Glot500: Scaling Multilingual Corpora and Language Models to 500 Languages

Ayyoob Imani[*1,2], **Peiqin Lin**[*1,2], **Amir Hossein Kargaran**[1,2], **Silvia Severini**[1], **Masoud Jalili Sabet**[1], **Nora Kassner**[1,2], **Chunlan Ma**[1,2], **Helmut Schmid**[1], **André F. T. Martins**[3,4,5], **François Yvon**[6] and **Hinrich Schütze**[1,2]
[1]CIS, LMU Munich, Germany [2]Munich Center for Machine Learning (MCML), Germany
[3]Instituto Superior Técnico (Lisbon ELLIS Unit) [4]Instituto de Telecomunicações
[5]Unbabel [6]Sorbonne Université, CNRS, ISIR, France
{ayyoob, linpq, amir, silvia}@cis.lmu.de

Abstract

The NLP community has mainly focused on scaling Large Language Models (LLMs) *vertically*, i.e., making them better for about 100 languages. We instead scale LLMs *horizontally*: we create, through continued pretraining, Glot500-m, an LLM that covers 511 predominantly low-resource languages. An important part of this effort is to collect and clean Glot500-c, a corpus that covers these 511 languages and allows us to train Glot500-m. We evaluate Glot500-m on five diverse tasks across these languages. We observe large improvements for both high-resource and low-resource languages compared to an XLM-R baseline. Our analysis shows that no single factor explains the quality of multilingual LLM representations. Rather, a combination of factors determines quality including corpus size, script, "help" from related languages and the total capacity of the model. Our work addresses an important goal of NLP research: we should not limit NLP to a small fraction of the world's languages and instead strive to support as many languages as possible to bring the benefits of NLP technology to all languages and cultures. Code, data and models are available at https://github.com/cisnlp/Glot500.

1 Introduction

The NLP community has mainly focused on scaling Large Language Models (LLMs) *vertically*, i.e., deepening their understanding of high-resource languages by scaling up parameters and training data. While this approach has revolutionized NLP, the achievements are largely limited to high-resource languages. Examples of "vertical" LLMs are GPT3 (Brown et al., 2020), PaLM (Chowdhery et al., 2022) and Bloom (BigScience et al., 2022). In this paper, we create Glot500-m, a model that instead focuses on scaling multilingual LLMs *horizontally*, i.e., scaling to a large number of languages the great majority of which is low-resource. As LLMs are essential for progress in NLP, lack of LLMs supporting low-resource languages is a serious impediment to bringing NLP to all of the world's languages and cultures. Our goal is to address this need with the creation of Glot500-m.[1]

Existing multilingual LLMs support only about 100 (Conneau et al., 2020) out of the 7000 languages of the world. These supported languages are the ones for which large amounts of training data are available through projects such as Oscar (Suárez et al., 2019) and the Wikipedia dumps.[2] Following Siddhant et al. (2022), we refer to the 100 languages covered by XLM-R (Conneau et al., 2020) as **head languages** and to the remaining languages as **tail languages**. This terminology is motivated by the skewed distribution of available data per language: for the best-resourced languages there are huge corpora available, but for the long tail of languages, only small corpora exist. This is a key problem we address: the availability of data for tail languages is limited compared to head languages. As a result, tail languages have often been ignored by language technologies (Joshi et al., 2020).

Although there exists some work on machine translation for a large number of tail languages (Costa-jussà et al., 2022; Bapna et al., 2022), existing LLMs for tail languages are limited to a relatively small number of languages (Wang et al., 2019; Alabi et al., 2022; Wang et al., 2022). In this paper, we address this gap. Our work has three parts. (i) **Corpus collection.** We collect Glot2000-c, a corpus covering thousands of tail languages. (ii) **Model training.** Using Glot500-c, a subset of Glot2000-c, we train Glot500-m, an LLM covering 511 languages. (iii) **Validation.** We conduct an extensive evaluation of the quality of Glot500-m's

[*]Equal contribution.

[1]In concurrent work, Adebara et al. (2022) train a multilingual model for 517 African languages on a 42 gigabyte corpus, but without making the model available.

[2]https://dumps.wikimedia.org/

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 1082–1117
July 9-14, 2023

representations of tail languages on a diverse suite of tasks.

In more detail, **corpus collection** considers three major sources: websites that are known to publish content in specific languages, corpora with classified multilingual content and datasets published in specific tail languages. The resulting dataset Glot2000-c comprises 700GB in 2266 languages collected from ≈150 sources. After cleaning and deduplication, we create the subset Glot500-c, consisting of 511 languages and 534 *language-scripts* (where we define a language-script as a combination of ISO 639-3[3] and script) to train Glot500-m. Our criterion for including a language-script in Glot500-c is that it includes more than 30,000 sentences.

Model training. To train Glot500-m, we employ vocabulary extension and continued pretraining. XLM-R's vocabulary is extended with new tokens trained on Glot500-c. We then perform continued pretraining of XLM-R with the MLM objective (Devlin et al., 2019).

Validation. We comprehensively evaluate Glot500-m on a diverse suite of natural language understanding, sequence labeling and multilingual tasks for hundreds of languages. The results demonstrate that Glot500-m performs better than XLM-R-B (XLM-R-base) for tail languages by a large margin while performing comparably (or better) for head languages.

Previous work on multilinguality has been hindered by the lack of LLMs supporting a large number of languages. This limitation has led to studies being conducted in settings dissimilar from real-world scenarios. For example, Dufter and Schütze (2020) use synthetic language data. And the curse of multilinguality has been primarily studied for a set of high-resource languages (Conneau et al., 2020). By creating Glot500-m, we can investigate these issues in a more realistic setting. We make code, data and trained models available to foster research by the community on how to include hundreds of languages that are currently ill-served by NLP technology.

Contributions. (i) We train the multilingual model Glot500-m on a 600GB corpus, covering more than 500 diverse languages, and make it publicly available at https://github.com/cisnlp/Glot500. (ii) We collect and clean Glot500-c, a corpus that covers these diverse languages and allows us to train Glot500-m, and will make as much of it publicly available as possible. (iii) We evaluate Glot500-m on pseudoperplexity and on five diverse tasks across these languages. We observe large improvements for low-resource languages compared to an XLM-R baseline. (iv) Our extensive analysis shows that no single factor explains the quality of multilingual LLM representations. Rather, a combination of factors determines quality including corpus size, script, "help" from related languages and the total capacity of the model. (v) Our work addresses an important goal of NLP research: we should not limit NLP to a relatively small number of high-resource languages and instead strive to support as many languages as possible to bring the benefits of NLP to all languages and cultures.

2 Related Work

Training multilingual LLMs using the masked language modeling (MLM) objective is effective to achieve cross-lingual representations (Devlin et al., 2019; Conneau et al., 2020). These models can be further improved by incorporating techniques such as discriminative pre-training (Chi et al., 2022) and the use of parallel data (Yang et al., 2020; Chi et al., 2021). However, this primarily benefits a limited set of languages with large corpora.

Recent research has attempted to extend existing LLMs to languages with limited resources. Wang et al. (2019) propose vocabulary extension; Ebrahimi and Kann (2021) investigate adaptation methods, including MLM and Translation Language Model (TLM) objectives and adapters; Alabi et al. (2022) adapt XLM-R to 17 African languages; Wang et al. (2022) expand language models to low-resource languages using bilingual lexicons.

Alternatively, parameter-efficient fine-tuning adapts pre-trained models to new languages by training a small set of weights effectively (Zhao et al., 2020; Pfeiffer et al., 2021; Ansell et al., 2022). Pfeiffer et al. (2022) address the "curse of multilinguality" by sharing a part of the model among all languages and having separate modules for each language. We show that the common perception that multilinguality increases as we add more languages, until, from some point, it starts decreasing, is naive. The amount of available data per language and the similarity between languages also play important roles (§6.8).

Another approach trains LLMs from scratch for a limited number of tail languages; e.g., AfriBERTa

[3] https://iso639-3.sil.org/code_tables/639

(Ogueji et al., 2021a) and IndicNLPSuite (Kakwani et al., 2020) are LLMs for 11 African languages and 11 Indic languages. In concurrent work, Adebara et al. (2022) train a multilingual model for 517 African languages on a 42 GB corpus, but without making the model available and with an evaluation on a smaller number of languages than ours.

Closely related to our work on corpus creation, Bapna et al. (2022) and Costa-jussà et al. (2022) also create NLP resources for a large number of tail languages. They train a language identifier model and extract textual data for tail languages from large-scale web crawls. This approach is effective, but it requires significant computational resources and native speakers for all tail languages. This is hard to do outside of large corporations. Bapna et al. (2022) have not made their data available. Costa-jussà et al. (2022) have only released a portion of their data in around 200 languages.

A key benefit of "horizontally" scaled multilingual LLMs is transfer from high- to low-resource languages. Our evaluation suggests that Glot500-m excels at this, but this is not the main focus of our paper. There is a large body of work on crosslingual transfer: (Artetxe and Schwenk, 2019; Imani-Googhari et al., 2022; Lauscher et al., 2020; Conneau et al., 2020; Turc et al., 2021; Fan et al., 2021; Severini et al., 2022; Choenni and Shutova, 2022; Wang et al., 2023), inter alia.

3 Glot2000-c

3.1 Data Collection

One of the major challenges in developing NLP technologies for tail languages is the scarcity of high-quality training data. In this work, we propose a lightweight methodology that is easily replicable for academic labs. We identify tail language data previously published by researchers, publishers and translators and then crawl or download them. By crawling a few websites and compiling data from around 150 different datasets, we amass more than 700GB of text in 2266 languages. We will refer to these sources of data as *data sources*. Our data covers many domains, including religious texts, news articles and scientific papers. Some of the data sources are high-quality, verified by native speakers, translators and linguists. Others are less reliable such as web crawls and Wikipedia dumps. It is therefore necessary to clean the data. For a list of data sources, see §C.

3.2 Language-Scripts

Some languages are written in multiple scripts; e.g., Tajik is written in both Cyrillic and Arabic scripts. Some data sources indicate the script, but others either do not or provide mixed text in multiple scripts. We detect the script for each sentence and treat each language-script as a separate entity.

3.3 Ngram LMs and Language Divergence

We train a 3-gram character-level language model M_i for each language-script L_i, using KenLM (Heafield, 2011). We refer to the perplexity calculated for the corpus of language L_i using language model M_j as $\mathcal{PP}(M_j, L_i)$. Similar to Gamallo et al. (2017), we define a perplexity-based divergence measure of languages L_i and L_j as:

$$\mathcal{D}_{L_i,L_j} = \max\left(\mathcal{PP}(M_j, L_i), \mathcal{PP}(M_i, L_j)\right)$$

We use $\mathcal{D}$ to filter out noisy data in §3.4 and study the effect of similar languages in LLM training in §6.7 and §6.8. For more details, see §A.

3.4 Data Cleaning

To remove noise, we use chunk-level and corpus-level filters.

While some sources are sentence-split, others provide multiple sentences (e.g., a paragraph) as one chunk. Chunk-level filters process each chunk of text from a data source as a unit, without sentence-splitting. Some chunk-level filters are based on the notion of word: we use white space tokenization when possible and otherwise resort to sentencePiece (Kudo and Richardson, 2018) trained by Costa-jussà et al. (2022).

As chunk-level filters, we employ the **sentence-level filters** SF1–SF5 from BigScience ROOTS (Laurençon et al., 2022).

SF1 Character repetition. If the ratio of repeated characters is too high, it is likely that the sentence has not enough textual content.

SF2 Word repetition. A high ratio of repeated words indicates non-useful repetitive content.

SF3 Special characters. Sentences with a high ratio of special characters are likely to be crawling artifacts or computer code.

