Combining static and contextualised multilingual embeddings

Anonymous ACL submission

Abstract

Static and contextual multilingual embeddings have complementary strengths. Static embeddings, while less expressive than contextual language models, can be more straightforwardly aligned across multiple languages. Contextual language models are more powerful. We combine the strengths of static and contextual models to improve multilingual representations. We extract static embeddings for 40 languages from XLM-R, validate those embeddings with cross-lingual word retrieval, and then align them using VecMap. This results in high-quality, highly multilingual static embeddings. Then we apply a novel continued pre-training approach to XLM-R, leveraging the high quality alignment of our static embeddings to better align the representation space of XLM-R. We show positive results for multiple complex semantic tasks. We will release the static embeddings and the continued pre-training code.

1 Introduction

Multilingual contextual encoders like XLM-R (Conneau et al., 2020a) and mBERT (Devlin et al., 2019), despite being trained without parallel data, exhibit "surprising" cross-linguality (Wu and Dredze, 2019; Conneau et al., 2020b) and have demonstrated strong performance on multilingual and cross-lingual tasks (e.g., Hu et al., 2020; Lauscher et al., 2020; Kurfalı and Östling, 2021; Turc et al., 2021). However, their *language-neutrality*, meaning how well languages are aligned with each other, has clear limits (Libovický et al., 2020; Cao et al., 2020, inter alia). In particular, more typologically distant language pairs tend to be less well-aligned than more similar ones, affecting transfer performance.

By contrast, cross-lingual alignment is well-studied for static embeddings (e.g., Mikolov et al., 2013; Vulić et al., 2020), and they can be aligned using simple transformation matrices, resulting in high quality multilingual embeddings. However, static embeddings are considerably less expressive than contextual models and have in many applications been superseded by them.

This paper aims to combine the strengths of static and contextual models, and explore how they may benefit from each other. Our method requires no parallel corpus. Monolingual static embeddings have been extracted from BERT by Gupta and Jaggi (2021). We show that their approach can be applied to multilingual embeddings. To our knowledge, we are the first to explore the extraction of static embeddings from a multilingual contextual model. We distill static embeddings for 40 languages from XLM-R, showing that the resulting embeddings are already somewhat cross-lingually aligned, but that their alignment can be improved using established tools (Section 3). These vectors are of high monolingual and cross-lingual quality despite being distilled using only 1M sentences per language. Second, we present a novel continued pre-training approach for the contextual model, combining masked language modelling (MLM) with an alignment loss that leverages the well-aligned static embeddings (Section 4). This results in improved multilingual contextualised embeddings which work well for complex semantic tasks.

2 Contextual and Static Embeddings

XLM-R (Conneau et al., 2020a) and mBERT (Devlin et al., 2019) have been successful in multi- and cross-lingual transfer despite being trained only on monolingual corpora. However, the 100 languages in XLM-R—or 104 in mBERT—are not represented equally well (cf. Wu and Dredze, 2020), either in terms of data size or downstream performance. Both Singh et al. (2019) and Libovický et al. (2020) found that mBERT clusters its representations of languages in a way that mirrors typological language family trees. However, representations being well-aligned across languages is related to better cross-lingual transfer performance, so this property limits the model's transfer ability especially for more distant language pairs.

In comparison, static embeddings are far less resource-intensive than contextual models, both at training and inference time. They can be trained with smaller data and achieve good representation quality where a Transformer model would be undertrained. Where time, data, or computational resources are limited, this makes static embeddings an attractive approach. Also, some NLP tasks rely on static embeddings in their formulation, such as lexical evaluation tasks, approaches comparing vector spaces to detect domain shift (Beyer et al., 2020) or linguistic change (Shoemark et al., 2019), or some bias detection and removal tasks (e.g., Kaneko and Bollegala, 2019; Manzini et al., 2019). Finally, importantly for us, cross-lingual alignment has been studied extensively in static embeddings (e.g., Artetxe et al., 2018a,b; Joulin et al., 2018). Especially those languages that are ill-represented in the massively multilingual model can benefit from using static embeddings. In summary, static and contextual representations have complementary strengths.

3 Static Embeddings from XLM-R

Gupta and Jaggi (2021) extracted English static embeddings from BERT and RoBERTa. They showed that their CBOW-like training scales better with more data and outperforms an aggregation approach to extracting static embeddings (Bommasani et al., 2020). In their system, X2Static, the context vector from which to predict the target word is given by the average of all vectors in the sentence without the target word. The method uses ten negative samples per target and calculates the loss based on similarity scores. However, they only evaluated their method on English. We are the first to extract static embeddings from a *multilingual* contextual model.

