Bitext Mining Using Distilled Sentence Representations for Low-Resource Languages

Kevin Heffernan
Meta AI
kevinheffernan@fb.com

Onur Çelebi
Meta AI
celebio@fb.com

Holger Schwenk
Meta AI
schwenk@fb.com

Abstract

Scaling multilingual representation learning beyond the hundred most frequent languages is challenging, in particular to cover the long tail of low-resource languages. We move away from the popular one-for-all multilingual models and focus on training multiple language (family) specific representations, but most prominently enable all languages to still be encoded in the same representational space. We focus on teacher-student training, allowing all encoders to be mutually compatible for bitext mining, and enabling fast learning of new languages. We also combine supervised and self-supervised training, allowing encoders to take advantage of monolingual training data.

Our approach significantly outperforms the original LASER encoder. We study very low-resource languages and handle 44 African languages, many of which are not covered by any other model. For these languages, we train sentence encoders and mine bitexts. Adding these mined bitexts yielded an improvement of 3.8 BLEU for NMT into English.

1 Introduction

There is increasing interest in multilingual sentence representations since they promise an appealing approach to extend NLP tasks to a large number of languages, without the need to separately train a language-specific model. Most of the current works on multilingual sentence representations have focused on training one model which handles all languages of interest, e.g. (Artetxe and Schwenk, 2019b; Feng et al., 2020; Reimers and Gurevych, 2020; Ramesh et al., 2022). The main motivation is that languages with limited resources will benefit from the fact that the same model has learned other (similar) languages. Zero-shot performance is of particular interest: the model generalizes well to a new language although it has never seen training data in that language. Training massively multilingual models faces several problems with increasing number of languages: how to make sure that all languages are learned, how to to account for the large imbalance of available training, or the high computational complexity. Reimers and Gurevych (2020) proposed a teacher-student approach to extend an existing (monolingual) sentence embedding space to new languages. We build on this generic idea and propose multiple improvements which significantly improve performance, namely different teacher and student architectures, several supervised and unsupervised training criteria, and language-specific encoders. We also investigate challenges when handling low-resources languages, showcased by training models for 50 African languages. To the best of our knowledge, many of these languages are not handled by any other sentence encoder or pretrained model. We dispose test data for 44 out of 50 languages, mine bitexts against 21.5 billion English sentences, and train SMT models to translate into English.

Multilingual sentence embeddings have many applications which is reflected by several metrics to evaluate them, e.g. the XTREME bench mark (Hu et al., 2020a; Ruder et al., 2021). In this work, we focus on the use of multilingual sentence embeddings for similarity-based bitext mining, as proposed by Artetxe and Schwenk (2019a), and on using these mined bitexts to improve NMT. Consequently, our primary metric is NMT performance. However, mining and NMT training is computationally expensive and it is intractable to systematically perform this evaluation for many different sentence encoder variants. As an evaluation proxy, we use multilingual similarity search error rate. In contrast to previous work which used the Tatoeba test set, e.g. (Artetxe and Schwenk, 2019b; Hu et al., 2020b; Reimers and Gurevych, 2020), we switch to the FLORES evaluation benchmark, which contains high-quality human translated texts from Wikipedia (Goyal et al., 2021) and covers many low-resource languages.

Findings of the Association for Computational Linguistics: EMNLP 2022, pages 2101–2112

The contributions of this work can be summarized as follows: 1) we move away from the popular *one-for-all approach* and train multiple, mutually compatible language (family) specific encoders; 2) we explore several variants and improvements of teacher-student training for multilingual sentence representations (section 3), and propose a new approach which combines supervised teacher-student with self-supervised MLM training to better handle very low-resource languages (subsection 5.3); 3) the new model substantially improves 12 languages which were badly handled by the original LASER encoder (subsection 5.1); and 4) we train sentence encoders for 50 African languages, mine bitexts, and train NMT systems (section 6). To the best of our knowledge, many of these languages are not handled by any other NMT system.

2 Related work

Multilingual sentence representation Examples of approaches to learn multilingual representations are multilingual BERT (m-BERT) which covers 104 languages (Devlin et al., 2019), XLM (Conneau and Lample, 2019), and XLM-R which was trained on 100 languages using crawled web data (Conneau et al., 2020). However, as these approaches do not take into account a sentence-level objective during training, they can result in poor performance when applied to tasks which use sentence representations such as bitext retrieval (Hu et al., 2020b). In order to address this, methods such as SentenceBERT (SBERT) make use of a Siamese network to better model sentence representations (Reimers and Gurevych, 2019). LaBSE (Feng et al., 2020) uses a dual-encoder approach with a transformer-based architecture and additive margin softmax loss (Yang et al., 2019). It covers 109 languages, and is pre-trained using a masked language modelling (MLM) and translation language modelling (TLM) objective (Conneau et al., 2020). LabSE was used to mine bitexts in eleven Indian languages (Ramesh et al., 2022). Another popular multilingual sentence embedding model is LASER (Artetxe and Schwenk, 2019b). It is based on a LSTM encoder/decoder architecture with a fixed-size embedding layer and no pre-training. LASER covers 93 languages.

When learning a multilingual embedding space, a limitation of many existing approaches is that they require training a new model from scratch each time a language is to be added. However, there have been various methods proposed to address this. Wang et al. (2020) provide one such technique which extends m-BERT to low-resource languages by increasing the size of the existing vocabulary, and then continuing self-supervised training using monolingual data for a low-resource language. Another example by Reimers and Gurevych (2020) uses multilingual distillation. In this supervised teacher-student approach, the teacher is a monolingual model pre-trained on English (SBERT), and the student is a pre-trained multilingual model (XLM-R). Using bitexts, the student then extends the embedding space to the desired language(s) by applying regression loss between the English sentence representation of the teacher, and the target language sentence representation of the student.

Scaling multilinguality Several recent works have addressed the challenges faced when scaling multilingual models to a hundred languages and beyond, namely massively multilingual NMT systems (Fan et al., 2020; Arivazhagan et al., 2019; NLLB Team et al., 2022). A recent study explored the extension to more than a thousand languages (Siddhant et al., 2022; Bapna et al., 2022). Training NMT models for a large number of languages faces many challenges and a large variety of architectures have been explored (Ma et al., 2021; Wang et al., 2022; Escolano et al., 2021; NLLB Team et al., 2022). To the best of our knowledge, similar modelling techniques were not yet considered to train (massively) multilingual sentence encoders.

Resources for African languages Collecting resources, training NMT systems, or performing evaluations for African languages is the focus of several works (Dabre and Sukhoo, 2022; Emezue and Dossou, 2020; Siminyu et al., 2021; Abbott and Martinus, 2019; Azunre et al., 2021; Hacheme, 2021; Nekoto et al., 2020). The Masakhane project[1] aims at providing resources to both strengthen and spur NLP research in African languages. A workshop focused on the evaluation of African languages will be held at EMNLP'22.[2] In the framework of the data track, several parallel corpora were made available. In general, the number of languages covered is well below the 44 languages we evaluate in this work.