SF4 Insufficient number of words. Since training language models requires enough context, very small chunks of text are not useful.

SF5 Deduplication. If two sentences are identical after eliminating punctuation and white space, one is removed.

	langs	*scripts*	*sents'*	*median s'*
Glot2000-c	2266	35	2.3B	8K
Glot500-c	511	30	1.5B	120K
Costa-jussà et al. (2022)	134	-	2.4B	3.3M
Bapna et al. (2022)	1503	-	1.7B	25K

Table 1: Statistics for Glot2000-c, Glot500-c and existing multilingual datasets: number of languages, scripts, sentences' and median number of sentences' per language-script.

In the rest of the paper, we refer to a chunk as a **sentence'**. A sentence' can consist of a short segment, a complete sentence or a chunk (i.e., several sentences).

Corpus-level filters detect if the corpus of a language-script is noisy; e.g., the corpus is in another language or consists of non-meaningful content such as tabular data. We employ filters CF1 and CF2.

CF1 In case of **mismatch between language and script**, the corpus is removed; e.g., Chinese written in Arabic is unlikely to be Chinese.

CF2 Perplexity mismatch. For each language-script L1, we find its closest language-script L2: the language-script with the lowest perplexity divergence (§3.3). If L1 and L2 are not in the same typological family, we check L1/L2 manually and take appropriate action such as removing the corpus (e.g., if it is actually English) or correcting the ISO code assigned to the corpus.

3.5 Training Data: Glot500-c

Among the 2000+ language-scripts that we collected data for, after cleaning, most have too little data for pretraining LLMs. It is difficult to quantify the minimum amount needed for pretraining. Therefore, we pick a relatively high "safe" threshold, 30,000 sentences', for inclusion of language-scripts in model training. This allows us to train the model effectively and cover many low-resource languages. Table 1 gives Glot500-c statistics. See §B for a list of language-scripts. We train Glot500-m on Glot500-c; note that while Glot500-c focuses on tail languages, it contains some data in head languages which we include in Glot500-m training to prevent catastrophic forgetting.

We divide the corpus for each language into train/dev/test, reserving 1000 sentences' each for dev and test and using the rest for train. We pick 1000 parallel verses if we have a Bible translation and add 500 each to test and dev. These parallel verses convey identical meanings and facilitate crosslingual evaluation. We pretrain the model using only the training data.

	XLM-R-B	XLM-R-L	Glot500-m
Model Size	278M	560M	395M
Vocab Size	250K	250K	401K
Transformer Size	86M	303M	86M

Table 2: Model sizes. Glot500-m and XLM-R-B have the same transformer size, but Glot500-m has a larger vocabulary, resulting in an overall larger model.

4 Glot500-m

4.1 Vocabulary Extension

To extend XLM-R's vocabulary, we use SentencePiece (Kudo and Richardson, 2018) with a unigram language model (Kudo, 2018) to train a tokenizer with a vocabulary size of 250K on Glot500-c. We sample data from different language-scripts according to a multinomial distribution, with α=.3. The amount we sample for head languages is the same as tail languages with the lowest amount; this favors tail languages – head languages are already well learned by XLM-R. We merge the obtained tokens with XLM-R's vocabulary. About 100K new tokens were in fact old tokens, i.e., already part of XLM-R's vocabulary. We take the probabilities of the (genuinely) new tokens directly from SentencePiece. After adding the 151K new tokens to XLM-R's vocabulary (which has size 250K), the vocabulary size of Glot500-m is 401K.

We could also calculate probabilities of existing and new tokens over a mixture of original XLM-R training corpus and Glot500-c (Chung et al., 2020). For head languages, the percentage of changed tokens using the new tokenizer compared to the original tokenizer ranges from 0.2% to 50%. However, we found no relationship between percentage of changed tokens and change in performance on downstream tasks. Thus, there was little effect of tokenization in our experiments.

4.2 Continued Pretraining

We create Glot500-m by continued pretraining of XLM-R-B with the MLM objective. The optimizer used is Adam with betas (0.9, 0.999). Initial learning rate: 5e-5. Each training step contains a batch of 384 training samples randomly picked from all language-scripts. The sampling strategy across language-scripts is the same as for vocabu-

	|head|	|tail|	measure (%)
Sentence Retrieval Tatoeba	70	28	Top10 Acc.
Sentence Retrieval Bible	94	275	Top10 Acc.
Text Classification	90	264	F1
NER	89	75	F1
POS	63	28	F1
Roundtrip Alignment	85	288	Accuracy

Table 3: Evaluation tasks and measures. |head|/|tail|: number of head/tail language-scripts

lary extension (§4.1). We save checkpoints every 10K steps and select the checkpoint with the best average performance on downstream tasks by early stopping. Table 2 lists the sizes of XLM-R-B, XLM-R-L and Glot500-m. Except for a larger vocabulary (§4.1), Glot500-m has the same size as XLM-R-B. We train Glot500-m on a server with eight NVIDIA RTX A6000 GPUs for two weeks.

Similar to XLM-R, we concatenate sentences' of a language-script and feed them as a stream to the tokenizer. The resulting output is then divided into chunks of 512 tokens and fed to the model.

5 Experimental Setup

For most tail languages, there are no manually labeled evaluation data. We therefore adopt a mixed evaluation strategy: based partly on human labels, partly on evaluation methods that are applicable to many languages without requiring gold data. Table 3 lists all our evaluation tasks.

Perplexity Following Salazar et al. (2020), we calculate pseudoperplexity (PPPL) over the held-out test set. PPPL is based on masking tokens one-by-one (not left to right). Salazar et al. (2020) give evidence that PPPL is a better measure of linguistic acceptability compared to standard left-to-right perplexity.

Roundtrip Alignment For assessing the quality of multilingual representations for a broad range of tail languages without human gold data, we adopt roundtrip evaluation (Dufter et al., 2018). We first word-align sentences' in a parallel corpus based on the multilingual representations of an LLM. We then start from a word w in a sentence' in language-script L1, follow the alignment links to its translations in language-script L2, then the alignment links from L2 to L3 and so on, until in the end we follow alignment links back to L1. If this "roundtrip" gets us back to w, then it indicates that the LLM has similar representations for the meaning of w in language-scripts L1, L2, L3, etc. In other words, the cross-lingual quality of representations is high. Vice versa, failure to get back to w is a sign of poor multilingual representations.

We use SimAlign (Jalili Sabet et al., 2020) and align on the sub-word level on the Bible part of test, based on the representations of the LLM computed by transformer layer 8 as suggested in the original paper. We use intersection symmetrization: each word in a sentence' is aligned to at most one word in the other sentence'.

As evaluation measure we compute the percentage of roundtrips that were successes, i.e., the roundtrip starts at w in L1 and returns back to w. For each language-script in test, we randomly select three language-scripts as intermediate points L2, L3, L4. Since the intermediate points influence the results, we run the experiment five times with different intermediate points and report the average. All models are evaluated with the same five sets of three intermediate language-scripts.

Sequence Labeling We consider two sequence labeling tasks: Named Entity Recognition (NER) and Part-Of-Speech (POS) tagging. We use the WikiANN dataset (Pan et al., 2017) for NER and version v2.11 of Universal Dependencies (UD) (de Marneffe et al., 2021) for POS. Since training data does not exist for some languages, we finetune on English (with early stopping based on dev) and evaluate zero-shot transfer on all languages covered by WikiANN/UD. We set the learning rate to 2e-5 with Adam.

Sentence Retrieval Following (Hu et al., 2020), we use up to 1000 English-aligned sentences' from Tatoeba (Artetxe and Schwenk, 2019) to evaluate SentRetr (sentence retrieval). We also use 500 English-aligned sentences' from the Bible part of test. We find nearest neighbors using cosine similarity based on the average word embeddings in layer $l = 8$ – following Jalili Sabet et al. (2020) – and compute top10 accuracy. For fair comparison and because the architectures are the same, we do not optimize the hyperparameter l for Glot500-m and XLM-R-B.

Text Classification We evaluate on Taxi1500 (Ma et al., 2023). It provides gold data for text classification with six classes in a large number of language-scripts of which Glot500-m supports 354. We finetune on English (with early stopping on dev) and evaluate zero-shot on test of the target language-script. Learning rate: 2e-5, batch size:

16 (following Ma et al. (2023)).

6 Experiments

In this section, we discuss aggregate results. For detailed results, see §D and §E.

6.1 Results

Table 4 gives results. Glot500-m outperforms XLM-R-B on all tasks for both head and tail language-scripts, except for POS on head. That Glot500-m outperforms XLM-R-B is expected for tail language-scripts (i.e., those not covered by XLM-R). For these language-scripts the improvement margin is large. Outperformance may seem counterintuitive for head language-scripts (those covered by XLM-R) since Glot500-m has the same number of (non-embedding) parameters as XLM-R-B. Since the number of covered languages has greatly increased, leaving less capacity per language, we might expect underperformance. There are a few possible explanations. First, XLM-R may be undertrained, and the inclusion of more head language training data may improve their representations. Second, having more languages may improve multilinguality by allowing languages to synergize and enhance each other's representations and cross-lingual transfer. Third, there are languages similar to head languages among the tail languages, which in turn aids head languages.

The gap between Glot500-m and the baselines for tail language-scripts in sequence labeling is smaller. These tasks do not require as deep an understanding of language and thus transfer from head to tail language-scripts is easier through shared tokens.

Glot500-m also outperforms XLM-R-L for tail language-scripts (all tasks) and head language-scripts (3 tasks). This suggests that scaling up size is not the only way for improvements. We can also improve the quality of multilingual LLM representations by increasing the number of languages.

6.2 Language Coverage

Table 5 compares Glot500-m vs. XLM-R-B on pseudoperplexity. For fair comparison we use word-level normalization. For 69 head language-scripts, Glot500-m underperforms XLM-R-B. This is expected as Glot500-m's training data is small for these language-scripts. Glot500-m outperforms XLM-R-B for 420 tail language-scripts.

There are eight tail language-scripts for which

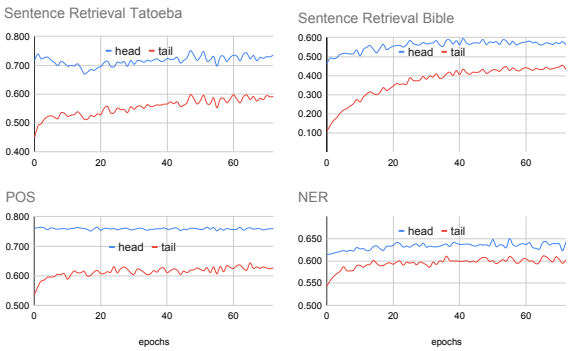

Figure 1: Progression of training for sentence retrieval and sequence labeling. x-axis: epochs/10K. The improvement is fast in the beginning for tail languages, then gets slower and and reaches a plateau. This pattern is partially observed for head languages.

Glot500-m performs worse than XLM-R-B. Five are tail languages with a similar head language where the two share a macro-language: ekk/Standard Estonian (est/Estonian), aln/Gheg Albanian (sqi/Albanian), nob/Norwegian Bokmal (nor/Norwegian), hbs/Serbo-Croatian (srp/Serbian), lvs/Standard Latvian (lav/Latvian). Since XLM-R-B's pretraining corpus is large for the five head languages, its performance is good for the close tail languages.

The other three languages all have a unique script: sat/Santali (Ol Chiki script), div/Dhivehi (Thaana script), iku/Inuktitut (Inuktitut syllabics). For these languages, XLM-R-B's tokenizer returns many UNK tokens since it is not trained on these scripts, resulting in an unreasonably optimistic estimate of pseudoperplexity by our implementation.