3.1 Extraction and Alignment Process

We choose 40 languages for static embeddings extraction. See Appendix A for the full list. As the multilingual contextual model, we use XLM-R. Due to the large number of languages and due to having limited data for some of them, we decided to use only up to 1M sentences per language for extraction. From preliminary experimentation with English, German and French, we determined how best to extract multilingual embeddings from the model: First, using X2Static (Gupta and Jaggi, 2021) worked better than aggregation (Bommasani et al., 2020) even with a small amount of data. One important difference with Gupta and Jaggi's work is that for our task the sentence-level variant of X2Static yielded better results than the paragraph-level version. Crucially, we also found that embeddings extracted from layer 6 of XLM-R performed noticeably better than embeddings extracted from the output layer. The latter fits with findings for mBERT by Muller et al. (2021) that the middle layers are more multilingually aligned.

Model	en-xx	xx-en
fasttext$_{unsup}$	54.71	58.26
X2S-M	52.11	59.00
X2S-MA	58.41	65.60
MUSE (Conneau et al., 2018)	58.88	65.21
RCSLS (Joulin et al., 2018)	**67.47**	**71.70**

Table 1: Results from MUSE BLI tasks. Scores are averaged over those language pairs present in all models. Even before alignment (X2S-M), the embeddings derived from XLM-R are competitive with fasttext vectors aligned using unsupervised VecMap (fasttext$_{unsup}$). After alignment and selection (X2S-MA), they are on-par with the supervised embeddings released by MUSE despite using much smaller data to train. We show per-language results in Table 5.

For the full set of embeddings, we used up to 1M sentences per language from the reconstructed CC100 corpus by Wenzek et al. (2020). We filtered out headlines and too-short sentences heuristically. See Appendix B for data sampling and processing details. We refer to the newly extracted embeddings as **X2S-M** for **X2S**tatic-**M**ultilingual.

In a second step, we align X2S-M using VecMap (Artetxe et al., 2018a) and a set of unsupervised dictionaries that we had previously induced from experiments aligning fasttext vectors (Bojanowski et al., 2017) with unsupervised VecMap (Artetxe et al., 2018b). We refer to the aligned embeddings as **X2S-MA** (**X2S**tatic-**M**ultilingually-**A**ligned).

3.2 Embedding Evaluation

We validate our embeddings using the MUSE benchmark (Conneau et al., 2018), which includes bilingual dictionary induction (BLI) tasks for 28 of the 40 languages we use, and on SemEval 2017 Task 2 (Camacho-Collados et al., 2017), monolingual and cross-lingual word similarity. Addition-

Model	cross-lingual	monolingual
fasttext$_{unsup}$	0.712	**0.743**
X2S-M	0.708	0.699
X2S-MA	0.713	0.706
MUSE	0.707	0.728
RCSLS	**0.714**	0.718

Table 2: Average monolingual and cross-lingual scores on SemEval 2017 Task 2 (Camacho-Collados et al., 2017). See Tables 6 and 7 for detailed results.

ally, we conduct a comparative evaluation of the supervised MUSE embeddings and the supervised RCSLS embeddings from Joulin et al. (2018). For the majority of languages, alignment improves BLI by at least a few points, with differences as large as 17 points for Bengali and Hindi (see Table 5). Such large gaps underline the fact that the alignment of XLM-R is suboptimal for these languages. Notable exceptions are Korean, Thai, Tagalog, and Vietnamese, where the embeddings showed some success before alignment but were not useful afterwards. It may be that the induced dictionaries did not work well for these languages or that the static embedding spaces were too different (cf. Vulić et al., 2020). In these cases, we use the "unaligned" embeddings for further experiments.

Tables 1 and 2 show that after alignment and selection (X2S-MA), our vectors perform similarly to the supervised embeddings released by MUSE. We also contrast X2S-M and X2S-MA against the fasttext embeddings that were used to induce the dictionaries mentioned above. On the cross-lingual tasks, X2S-MA performs on par with the fasttext embeddings; on the monolingual tasks, fasttext clearly outperforms X2S-M and X2S-MA. Note, however, that SemEval Task 2 only contains data for five of the 40 languages we experiment with.

4 Cross-Linguality Transfer to XLM-R

Since our static embeddings are of reasonably high quality after extraction and their cross-linguality can be further improved using established methods, we now ask whether the language neutrality of the Transformer model can in turn be improved via indirect transfer from our aligned static embeddings.

4.1 Continued Pre-Training

Our approach for transfer from the static embeddings is based on mixing an alignment loss with masked language modelling (MLM). For the alignment loss, we sample word-vector pairs from our static embeddings, encode the word using the contextual model, and mean-pool the contextual representations over the subword tokens. We then compare this representation to the sampled static vector using one of two loss terms:

1) MSE. We use mean squared error (MSE), i.e., an element-wise comparison of the static and contextual representations. This works only if the static vector dimension matches the model's hidden size.