[1] https://www.masakhane.io/

[2] https://www.statmt.org/wmt22/large-scale-multilingual-translation-task.html

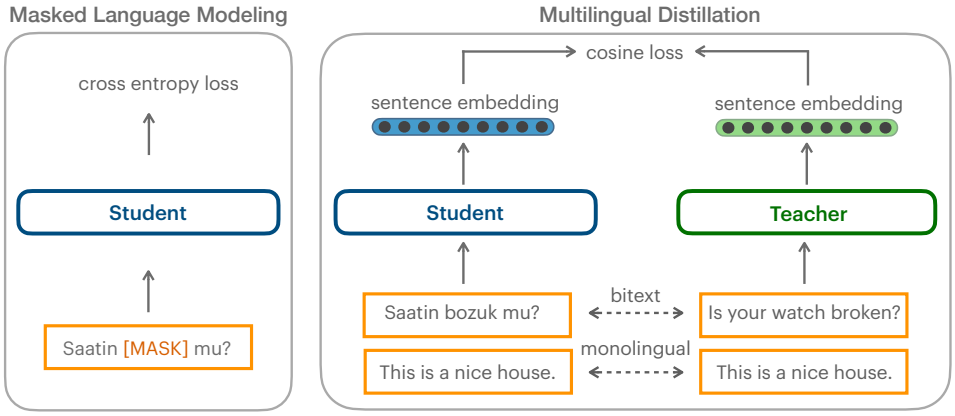

Figure 1: Architecture of our teacher-student approach.

3 Architecture

The overall architecture of our approach is summarized in Figure 1. The teacher is an improved LASER encoder. Compared to the original training procedure described in Artetxe and Schwenk (2019b), we use SPM instead of BPE preprocessing, up-sampling of low-resource languages, and a new implementation in `fairseq`. All the other parameters are unchanged, namely a 5-layer BiLSTM encoder, the 1024 dimensional sentence embeddings are obtained by max-pooling over the last layer, and training is performed for 93 languages with public resources obtained from OPUS. The reader is referred to Artetxe and Schwenk (2019b) for details on the original LASER training procedure. We use this new multilingual sentence encoder as the teacher in all our experiments and in this work refer to our teacher as LASER2, and student models as LASER3. The code to train the teacher or student models is freely available in the fairseq github repository.[3]

Training of the students follows the general idea of a teacher-student approach as initially proposed by Reimers and Gurevych (2020), but with several important differences. We want to scale encoder training and bitext mining well beyond the roughly 100 languages handled by current multilingual encoders. This may include languages which are not covered by existing pretrained models, and retraining them would be computationally very expensive. Also, some languages may be written in a new script which is not covered. Therefore, we made the following design choices: (1) We do not initialize the student with some pretrained model, e.g. XLM-R, but use a random initialization; (2) The student is trained on 2M sentences of English monolingual data, and we also add 2M sentence of English-Spanish bitexts from CCMatrix (Schwenk et al., 2021) to better align with the teacher's multilingual embedding space; (3) Instead of one massively multilingual model, we train multiple students for a small subset of (similar) languages, or even a single language; (4) Use of separate SPM vocabularies for teacher and student, better accommodating scripts and tokens in the student languages which were unseen by the teacher (cf. subsection 5.2); (5) Optimization of the cosine loss between the teacher and student embedding, since this is the relevant metric for bitext mining (cf. Figure 1 above); (6) Jointly train distillation alongside a MLM criterion to benefit additional learning from monolingual data in a foreign language (cf. Figure 1, and subsection 5.3); (7) Addition of curriculum learning in the form of *progressive distillation*. In this strategy, instead of sending the entire sentence pairs all at once, we send incremental versions of the respective sentence pairs to both teacher and student, which we found to be helpful for some particularly challenging low-resource languages.

Our motivation of using a total of 4M English sentences is to *"anchor"* the student encoder to the embedding space, and hopefully be able to learn new languages with a limited amount of parallel texts. In initial experiments, we used a 6-layer BiLSTM encoder architecture as in Artetxe and Schwenk (2019b), but we saw consistent improvements by switching to a 12-layer transformer. We keep the same student architecture for all languages (L=12, H=1024, A=4, 250M

[3] https://github.com/facebookresearch/fairseq/tree/nllb/examples/nllb/laser_distillation

params). Teacher-student training was performed on 16 GPUs, ADAM optimizer, a learning rate of 0.0005 and with a batch size of 2,000. When we minimize the cosine distance only, max-pooling of the transformer outputs to achieve the fixed-size sentence representations worked best, compared to using a special token like `[CLS]`. For curriculum learning using *progressive distillation*, we incrementally send a percentage of subwords from each sentence pair (e.g. 10%, 20%, ..., 100%). We experimented sending various incremental percentages of the sentence pairs to both teacher and student (e.g. 20%, 40%), but found 10% increments to perform best.

4 Training and evaluation resources

Evaluation data Creating high-quality evaluation data for low-resource languages is challenging. In this work we evaluate our approach on two publicly available corpora: Tatoeba and FLORES. The Tatoeba corpus is a test set covering 112 languages (Artetxe and Schwenk, 2019b), and contains up to 1000 sentences for each language pair. FLORES is a freely available N-way parallel test with 1012 sentences the devtest set (Goyal et al., 2021).[4] It initially covered 101 languages, and was recently extended to 200 languages (NLLB Team et al., 2022), including 44 African languages on which we report results in this paper.

Monolingual data comes mostly from Common Crawl and other public sources like ParaCrawl,[5] and some additional targeted crawling for several low-resource languages. We have extended and improved fastText LID (Grave et al., 2018) to include additional languages considered in this work. We trained this new LID model on publicly available monolingual data and evaluated it on FLORES. Preprocessing includes: sentence splitting, filtering of sentences in the wrong script or with more than 20% of numbers or punctuation, LID and deduplication, as well as LM filtering on English.

Bitexts are obtained from OPUS[6] (Tiedemann, 2012) and used to train the sentence encoders and NMT systems. The amount of available resources is summarized in Table 4.