Glot500-m's token-level normalized pseudoperplexity ranges from 1.95 for lhu/Lahu to 94.4 for tok/Toki Pona. The average is 13.5, the median 10.6. We analyze the five language-scripts with the highest pseudoperplexity: tok_Latn, luo_Latn, acm_Arab, ach_Latn, and teo_Latn.

tok/Toki Pona is a constructed language. According to Wikipedia: "Essentially identical concepts can be described by different words as the choice relies on the speaker's perception and experience." This property can result in higher variability and higher perplexity.

acm/Mesopotamian Arabic contains a large number of tweets in raw form. This may result in difficult-to-predict tokens in test.

luo/Luo, ach/Acoli and teo/Teso are related Nilotic languages spoken in Kenya, Tanzania, Uganda and South Sudan. Their high perplex-

	tail			head			all		
	XLM-R-B	XLM-R-L	Glot500-m	XLM-R-B	XLM-R-L	Glot500-m	XLM-R-B	XLM-R-L	Glot500-m
Pseudoperplexity	304.2	168.6	**12.2**	12.5	**8.4**	11.8	247.8	136.4	**11.6**
Sentence Retrieval Tatoeba	32.6	33.6	**59.8**	66.2	71.1	**75.0**	56.6	60.4	**70.7**
Sentence Retrieval Bible	7.4	7.1	**43.2**	54.2	58.3	**59.0**	19.3	20.1	**47.3**
Text Classification	13.7	13.9	**46.6**	51.3	**60.5**	54.7	23.3	25.8	**48.7**
NER	47.5	51.8	**60.7**	61.8	**66.0**	63.9	55.3	59.5	**62.4**
POS	41.7	43.5	**62.3**	76.4	**78.4**	76.0	65.8	67.7	**71.8**
Roundtrip Alignment	2.6	3.1	**4.5**	3.4	4.1	**5.5**	2.8	3.3	**4.7**

Table 4: Evaluation of XLM-R base and large (XLM-R-B and XLM-R-L) and Glot500-m on pseudoperplexity and six multilingual tasks across 5 seeds. Each number is an average over head, tail and all language-scripts. See §D, §E for results per task and language-script. Glot500-m outperforms XLM-R-B in all tasks for head (except for POS) and tail language-scripts and XLM-R-L for tail language-scripts. Best result per row/column group in bold.

	head	tail
Glot500-m is better	37	420
XLM-R-B is better	69	8

Table 5: Pseudoperplexity Glot500-m vs XLM-R-B. Glot500-m's worse performance on head can be attributed to smaller training corpora and the relative difficulty of learning five times more languages with the same number of (non-embedding) parameters. Glot500-m performs better on almost all tail language-scripts. §6.2 discusses the eight exceptions.

ity could be related to the fact that they are tonal languages, but the tones are not orthographically indicated. Another possible explanation is that the training data is dominated by one subcorpus (Jehova's Witnesses) whereas the test data are dominated by PBC. There are orthographic differences between the two, e.g., "dong" (JW) vs. "doŋ" (PBC) for Acoli. These three languages are also spoken over a large area in countries with different standard languages, which could increase variability.

Our analysis is not conclusive. We note however that the gap between the three languages and the next most difficult languages in terms of pseudoperplexity is not large. So maybe Luo, Acoli and Teso are simply (for reasons still to be determined) languages that have higher perplexity than others.

6.3 Training Progression

To analyze the training process, we evaluate Glot500-m on sequence labeling and SentRetr at 10,000-step intervals. Figure 1 shows that performance improves rapidly at the onset of training, but then the rate of improvement slows down. This trend is particularly pronounced for tail languages in SentRetr. In comparison, sequence labeling is relatively straightforward, with the baseline (XLM-R-B, epoch 0) achieving high performance by correctly transferring prevalent classes such as *verb* and *noun* through shared vocabulary, resulting in a smaller improvement of Glot500-m vs. XLM-R-B.

For SentRetr, we observe larger improvements for the Bible than for Tatoeba. This is likely due to the higher proportion of religious data in Glot500-c, compared to XLM-R's training data (i.e., CC100).

The average performance on downstream tasks peaks at 480K steps. We have taken a snapshot of Glot500-m at this stage and released it.

6.4 Analysis across Language-Scripts

To analyze the effect of language-scripts, we select five tail language-scripts each with the largest and smallest gain when comparing Glot500-m vs. XLM-R-B for SentRetr and sequence labeling.

Table 6 shows that Glot500-m improves languages with scripts not covered by XLM-R (e.g., div/Dhivehi, Thaana script, see §6.2) by a large margin since XLM-R simply regards the uncovered scripts as unknown tokens and cannot compute meaningful representations for the input. The large amount of data we collected in Glot500-c also contributes to the improvement for tail languages, e.g., for tat_Cyrl (Tatar) in SentRetr Tatoeba and mlt_Latn (Maltese) in POS. See §6.7 for a detailed analysis of the effect of corpus size.

On the other hand, Glot500-m achieves just comparable or even worse results for some language-scripts. We see at least three explanations. (i) As discussed in §6.2, some tail languages (e.g., nob/Norwegian Bokmal) are close to a head language (e.g., nor/Norwegian), so Glot500-m has no advantage over XLM-R-B. (ii) A language is at the low end of our corpus size range (i.e., 30,000 sentences'). Example: xav_Latn, Xavánte. (iii) Some languages are completely distinct from all other languages in Glot500-c, thus without support from any similar language. An example is mau_Latn, Huautla Mazatec. Glot500-m has a much harder

		language-script	XLMR	Glot500	gain		language-script	XLMR	Glot500	gain
high end	SentRetr Tatoeba	tat C Tatar	10.3	70.3	60.0	SentRetr Bible	uzn C Northern Uzbek	5.4	87.0	81.6
		nds L Low German	28.8	77.1	48.3		crs L Seselwa Creole	7.4	80.6	73.2
		tuk L Turkmen	16.3	63.5	47.3		srn L Sranan Tongo	6.8	79.8	73.0
		ile L Interlingue	34.6	75.6	41.0		uzb C Uzbek	6.2	78.8	72.6
		uzb C Uzbek	25.2	64.5	39.3		bcl L Central Bikol	10.2	79.8	69.6
low end		dtp L Kadazan Dusun	5.6	21.1	15.5		xav L Xavánte	2.2	5.0	2.8
		kab L Kabyle	3.7	16.4	12.7		mau L Huautla Mazatec	2.4	3.6	1.2
		pam L Pampanga	4.8	11.0	6.2		ahk L Akha	3.0	3.2	0.2
		lvs L Standard Latvian	73.4	76.9	3.5		aln L Gheg Albanian	67.8	67.6	-0.2
		nob L Bokmål	93.5	95.7	2.2		nob L Bokmål	82.8	79.2	-3.6
high end	NER	div T Dhivehi	0.0	50.9	50.9	POS	mlt L Maltese	21.3	80.3	59.0
		che C Chechen	15.3	61.2	45.9		sah C Yakut	21.9	76.9	55.0
		mri L Maori	16.0	58.9	42.9		sme L Northern Sami	29.6	73.6	44.1
		nan L Min Nan	42.3	84.9	42.6		yor L Yoruba	22.8	64.2	41.4
		tgk C Tajik	26.3	66.4	40.0		quc L K'iche'	28.5	64.1	35.6
low end		zea L Zeeuws	68.1	67.3	-0.8		lzh H Literary Chinese	11.7	18.4	6.7
		vol L Volapük	60.0	59.0	-1.0		nap L Neapolitan	47.1	50.0	2.9
		min L Minangkabau	42.3	40.4	-1.8		hyw A Western Armenian	79.1	81.1	2.0
		wuu H Wu Chinese	28.9	23.9	-5.0		kmr L Northern Kurdish	73.5	75.2	1.7
		lzh H Literary Chinese	15.7	10.3	-5.4		aln L Gheg Albanian	54.7	51.2	-3.5

Table 6: Results for five tail language-scripts each with the largest (high end) and smallest (low end) gain Glot500-m vs. XLM-R-B for four tasks. Glot500-m's gain over XLM-R-B is large at the high end and small or slightly negative at the low end. L = Latin, C = Cyrillic, H = Hani, A = Armenian, T = Thaana

lang-script		XLM-R-B	Glot500-m	gain
uig_Arab	head	45.8	56.2	10.4
uig_Latn	tail	9.8	62.8	53.0
hin_Deva	head	67.0	76.6	9.6
hin_Latn	tail	13.6	43.2	29.6
uzb_Latn	head	54.8	67.6	12.8
uzb_Cyrl	tail	6.2	78.8	72.6
kaa_Cyrl	tail	17.6	73.8	56.2
kaa_Latn	tail	9.2	43.4	34.2
kmr_Cyrl	tail	4.0	42.4	38.4
kmr_Latn	tail	35.8	63.0	27.2
tuk_Cyrl	tail	13.6	65.0	51.4
tuk_Latn	tail	9.6	66.2	56.6

Table 7: Sentence Retrieval Bible performance of Glot500-m and XLM-R-B for six languages with two scripts: Uighur (uig), Hindi (hin), Uzbek (uzb), Kara-Kalpak (kaa), Northern Kurdish (kmr), Turkmen (tuk). Glot500-m clearly outperforms XLM-R-B with large differences for tail language-scripts.

time learning good representations in these cases.

6.5 Languages with Multiple Scripts

Table 7 compares SentRetr performance XLM-R-B vs. Glot500-m for six languages with two scripts. Unsurprisingly, XLM-R performs much better for a language-script it was pretrained on ("head") than on one that it was not ("tail"). We can improve the performance of a language, even surpassing the language-script covered by XLM-R, if we collect enough data for its script not covered by XLM-R. For languages with two scripts not covered by XLM-R, the performance is better for the script for which we collect a larger corpus. For example, kaa_Cyrl (Kara-Kalpak) has about three times as much data as kaa_Latn. This explains why kaa_Cyrl outperforms kaa_Latn by 30%.

Dufter and Schütze (2020) found that, after training a multilingual model with two scripts for English (natural English and "fake English"), the model performed well at zero-shot transfer if the capacity of the model was of the right size (i.e., not too small, not too large). Our experiments with real data show the complexity of the issue: even if there is a "right" size for an LLM that supports both full acquisition of languages and multilingual transfer, this size is difficult to determine and it may be different for different language pairs in a large horizontally scaled model like Glot500-m.

6.6 Analysis across Language Families

Table 8 compares SentRetr performance Glot500-m vs. XLM-R-B for seven language families that have ten or more language-scripts in Glot500-c. We assign languages to families based on Glottolog.[4] Generally, XLM-R has better performance the more language-scripts from a language family are represented in its training data; e.g., performance is better for indo1319 and worse for maya1287. The results suggest that Glot500-m's improvement over

[4] http://glottolog.org/glottolog/family

family	$\|L_G\|$	$\|L_X\|$	XLM-R-B	Glot500-m	gain
indo1319	91	50	41.5	61.4	19.9
atla1278	69	2	5.5	45.2	39.6
aust1307	53	6	13.7	47.0	33.2
turk1311	22	7	20.1	62.9	42.8
sino1245	22	2	7.6	38.9	31.3
maya1287	15	0	3.8	20.3	16.4
afro1255	12	5	13.0	34.3	21.4

Table 8: Average Sentence Retrieval Bible performance of Glot500-m and XLM-R-B for seven language families. The difference in coverage of a family by Glot500-m vs. XLM-R-B is partially predictive of the performance difference. $|L_G|/|L_X|$: number of language-scripts from family covered by Glot500-m/XLM-R.

lang-script	Glot+1	Glot500-m
rug_Latn, Roviana	**51.0**	49.0
yan_Latn, Mayangna/Sumo	**46.4**	31.8
wbm_Latn, Wa/Va	**49.6**	46.4
ctd_Latn, Tedim Chin	47.4	**59.4**
quh_Latn, Southern Quechua	33.4	**56.2**
tat_Cyrl, Tatar	58.8	**67.2**

Table 9: Performance on Sentence Retrieval Bible of continued pretraining on just one language-script (Glot+1) vs. on Glot500-c (Glot500-m). Glot500-m underperforms on the top three and outperforms on the bottom three. Our explanation is that the second group is supported by closely related languages in Glot500-c; e.g., for Southern Quechua (quh), Glot500-m also covers closely related Cuzco Quechua (quz). For the first group this is not the case; e.g., the Wa language (wbm) has no close relative in Glot500-c.