2) DCCA. The second option is a correlation loss (deep canonical correlation analysis; Andrew et al., 2013; implementation from Arjmand, 2020): Standard CCA (Hotelling, 1936) takes two continuous representations of related data and linearly transforms them to create two maximally correlated views. In deep CCA, the linear transformations are replaced by deep networks, which can be optimised on mini-batches. In our case, we treat the contextual model as one of the two deep models, and replace the other with the static embeddings. We back-propagate the loss only to the deep model.

We train with two sets of static vectors: Fasttext aligned with unsupervised VecMap (fasttext$_{unsup}$), and our aligned and selected X2S-MA vectors. The former have 300 dimensions and so can only be used with DCCA; the latter have 768 dimensions and can thus be used with either loss.

Additionally, we use MLM during training to ensure that the model retains its contextual capabilities. See Appendix C for training details. As a second baseline, we also continue the pre-training with only MLM on our selected languages for the same number of update steps. This ensures that any improvements from our proposed model are not merely a result of carrying out further MLM training in these languages.

4.2 Downstream Tasks

For our downstream evaluation tasks, we follow the fine-tuning procedures shown in the repository for Hu et al. (2020) for better comparability. We use a zero-shot transfer setting, i.e., we fine-tune only on English data but evaluate on all test sets. We report mean F1 score over all test sets and three fine-tuning runs for all tasks except Tatoeba, which uses accuracy as its metric and no fine-tuning.

Question Answering. We use two extractive QA tasks, XQuAD (Artetxe et al., 2020) and TyDiQA-GoldP (Clark et al., 2020). For XQuAD, the

Model	XQuAD	TyDiQA	PAN-X	UD-POS	Tatoeba	avg
XLM-R	70.51	48.91	60.40	72.92	50.35	60.62
+MLM	70.50	48.15	61.80	**72.97**	60.87	62.86
+fasttext$_{DCCA}$	70.84	**52.47**	61.84	72.09	59.99	63.45
+X2S-MA$_{MSE}$	70.42	49.20	62.62	72.95	10.05	53.05
+X2S-MA$_{DCCA}$	**70.92**	51.02	**62.73**	72.09	**68.06**	**64.96**

Table 3: Downstream evaluation results. For the QA and sequence tagging tasks, we report F1 scores averaged over three fine-tuning runs. For Tatoeba we report accuracy. +fasttext$_{DCCA}$ means continued pre-training was done using MLM and DCCA with the aligned fasttext vectors, and analogously for +X2S-MA$_{MSE}$ and +X2S-MA$_{DCCA}$. See appendix Tables 8-12 for per-language results.

SQuAD v1.1 (Rajpurkar et al., 2016) training set is used. TyDiQA includes its own training set.

Sequence Labelling. We experiment with the PAN-X (Pan et al., 2017) named entity recognition and the UD-POS part-of-speech tagging tasks. The annotated data for UD-POS are taken from Universal Dependencies v2.5 (Zeman et al., 2019).

Tatoeba is a sentence retrieval task compiled by Artetxe and Schwenk (2019). It does not need fine-tuning, instead using the cosine similarity of the mean-pooled layer 7 hidden states for retrieval.

4.3 Results and Discussion

Table 3 shows our downstream task results along with the average over all evaluated tasks. As expected, our second baseline with additional MLM in the affected languages can improve slightly over the unmodified XLM-R. However, our proposed training with a DCCA loss improves further over both baselines, except on UD-POS. This shows that the improvement is not merely a result of specialisation on the task languages, but that our alignment loss improves the model's language-neutrality.

Although the fasttext$_{unsup}$ vectors performed very well in Section 3.2, using them in continued pre-training is less effective than using X2S-MA. X2S-MA has the advantage of having the same dimension as the model hidden size, as well as being derived from XLM-R itself, both of which likely make it easier to transfer their alignment signal to the contextual model.

While both Tatoeba and the QA tasks favour DCCA, PAN-X improves regardless of the alignment loss used with X2S-MA, and UD-POS performance even degrades when using DCCA. We speculate that this is caused by the different task types requiring different strengths of the model. Further, UD-POS is a syntactic task, and the strength of the static embeddings is semantic.

The sentence retrieval task is highly sensitive to changes in the representation, whereas the tasks using fine-tuning are more stable. It may be that although the continued pre-training with DCCA improves the alignment of XLM-R, fine-tuning for tasks on English data then primarily changes the English representation space again, leading to forgetting. This prompts the question whether the model could in future benefit from using the alignment loss alongside fine-tuning. Additionally, the static embeddings may be improved further by training them on more data per language, leading to an even better signal for XLM-R. Recent work also shows that some outlier dimensions in contextual models can obscure representational quality, suggesting that "accounting for rogue dimensions" (Timkey and van Schijndel, 2021, p.4527) when learning static embeddings may help as well.