[4] https://github.com/facebookresearch/flores

[5] https://paracrawl.eu

[6] https://opus.nlpl.eu/

4.1 Multilingual similarity search

As the end goal for our mined bitexts is to improve the quality of a translation system, our main evaluation is NMT quality. However, given the expense involved in both mining and training NMT systems, it is not tractable to perform such an evaluation each time a new encoder is trained. Therefore, as a proxy metric for our encoders we use a mining-based multilingual similarity search error rate, in this work referred to as `xsim`. Unlike cosine accuracy which aligns source and target embeddings with the highest cosine similarity, `xsim` aligns based on the highest margin-based score, which has shown to be helpful for mining (Artetxe and Schwenk, 2019a). In this work, we score using the *ratio* margin R, defined as:

$$R(x,y) = \frac{cos(x,y)}{\sum_{z \in NN_k(x)} \frac{cos(x,z)}{2k} + \sum_{z \in NN_k(y)} \frac{cos(y,z)}{2k}}$$

where x and y are the source and target sentences, and $NN_k(x)$ denotes the k nearest neighbors of x. We used $k = 4$. The `xsim` score is then defined as the error rate of wrongly aligned sentences in our test set, searching in English (i.e. xx $\rightarrow$ eng). The `xsim` evaluation tool is freely available.[7]

5 Experimental evaluation: multilingual similarity search

In this section, we provide some evaluations of our proposed multilingual distillation approach, based on multilingual similarity search.

5.1 Improving LASER

LASER has been shown to perform well on many languages already. Rather than focusing on marginal improvements for these languages, we instead selected languages for which the original LASER encoder had an average accuracy of less than 90% on the Tatoeba test set. However, as the Tatoeba test set is translated by volunteers, contains a majority of easy confusable short sentences, and for some languages has much less than 1000 sentences, we propose in this work to instead primarily rely on the FLORES dataset as the ground truth. This dataset is of a higher quality as a result of professional human annotation, and contains the same number of sentences across languages.

[7] https://github.com/facebookresearch/LASER/blob/main/tasks/xsim

ISO	Language	FLORES (1012 sents N-way)			Tatoeba			
		LASER	LASER3	LaBSE	LASER	LASER3	LaBSE	# Sents
amh	Amharic	57.4	0.1	0	51.2	10.7	5.4	168
bel	Belarusian	40.4	0.3	0	29.5	5.1	3.1	1000
gle	Irish	92.5	0.8	0	94.9	15.8	3.6	1000
hye	Armenian	75.6	0.2	0	60.2	8.0	3.8	742
kat	Georgian	61.0	1.8	0	60.6	20.9	3.5	746
kaz	Kazakh	63.3	0.5	0.2	79.3	16.5	8.3	575
khm	Khmer	64.4	2.1	2.1	74.0	43.6	15	722
swh	Swahili	0.8	0.1	0	36.7	16.4	8.7	390
tam	Tamil	40.8	0.2	0	22.8	36.8	6.5	307
tel	Telugu	6.8	0.2	0	16.2	15.8	1.3	234
urd	Urdu	6.6	0.2	0.1	12.4	9.0	3.6	1000
uzb	Uzbek	79.9	0.2	0.1	77.3	18.2	10.5	428
	Average	49.1	0.6	0.2	51.3	18.1	6.1	

Table 1: Comparison of LASER, LASER3, and LaBSE on FLORES and Tatoeba test sets (`xsim` error rates).

Also, FLORES is N-way parallel and the results are comparable among languages. To illustrate this difference between datasets, we provide results in Table 1 for the same languages across both test sets.

In all instances on FLORES, we observe significant improvements upon the original LASER encoders using our proposed teacher-student approach, and also achieve competitive results to the much larger *one-for-all* model LaBSE (average difference of 0.4% `xsim` error rate) We also notice that there is a considerable difference between both test sets. For example, on FLORES we report an `xsim` of 0.1% for Swahili, but an `xsim` of 16.4% of Tatoeba. To see if this phenomenon occurs with other representations, we also included LaBSE, for which we observed a similar effect. This stark difference further suggests that Tatoeba is a less reliable benchmark for evaluating sentence encoders. In particular, Tatoeba mainly contains very short sentences which can create a strong bias towards a particular model or training corpus. Given this observation, in the rest of this work we move away from Tatoeba and instead evaluate on FLORES. We hope that other existing approaches and future work will also adopt evaluation on FLORES using a margin criterion.

Although we also hoped to show a comparison to a similar distillation method by Reimers and Gurevych (2020), their existing results were reported on Tatoeba (which as shown above is not very reliable to compare against), and results were not reported using the margin score (cf. subsection 4.1). We attempted to evaluate their reported models on FLORES using `xsim`, but their model is not available. We also attempted to reproduce the author's result by training new models using the provided code, but as we were not able to obtain the original training data used, we were unfortunately not able to reach a reasonably close outcome in order to make a fair comparison.

5.2 Language-specific encoders

In our first experiments, we used the same preprocessing and SPM vocabulary for each student as the LASER2 SPM teacher: a 50k SPM vocabulary which was trained on all LASER2 languages. On one hand, using a massively multilingual SPM vocabulary is expected to improve the generalization among languages, and it is the only possible solution when training a massively multilingual model which handles all languages. On the other hand, low-resource languages may be badly modeled in a joint SPM vocabulary, i.e. mostly by very short SPM tokens, despite the use of up-sampling strategies. Our approach to train multiple sentence encoders, each one specific to a small number of languages, opens the possibility to train and use specific SPM vocabularies for each subset of a small number of languages. Table 2 summarizes the results for these different training strategies for two example languages: Amharic (amh) and Tigrinya (tir). Both are part of the family of Semitic languages, and use their own specific Ge'ez script. Other major languages from this family are Aramaic, Arabic and Hebrew, Maltese and Tigre, all

Training	SPM	#train	amh	tir
LASER2	50k joint	220M	34.9	92.9
Semitic	50k joint	9M	0.2	1.19
Ge'ez	8k specific	0.7M	0.1	0.89
LaBSE	501k joint	≈ 6B	0	13.74

Table 2: `xsim` error rates on FLORES devtest for Amharic and Tigrinya and different training strategies (see text for details). The amount of training data excludes 4M sentences of English for our models.

Approach	`xsim`
LASER	70.65
LaBSE	26.19
LASER3	21.05
+MLM	12.65
+MLM + Curriculum learning	5.93

Table 3: Comparison of LASER and LaBSE to Wolof student models trained with and without MLM and curriculum (`xsim` error rates).

using their own specific script.

Amharic was part of the 93 languages LASER2 was trained on, but the `xsim` error rate is rather high, and LASER2 generalizes badly to Tigrinya. We first trained a specific encoder for three Semitic languages: Amharic, Tigrinya and Maltese. We only added Maltese, which uses a Latin script, in order to avoid a multitude of different scripts to be learnt by one encoder. This yields a significant `xsim` improvement to 0.2 and 1.19% respectively, highlighting the usefulness of teacher-student training and specific encoders for a small set of similar languages. We then trained an encoder for Amharic and Tigrinya only, paired with English as in all our experiments, and a specific 8k SPM vocabulary to better support the Ge'ez script. This brought `xsim` down to 0.1% and 0.89%, respectively although we use less training data. Our best model is on par with LaBSE, which was trained on Amharic only, and significantly outperforms it for Tigrinya.