XLM-R is the larger, the better our training corpus Glot500-c's coverage is of a family.

6.7 Effect of Amount of Training Data

We examine correlation between pretraining corpus size and Glot500-m zero-shot performance. We focus on SentRetr Bible (§5) since it supports the most head and tail languages. We find that Pearson's $r = .34$, i.e., corpus size and performance are moderately, but clearly correlated. We suspect that the correlation is not larger because, in addition to corpus size of language l itself, corpus size of languages closely related to l is also an important factor (see §6.4 for a similar finding for Norwegian). We therefore also compute Pearson's r between (i) performance of language l on SentRetr Bible and (ii) joint corpus size of l and its k nearest neighbors (according to perplexity divergence, §3.3). In this case, Pearson's $r = .44$ (for both $k = 3$ and $k = 4$), indicating that the corpus size of nearest neighbor languages does play a role.

6.8 Support through Related Languages

Building on §6.7, there is another way we can investigate the positive effect of closely related languages on performance: We can compare performance (again on SentRetr Bible) of continued pretraining on just one language (we refer to this model as Glot+1) vs. on all 511 languages represented in Glot500-c (i.e., Glot500-m). Table 9 presents results for six language-scripts selected from various language families and suggests that some languages do not receive support from related languages (top three). In that case, Glot+1 can fully concentrate on learning the isolated language and does better than Glot500-c. Other languages (bottom three) do receive support from related languages. For example, Southern Quechua (quh) seems to receive support in Glot500-m from closely related Cuzco Quechua (quz), resulting in Glot500-m outperforming Glot+1.

7 Conclusion and Future Work

We collect and data-clean Glot500-c, a large corpus of hundreds of usually neglected tail (i.e., long-tail) languages and create Glot500-m, an LLM that is trained on Glot500-c and covers these languages. We evaluate Glot500-m on six tasks that allow us to evaluate almost all languages. We observe large improvements for both head and tail languages compared to XLM-R. Our analysis shows that no single factor fully explains the quality of the representation of a language in a multilingual model. Rather, a combination of factors is important, including corpus size, script, "help" from related languages and the total capacity of the model.

This work is the first to create a language model on a dataset of several hundreds of gigabytes and to make it publicly available for such a large and diverse number of low-resource languages. In future research, we would like to train larger models to further investigate the effect of model size, distill highly multilingual models for resource-efficient deployment, explore alternatives to continued pretraining and use models for more tail language downstream tasks.

Limitations

(1) We did not perform any comprehensive hyperparameter search, which would have further consolidated our results. This decision was made due to the high cost of training multiple models. (2) Compared to current very large models, Glot500-m

is comparatively small. (3) Although we have tried to minimize the amount of noise in our data, some noise is still present.

Ethics Statement

There are two issues worth mentioning in regards to this project. First, it was not feasible for us to thoroughly examine the content of the data for all languages, thus we cannot confirm the absence of discrimination based on factors such as race or sexuality. The data was solely utilized as a textual corpus, and the content should not be interpreted as an endorsement by our team. If the model is subsequently utilized for generation, it is possible that the training data may be reflected in the generated output. However, addressing potential biases within the data is an area for future research. Second, it is important to note that while the data sources utilized in this study do not explicitly prohibit the reuse of data for research purposes, some sources do have copyright statements indicating that such use is permissible while others do not. Additionally, certain sources prohibit the redistribution of data. As such, data from these sources is omitted from the published version of Glot2000-c.

Acknowledgements

We would like to thank Renhao Pei, Yihong Liu, Verena Blaschke, and the anonymous reviewers. This work was funded by the European Research Council (grants #740516 and #758969) and EU's Horizon Europe Research and Innovation Actions (UTTER, contract 101070631).

References

Solomon Teferra Abate, Michael Melese, Martha Yifiru Tachbelie, Million Meshesha, Solomon Atinafu, Wondwossen Mulugeta, Yaregal Assabie, Hafte Abera, Binyam Ephrem, Tewodros Abebe, Wondimagegnhue Tsegaye, Amanuel Lemma, Tsegaye Andargie, and Seifedin Shifaw. 2018. Parallel corpora for bi-lingual English-Ethiopian languages statistical machine translation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3102–3111, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Ahmed Abdelali, Hamdy Mubarak, Younes Samih, Sabit Hassan, and Kareem Darwish. 2021. QADI: Arabic dialect identification in the wild. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 1–10, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

Kathrein Abu Kwaik, Motaz Saad, Stergios Chatzikyriakidis, and Simon Dobnik. 2018. Shami: A corpus of Levantine Arabic dialects. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Ife Adebara, AbdelRahim Elmadany, Muhammad Abdul-Mageed, and Alcides Alcoba Inciarte. 2022. SERENGETI: Massively multilingual language models for Africa. *arXiv preprint arXiv:2212.10785*.

David Adelani, Jesujoba Alabi, Angela Fan, Julia Kreutzer, Xiaoyu Shen, Machel Reid, Dana Ruiter, Dietrich Klakow, Peter Nabende, Ernie Chang, Tajuddeen Gwadabe, Freshia Sackey, Bonaventure F. P. Dossou, Chris Emezue, Colin Leong, Michael Beukman, Shamsuddeen Muhammad, Guyo Jarso, Oreen Yousuf, Andre Niyongabo Rubungo, Gilles Hacheme, Eric Peter Wairagala, Muhammad Umair Nasir, Benjamin Ajibade, Tunde Ajayi, Yvonne Gitau, Jade Abbott, Mohamed Ahmed, Millicent Ochieng, Anuoluwapo Aremu, Perez Ogayo, Jonathan Mukiibi, Fatoumata Ouoba Kabore, Godson Kalipe, Derguene Mbaye, Allahsera Auguste Tapo, Victoire Memdjokam Koagne, Edwin Munkoh-Buabeng, Valencia Wagner, Idris Abdulmumin, Ayodele Awokoya, Happy Buzaaba, Blessing Sibanda, Andiswa Bukula, and Sam Manthalu. 2022. A few thousand translations go a long way! leveraging pre-trained models for African news translation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3053–3070, Seattle, United States. Association for Computational Linguistics.

David Adelani, Dana Ruiter, Jesujoba Alabi, Damilola Adebonojo, Adesina Ayeni, Mofe Adeyemi, Ayodele Esther Awokoya, and Cristina España-Bonet. 2021. The effect of domain and diacritics in Yoruba–English neural machine translation. In *Proceedings of Machine Translation Summit XVIII: Research Track*, pages 61–75, Virtual. Association for Machine Translation in the Americas.

Rodrigo Agerri, Xavier Gómez Guinovart, German Rigau, and Miguel Anxo Solla Portela. 2018. Developing new linguistic resources and tools for the Galician language. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Jesujoba O. Alabi, David Ifeoluwa Adelani, Marius Mosbach, and Dietrich Klakow. 2022. Adapting pre-trained language models to African languages via multilingual adaptive fine-tuning. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4336–4349, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Israa Alsarsour, Esraa Mohamed, Reem Suwaileh, and Tamer Elsayed. 2018. DART: A large dataset of dialectal Arabic tweets. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Antonios Anastasopoulos, Alessandro Cattelan, Zi-Yi Dou, Marcello Federico, Christian Federmann, Dmitriy Genzel, Franscisco Guzmán, Junjie Hu, Macduff Hughes, Philipp Koehn, Rosie Lazar, Will Lewis, Graham Neubig, Mengmeng Niu, Alp Öktem, Eric Paquin, Grace Tang, and Sylwia Tur. 2020. TICO-19: the translation initiative for COvid-19. In *Proceedings of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020*, Online. Association for Computational Linguistics.

Alan Ansell, Edoardo Ponti, Anna Korhonen, and Ivan Vulić. 2022. Composable sparse fine-tuning for cross-lingual transfer. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1778–1796, Dublin, Ireland. 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.

Niyati Bafna. 2022. Empirical models for an indic language continuum.

Marta Bañón, Pinzhen Chen, Barry Haddow, Kenneth Heafield, Hieu Hoang, Miquel Esplà-Gomis, Mikel L. Forcada, Amir Kamran, Faheem Kirefu, Philipp Koehn, Sergio Ortiz Rojas, Leopoldo Pla Sempere, Gema Ramírez-Sánchez, Elsa Sarrías, Marek Strelec, Brian Thompson, William Waites, Dion Wiggins, and Jaume Zaragoza. 2020. ParaCrawl: Web-scale acquisition of parallel corpora. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4555–4567, Online. Association for Computational Linguistics.

Marta Bañón, Miquel Esplà-Gomis, Mikel L. Forcada, Cristian García-Romero, Taja Kuzman, Nikola Ljubesic, Rik van Noord, Leopoldo Pla Sempere, Gema Ramírez-Sánchez, Peter Rupnik, Vít Suchomel, Antonio Toral, Tobias van der Werff, and Jaume Zaragoza. 2022. Macocu: Massive collection and curation of monolingual and bilingual data: focus on under-resourced languages. In *Proceedings of the 23rd Annual Conference of the European Association for Machine Translation, EAMT 2022, Ghent, Belgium, June 1-3, 2022*, pages 301–302. European Association for Machine Translation.

Ankur Bapna, Isaac Caswell, Julia Kreutzer, Orhan Firat, Daan van Esch, Aditya Siddhant, Mengmeng Niu, Pallavi Baljekar, Xavier Garcia, Wolfgang Macherey, et al. 2022. Building machine translation systems for the next thousand languages. *arXiv preprint arXiv:2205.03983*.