5 Conclusions

We have extracted high-quality, highly multilingual static embeddings from XLM-R using a modified version of X2Static and only 1M sentences of data per language. Our vectors have reasonable cross-lingual quality immediately after extraction, but we are able to improve their performance using alignment with dictionaries induced from fasttext vectors using VecMap. No parallel corpus was needed for this process. Our final models perform competitively with supervised vectors from MUSE, and outperform both MUSE and RCSLS—or provide models at all—for a number of lower- and medium-resource languages.

Further, we have proposed a novel continued pre-training approach that pairs an alignment loss with MLM. Using this approach and particularly the DCCA loss, we can improve the language-neutrality of XLM-R, benefitting downstream performance on semantic tasks.

References

Galen Andrew, Raman Arora, Jeff Bilmes, and Karen Livescu. 2013. Deep canonical correlation analysis. In *Proceedings of Machine Learning Research*, volume 28 (3), pages 1247–1255, Atlanta, Georgia, USA. PMLR.

Armin Arjmand. 2020. Dgcca-pytorch.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018a. Generalizing and improving bilingual word embedding mappings with a multi-step framework of linear transformations.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018b. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 789–798, Melbourne, Australia. Association for Computational Linguistics.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637, Online. Association for Computational Linguistics.

Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Anne Beyer, Göran Kauermann, and Hinrich Schütze. 2020. Embedding space correlation as a measure of domain similarity. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2431–2439, Marseille, France. European Language Resources Association.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information.

Rishi Bommasani, Kelly Davis, and Claire Cardie. 2020. Interpreting Pretrained Contextualized Representations via Reductions to Static Embeddings. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4758–4781, Online. Association for Computational Linguistics.

Jose Camacho-Collados, Mohammad Taher Pilehvar, Nigel Collier, and Roberto Navigli. 2017. SemEval-2017 task 2: Multilingual and cross-lingual semantic word similarity. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 15–26, Vancouver, Canada. Association for Computational Linguistics.

Steven Cao, Nikita Kitaev, and Dan Klein. 2020. Multilingual alignment of contextual word representations.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics*, 8:454–470.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020a. 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, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data.

Alexis Conneau, Shijie Wu, Haoran Li, Luke Zettlemoyer, and Veselin Stoyanov. 2020b. Emerging cross-lingual structure in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6022–6034, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

David M. Eberhard, Gary F. Simons, and Charles D. Fennig, editors. 2021. *Ethnologue: Languages of the World. Twenty-fourth edition.* SIL International. Online version.

Prakhar Gupta and Martin Jaggi. 2021. Obtaining better static word embeddings using contextual embedding models. 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 5241–5253, Online. Association for Computational Linguistics.

Harold Hotelling. 1936. Relations between two sets of variates. *Biometrika*, 28:321–377.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization. *CoRR*, abs/2003.11080.

Armand Joulin, Piotr Bojanowski, Tomas Mikolov, Hervé Jégou, and Edouard Grave. 2018. Loss in translation: Learning bilingual word mapping with a retrieval criterion. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2979–2984, Brussels, Belgium. Association for Computational Linguistics.

Sun Junyi. 2013. jieba.

Masahiro Kaneko and Danushka Bollegala. 2019. Gender-preserving debiasing for pre-trained word embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1641–1650, Florence, Italy. Association for Computational Linguistics.

Murathan Kurfalı and Robert Östling. 2021. Probing multilingual language models for discourse. In *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*, pages 8–19, Online. Association for Computational Linguistics.

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

Jindřich Libovický, Rudolf Rosa, and Alexander Fraser. 2020. On the language neutrality of pre-trained multilingual representations. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1663–1674, Online. Association for Computational Linguistics.

Thomas Manzini, Lim Yao Chong, Alan W Black, and Yulia Tsvetkov. 2019. Black is to criminal as caucasian is to police: Detecting and removing multiclass bias in word embeddings. 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 615–621, Minneapolis, Minnesota. Association for Computational Linguistics.

Paul McCann. 2020. fugashi, a tool for tokenizing Japanese in python. In *Proceedings of Second Workshop for NLP Open Source Software (NLP-OSS)*, pages 44–51, Online. Association for Computational Linguistics.

Tomas Mikolov, Quoc V. Le, and Ilya Sutskever. 2013. Exploiting similarities among languages for machine translation.

Benjamin Muller, Yanai Elazar, Benoît Sagot, and Djamé Seddah. 2021. First align, then predict: Understanding the cross-lingual ability of multilingual BERT. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2214–2231, Online. Association for Computational Linguistics.

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

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Rudolf Rosa. 2018. Plaintext wikipedia dump 2018. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Ming Rui. 2020. Icu-tokenizer.

Philippa Shoemark, Farhana Ferdousi Liza, Dong Nguyen, Scott Hale, and Barbara McGillivray. 2019. Room to Glo: A systematic comparison of semantic change detection approaches with word embeddings. 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 66–76, Hong Kong, China. Association for Computational Linguistics.