5.3 Joint training

In order to highlight the effect of jointly training our students with masked language modelling and curriculum learning, we chose a very low-resource language with little bitexts available to use for distillation alone: Wolof. As with previous students, we trained Wolof alongside closely related Senegambian languages: Fulah, Bassari, and Wamey, but the joint training strategies are only applied to Wolof. In total we used 21k bitexts, and an additional 94k of monolingual data for Wolof.

We observe a large reduction in `xsim` when using joint training (see Table 3). For example, we see a 40% relative reduction when adding the MLM criterion (21.05 → 12.65), and a further decrease of 12.65 → 5.93 when also adding in curriculum learning. As we also observed a similar effect for other languages, the results above suggest that jointly training distillation alongside masked language modelling and curriculum learning is particularly beneficial for such low resource languages.

6 Encoding and mining very low-resource languages

About 1.2 billion people are living in Africa, and with an estimated number of 2000 languages, Africa is home to approximately one-third of the world's languages. However, to the best of our knowledge, only twenty-two African languages are currently handled by public MT systems. Most of the African languages are considered as very low-resource languages, i.e. less than 100 thousand sentence of bitexts are publicly available. Those resources are mainly religious texts, e.g. Bible translations, which can lead to a domain mismatch when directly training NMT systems on this data.

In this section, we investigate the challenges to train sentence encoders for 50 African languages, perform bitext mining, and train NMT models to translate these African languages into English.

6.1 Choice of African languages

We tried to cover as many African languages as possible. The main limitation was the availability of high-quality test sets to evaluate our models. FLORES-200 covers 44 African languages. We added 6 languages for which we have no FLORES test sets, namely Acholi, Luba, Luvale, Tiv, Venda and Zande, but sufficient resources to train sentence encoders and NMT systems. Statistics for the 44 languages with test data are given in Table 4.

6.2 Encoder training and evaluation

We have explored several techniques to train sentence encoders on multiple languages, grouped into *"families"* in different ways. The largest family of African languages are by far Bantu languages. Other language families include Chadic, Cushtic, Kwa, Mande, Nilotic, Semitic and Senegambian.

ISO	Language	Bitexts [k]	Mono [k]	xsim [%] LabSE	xsim [%] LASER3	Mined [k]	BLEU xxx/eng public	BLEU xxx/eng +mined
afr	**Afrikanns**	2061	189396	**0.00**	0.00	24240	50.59	54.80
aka	Akan	13	163	27.47	0.40	533	0.15	2.05
amh	**Amharic**	448	20959	**0.00**	0.10	9267	14.67	26.60
bam	Bambara	16	347	40.81	4.64	656	0.58	3.62
bem	Bemba	700	2302	12.15	0.10	2166	15.19	17.70
cjk	Chokwe	40	356	34.39	25.20	839	0.00	1.98
dik	Dinka	25	643	38.04	21.84	571	0.00	2.43
dyu	Dyula	67	239	47.04	21.05	1177	0.26	1.12
ewe	Ewe	642	3858	38.64	1.28	3057	11.30	11.21
fon	Fon	44	1277	48.52	14.43	1009	1.05	2.33
fuv	**Fulfulde**	26	1223	32.21	28.46	4509	0.00	6.42
hau	**Hausa**	416	39242	**0.30**	0.49	8454	18.92	29.36
ibo	**Igbo**	524	8124	**0.00**	0.20	5618	17.38	21.00
kam	**Kamba**	58	130	27.47	15.32	948	1.37	2.46
kau_Arab	Kanuri	6	20020	75.00	60.18	3866	0.00	1.05
kau_Latn	Kanuri	11	607	37.85	4.64	307	0.00	2.52
kik	Kikuyu	119	148	27.27	1.28	1416	5.26	8.04
kin	**Kinyarwanda**	2012	12657	**0.20**	0.30	8385	17.42	20.24
kmb	Kimbundu	101	269	35.47	7.31	875	1.99	2.67
kon	Kongo	229	481	24.31	0.99	1497	7.55	9.09
lin	**Lingala**	1038	2192	22.92	0.40	2632	15.99	16.87
lua	Luba-Kasai	325	1481	24.90	1.98	1635	6.83	8.01
lug	**Luganda**	304	3985	13.34	1.09	2901	9.07	12.23
luo	**Luo**	158	1714	35.38	0.49	2244	6.60	11.37
nso	**Northen Sotho**	624	3234	0.30	0.20	2526	22.94	27.54
nus	Nuer	21	128	50.30	7.11	785	0.00	2.97
nya	**Chewa; Nyanja**	867	4424	**0.00**	0.20	6301	17.54	22.27
orm	**Oromo**	177	7576	45.65	0.49	1916	5.54	9.42
run	Rundi	665	3864	0.10	0.49	3428	12.25	15.83
sna	**Shona**	826	13357	**0.30**	0.30	5959	19.27	22.77
som	**Somali**	179	78010	**0.20**	0.69	4935	5.13	21.07
sot	Sotho	1515	8156	**0.00**	0.10	6326	23.28	30.66
ssw	**Swati**	436	1424	2.08	0.40	1407	6.80	14.78
swh	**Swahili**	1871	55400	**0.00**	0.10	14238	31.19	38.57
tir	Tigrinya	115	4789	13.74	0.89	2380	3.54	12.04
tsn	**Tswana**	899	5154	1.28	1.19	4298	19.58	20.18
tso	**Tsonga**	851	4434	22.73	0.79	3294	21.96	23.25
tum	Tumbuka	585	1565	5.53	1.48	2966	8.92	10.92
twi	Twi	630	5520	24.41	0.69	2726	14.51	14.65
umb	**Umbundu**	233	795	36.96	12.45	1299	1.97	3.12
wol	**Wolof**	9	2817	**26.19**	5.93	808	0.00	2.94
xho	**Xhosa**	1176	27718	**0.10**	0.20	6315	26.22	31.57
yor	**Yoruba**	518	41730	**0.69**	3.66	5867	12.49	15.61
zul	**Zulu**	1758	20477	**0.10**	0.20	9167	29.19	33.62

Table 4: List of African languages, available resources and result summary (on FLORES devtest). Languages in bold are handled at WMT'22. LaBSE's `xsim` error rates in bold correspond to languages it was trained on.

We first trained one encoder on all African languages and then tried to improve them by using smaller language family specific models. Unfortunately, several language families have a very small total amount of bitext training data, in particular Mande languages (83k).We were not able to train language specific encoders for these families which performed better than when trained together with all other African languages. The following languages were trained separately: 1) Semitic: amh and tir; 2) Kwa languages: aka, ewe, fon and twi; and 3) Wolof (using MLM training).