Workshop BigScience, :, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Biderman, Albert Webson, Pawan Sasanka Ammanamanchi, Thomas Wang, Benoît Sagot, Niklas Muennighoff, Albert Villanova del Moral, Olatunji Ruwase, Rachel Bawden, Stas Bekman, Angelina McMillan-Major, Iz Beltagy, Huu Nguyen, Lucile Saulnier, Samson Tan, Pedro Ortiz Suarez, Victor Sanh, Hugo Laurençon, Yacine Jernite, Julien Launay, Margaret Mitchell, Colin Raffel, Aaron Gokaslan, Adi Simhi, Aitor Soroa, Alham Fikri Aji, Amit Alfassy, Anna Rogers, Ariel Kreisberg Nitzav, Canwen Xu, Chenghao Mou, Chris Emezue, Christopher Klamm, Colin Leong, Daniel van Strien, David Ifeoluwa Adelani, Dragomir Radev, Eduardo González Ponferrada, Efrat Levkovizh, Ethan Kim, Eyal Bar Natan, Francesco De Toni, Gérard Dupont, Germán Kruszewski, Giada Pistilli, Hady Elsahar, Hamza Benyamina, Hieu Tran, Ian Yu, Idris Abdulmumin, Isaac Johnson, Itziar Gonzalez-Dios, Javier de la Rosa, Jenny Chim, Jesse Dodge, Jian Zhu, Jonathan Chang, Jörg Frohberg, Joseph Tobing, Joydeep Bhattacharjee, Khalid Almubarak, Kimbo Chen, Kyle Lo, Leandro Von Werra, Leon Weber, Long Phan, Loubna Ben allal, Ludovic Tanguy, Manan Dey, Manuel Romero Muñoz, Maraim Masoud, María Grandury, Mario Šaško, Max Huang, Maximin Coavoux, Mayank Singh, Mike Tian-Jian Jiang, Minh Chien Vu, Mohammad A. Jauhar, Mustafa Ghaleb, Nishant Subramani, Nora Kassner, Nurulaqilla Khamis, Olivier Nguyen, Omar Espejel, Ona de Gibert, Paulo Villegas, Peter Henderson, Pierre Colombo, Priscilla Amuok, Quentin Lhoest, Rheza Harliman, Rishi Bommasani, Roberto Luis López, Rui Ribeiro, Salomey Osei, Sampo Pyysalo, Sebastian Nagel, Shamik Bose, Shamsuddeen Hassan Muhammad, Shanya Sharma, Shayne Longpre, Somaieh Nikpoor, Stanislav Silberberg, Suhas Pai, Sydney Zink, Tiago Timponi Torrent, Timo Schick, Tristan Thrush, Valentin Danchev, Vassilina Nikoulina, Veronika Laippala, Violette Lepercq, Vrinda Prabhu, Zaid Alyafeai, Zeerak Talat, Arun Raja, Benjamin Heinzerling, Chenglei Si, Davut Emre Taşar, Elizabeth Salesky, Sabrina J. Mielke, Wilson Y. Lee, Abheesht Sharma, Andrea Santilli, Antoine Chaffin, Arnaud Stiegler, Debajyoti Datta, Eliza Szczechla, Gunjan Chhablani, Han Wang, Harshit Pandey, Hendrik Strobelt, Jason Alan Fries, Jos Rozen, Leo Gao, Lintang Sutawika, M Saiful Bari, Maged S. Al-shaibani, Matteo Manica, Nihal Nayak, Ryan Teehan, Samuel Albanie, Sheng Shen, Srulik Ben-David, Stephen H. Bach, Taewoon Kim, Tali Bers, Thibault Fevry, Trishala Neeraj, Urmish Thakker, Vikas Raunak, Xiangru Tang, Zheng-Xin Yong, Zhiqing Sun, Shaked Brody, Yallow Uri, Hadar Tojarieh, Adam Roberts, Hyung Won Chung, Jaesung Tae, Jason Phang, Ofir Press, Conglong Li, Deepak Narayanan, Hatim Bourfoune, Jared Casper, Jeff Rasley, Max Ryabinin, Mayank Mishra, Minjia Zhang, Mohammad Shoeybi, Myriam Peyrounette, Nicolas Patry, Nouamane Tazi, Omar Sanseviero, Patrick von

Platen, Pierre Cornette, Pierre François Lavallée, Rémi Lacroix, Samyam Rajbhandari, Sanchit Gandhi, Shaden Smith, Stéphane Requena, Suraj Patil, Tim Dettmers, Ahmed Baruwa, Amanpreet Singh, Anastasia Cheveleva, Anne-Laure Ligozat, Arjun Subramonian, Aurélie Névéol, Charles Lovering, Dan Garrette, Deepak Tunuguntla, Ehud Reiter, Ekaterina Taktasheva, Ekaterina Voloshina, Eli Bogdanov, Genta Indra Winata, Hailey Schoelkopf, Jan-Christoph Kalo, Jekaterina Novikova, Jessica Zosa Forde, Jordan Clive, Jungo Kasai, Ken Kawamura, Liam Hazan, Marine Carpuat, Miruna Clinciu, Najoung Kim, Newton Cheng, Oleg Serikov, Omer Antverg, Oskar van der Wal, Rui Zhang, Ruochen Zhang, Sebastian Gehrmann, Shachar Mirkin, Shani Pais, Tatiana Shavrina, Thomas Scialom, Tian Yun, Tomasz Limisiewicz, Verena Rieser, Vitaly Protasov, Vladislav Mikhailov, Yada Pruksachatkun, Yonatan Belinkov, Zachary Bamberger, Zdeněk Kasner, Alice Rueda, Amanda Pestana, Amir Feizpour, Ammar Khan, Amy Faranak, Ana Santos, Anthony Hevia, Antigona Unldreaj, Arash Aghagol, Arezoo Abdollahi, Aycha Tammour, Azadeh HajiHosseini, Bahareh Behroozi, Benjamin Ajibade, Bharat Saxena, Carlos Muñoz Ferrandis, Danish Contractor, David Lansky, Davis David, Douwe Kiela, Duong A. Nguyen, Edward Tan, Emi Baylor, Ezinwanne Ozoani, Fatima Mirza, Frankline Ononiwu, Habib Rezanejad, Hessie Jones, Indrani Bhattacharya, Irene Solaiman, Irina Sedenko, Isar Nejadgholi, Jesse Passmore, Josh Seltzer, Julio Bonis Sanz, Livia Dutra, Mairon Samagaio, Maraim Elbadri, Margot Mieskes, Marissa Gerchick, Martha Akinlolu, Michael McKenna, Mike Qiu, Muhammed Ghauri, Mykola Burynok, Nafis Abrar, Nazneen Rajani, Nour Elkott, Nour Fahmy, Olanrewaju Samuel, Ran An, Rasmus Kromann, Ryan Hao, Samira Alizadeh, Sarmad Shubber, Silas Wang, Sourav Roy, Sylvain Viguier, Thanh Le, Tobi Oyebade, Trieu Le, Yoyo Yang, Zach Nguyen, Abhinav Ramesh Kashyap, Alfredo Palasciano, Alison Callahan, Anima Shukla, Antonio Miranda-Escalada, Ayush Singh, Benjamin Beilharz, Bo Wang, Caio Brito, Chenxi Zhou, Chirag Jain, Chuxin Xu, Clémentine Fourrier, Daniel León Periñán, Daniel Molano, Dian Yu, Enrique Manjavacas, Fabio Barth, Florian Fuhrimann, Gabriel Altay, Giyaseddin Bayrak, Gully Burns, Helena U. Vrabec, Imane Bello, Ishani Dash, Jihyun Kang, John Giorgi, Jonas Golde, Jose David Posada, Karthik Rangasai Sivaraman, Lokesh Bulchandani, Lu Liu, Luisa Shinzato, Madeleine Hahn de Bykhovetz, Maiko Takeuchi, Marc Pàmies, Maria A Castillo, Marianna Nezhurina, Mario Sänger, Matthias Samwald, Michael Cullan, Michael Weinberg, Michiel De Wolf, Mina Mihaljcic, Minna Liu, Moritz Freidank, Myungsun Kang, Natasha Seelam, Nathan Dahlberg, Nicholas Michio Broad, Nikolaus Muellner, Pascale Fung, Patrick Haller, Ramya Chandrasekhar, Renata Eisenberg, Robert Martin, Rodrigo Canalli, Rosaline Su, Ruisi Su, Samuel Cahyawijaya, Samuele Garda, Shlok S Deshmukh, Shubhanshu Mishra, Sid Kiblawi, Simon Ott, Sinee Sang-aroonsiri, Srishti Kumar, Stefan Schweter, Sushil Bharati, Tanmay Laud, Théo Gigant, Tomoya Kainuma, Wojciech Kusa, Yanis Labrak, Yash Shailesh Bajaj, Yash Venkatraman, Yifan Xu, Yingxin Xu, Yu Xu, Zhe Tan, Zhongli Xie, Zifan Ye, Mathilde Bras, Younes Belkada, and Thomas Wolf. 2022. BLOOM: a 176b-parameter open-access multilingual language model.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

José Camacho-Collados, Claudio Delli Bovi, Alessandro Raganato, and Roberto Navigli. 2016. A large-scale multilingual disambiguation of glosses. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1701–1708, Portorož, Slovenia. European Language Resources Association (ELRA).

Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. 2021. InfoXLM: An information-theoretic framework for cross-lingual language model pre-training. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3576–3588, Online. Association for Computational Linguistics.

Zewen Chi, Shaohan Huang, Li Dong, Shuming Ma, Bo Zheng, Saksham Singhal, Payal Bajaj, Xia Song, Xian-Ling Mao, Heyan Huang, and Furu Wei. 2022. XLM-E: Cross-lingual language model pre-training via ELECTRA. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6170–6182, Dublin, Ireland. Association for Computational Linguistics.

Rochelle Choenni and Ekaterina Shutova. 2022. Investigating language relationships in multilingual sentence encoders through the lens of linguistic typology. *Computational Linguistics*, 48(3):635–672.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Hyung Won Chung, Dan Garrette, Kiat Chuan Tan, and Jason Riesa. 2020. Improving multilingual models with language-clustered vocabularies. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4536–4546, Online. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised

cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal dependencies. *Computational Linguistics*, 47(2):255–308.

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.

Philipp Dufter and Hinrich Schütze. 2020. Identifying elements essential for BERT's multilinguality. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4423–4437, Online. Association for Computational Linguistics.

Philipp Dufter, Mengjie Zhao, Martin Schmitt, Alexander Fraser, and Hinrich Schütze. 2018. Embedding learning through multilingual concept induction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1520–1530, Melbourne, Australia. Association for Computational Linguistics.

Jonathan Dunn. 2020. Mapping languages: the corpus of global language use. *Lang. Resour. Evaluation*, 54(4):999–1018.

Eberhard, David M., Gary F. Simons, and Charles D. Fennig (eds.). 2022. Ethnologue: Languages of the world. twenty-fifth edition.

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

Mahmoud El-Haj. 2020. Habibi - a multi dialect multi national Arabic song lyrics corpus. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1318–1326, Marseille, France. European Language Resources Association.

Mahmoud El-Haj, Paul Rayson, and Mariam Aboelezz. 2018. Arabic dialect identification in the context of bivalency and code-switching. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

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, Michael Auli, and Armand Joulin. 2021. Beyond english-centric multilingual machine translation. *J. Mach. Learn. Res.*, 22:107:1–107:48.

Pablo Gamallo, Jose Ramom Pichel, and Iñaki Alegria. 2017. A perplexity-based method for similar languages discrimination. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 109–114, Valencia, Spain. Association for Computational Linguistics.

Dirk Goldhahn, Thomas Eckart, and Uwe Quasthoff. 2012. Building large monolingual dictionaries at the leipzig corpora collection: From 100 to 200 languages. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation, LREC 2012, Istanbul, Turkey, May 23-25, 2012*, pages 759–765. European Language Resources Association (ELRA).

Santiago Góngora, Nicolás Giossa, and Luis Chiruzzo. 2021. Experiments on a Guarani corpus of news and social media. In *Proceedings of the First Workshop on Natural Language Processing for Indigenous Languages of the Americas*, pages 153–158, Online. Association for Computational Linguistics.

Santiago Góngora, Nicolás Giossa, and Luis Chiruzzo. 2022. Can we use word embeddings for enhancing Guarani-Spanish machine translation? In *Proceedings of the Fifth Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 127–132, Dublin, Ireland. Association for Computational Linguistics.

Thamme Gowda, Zhao Zhang, Chris Mattmann, and Jonathan May. 2021. Many-to-English machine translation tools, data, and pretrained models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 306–316, Online. Association for Computational Linguistics.

Tahmid Hasan, Abhik Bhattacharjee, Md. Saiful Islam, Kazi Samin Mubasshir, Yuan-Fang Li, Yong-Bin Kang, M. Sohel Rahman, and Rifat Shahriyar. 2021. Xl-sum: Large-scale multilingual abstractive summarization for 44 languages. In *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 4693–4703. Association for Computational Linguistics.

Kenneth Heafield. 2011. KenLM: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197, Edinburgh, Scotland. Association for Computational Linguistics.

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 generalisation. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 4411–4421. PMLR.

Ayyoob ImaniGooghari, Silvia Severini, Masoud Jalili Sabet, François Yvon, and Hinrich Schütze. 2022. Graph-based multilingual label propagation for low-resource part-of-speech tagging. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1577–1589, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. IndicNLPSuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4948–4961, Online. Association for Computational Linguistics.