Jasdeep Singh, Bryan McCann, Richard Socher, and Caiming Xiong. 2019. BERT is not an interlingua and the bias of tokenization. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*, pages 47–55, Hong Kong, China. Association for Computational Linguistics.

Milan Straka and Jana Straková. 2017. Tokenizing, POS tagging, lemmatizing and parsing UD 2.0 with UDPipe. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 88–99, Vancouver, Canada. Association for Computational Linguistics.

Text Analysis and Knowledge Engineering Lab. 2021. spacy-udpipe.

William Timkey and Marten van Schijndel. 2021. All bark and no bite: Rogue dimensions in transformer language models obscure representational quality. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4527–4546, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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

Ivan Vulić, Sebastian Ruder, and Anders Søgaard. 2020. Are all good word vector spaces isomorphic? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3178–3192, Online. Association for Computational Linguistics.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France. European Language Resources Association.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi 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 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. 2019. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 833–844, Hong Kong, China. Association for Computational Linguistics.

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.

Daniel Zeman, Joakim Nivre, et al. 2019. Universal dependencies 2.5. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

A List of Languages

We list all languages used in our experiments in Table 4.

Language	Code	Family
Afrikaans	af	IE: Germanic
Arabic	ar	Semitic
Bulgarian	bg	IE: Slavic
Bengali	bn	IE: Indo-Aryan
German	de	IE: Germanic
Greek	el	IE: Greek
English	en	IE: Germanic
Spanish	es	IE: Romance
Estonian	et	Uralic
Basque	eu	Isolate
Farsi	fa	IE: Iranian
Finnish	fi	Uralic
French	fr	IE: Romance
Hebrew	he	Semitic
Hindi	hi	IE: Indo-Aryan
Hungarian	hu	Uralic
Indonesian	id	Malayo-Polynesian
Italian	it	IE: Romance
Japanese	ja	Japonic
Javanese	jv	Malayo-Polynesian
Georgian	ka	Kartvelian
Kazakh	kk	Turkic
Korean	ko	Koreanic
Malayalam	ml	Dravidian
Marathi	mr	IE: Indo-Aryan
Malay	ms	Malayo-Polynesian
Burmese	my	Sino-Tibetan
Dutch	nl	IE: Germanic
Portuguese	pt	IE: Romance
Russian	ru	IE: Slavic
Swahili	sw	Niger-Congo
Tamil	ta	Dravidian
Telugu	te	Dravidian
Thai	th	Kra-Dai
Tagalog	tl	Malayo-Polynesian
Turkish	tr	Turkic
Urdu	ur	IE: Indo-Aryan
Vietnamese	vi	Mon-Khmer
Yoruba	yo	Niger-Congo
Mandarin	zh	Sino-Tibetan

Table 4: List of languages used with their ISO codes and language families (Eberhard et al., 2021). IE stands for Indo-European.

B Data Sampling and Processing Details for X2S-M

Data Sampling. After sampling data from the reconstructed CC100 corpus (Wenzek et al., 2020), we do sentence segmentation and tokenisation (see the list of languages and tools below), then filter the data heuristically: Like Bommasani et al. (2020), we discard sentences with fewer than seven tokens. We also keep only sentences from paragraphs with at least two sentences, avoiding, for example, headlines.

Segmentation and Tokenisation Tools. af, ar, bg, de, en, el, es, et, eu, fa, fi, fr, he, hi, hu, id, it, ko, mr, nl, pt, ru, ta, te, tr, ur, vi: UDPipe (Straka and Straková, 2017; Text Analysis and Knowledge Engineering Lab, 2021) for both sentence segmentation and tokenisation. ja: ICU-tokenizer (Rui, 2020) for sentence segmentation, fugashi (McCann, 2020) for tokenisation. zh: ICU-tokenizer for sentence segmentation, jieba (Junyi, 2013) for tokenisation. bn, jv, ka, kk, ml, ms, my, sw, th, tl, yo: ICU-tokenizer for both.

C Continued Pre-Training Details

We start from the XLM-R$_{\text{BASE}}$ checkpoint, which has 270M parameters. At each training step, we mix samples from a text dataset with samples from our static embeddings, computing both a language modelling and an alignment loss. We use an effective batch size of 64 for MLM and 1024 for the alignment loss. The data for MLM is sampled from concatenated Wikipedia data of all 40 languages. For this corpus, 100k paragraphs per language were taken from Rosa (2018). Each model is trained for 7500 update steps, corresponding to roughly four epochs over our set of static embeddings. We use the default hyperparameters for language modelling in Huggingface Transformers (Wolf et al., 2020). The final checkpoints are selected based on the MLM loss over a separate validation set. Each training run was done on a single Nvidia GeForce GTX 1080 Ti GPU.