Table 4 provides the `xsim` scores for all languages for which we have FLORES devtest data. We always use the LASER2 teacher model for English and not the individual student models (which were also trained on English). This ensures that all students are mutually compatible and simplifies mining. For comparison, we also evaluated the publicly available LaBSE model[8] on our test data. LaBSE was trained on a total of 109 languages which includes 14 African languages (in bold in

[8] https://huggingface.co/sentence-transformers/LaBSE

Table 4). LaBSE performs very well on all of them, except Wolof which has xsim of 26.2%. Our encoder for Wolof achieves 5.9% xsim error. LabSE generalizes well to 4 other languages (nso, run, ssw and tsn). LabSE's xsim scores for the other languages are rather high. Our LASER3 sentence encoders have xsim error rates below 5% for 34 languages. The most difficult languages are: cjk, dik, dyu, fon, fuv, kam, kau, kmb and umb. For most of them, we have a very limited amount of bitexts (less than 50k). In the appendix, we provide the xsim error rates among all African languages as well as against French. This demonstrates that the student encoders are mutually compatible among each others, and with other languages of the LASER2 teacher.

6.3 Bitext evaluation

We now turn to using these encoders for bitext mining. We follow exactly the same margin-based mining procedure as described in Schwenk et al. (2021). Our main source of monolingual data was Common Crawl, complemented by targeted crawling (see section 4 for details on preprocessing). The amount of unique sentences is given in Table 4 in the column "Mono [k]". We mine against 21.5 billion unique sentences in English.

NMT training To evaluate the quality of the mined bitexts, we add the mined bitexts to the available public bitexts and train NMT systems, translating from foreign into English, and compare the BLEU scores with baseline models which were trained on freely available bitexts only. We train NMT systems to translate separately from each language into English. We hope that this gives us signals on the quality of the mined bitexts. For simplicity, we use the same architecture for all languages: a 6 layer transformer for the encoder and decoder, 8 attention heads, ffn=4096 and 512-dimensional embeddings. Models were trained for 100 epochs on 32 GPUs The results are summarized in Table 4, last three columns.

Analysis. We observe significant gains in the BLEU scores for several languages, e.g. fuv, sot, ssw, swh, tir and xho, improve by more than 5 points BLEU, and amh, hau and som by more than 10 BLEU. The most impressive result is obtained for Somali: training an NMT system on the available 179k bitexts yields 5.1 BLEU. This is then improved to 21.1 BLEU by adding 4.9M mined bitexts. We also obtain a nice result on Fulfude: publicly available bitexts are extremely limited (26k) and we are able to reach 6.5 BLEU using mined bitexts, despite a sentence encoder with a high xsim error rate of 28.46%. There are 13 languages with BLEU scores below 5%: aka, bam, cjk, dik, dyu, fon, kam, kau_Arab, kau_Latn, kmb, nus, umb and wol. The sentence encoders for most of these languages need to be improved further, but the limiting factor is often the amount of available monolingual data - we simply have not enough data to mine in. A typical example is Akan (aka): we have a very good sentence encoder - since it was trained jointly with the other Kwa languages, but only 163k sentences of monolingual data. There is not much mining can do here. In average over all 44 languages, BLEU improved from 11.0 to 14.8.

These results should not be considered as the best possible MT performance which can be obtained with the available resources. We made no attempt to optimize the precision/recall trade-off of the mining individually for each language pair, and use the same the margin threshold of 1.06, nor did we adapt the NMT architecture and parameters to the amount of bitexts. Significant improvements in the BLEU scores can be obtained by training a massively multilingual NMT system as demonstrated in NLLB Team et al. (2022). In that work, the same underlying teacher-student approach was used to train student for more than 148 languages and mine more than 880 million sentences of bitexts. Ablation studies have shown that mined data brought an improvement of 12.5 chrF++ when translating into English, averaged over all 200 languages.

7 Acknowledgements

For their helpful contributions to this work, we would like to thank: Bapi Akula, Pierre Andrews, Angela Fan, Cynthia Gao, Kenneth Heafield, Philipp Koehn, Janice Lam, Alex Mourachko, Christophe Rogers and Guillaume Wenzek.

8 Conclusion

Multilingual sentence representations are key to extend NLP approaches to more languages, and they are the underlying engine for distance-based bitext mining, which turned out to be crucial to scale NMT to more languages. In this work, we attack the challenge to scale the LASER encoder and cover 50 African languages. To the best of our knowledge, only 14 African languages are handled and evaluated by current multilingual encoders.

We achieve this by moving away from a *one-for-all* approach to an improved teacher-student training of several encoders, each one trained on a small subset of languages. This enabled us to better adapt the encoders to language specificities, e.g. a particular writing script, while maintaining mutual compatibility. Our new models significantly outperform the original LASER model on the FLORES test set, and we are on par or better than all other publicly available multilingual sentence encoders, namely LaBSE. We were also able to integrate monolingual data by jointly minimizing a cosine and MLM loss. We showcase the potential of this technique for the Wolof language, reducing the `xsim` error rate from 21.1% down to 5.9%. We performed bitext mining for 44 African languages. Adding this data yielded an average improvement of 3.8 point BLEU for translation into English. Training code, the models and the mined bitexts are freely available.[9]

9 Limitations

Our new student-teacher approach to independently train multiple, but mutually compatible sentence encoders, enabled us to attack many low-resource African languages. However, not all have a sufficient low `xsim` error rate to mine high-quality bitexts. It is difficult to say whether these error rates are the result of missing resources to train these encoders, or inherent to the characteristics of each language. Several low-resource languages are also lacking a well-defined and standard writing system: they may be written in different standards or scripts in different regions. This obviously complicates training sentence encoders.

In addition, bitext mining by itself reaches its limits when not enough monolingual data is available, independent of the mining approach which is applied. On one hand, it could be of course that we were simply unable to locate and crawl this monolingual data. On the other hand, however, by handling more and more (very) low-resource languages, we might be faced with languages which are mainly spoken.

References

Jade Abbott and Laura Martinus. 2019. Benchmarking neural machine translation for Southern African languages. In *Proceedings of the 2019 Workshop on Widening NLP*, pages 98–101, Florence, Italy. Association for Computational Linguistics.

Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Dmitry Lepikhin, Melvin Johnson, Maxim Krikun, Mia Xu Chen, Yuan Cao, George F. Foster, Colin Cherry, Wolfgang Macherey, Zhifeng Chen, and Yonghui Wu. 2019. Massively multilingual neural machine translation in the wild: Findings and challenges. *http://arxiv.org/abs/1907.05019*.

Mikel Artetxe and Holger Schwenk. 2019a. Margin-based Parallel Corpus Mining with Multilingual Sentence Embeddings. In *ACL*, pages 3197–3203.

Mikel Artetxe and Holger Schwenk. 2019b. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *TACL*, pages 597–610.