Fajri Koto and Ikhwan Koto. 2020. Towards computational linguistics in Minangkabau language: Studies on sentiment analysis and machine translation. In *Proceedings of the 34th Pacific Asia Conference on Language, Information and Computation*, pages 138–148, Hanoi, Vietnam. Association for Computational Linguistics.

Julia Kreutzer, Isaac Caswell, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, Monang Setyawan, Supheakmungkol Sarin, Sokhar Samb, Benoît Sagot, Clara Rivera, Annette Rios, Isabel Papadimitriou, Salomey Osei, Pedro Ortiz Suarez, Iroro Orife, Kelechi Ogueji, Andre Niyongabo Rubungo, Toan Q. Nguyen, Mathias Müller, André Müller, Shamsuddeen Hassan Muhammad, Nanda Muhammad, Ayanda Mnyakeni, Jamshidbek Mirzakhalov, Tapiwanashe Matangira, Colin Leong, Nze Lawson, Sneha Kudugunta, Yacine Jernite, Mathias Jenny, Orhan Firat, Bonaventure F. P. Dossou, Sakhile Dlamini, Nisansa de Silva, Sakine Çabuk Ballı, Stella Biderman, Alessia Battisti, Ahmed Baruwa, Ankur Bapna, Pallavi Baljekar, Israel Abebe Azime, Ayodele Awokoya, Duygu Ataman, Orevaoghene Ahia, Oghenefego Ahia, Sweta Agrawal, and Mofetoluwa Adeyemi. 2022. Quality at a glance: An audit of web-crawled multilingual datasets. *Transactions of the Association for Computational Linguistics*, 10:50–72.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75, Melbourne, Australia. Association for Computational Linguistics.

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

Anoop Kunchukuttan, Pratik Mehta, and Pushpak Bhattacharyya. 2018. The IIT Bombay English-Hindi parallel corpus. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Hugo Laurençon, Lucile Saulnier, Thomas Wang, Christopher Akiki, Albert Villanova del Moral, Teven Le Scao, Leandro Von Werra, Chenghao Mou, Eduardo González Ponferrada, Huu Nguyen, et al. 2022. The BigScience ROOTS Corpus: A 1.6 TB Composite Multilingual Dataset. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

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.

Colin Leong, Joshua Nemecek, Jacob Mansdorfer, Anna Filighera, Abraham Owodunni, and Daniel Whitenack. 2022. Bloom library: Multimodal datasets in 300+ languages for a variety of downstream tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 8608–8621. Association for Computational Linguistics.

Chunlan Ma, Ayyoob ImaniGooghari, Haotian Ye, Ehsaneddin Asgari, and Hinrich Schütze. 2023. Taxi1500: A multilingual dataset for text classification in 1500 languages.

Martin Majliš. 2011. W2C – web to corpus – corpora. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Jamshidbek Mirzakhalov, Anoop Babu, Duygu Ataman, Sherzod Kariev, Francis Tyers, Otabek Abduraufov, Mammad Hajili, Sardana Ivanova, Abror Khaytbaev, Antonio Laverghetta Jr., Bekhzodbek Moydinboyev, Esra Onal, Shaxnoza Pulatova, Ahsan Wahab, Orhan Firat, and Sriram Chellappan. 2021. A large-scale study of machine translation in Turkic languages. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5876–5890, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Steven Moran, Christian Bentz, Ximena Gutierrez-Vasques, Olga Pelloni, and Tanja Samardzic. 2022. TeDDi sample: Text data diversity sample for language comparison and multilingual NLP. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 1150–1158, Marseille, France. European Language Resources Association.

Makoto Morishita, Jun Suzuki, and Masaaki Nagata. 2020. JParaCrawl: A large scale web-based English-Japanese parallel corpus. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3603–3609, Marseille, France. European Language Resources Association.

Toshiaki Nakazawa, Hideya Mino, Isao Goto, Raj Dabre, Shohei Higashiyama, Shantipriya Parida, Anoop Kunchukuttan, Makoto Morishita, Ondřej Bojar, Chenhui Chu, Akiko Eriguchi, Kaori Abe, Yusuke Oda, and Sadao Kurohashi. 2022. Overview of the 9th workshop on Asian translation. In *Proceedings of the 9th Workshop on Asian Translation*, pages 1–36, Gyeongju, Republic of Korea. International Conference on Computational Linguistics.

Toshiaki Nakazawa, Hideki Nakayama, Chenchen Ding, Raj Dabre, Shohei Higashiyama, Hideya Mino, Isao Goto, Win Pa Pa, Anoop Kunchukuttan, Shantipriya Parida, Ondřej Bojar, Chenhui Chu, Akiko Eriguchi, Kaori Abe, Yusuke Oda, and Sadao Kurohashi. 2021. Overview of the 8th workshop on Asian translation. In *Proceedings of the 8th Workshop on Asian Translation (WAT2021)*, pages 1–45, Online. Association for Computational Linguistics.

Graham Neubig. 2011. The Kyoto free translation task. http://www.phontron.com/kftt.

Kelechi Ogueji, Yuxin Zhu, and Jimmy Lin. 2021a. Small data? no problem! exploring the viability of pretrained multilingual language models for low-resourced languages. In *Proceedings of the 1st Workshop on Multilingual Representation Learning*, pages 116–126, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Kelechi Ogueji, Yuxin Zhu, and Jimmy Lin. 2021b. Small data? no problem! exploring the viability of pretrained multilingual language models for low-resourced languages. In *Proceedings of the 1st Workshop on Multilingual Representation Learning*, pages 116–126.

Chester Palen-Michel, June Kim, and Constantine Lignos. 2022. Multilingual open text release 1: Public domain news in 44 languages. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2080–2089, Marseille, France. European Language Resources Association.

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

Jonas Pfeiffer, Naman Goyal, Xi Lin, Xian Li, James Cross, Sebastian Riedel, and Mikel Artetxe. 2022. Lifting the curse of multilinguality by pre-training modular transformers. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3479–3495, Seattle, United States. Association for Computational Linguistics.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2021. UNKs everywhere: Adapting multilingual language models to new scripts. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10186–10203, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Roberts Rozis and Raivis Skadiņš. 2017. Tilde MODEL - multilingual open data for EU languages. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 263–265, Gothenburg, Sweden. Association for Computational Linguistics.

Hassan Sajjad, Ahmed Abdelali, Nadir Durrani, and Fahim Dalvi. 2020. AraBench: Benchmarking dialectal Arabic-English machine translation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5094–5107, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Julian Salazar, Davis Liang, Toan Q. Nguyen, and Katrin Kirchhoff. 2020. Masked language model scoring.

In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2699–2712, Online. Association for Computational Linguistics.

Holger Schwenk, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. 2021. WikiMatrix: Mining 135M parallel sentences in 1620 language pairs from Wikipedia. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1351–1361, Online. Association for Computational Linguistics.

Silvia Severini, Ayyoob Imani, Philipp Dufter, and Hinrich Schütze. 2022. Towards a broad coverage named entity resource: A data-efficient approach for many diverse languages. *arXiv preprint arXiv:2201.12219*.

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. *arXiv preprint arXiv:2201.03110*.

Anil Kumar Singh. 2008. Named entity recognition for south and south East Asian languages: Taking stock. In *Proceedings of the IJCNLP-08 Workshop on Named Entity Recognition for South and South East Asian Languages*.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous pipeline for processing huge corpora on medium to low resource infrastructures. In *7th Workshop on the Challenges in the Management of Large Corpora (CMLC-7)*. Leibniz-Institut für Deutsche Sprache.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in opus. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey. European Language Resources Association (ELRA).

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

Hai Wang, Dian Yu, Kai Sun, Jianshu Chen, and Dong Yu. 2019. Improving pre-trained multilingual model with vocabulary expansion. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 316–327, Hong Kong, China. Association for Computational Linguistics.

Mingyang Wang, Heike Adel, Lukas Lange, Jannik Strötgen, and Hinrich Schütze. 2023. NLNDE at semeval-2023 task 12: Adaptive pretraining and source language selection for low-resource multilingual sentiment analysis. *CoRR*, abs/2305.00090.

Xinyi Wang, Sebastian Ruder, and Graham Neubig. 2022. Expanding pretrained models to thousands more languages via lexicon-based adaptation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 863–877, Dublin, Ireland. Association for Computational Linguistics.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020a. Ccnet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of The 12th Language Resources and Evaluation Conference, LREC 2020, Marseille, France, May 11-16, 2020*, pages 4003–4012. European Language Resources Association.

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

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

Jian Yang, Shuming Ma, Dongdong Zhang, Shuangzhi Wu, Zhoujun Li, and Ming Zhou. 2020. Alternating language modeling for cross-lingual pre-training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9386–9393.

Rodolfo Zevallos, John Ortega, William Chen, Richard Castro, Núria Bel, Cesar Toshio, Renzo Venturas, Hilario Aradiel, and Nelsi Melgarejo. 2022. Introducing QuBERT: A large monolingual corpus and BERT model for Southern Quechua. In *Proceedings of the Third Workshop on Deep Learning for Low-Resource Natural Language Processing*, pages 1–13, Hybrid. Association for Computational Linguistics.

Mengjie Zhao, Tao Lin, Fei Mi, Martin Jaggi, and Hinrich Schütze. 2020. Masking as an efficient alternative to finetuning for pretrained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2226–2241, Online. Association for Computational Linguistics.

A N-grams LMs and Language Divergence

Perplexity and Language Divergence. Perplexity measures how well a model predicts a sample test data. Assuming a test data contains sequences of

characters $S = ch_1, ch_2, \cdots, ch_T$, perplexity ($\mathcal{PP}$) of S given an n-gram character level language model M is computed as follows:

$$\mathcal{PP}(S, M) = \sqrt[T]{\prod_{t=1}^{T} \frac{1}{\mathbb{P}\left(ch_t \mid ch_1^{t-1}\right)}} \quad (1)$$

where $\mathbb{P}\left(ch_t \mid ch_1^{t-1}\right)$ is computed as by dividing the observed frequency (C) of $ch_1^{t-1}ch_i$ by the observed frequency of ch_1^{t-1} in M training data:

$$\mathbb{P}\left(ch_t \mid ch_1^{t-1}\right) = \frac{C\left(ch_1^{t-1}ch_t\right)}{C\left(ch_1^{t-1}\right)} \quad (2)$$

Given the definition of perplexity, we can determine how well a trained language model on language L_1 predicts the test text of language L_2 and vice-versa. The divergence between two languages is computed with the maximum of the perplexity values in both directions. Two reasons lead to the use of max: first, a symmetrical divergence is required, and second, languages differ in their complexity, so one direction of computing perplexity may result in a much lower perplexity than another. Thus, comparing perplexity results becomes difficult. As an example, the Kuanua language (ksd_Latn) has short words and a simple structure, which results in 3−gram models getting lower perplexity on its text compared to other languages. The lower the perplexity the smaller the divergence between languages. The divergence ($\mathcal{D}$) between language L_i and L_j with trained language models of M_{L_z} and test texts of S_{L_z}, where L_z is the corresponding language, computed as follows:

$$\mathcal{D}_{L_i,L_j} = \max\left(\mathcal{PP}(S_{L_i}, M_{L_j}), \mathcal{PP}(S_{L_j}, M_{L_i})\right) \quad (3)$$

Runs and Data. The data used to train and test the character level n-gram models is the same data used for the training and testing of the Glot500-m. The training of the models was limited to 100,000 sentences' per language-script. We use KenLM library (Heafield, 2011) to build n-gram models. This library uses an interpolated modified Kneser-Ney smoothing for estimating the unseen n-grams. Our evaluation has been performed over 7 n-gram models ($3 \leq n \leq 9$).