Model	af	ar	bg	bn	de	el	es	et	fa	fi
fasttext$_{unsup}$	34.43	44.04	53.13	28.90	73.38	55.01	78.70	43.65	36.83	48.24
X2S-M	58.48	30.23	50.91	18.03	64.52	42.08	74.07	44.82	32.17	49.21
X2S-MA	**60.69**	44.17	57.99	**34.61**	71.51	52.98	78.00	52.88	41.01	54.02
MUSE	–	44.80	52.40	–	73.67	52.37	82.67	41.77	–	53.77
RCSLS	38.13	**57.95**	**61.70**	32.17	**78.37**	**59.80**	**85.43**	**53.30**	**44.80**	**65.87**

Model	fr	he	hi	hu	id	it	ja	ko	ms	nl
fasttext$_{unsup}$	78.89	49.82	43.29	56.67	65.15	75.83	**42.73**	0.03	40.81	73.35
X2S-M	72.18	35.96	32.73	54.15	67.82	70.23	31.57	26.70	56.44	69.54
X2S-MA	77.36	49.87	**49.94**	60.16	**73.79**	76.52	42.53	25.83	**63.64**	75.08
MUSE	82.67	49.10	–	59.37	67.67	78.23	–	–	–	75.43
RCSLS	**84.43**	**59.21**	45.71	**70.00**	72.87	**81.90**	–	**47.01**	–	**80.07**

Model	pt	ru	ta	th	tl	tr	vi	zh
fasttext$_{unsup}$	69.60	49.96	27.09	0.00	0.00	44.85	0.00	33.80
X2S-M	75.76	46.11	16.97	**29.37**	**53.42**	50.42	46.39	35.65
X2S-MA	77.38	53.47	**31.23**	28.58	53.12	51.97	46.89	44.80
MUSE	80.77	58.87	–	–	–	53.05	48.20	–
RCSLS	**83.87**	**65.60**	26.75	26.67	27.73	**62.49**	**60.03**	**50.63**

Table 5: Cross-lingual MUSE results, per language with English, averaged over both directions.

Model	de-en	de-es	de-fa	de-it	en-es	en-fa	en-it	es-fa	es-it	fa-it	avg
fasttext$_{unsup}$	**0.74**	**0.75**	**0.69**	**0.72**	**0.73**	0.69	0.71	0.70	**0.74**	0.66	0.712
X2S-M	0.71	0.73	0.66	0.70	0.72	0.69	0.72	**0.73**	**0.74**	0.69	0.708
X2S-MA	0.72	0.72	0.67	0.70	**0.73**	0.71	0.73	0.72	**0.74**	0.69	0.713
MUSE	0.71	0.70	–	0.68	0.71	–	0.71	–	0.73	–	0.707
RCSLS	**0.74**	0.71	0.67	0.69	**0.73**	**0.73**	**0.74**	0.71	0.73	**0.70**	**0.714**

Table 6: Full cross-lingual results from SemEval 2017 Task 2 (Camacho-Collados et al., 2017).

Model	de	en	es	fa	it
fasttext$_{unsup}$	**0.80**	0.71	**0.76**	**0.72**	**0.73**
X2S-M	0.73	0.70	0.73	0.65	0.68
X2S-MA	0.73	**0.72**	0.72	0.66	0.70
MUSE (Conneau et al., 2018)	0.73	**0.72**	0.74	–	0.72
RCSLS (Joulin et al., 2018)	0.73	**0.72**	0.74	0.66	**0.73**

Table 7: Full monolingual results from SemEval 2017 Task 2 (Camacho-Collados et al., 2017).

Model	ar	de	el	en	es	hi	ru	th	tr	vi	zh
XLM-R	65.34	74.47	72.57	83.21	76.98	67.72	74.31	67.66	**68.55**	73.66	51.09
+MLM	64.93	74.73	72.52	83.66	76.75	68.00	74.30	67.76	67.86	73.35	51.68
+fasttext$_{DCCA}$	65.50	74.77	**73.78**	83.66	76.75	68.84	**75.06**	67.35	68.30	**74.18**	51.00
+X2S-MA$_{MSE}$	64.73	74.01	72.87	83.51	76.36	67.82	74.46	**67.77**	68.04	73.78	51.30
+X2S-MA$_{DCCA}$	**65.91**	**74.83**	73.05	**84.07**	**77.00**	**69.29**	74.26	66.99	**68.55**	73.98	**52.20**

Table 8: XQuAD results (F1) per language. Averaged over three fine-tuning runs with different random seeds.

Model	ar	bn	en	fi	id	ko	ru	sw	te
XLM-R	57.43	37.20	62.74	53.87	68.04	20.67	52.25	54.16	33.80
+MLM	57.89	35.48	62.38	51.70	66.06	21.08	52.64	54.76	31.40
+fasttext$_{DCCA}$	**60.96**	**43.20**	**63.79**	56.52	**70.72**	**23.58**	**55.57**	**55.37**	**42.56**
+X2S-MA$_{MSE}$	57.46	37.59	61.16	52.95	66.77	21.73	51.63	53.10	**40.43**
+X2S-MA$_{DCCA}$	58.58	42.69	63.48	**56.78**	69.02	23.11	54.55	54.90	36.04

Table 9: TyDiQA results (F1) per language. Averaged over three fine-tuning runs with different random seeds.