Paul Azunre, Salomey Osei, Salomey Addo, Lawrence Asamoah Adu-Gyamfi, Stephen Moore, Bernard Adabankah, Bernard Opoku, Clara Asare-Nyarko, Samuel Nyarko, Cynthia Amoaba, et al. 2021. NLP for ghanaian languages. *arXiv preprint arXiv:2103.15475*.

Ankur Bapna, Isaac Caswell, Julia Kreutzer, Orhan Firat, Daan van Esch, Aditya Siddhant, Mengmeng Niu, Pallavi Baljekar, Xavier Garcia, Wolfgang Macherey, Theresa Breiner, Vera Axelrod, Jason Riesa, Yuan Cao, Mia Xu Chen, Klaus Macherey, Maxim Krikun, Pidong Wang, Alexander Gutkin, Apurva Shah, Yanping Huang, Zhifeng Chen, Yonghui Wu, and Macduff Hughes. 2022. Building machine translation systems for the next thousand languages. *https://arxiv.org/abs/2205.03983*.

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 *ACL*, pages 8440–8451.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *NeurIPS*.

Raj Dabre and Aneerav Sukhoo. 2022. Morisienmt: A dataset for mauritian creole machine translation. *arXiv preprint arXiv:2206.02421*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, pages 4171–4186.

Chris Chinenye Emezue and Femi Pancrace Bonaventure Dossou. 2020. Ffr v1. 1: Fon-french neural machine translation. In *Proceedings of the The Fourth Widening Natural Language Processing Workshop*, pages 83–87.

[9] https://github.com/facebookresearch/LASER/

Carlos Escolano, Marta R. Costa-jussà, José A. R. Fonollosa, and Mikel Artetxe. 2021. Multilingual machine translation: Closing the gap between shared and language-specific encoder-decoders. In *EACL*, pages 944–948, Online. ACL.

Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, Naman Goyal, Tom Birch, Vitaliy Liptchinsky, Sergey Edunov, Edouard Grave, Michael Auli, and Armand Joulin. 2020. Beyond english-centric multilingual machine translation. *JMLR*.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhaga, and Wei Wang. 2020. Language-agnostic BERT sentence embedding. https://arxiv.org/abs/2007.01852.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzman, and Angela Fan. 2021. The FLORES-101 evaluation benchmark for low-resource and multilingual machine translation. https://arxiv.org/abs/2106.03193.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. https://arxiv.org/abs/1802.06893.

Gilles Hacheme. 2021. English2gbe: A multilingual machine translation model for {Fon/Ewe}gbe. *CoRR*, abs/2112.11482.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020a. XTREME: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization. In *ICML*, pages 4411–4421.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020b. XTREME: a massively multilingual multi-task benchmark for evaluating cross-lingual generalization. In https://arxiv.org/abs/2003.11080.

Shuming Ma, Li Dong, Shaohan Huang, Dongdong Zhang, Alexandre Muzio, Saksham Singhal, Hany Hassan Awadalla, Xia Song, and Furu Wei. 2021. DeltaLM: encoder-decoder pre-training for language generation and translation by augmenting pretrained multilingual encoders. https://arxiv.org/abs/2106.13736.

Wilhelmina Nekoto, Vukosi Marivate, Tshinondiwa Matsila, Timi E. Fasubaa, Tajudeen Kolawole, Taiwo Fagbohungbe, Solomon Oluwole Akinola, Shamsuddeen Hassan Muhammad, Salomon Kabongo, Salomey Osei, Sackey Freshia, Rubungo Andre Niyongabo, Ricky Macharm, Perez Ogayo, Orevaoghene Ahia, Musie Meressa, Mofe Adeyemi, Masabata Mokgesi-Selinga, Lawrence Okegbemi, Laura Jane Martinus, Kolawole Tajudeen, Kevin Degila, Kelechi Ogueji, Kathleen Siminyu, Julia Kreutzer, Jason Webster, Jamiil Toure Ali, Jade Z. Abbott, Iroro Orife, Ignatius Ezeani, Idris Abdulkabir Dangana, Herman Kamper, Hady Elsahar, Goodness Duru, Ghollah Kioko, Espoir Murhabazi, Elan Van Biljon, Daniel Whitenack, Christopher Onyefuluchi, Chris Emezue, Bonaventure Dossou, Blessing Sibanda, Blessing Itoro Bassey, Ayodele Olabiyi, Arshath Ramkilowan, Alp Öktem, Adewale Akinfaderin, and Abdallah Bashir. 2020. Participatory research for low-resourced machine translation: A case study in african languages. *CoRR*, abs/2010.02353.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation.

Gowtham Ramesh, Sumanth Doddapaneni, Aravinth Bheemaraj, Mayank Jobanputra, Raghavan AK, Ajitesh Sharma, Sujit Sahoo, Harshita Diddee, Mahalakshmi J, Divyanshu Kakwani, Navneet Kumar, Aswin Pradeep, Srihari Nagaraj, Kumar Deepak, Vivek Raghavan, Anoop Kunchukuttan, Pratyush Kumar, and Mitesh Shantadevi Khapr. 2022. Samanantar: The largest publicly available parallel corpora collection for 11 indic languages. *TACL*, 10:145–162.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese bert-networks. In *EMNLP*, pages 3982–3992.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *EMNLP*, pages 4512–4525.

Sebastian Ruder, Noah Constant, Jan Botha, Aditya Siddhant, Orhan Firat, Jinlan Fu, Pengfei Liu, Junjie Hu, Dan Garrette, Graham Neubig, and Melvin Johnson. 2021. XTREME-R: Towards more challenging and nuanced multilingual evaluation. In *EMNLP*, pages 10215–10245.

Holger Schwenk, Guillaume Wenzek, Sergey Edunov, Edouard Grave, Armand Joulin, and Angela Fan. 2021. CCMatrix: Mining billions of high-quality parallel sentences on the web. In *ACL*, page 6490–6500.

Aditya Siddhant, Ankur Bapna, Orhan Firat, Yuan Cao, Mia Xu Chen, Isaac Caswell, and Xavier Garcia.

2022. Towards the next 1000 languages in multilingual machine translation: Exploring the synergy between supervised and self-supervised learning. https://arxiv.org/abs/2201.03110.

Kathleen Siminyu, Godson Kalipe, Davor Orlic, Jade Abbott, Vukosi Marivate, Sackey Freshia, Prateek Sibal, Bhanu Neupane, David I. Adelani, Amelia Taylor, Jamiil Toure ALI, Kevin Degila, Momboladji Balogoun, Thierno Ibrahima DIOP, Davis David, Chayma Fourati, Hatem Haddad, and Malek Naski. 2021. AI4D – african language program. https://arxiv.org/abs/2104.02516.

J. Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *LREC*.