Baseline and Evaluation. Language family trees were used as a baseline for evaluating the divergence measures of the proposed approach. We obtained language family tree data from Ethnologue online version (Eberhard et al., 2022). For each language, the family tree follows the general order from largest typological language family group to smallest. There is only one family tree for each language in the baseline data. Nodes in the family tree represent typological language family groups. Each node only has one parent, so if a node is common in the family tree of two languages, its parent is also common. We evaluate our perplexity method on the following binary classification task: Do the majority of a language L_z's k nearest neighbors belong to the same typological language family group as L_z? Assuming languages L_i and L_j, with the following family trees:

T_{L_i} : ❶ → ❷ → ❸ → ❹ → ❺ → ❻
T_{L_j} : ❶ → ❷ → ❼ → ❽

These 2 languages belong to the same typological family group with family tree levels of $l \in \{1, 2\}$, but not with family tree levels of $l = 3$ and higher.

Result. When it comes to language families, the majority of studies only refer to the largest typological language family group (level $l = 1$). Here, we also assess our methodology for other levels. The results of classification accuracy for 3−gram model, $k \in \{1, 3, 7, 13, 21\}$ and $l \in \{1, 2, 3, \max\}$ are shown in Table 10. In cases where the maximum level of a tree is less than the l parameter, the maximum level for that language is used. Languages without a family or no other family member in our data are excluded. We only report the 3−gram model results as it gets the best results in most configurations among other n-gram models. With increasing l, the accuracy decreases, since more languages fall outside the same typological family. As k increases, the accuracy decreases, because languages with faraway neighbors are being included but the number of languages in the language typological group family will remain the same. There are times when languages have a lot of loan words from other languages because of geological proximity or historical reasons (e.g, colonization), which makes them similar to the languages they borrowed words from in our method. However they are different when it comes to their typological families and our method fails in these cases. Aymara (Macrolanguage: aym_Latn) and Quechua (Macrolanguage: que_Latn), for example, had a great deal of contact and influence on each other, but they do not belong to the same typological group. As well, some of the typological families are not that large, which makes our results worse when k increases. This is

the case, for instance, of the Tarascan typological family which only has two members.

model	l	k	accuracy (%)
3-gram	1	1	84.45
3-gram	1	3	75.77
3-gram	1	7	69.08
3-gram	1	13	62.75
3-gram	1	21	55.33
3-gram	2	1	79.75
3-gram	2	3	67.63
3-gram	2	7	59.49
3-gram	2	13	51.36
3-gram	2	21	42.68
3-gram	3	1	75.05
3-gram	3	3	60.22
3-gram	3	7	49.55
3-gram	3	13	38.34
3-gram	3	21	29.84
3-gram	max	1	59.31
3-gram	max	3	36.89
3-gram	max	7	18.81
3-gram	max	13	6.87
3-gram	max	21	2.89

Table 10: Detecting the typological relatedness of language with n-gram divergence: (Eq. 3); l: level of typological language family group; k: number of nearest language neighbors.

B Languages

The list of languages used to train Glot500-m with the amount of available data for each language is available in Tables 11, 12 and 13.

On Macrolanguages The presence of language codes that are supersets of other language codes within datasets is not uncommon (Kreutzer et al., 2022). This issue becomes more prevalent in extensive collections. Within the ISO 639-3 standard, these languages are referred to as macrolanguages. When confronted with macrolanguages, if it is not feasible to ascertain the specific individual language contained within a dataset, the macrolanguage code is retained. Consequently, it is possible that in Glot2000-c and Glot500-c both the corpora for the macrolanguage and its individual languages have been included.

C List of data sources

The datasets and repositories used in this project involve: AI4Bharat,[5] AIFORTHAI-LotusCorpus,[6] Add (El-Haj et al., 2018), AfriBERTa (Ogueji et al., 2021b), AfroMAFT (Adelani et al., 2022; Xue et al., 2021), Anuvaad,[7] AraBench (Sajjad et al., 2020), AUTSHUMATO,[8] Bloom (Leong et al., 2022), CC100 (Conneau et al., 2020; Wenzek et al., 2020a), CCNet (Wenzek et al., 2020b), CMU_Haitian_Creole,[9] CORP.NCHLT,[10] Clarin,[11] DART (Alsarsour et al., 2018), Earthlings (Dunn, 2020), FFR,[12] Flores200 (Costa-jussà et al., 2022), GiossaMedia (Góngora et al., 2022, 2021), Glosses (Camacho-Collados et al., 2016), Habibi (El-Haj, 2020), HinDialect (Bafna, 2022), HornMT,[13] IITB (Kunchukuttan et al., 2018), IndicNLP (Nakazawa et al., 2021), Indiccorp (Kakwani et al., 2020), isiZulu,[14] JParaCrawl (Morishita et al., 2020), KinyaSMT,[15] LeipzigData (Goldhahn et al., 2012), Lindat,[16] Lingala_Song_Lyrics,[17] Lyrics,[18] MC4 (Raffel et al., 2020), MTData (Gowda et al., 2021), MaCoCu (Bañón et al., 2022), Makerere MT Corpus,[19] Masakhane community,[20] Mburisano_Covid,[21] Menyo20K (Adelani et al., 2021), Minangkabau corpora (Koto and Koto, 2020), MoT (Palen-Michel et al., 2022), NLLB_seed (Costa-jussà et al., 2022), Nart/abkhaz,[22] OPUS (Tiedemann, 2012), OSCAR (Suárez et al., 2019), ParaCrawl (Bañón et al., 2020), Parallel Corpora for Ethiopian Lan-

[5] https://ai4bharat.org/

[6] https://github.com/korakot/corpus/releases/download/v1.0/AIFORTHAI-LotusCorpus.zip

[7] https://github.com/project-anuvaad/anuvaad-parallel-corpus

[8] https://autshumato.sourceforge.net/

[9] http://www.speech.cs.cmu.edu/haitian/text/

[10] https://repo.sadilar.org/handle/20.500.12185/7

[11] https://www.clarin.si/

[12] https://github.com/bonaventuredossou/ffr-v1/tree/master/FFR-Dataset

[13] https://github.com/asmelashteka/HornMT

[14] https://zenodo.org/record/5035171

[15] https://github.com/pniyongabo/kinyarwandaSMT

[16] https://lindat.cz/faq-repository

[17] https://github.com/espoirMur/songs_lyrics_webscrap

[18] https://lyricstranslate.com/

[19] https://zenodo.org/record/5089560

[20] https://github.com/masakhane-io/masakhane-community

[21] https://repo.sadilar.org/handle/20.500.12185/536

[22] https://huggingface.co/datasets/Nart/abkhaz_text

Language-Script	|Sent|	Family	Head	Language-Script	|Sent|	Family	Head	Language-Script	|Sent|	Family	Head
hbs_Latn	63411156	indo1319		vec_Latn	514240	indo1319		swh_Latn	95776	atla1278	yes
mal_Mlym	48098273	drav1251	yes	jpn_Jpan	510722	japo1237	yes	alt_Cyrl	95148	turk1311	
aze_Latn	46300705		yes	lus_Latn	509250	sino1245		rmn_Grek	94533	indo1319	
guj_Gujr	45738685	indo1319	yes	crs_Latn	508755	indo1319		miq_Latn	94343	misu1242	
ben_Beng	43514870	indo1319	yes	kqn_Latn	507913	atla1278		kaa_Cyrl	88815	turk1311	
kan_Knda	41836495	drav1251	yes	ndo_Latn	496613	atla1278		kos_Latn	88603	aust1307	
tel_Telu	41580525	drav1251	yes	snd_Arab	488730	indo1319	yes	grn_Latn	87568		
mlt_Latn	40654838	afro1255		yue_Hani	484700	sino1245		lhu_Latn	87255	sino1245	
fra_Latn	39197581	indo1319	yes	tiv_Latn	483064	atla1278		lzh_Hani	86035	sino1245	
spa_Latn	37286756	indo1319	yes	kua_Latn	473535	atla1278		ajp_Arab	83297	afro1255	
eng_Latn	36122761	indo1319	yes	kwy_Latn	473274	atla1278		cmn_Hani	80745	sino1245	yes
fil_Latn	33493255	aust1307	yes	hin_Latn	466175	indo1319		gcf_Latn	80737	indo1319	
nob_Latn	32869205	indo1319		iku_Cans	465011			rmn_Cyrl	79925	indo1319	
rus_Cyrl	31787973	indo1319	yes	kal_Latn	462430	eski1264		kjh_Cyrl	79262	turk1311	
deu_Latn	31015993	indo1319	yes	tdt_Latn	459818	aust1307		rng_Latn	78177	atla1278	
tur_Latn	29184662	turk1311	yes	gsw_Latn	449240	indo1319		mgh_Latn	78117	atla1278	
pan_Guru	29052537	indo1319	yes	mfe_Latn	447435	indo1319		xmv_Latn	77896	aust1307	
mar_Deva	28748897	indo1319	yes	swc_Latn	446378	atla1278		ige_Latn	77114	atla1278	
por_Latn	27824391	indo1319	yes	mon_Latn	437950	mong1349		rmy_Latn	76991	indo1319	
nld_Latn	25061426	indo1319	yes	mos_Latn	437666	atla1278		srm_Latn	76884	indo1319	
ara_Arab	24524122		yes	kik_Latn	437228	atla1278		bak_Latn	76809	turk1311	
zho_Hani	24143786		yes	cnh_Latn	436667	sino1245		gur_Latn	76151	atla1278	
ita_Latn	23539857	indo1319	yes	gil_Latn	434529	aust1307		idu_Latn	75106	atla1278	
ind_Latn	23018106	aust1307	yes	pon_Latn	434522	aust1307		yom_Latn	74818	atla1278	
ell_Grek	22033282	indo1319	yes	umb_Latn	431589	atla1278		tdx_Latn	74430	aust1307	
bul_Cyrl	21823004	indo1319	yes	lvs_Latn	422952	indo1319		mzn_Arab	73719	indo1319	
swe_Latn	20725883	indo1319	yes	sco_Latn	411591	indo1319		cfm_Latn	70227	sino1245	
ces_Latn	20376340	indo1319	yes	ori_Orya	410827		yes	zpa_Latn	69237	otom1299	
isl_Latn	19547941	indo1319	yes	arg_Latn	410683	indo1319		kbd_Cyrl	67914	abkh1242	
pol_Latn	19339945	indo1319	yes	kur_Latn	407169	indo1319	yes	lao_Laoo	66966	taik1256	yes
ron_Latn	19190217	indo1319	yes	dhv_Latn	405711	aust1307		nap_Latn	65826	indo1319	
dan_Latn	19174573	indo1319	yes	luo_Latn	398974	nilo1247		qub_Latn	64973	quec1387	
hun_Latn	18800025	ural1272	yes	lun_Latn	395764	atla1278		oke_Latn	64508	atla1278	
tgk_Cyrl	18659517	indo1319		nzi_Latn	394247	atla1278		ote_Latn	64224	otom1299	
srp_Latn	18371769	indo1319	yes	gug_Latn	392227	tupi1275		bsb_Latn	63634	aust1307	
fas_Arab	18277593		yes	bar_Latn	387070	indo1319		ogo_Latn	61901	atla1278	
ceb_Latn	18149215	aust1307		bci_Latn	384059	atla1278		abn_Latn	61830	atla1278	
heb_Hebr	18128962	afro1255	yes	chk_Latn	380596	aust1307		ldi_Latn	61827	atla1278	
hrv_Latn	17882932	indo1319	yes	roh_Latn	377067	indo1319		ayr_Latn	61570	ayma1253	
glg_Latn	17852274	indo1319	yes	aym_Latn	373329	ayma1253		gom_Deva	61140	indo1319	
fin_Latn	16730388	ural1272	yes	yap_Latn	358929	aust1307		bba_Latn	61123	atla1278	
slv_Latn	15719210	indo1319	yes	ssw_Latn	356561	atla1278		aln_Latn	60989	indo1319	
vie_Latn	15697827	aust1305	yes	quz_Latn	354781	quec1387		leh_Latn	59944	atla1278	
mkd_Cyrl	14717004	indo1319	yes	sah_Cyrl	352697	turk1311		ban_Latn	59805	aust1307	
slk_Latn	14633631	indo1319	yes	tsn_Latn	350954	atla1278		ace_Latn	59333	aust1307	
nor_Latn	14576191	indo1319	yes	lmo_Latn	348135	indo1319		pes_Arab	57511	indo1319	yes
est_Latn	13600579		yes	ido_Latn	331239	arti1236		skg_Latn	57228	aust1307	
ltz_Latn	12997242	indo1319		abk_Cyrl	321578	abkh1242		ary_Arab	56933	afro1255	
eus_Latn	12775959		yes	zne_Latn	318871	atla1278		hus_Latn	56176	maya1287	
lit_Latn	12479626	indo1319	yes	quy_Latn	311040	quec1387		glv_Latn	55641	indo1319	
kaz_Cyrl	12378727	turk1311	yes	kam_Latn	310659	atla1278		fat_Latn	55609	atla1278	
lav_Latn	12143980	indo1319	yes	bbc_Latn	310420	aust1307		frr_Latn	55254	indo1319	
bos_Latn	11014744	indo1319	yes	vol_Latn	310399	arti1236		mwn_Latn	54805	atla1278	
epo_Latn	8737198	arti1236	yes	wal_Latn	309873	gong1255		mai_Deva	54687	indo1319	
cat_Latn	8648271	indo1319	yes	uig_Arab	307302	turk1311	yes	dua_Latn	53392	atla1278	
tha_Thai	7735209	taik1256	yes	vmw_Latn	306899	atla1278		dzo_Tibt	52732	sino1245	
ukr_Cyrl	7462046	indo1319	yes	kwn_Latn	305362	atla1278		ctd_Latn	52135	sino1245	
tgl_Latn	7411064	aust1307	yes	pam_Latn	303737	aust1307		nnb_Latn	52041	atla1278	
sin_Sinh	7293178	indo1319	yes	seh_Latn	300243	atla1278		sxn_Latn	51749	aust1307	
gle_Latn	7225513	indo1319	yes	tsc_Latn	298442	atla1278		mps_Latn	50645	tebe1251	
hin_Deva	7046700	indo1319	yes	nyk_Latn	297976	atla1278		mny_Latn	50581	atla1278	
kor_Hang	6468444	kore1284	yes	kmb_Latn	296269	atla1278		gkp_Latn	50549	mand1469	
ory_Orya	6266475	indo1319		zai_Latn	277632	otom1299		kat_Latn	50424	kart1248	
urd_Arab	6009594	indo1319	yes	gym_Latn	274512	chib1249		bjn_Latn	49068	aust1307	
swa_Latn	5989369		yes	bod_Tibt	273489	sino1245		acr_Latn	48886	maya1287	
sqi_Latn	5526836	indo1319	yes	nde_Latn	269931	atla1278		dtp_Latn	48468	aust1307	
bel_Cyrl	5319675	indo1319	yes	fon_Latn	268566	atla1278		lam_Latn	46853	atla1278	
afr_Latn	5157787	indo1319	yes	ber_Latn	264426			bik_Latn	46561		
nno_Latn	4899103	indo1319		nbl_Latn	259158	atla1278		poh_Latn	46454	maya1287	
tat_Cyrl	4708088	turk1311		kmr_Latn	256677	indo1319		phm_Latn	45862	atla1278	