Model	af	ar	bg	bn	de	el	en	es	et	eu
XLM-R	74.88	46.12	77.18	67.96	74.34	72.97	**82.83**	74.52	70.44	57.75
+MLM	76.48	48.25	77.51	69.89	75.00	73.88	82.75	75.90	73.17	57.21
+fasttext$_{DCCA}$	**77.93**	47.58	**78.00**	67.27	**76.23**	75.34	82.82	**79.45**	74.06	61.43
+X2S-MA$_{MSE}$	76.87	47.86	77.79	**70.69**	75.58	**76.34**	82.72	77.87	73.96	**61.90**
+X2S-MA$_{DCCA}$	77.50	**53.03**	77.98	66.16	75.81	75.30	82.73	75.76	**74.67**	60.28
Model	**fa**	**fi**	**fr**	**he**	**hi**	**hu**	**id**	**it**	**ja**	**jv**
XLM-R	49.30	74.95	77.51	51.86	66.65	76.10	48.99	77.13	19.61	57.45
+MLM	47.72	75.52	**79.17**	53.63	**68.74**	76.94	50.62	77.48	18.28	58.32
+fasttext$_{DCCA}$	47.74	76.93	78.71	56.70	66.66	**77.27**	49.35	**78.56**	17.48	59.14
+X2S-MA$_{MSE}$	**55.45**	**76.30**	78.83	**57.81**	67.76	77.22	49.92	77.98	**20.53**	**63.28**
+X2S-MA$_{DCCA}$	50.56	76.20	78.88	54.91	67.86	76.83	**55.03**	78.13	17.94	58.42
Model	**ka**	**kk**	**ko**	**ml**	**mr**	**ms**	**my**	**nl**	**pt**	**ru**
XLM-R	65.60	45.45	48.07	60.50	61.31	62.54	53.09	79.45	77.67	63.42
+MLM	67.35	51.14	51.97	63.19	61.30	67.42	52.84	80.64	79.14	62.40
+fasttext$_{DCCA}$	67.88	51.49	47.48	51.92	**63.13**	57.89	46.19	**81.25**	79.48	64.41
+X2S-MA$_{MSE}$	**69.14**	**51.76**	**54.13**	**64.49**	62.96	**67.43**	**53.53**	80.82	78.90	**64.50**
+X2S-MA$_{DCCA}$	66.49	50.59	52.55	59.64	60.35	66.94	51.79	81.06	**80.45**	62.77
Model	**sw**	**ta**	**te**	**th**	**tl**	**tr**	**ur**	**vi**	**yo**	**zh**
XLM-R	63.96	54.64	48.66	3.60	71.46	74.68	54.31	68.58	34.91	**25.47**
+MLM	65.27	56.12	50.77	3.34	71.39	76.49	62.23	69.88	38.05	24.51
+fasttext$_{DCCA}$	**66.45**	57.31	53.63	3.42	**71.78**	**78.59**	56.52	**71.97**	**53.07**	21.26
+X2S-MA$_{MSE}$	66.35	**58.47**	53.66	3.22	70.49	77.09	60.26	69.90	37.00	24.33
+X2S-MA$_{DCCA}$	65.40	56.26	**54.61**	2.19	67.65	77.53	**63.47**	70.53	50.23	24.40

Table 10: PAN-X results (F1) per language. Averaged over three fine-tuning runs with different random seeds.

Model	**af**	**ar**	**bg**	**de**	**el**	**en**	**es**	**et**	**eu**
XLM-R	88.46	67.56	88.58	88.64	**87.79**	**95.85**	88.04	85.63	**69.38**
+MLM	88.75	68.21	**88.85**	88.57	87.37	95.71	88.51	85.88	69.05
+fasttext$_{DCCA}$	**88.96**	67.73	88.30	88.40	87.34	95.79	87.33	85.58	68.33
+X2S-MA$_{MSE}$	88.87	**68.43**	88.55	**88.72**	87.45	95.77	**88.61**	85.72	69.27
+X2S-MA$_{DCCA}$	88.50	67.45	88.11	88.22	87.26	95.69	87.87	**85.99**	68.34
Model	**fa**	**fi**	**fr**	**he**	**hi**	**hu**	**id**	**it**	**ja**
XLM-R	70.16	85.60	86.00	66.96	67.83	**83.14**	72.64	87.41	**24.23**
+MLM	70.14	**85.75**	86.50	**68.51**	68.14	83.07	**72.59**	88.46	23.59
+fasttext$_{DCCA}$	68.70	85.69	86.20	66.33	65.70	82.87	72.64	87.32	13.89
+X2S-MA$_{MSE}$	**70.46**	85.61	**86.76**	67.63	**69.30**	82.82	**72.59**	**88.61**	20.61
+X2S-MA$_{DCCA}$	68.81	85.74	86.38	66.34	66.01	82.89	72.82	87.43	14.12
Model	**kk**	**ko**	**mr**	**nl**	**pt**	**ru**	**ta**	**te**	**th**
XLM-R	76.74	53.06	82.95	89.42	86.21	**89.25**	62.12	**84.90**	42.36
+MLM	76.54	52.88	83.21	**89.45**	86.82	89.00	61.62	83.79	42.09
+fasttext$_{DCCA}$	**78.09**	52.86	82.86	89.35	85.70	89.11	**63.00**	84.21	41.54
+X2S-MA$_{MSE}$	76.55	**53.16**	**84.19**	**89.45**	**87.45**	89.17	61.44	84.60	**42.62**
+X2S-MA$_{DCCA}$	77.78	52.93	82.66	89.37	86.07	88.89	62.21	84.49	39.63