Hongyu Wang, Shuming Ma, Li Dong, Shaohan Huang, Dongdong Zhang, and Furu Wei. 2022. DeepNet: scaling transformers to 1,000 layers. https://arxiv.org/abs/2203.00555.

Zihan Wang, Karthikeyan K, Stephen Mayhew, and Dan Roth. 2020. Extending multilingual BERT to low-resource languages. In *EMNLP*, pages 2649–2656.

Yinfei Yang, Gustavo Hernandez Abrego, Steve Yuan, Qinlan Shen Mandy Guo, Daniel Cer, Brian Strope Yun-hsuan Sun and, and Ray Kurzweil. 2019. Improving multilingual sentence embedding using bi-directional dual encoder with additive margin softmax. In *IJCAI*, pages 5370–5378.

A Analysis of zero-short performance of multiple student encoders

We trained our students to minimize the xsim score of each language with respect to the English LASER2 teacher. In order to best consider specificities of languages, several independent student models were trained:

- Semitic languages: amh and tir
- Kwa languages: aka, ewe, fon, and twi
- Senegambian languages: wol (+bsc, cou and fuv)
- remaining 43 languages

This means that for instance the students of the Semitic family have never seen any data of all the other languages. The only link is the common teacher. Table 5 gives the xsim scores for all possible pairs. To limit the size of the table, we consider only the 30 best performing languages i.e., those with the smallest xsim scores. Please note that the table is not symmetric (e.g. eng $\rightarrow$ kon $= 1.48$, while kon $\rightarrow$ eng $= 0.99$).

We observe that the xsim scores amongst the African language pairs are higher than with English, but they stay rather low for most of the pairs (below 5%). As an example, let us consider the two student models for Semitic and Kwa languages. Both were trained on few languages with a small amount of bitexts. Still, we achieve reasonable xsim scores among them: aka $\rightarrow$ amh $= 2.87$, amh $\rightarrow$ tir $= 2.67$, or tir $\rightarrow$ twi $= 5.24$.

Zero-shot performance with French Finally, we also added the xsim scores of all languages with respect to French, encoded by the LASER2 teacher. Please note that none of the student models were trained to minimize the cosine distance to French embeddings. Nevertheless, we observe very low xsim scores, close to those with English. The average xsim error is 0.77 compared to 0.56 against English. There are even some pairs for which the xsim error rates to French are smaller than to English, namely ibo/fra, tsn/fra and zho/fra. This means that we can use our student encoders to mine against all languages supported by the LASER2 encoder.

