	50	50	50	50

Table B.1: Features of the data used for our experiments.

C Supplementary Results

Model	Method	BZD	GN	QUY	SHP
AWESoME	BL	65.38	58.51	63.98	62.80
+mBERT	+MLM-T	64.26	43.29	39.10	66.44
	+MLM-ST	65.43	43.46	37.20	65.63
	+TLM	54.25	34.62	30.38	62.10
AWESoME	BL	76.29	71.85	71.53	73.96
+XLM-R	+MLM-T	72.73	57.50	43.30	69.25
	+MLM-ST	73.08	60.28	44.88	70.48
	+TLM	71.88	49.76	36.11	69.23
FastAlign	Union	47.39	39.78	58.37	57.91
Giza++	Union	51.03	62.07	47.18	64.98

Table C.1: Development AER for each language and method.

Method	Heuristic	BZD	GN	QUY	SHP
FastAlign	grow-diagonal-final	54.56	49.64	60.51	56.11
	grow-diagonal	55.36	50.41	63.81	56.87
	intersection	57.11	52.89	66.92	61.67
	union	51.40	43.52	54.06	54.67
	reverse	52.21	51.51	61.27	58.41
Giza++	grow-diagonal-final	55.51	53.38	75.29	62.72
	grow-diagonal	59.33	58.41	80.06	69.53
	intersection	63.71	64.95	82.55	77.41
	union	55.61	49.92	66.01	60.84
	reverse	56.43	62.20	76.05	72.39

Table C.2: AER results on the test set for various growing heuristics.

		BZD			GN			QUY			SHP		
Model	Method	P	R	F	P	R	F	P	R	F	P	R	F
AWESoME (mBERT)	BL	41.8	23.4	30.0	50.6	29.0	36.9	49.4	24.7	33.0	64.0	28.7	39.6
	+MLM-T	42.7	24.4	31.1	68.6	39.7	50.3	69.1	43.5	53.4	66.0	30.6	41.8
	+MLM-ST	42.9	22.3	29.4	67.2	39.5	49.8	73.8	47.1	57.5	67.6	29.8	41.3
	+TLM	62.1	31.2	41.6	76.3	45.4	56.9	79.0	52.5	63.0	79.4	34.0	47.7
AWESoME (XLM-R)	BL	48.0	12.5	19.9	48.4	18.6	26.9	49.7	16.5	24.8	64.5	20.2	30.8
	+MLM-T	38.9	16.4	23.1	63.0	23.8	34.6	70.2	34.6	46.4	57.2	25.0	34.8
	+MLM-ST	40.8	15.5	22.5	65.7	24.3	35.4	74.8	34.4	47.1	56.5	23.7	33.4
	+TLM	50.0	16.8	25.1	76.9	28.1	41.2	83.2	43.1	56.8	77.0	23.9	36.5
FastAlign	Union	46.4	51.0	48.6	55.4	57.6	56.5	44.3	47.7	45.9	48.0	43.0	45.3
Giza++	Union	39.9	49.8	44.3	48.3	52.0	50.1	32.0	36.3	34.0	37.2	41.4	39.2
mBERT	+MLM-WT	-	-	-	76.3	49.4	60.0	70.8	43.6	54.0	-	-	-
XLM-R	+MLM-WT	-	-	-	66.4	31.5	42.7	75.0	38.8	51.2	-	-	-

Table C.3: Precision, recall, and F-measure for main test set results. All metrics are on a 0–100 scale (larger is better).

Num. Examples	+TLM	+MLM-WT	+MLM-ST	FastAlign
50	67.58	66.89	67.53	67.26
100	67.74	63.87	65.79	66.97
200	68.42	65.28	65.75	66.91
400	65.61	66.43	65.31	66.80
800	63.43	62.84	63.76	66.26
1600	61.81	59.82	63.34	65.91
3200	56.93	57.41	62.99	64.75
6400	50.59	52.24	61.83	64.84
12800	43.98	52.14	61.18	63.06
25600	39.87	48.69	56.80	59.72

Table C.4: AER for each method and subset used in the Subset Analysis.

+MLM-WT		+TLM		FastAlign	
Avg. Char	AER	Avg. Char	AER	Avg. Char	AER
13.20	64.13	14.31	68.97	14.31	65.99
30.29	63.13	31.20	61.61	31.20	64.89
41.49	63.39	42.51	55.95	42.51	63.89
50.19	62.19	51.45	54.73	51.45	64.19
57.47	61.01	59.23	53.70	59.23	63.89
64.20	59.07	66.44	49.12	66.44	62.15
70.83	59.24	73.30	50.04	73.30	63.38
77.12	57.06	80.09	48.12	80.09	63.56
82.63	57.63	87.02	48.10	87.02	63.15
89.31	55.77	94.31	47.63	94.31	62.96
96.66	55.54	102.30	46.76	102.30	63.78
104.76	54.40	111.48	46.07	111.48	62.99
113.76	53.24	122.29	45.56	122.29	62.43
124.33	51.07	135.93	45.62	135.93	61.87
137.03	51.36	154.86	44.35	154.86	63.31
152.70	50.43	195.18	42.55	195.18	62.23
174.88	50.25	-	-	-	-
212.44	51.10	-	-	-	-
319.76	49.22	-	-	-	-

Table C.5: AER for each method and length group used in the Length Analysis. Average Chars represents the average number of characters per example, for each group.

Figure C.1: AER using the development set, per layer, per language, for both mBERT and XLM-R.