Model	**tl**	**tr**	**ur**	**vi**	**yo**	**zh**
XLM-R	88.91	74.27	56.48	**58.59**	25.29	**32.08**
+MLM	**89.42**	74.20	56.58	58.21	24.38	32.06
+fasttext$_{DCCA}$	88.22	74.53	56.06	57.62	23.76	25.02
+X2S-MA$_{MSE}$	89.21	74.19	**57.45**	58.15	**25.45**	28.54
+X2S-MA$_{DCCA}$	87.44	**74.58**	56.79	57.68	24.55	25.80

Table 11: UD-POS results (F1) per language. Averaged over three fine-tuning runs with different random seeds.

Model	**af**	**ar**	**bg**	**bn**	**de**	**el**	**es**	**et**	**eu**
XLM-R	51.60	35.80	66.90	28.70	88.40	51.60	71.00	44.20	26.10
+MLM	65.60	46.50	74.70	41.70	91.90	61.10	79.00	55.80	38.60
+fasttext$_{DCCA}$	70.60	47.20	78.20	44.90	95.00	68.40	85.80	63.90	44.70
+X2S-MA$_{MSE}$	10.90	3.90	17.10	2.40	42.50	5.10	15.20	7.90	7.40
+X2S-MA$_{DCCA}$	**74.10**	**57.00**	**82.10**	**54.90**	**95.40**	**72.50**	**88.60**	**75.20**	**52.50**
Model	**fa**	**fi**	**fr**	**he**	**hi**	**hu**	**id**	**it**	**ja**
XLM-R	64.40	63.90	72.50	51.70	50.50	58.70	68.60	64.70	52.80
+MLM	73.50	74.60	77.90	65.10	69.10	69.90	81.10	73.40	64.20
+fasttext$_{DCCA}$	74.60	78.60	82.30	65.50	61.90	73.30	82.80	78.50	67.00
+X2S-MA$_{MSE}$	10.50	12.70	22.20	10.10	9.00	13.40	14.30	11.50	10.00
+X2S-MA$_{DCCA}$	**79.90**	**84.30**	**84.30**	**71.70**	**70.10**	**80.20**	**86.40**	**82.30**	**74.00**
Model	**jv**	**ka**	**kk**	**ko**	**ml**	**mr**	**nl**	**pt**	**ru**
XLM-R	15.12	37.13	33.22	50.10	54.73	38.00	76.80	76.60	69.80
+MLM	20.00	45.98	44.17	61.00	**64.19**	**50.70**	84.60	84.40	78.50
+fasttext$_{DCCA}$	16.10	30.56	53.39	40.40	14.56	35.40	87.20	88.30	83.00
+X2S-MA$_{MSE}$	5.37	4.96	6.09	10.50	4.51	5.30	17.80	19.70	12.50
+X2S-MA$_{DCCA}$	**22.93**	**63.81**	**62.26**	**63.20**	25.47	34.90	**89.30**	**90.40**	**85.60**
Model	**sw**	**ta**	**te**	**th**	**tl**	**tr**	**ur**	**vi**	**zh**
XLM-R	15.64	25.08	30.77	34.67	29.70	54.90	31.10	67.70	59.40
+MLM	23.59	36.16	37.61	51.28	39.90	65.20	**47.40**	77.50	75.60
+fasttext$_{DCCA}$	21.54	42.35	51.28	35.58	37.80	69.30	42.60	76.20	70.80
+X2S-MA$_{MSE}$	4.10	1.95	3.42	1.64	6.80	6.80	2.50	15.60	6.10
+X2S-MA$_{DCCA}$	**23.85**	**56.35**	**59.40**	**68.43**	**45.10**	**78.00**	45.90	**84.40**	**85.20**

Table 12: Tatoeba results (accuracy) per language.