	afr	aka	amh	bem	eng	ewe	fra	hau	ibo	kik	kin	kon	lin	lua	lug	luo	nso	nya	orm	run	sna	som	sot	ssw	swh	tir	tsn	tso	tum	twi	xho	zul
afr	0.00	5.73	1.48	3.16	0.00	8.89	0.20	3.46	3.36	3.95	2.47	6.92	3.06	7.61	8.10	3.46	2.27	3.26	10.2	4.45	2.47	6.52	1.48	3.36	4.45	7.71	4.55	3.06	6.52	4.94	1.28	1.19
aka	3.56	0.00	2.87	2.37	0.40	7.71	1.09	4.64	3.26	4.05	2.87	5.34	2.47	9.29	8.70	5.04	1.48	3.66	8.70	3.66	2.96	7.02	1.78	2.67	3.36	8.00	3.26	2.37	4.74	1.98	2.17	1.28
amh	1.09	4.74	0.00	1.88	0.10	5.83	0.20	3.36	2.37	2.17	1.28	4.25	2.08	6.92	6.82	2.47	1.09	1.98	5.04	2.67	0.99	4.45	0.40	1.58	4.15	2.67	2.96	2.77	4.15	2.08	0.49	0.69
bem	1.78	2.67	1.09	0.00	0.10	6.13	0.59	2.67	2.08	2.17	1.09	3.26	1.58	4.35	5.83	1.88	0.40	1.09	5.34	1.38	1.19	4.05	0.59	1.19	1.68	5.04	1.78	2.08	2.77	2.67	0.89	0.30
eng	0.00	0.99	0.10	0.10	0.00	2.77	0.00	0.79	0.49	0.69	0.30	1.48	0.20	1.78	2.57	0.79	0.10	0.20	1.28	0.40	0.20	1.98	0.10	0.40	0.89	0.59	1.38	0.69	1.09	0.40	0.10	0.10
ewe	5.04	7.02	4.74	4.45	1.28	0.00	1.68	6.03	6.82	5.73	4.35	7.71	4.64	12.1	11.0	6.13	4.35	4.45	13.1	6.32	4.74	10.1	2.96	4.25	10.8	10.8	5.04	4.25	6.72	5.63	2.96	3.75
fra	0.00	1.58	0.10	0.99	0.00	2.87	0.00	1.09	0.40	0.89	0.40	1.58	0.20	2.57	3.46	0.89	0.40	0.59	2.37	0.59	0.20	2.17	0.20	0.40	1.68	0.99	1.28	0.59	1.98	0.79	0.10	0.10
hau	1.58	3.56	1.88	1.78	0.49	6.82	0.59	0.00	2.08	2.17	2.08	4.45	1.88	6.23	5.34	3.06	1.38	1.28	5.34	2.37	1.68	3.85	0.89	1.68	4.05	5.83	2.57	2.37	3.56	2.57	1.19	0.99
ibo	1.88	3.56	1.58	2.17	0.20	6.62	0.30	2.17	0.00	2.77	2.17	3.85	2.08	6.72	5.04	3.66	1.19	2.27	5.73	2.77	2.17	4.35	1.48	2.17	2.87	5.73	2.67	2.27	5.04	2.37	1.38	0.99
kik	2.57	5.73	2.57	2.47	1.28	7.71	1.38	3.06	2.87	0.00	2.77	5.34	2.67	6.03	6.62	3.16	1.78	2.96	7.91	4.05	2.67	5.53	1.68	2.47	2.47	6.82	2.87	2.47	4.25	3.85	1.38	1.19
kin	1.78	2.67	0.99	1.58	0.30	6.32	0.49	2.67	1.68	2.57	0.00	2.67	1.09	5.34	4.94	2.27	0.59	1.38	4.45	1.78	1.48	3.56	0.89	1.19	1.88	3.16	2.57	1.28	2.77	1.98	0.59	0.59
kon	4.35	5.14	3.36	3.66	0.99	8.70	0.49	5.63	5.14	4.84	4.25	0.00	3.06	9.19	8.20	4.84	2.87	3.95	10.6	5.14	2.87	6.23	2.17	2.96	3.46	9.29	4.45	3.56	5.04	5.24	1.88	2.47
lin	1.68	2.67	1.28	1.28	0.40	4.84	0.40	2.57	2.27	2.96	1.19	2.37	0.10	5.53	5.34	1.88	0.99	1.28	5.04	1.98	0.99	4.35	0.59	1.28	2.17	3.95	2.08	1.68	2.87	1.68	0.40	0.49
lua	5.53	8.70	6.82	4.35	1.98	12.8	2.57	6.72	7.02	6.13	6.03	8.00	5.63	0.00	10.2	6.92	4.15	5.83	13.6	8.70	5.63	8.60	4.74	5.73	6.52	12.8	5.53	5.83	9.09	7.41	4.05	3.85
lug	3.95	8.99	4.15	4.15	1.09	10.4	1.88	6.82	5.43	5.04	4.74	7.41	4.84	11.3	0.00	5.43	3.85	4.25	12.7	6.82	3.75	8.30	2.47	3.95	5.73	13.1	4.74	6.03	8.00	6.62	2.87	3.36
luo	3.26	7.11	3.46	3.36	0.49	10.5	0.89	5.93	5.04	3.75	3.85	6.32	4.15	9.49	9.88	0.00	3.66	4.55	11.4	5.73	4.45	7.11	2.08	3.36	4.35	12.4	4.15	3.66	6.62	5.53	1.88	2.77
nso	0.59	2.17	0.69	0.99	0.20	3.85	0.49	1.78	1.68	1.38	0.69	2.37	0.99	4.35	3.46	1.68	0.00	0.59	3.95	1.48	0.69	3.16	0.40	0.49	0.89	2.27	1.78	1.28	2.96	1.28	0.40	0.30
nya	1.68	3.26	0.99	1.19	0.20	6.03	0.49	2.37	2.87	2.67	1.28	2.87	1.38	5.53	5.73	2.47	1.38	0.00	5.34	2.96	0.79	3.95	0.99	1.09	2.37	4.25	2.37	1.68	2.17	2.67	0.40	0.49
orm	5.04	8.60	2.77	4.05	0.49	12.7	1.48	7.31	5.34	5.63	6.03	8.60	5.63	13.5	13.5	8.30	4.74	5.43	0.00	8.79	6.82	9.19	4.05	6.13	4.55	10.6	5.93	5.73	9.09	6.52	3.95	3.16
run	2.27	3.46	1.09	1.19	0.49	6.52	0.49	2.47	2.27	2.37	0.79	3.26	1.78	7.51	5.73	2.77	0.99	2.08	6.62	0.00	1.09	4.55	0.40	1.28	2.77	4.74	2.27	1.68	2.87	2.17	0.99	0.49
sna	2.47	5.24	1.68	2.08	0.30	6.52	0.89	3.26	2.57	2.77	1.98	3.36	1.88	6.52	6.62	3.06	1.09	1.38	6.62	2.27	0.00	4.15	1.28	1.09	2.96	5.43	2.87	2.08	3.66	2.17	1.09	0.59
som	2.87	6.52	1.28	3.06	0.69	9.58	0.49	3.95	3.46	4.15	3.56	4.84	3.56	9.49	8.10	4.55	2.17	3.56	6.62	5.14	2.96	0.00	1.98	2.67	2.27	7.31	3.85	3.75	6.82	4.94	2.27	1.19
sot	1.09	2.57	0.69	0.69	0.10	4.64	0.00	1.78	0.79	1.58	0.79	1.98	0.69	4.94	3.95	0.99	0.40	0.79	3.56	1.38	0.79	3.06	0.00	0.69	1.58	3.06	1.48	1.38	2.57	1.58	0.20	0.10
ssw	1.88	3.26	1.28	1.58	0.40	5.14	0.49	2.57	2.37	2.57	1.19	2.57	1.98	5.53	4.94	2.27	0.99	1.28	5.93	2.27	1.09	4.15	1.19	0.00	2.37	4.35	2.08	1.88	3.16	2.57	0.69	0.69
swh	1.09	4.15	0.59	1.09	0.10	7.41	0.40	2.27	1.19	1.58	1.09	3.26	2.27	6.03	4.74	2.67	0.99	2.08	5.53	2.27	1.48	2.67	0.49	1.19	0.00	4.35	2.08	1.98	2.96	2.47	0.30	0.30
tir	3.06	6.42	1.38	3.36	0.89	8.30	1.09	4.64	4.15	3.85	2.08	5.73	3.16	9.39	10.5	4.55	2.47	3.46	7.71	4.45	2.27	8.20	1.28	3.16	8.20	0.00	3.95	3.56	7.02	5.24	1.48	1.58
tsn	2.57	4.45	2.08	2.27	1.19	6.32	1.28	3.95	2.77	2.96	2.87	3.75	2.37	5.93	5.93	3.46	1.88	2.27	6.32	3.06	2.37	4.55	1.88	2.17	2.47	4.45	0.10	2.27	4.15	3.56	1.88	1.48
tso	1.38	2.17	1.48	1.58	0.79	4.55	0.79	2.96	1.88	1.98	1.38	3.16	1.38	5.73	6.13	3.16	0.79	1.28	4.35	2.27	1.28	4.15	0.89	1.19	2.37	3.56	2.27	0.00	2.87	1.68	0.79	0.69
tum	3.36	5.53	3.26	2.57	1.48	7.11	1.48	4.15	4.45	3.66	2.87	5.24	3.46	8.40	8.20	4.15	2.37	2.47	8.89	3.75	3.16	6.52	2.08	2.47	4.25	8.00	3.66	2.67	0.00	3.46	1.98	1.78
twi	2.87	2.47	2.08	2.77	0.69	7.11	0.99	4.25	2.96	3.56	2.67	5.73	2.67	6.52	7.02	3.46	1.19	2.87	8.20	4.05	2.27	6.52	1.68	2.77	3.75	7.71	3.16	2.37	4.45	0.00	1.68	1.68
xho	1.19	3.26	0.40	1.09	0.20	5.04	0.20	2.47	1.58	1.48	1.19	2.17	1.09	4.35	4.84	1.68	0.59	1.28	4.94	2.08	0.89	3.66	0.59	0.69	1.78	3.56	2.27	1.48	2.77	1.48	0.00	0.40
zul	0.40	1.68	0.49	0.79	0.20	3.66	0.10	1.28	0.89	0.89	0.79	1.88	0.59	3.66	3.36	1.28	0.49	0.49	2.77	0.69	0.40	2.77	0.40	0.59	1.28	1.78	1.48	0.49	1.98	1.28	0.20	0.00
avg	2.32	4.39	1.89	2.20	0.56	6.88	0.77	3.52	2.95	3.00	2.29	4.25	2.41	6.83	6.60	3.30	1.71	2.40	6.78	3.41	2.15	5.14	1.42	2.14	3.37	5.94	3.02	2.56	4.34	3.19	1.35	1.26

Table 5: `xsim` matrix for subset of African languages, and English and French. All results are on FLORES devtest.