Table 11: List of languages used to train Glot500-m (Part I).

Language-Script	|Sent|	Family	Head	Language-Script	|Sent|	Family	Head	Language-Script	|Sent|	Family	Head
ast_Latn	4683554	indo1319		guc_Latn	249044	araw1281		hrx_Latn	45716	indo1319	
mon_Cyrl	4616960	mong1349	yes	mam_Latn	248348	maya1287		quh_Latn	45566	quec1387	
hbs_Cyrl	4598073	indo1319		nia_Latn	247406	aust1307		hyw_Cyrl	45379	indo1319	
hau_Latn	4368483	afro1255	yes	nyn_Latn	241992	atla1278		rue_Cyrl	45369	indo1319	
sna_Latn	4019596	atla1278		cab_Latn	240101	araw1281		eml_Latn	44630	indo1319	
msa_Latn	3929084		yes	top_Latn	239232	toto1251		acm_Arab	44505	afro1255	
som_Latn	3916769	afro1255	yes	tog_Latn	231969	atla1278		tob_Latn	44473	guai1249	
srp_Cyrl	3864091	indo1319	yes	mco_Latn	231209	mixe1284		ach_Latn	43974	nilo1247	
mlg_Latn	3715802		yes	tzh_Latn	230706	maya1287		vep_Latn	43076	ural1272	
zul_Latn	3580113	atla1278		pms_Latn	227748	indo1319		npi_Deva	43072	indo1319	
arz_Arab	3488224	afro1255		wuu_Hani	224088	sino1245		tok_Latn	42820	arti1236	
nya_Latn	3409030	atla1278		plt_Latn	220413	aust1307		sgs_Latn	42467	indo1319	
tam_Taml	3388255	drav1251	yes	yid_Hebr	220214	indo1319	yes	lij_Latn	42447	indo1319	
hat_Latn	3226932	indo1319		ada_Latn	219427	atla1278		myv_Cyrl	42147	ural1272	
uzb_Latn	3223485	turk1311	yes	iba_Latn	213615	aust1307		tih_Latn	41873	aust1307	
sot_Latn	3205510	atla1278		kek_Latn	209932	maya1287		tat_Latn	41640	turk1311	
uzb_Cyrl	3029947	turk1311		koo_Latn	209375	atla1278		lfn_Latn	41632	arti1236	
cos_Latn	3015055	indo1319		sop_Latn	206501	atla1278		cgg_Latn	41196	atla1278	
als_Latn	2954874	indo1319		kac_Latn	205542	sino1245		ful_Latn	41188	atla1278	
amh_Ethi	2862985	afro1255	yes	qvi_Latn	205447	quec1387		gor_Latn	41174	aust1307	
sun_Latn	2586011	aust1307	yes	cak_Latn	204472	maya1287		ile_Latn	40984	arti1236	
war_Latn	2584810	aust1307		kbp_Latn	202877	atla1278		ium_Latn	40683	hmon1336	
div_Thaa	2418687	indo1319		ctu_Latn	201662	maya1287		teo_Latn	40203	nilo1247	
yor_Latn	2392359	atla1278		kri_Latn	201087	indo1319		kia_Latn	40035	atla1278	
fao_Latn	2365271	indo1319		mau_Latn	199134	otom1299		crh_Cyrl	39985	turk1311	
uzn_Cyrl	2293672	turk1311		scn_Latn	199068	indo1319		crh_Latn	39896	turk1311	
smo_Latn	2290439	aust1307		tyv_Cyrl	198649	turk1311		enm_Latn	39809	indo1319	
bak_Cyrl	2264196	turk1311		ina_Latn	197315	arti1236		sat_Olck	39614	aust1305	
ilo_Latn	2106531	aust1307		btx_Latn	193701	aust1307		mad_Latn	38993	aust1307	
tso_Latn	2100708	atla1278		nch_Latn	193129	utoa1244		cac_Latn	38812	maya1287	
mri_Latn	2046850	aust1307		ncj_Latn	192962	utoa1244		hnj_Latn	38611	hmon1336	
hmn_Latn	1903898			pau_Latn	190529	aust1307		ksh_Latn	38130	indo1319	
asm_Beng	1882353	indo1319	yes	toj_Latn	189651	maya1287		ikk_Latn	38071	atla1278	
hil_Latn	1798875	aust1307		pcm_Latn	187594	indo1319		sba_Latn	38040	cent2225	
nso_Latn	1619354	atla1278		dyu_Latn	186367	mand1469		zom_Latn	37013	sino1245	
ibo_Latn	1543820	atla1278		kss_Latn	185868	atla1278		bqc_Latn	36881	mand1469	
kin_Latn	1521612	atla1278		afb_Arab	183694	afro1255		bim_Latn	36835	atla1278	
hye_Armn	1463123	indo1319	yes	urh_Latn	182214	atla1278		mdy_Ethi	36370	gong1255	
oci_Latn	1449128	indo1319		quc_Latn	181559	maya1287		bts_Latn	36216	aust1307	
lin_Latn	1408460	atla1278		new_Deva	181427	sino1245		gya_Latn	35902	atla1278	
tpi_Latn	1401844	indo1319		yao_Latn	179965	atla1278		ajg_Latn	35631	atla1278	
twi_Latn	1400979	atla1278		ngl_Latn	178498	atla1278		agw_Latn	35585	aust1307	
kir_Cyrl	1397566	turk1311	yes	nyu_Latn	177483	atla1278		kom_Cyrl	35249	ural1272	
pap_Latn	1360138	indo1319		kab_Latn	176015	afro1255		knv_Latn	35196		
nep_Deva	1317291	indo1319	yes	tuk_Cyrl	175769	turk1311		giz_Latn	35040	afro1255	
azj_Latn	1315834	turk1311		xmf_Geor	174994	kart1248		hui_Latn	34926	nucl1709	
bcl_Latn	1284493	aust1307		ndc_Latn	174305	atla1278		kpg_Latn	34900	aust1307	
xho_Latn	1262364	atla1278	yes	san_Deva	165616	indo1319	yes	zea_Latn	34426	indo1319	
cym_Latn	1244783	indo1319	yes	nba_Latn	163485	atla1278		aoj_Latn	34349	nucl1708	
gaa_Latn	1222307	atla1278		bpy_Beng	162838	indo1319		csy_Latn	34126	sino1245	
ton_Latn	1216118	aust1307		ncx_Latn	162558	utoa1244		azb_Arab	33758	turk1311	yes
tah_Latn	1190747	aust1307		qug_Latn	162500	quec1387		csb_Latn	33743	indo1319	
lat_Latn	1179913	indo1319	yes	rmn_Latn	162069	indo1319		tpm_Latn	33517	atla1278	
srn_Latn	1172349	indo1319		cjk_Latn	160645	atla1278		quw_Latn	33449	quec1387	
ewe_Latn	1161605	atla1278		arb_Arab	159884	afro1255	yes	rmy_Cyrl	33351	indo1319	
bem_Latn	1111969	atla1278		kea_Latn	158047	indo1319		ixl_Latn	33289	maya1287	
efi_Latn	1082621	atla1278		mck_Latn	157521	atla1278		mbb_Latn	33240	aust1307	
bis_Latn	1070170	indo1319		arn_Latn	155882	arau1255		pfl_Latn	33148	indo1319	
orm_Latn	1067699		yes	pdt_Latn	155485	indo1319		pcd_Latn	32867	indo1319	
haw_Latn	1062491	aust1307		her_Latn	154827	atla1278		tlh_Latn	32863	arti1236	
hmo_Latn	1033636	pidg1258		gla_Latn	152563	indo1319	yes	suz_Deva	32811	sino1245	
kat_Geor	1004297	kart1248	yes	kmr_Cyrl	151728	indo1319		gcr_Latn	32676	indo1319	
pag_Latn	983637	aust1307		mwl_Latn	150054	indo1319		jbo_Latn	32619	arti1236	
loz_Latn	964418	atla1278		nav_Latn	147702	atha1245		tbz_Latn	32264	atla1278	
fry_Latn	957422	indo1319	yes	ksw_Mymr	147674	sino1245		bam_Latn	32150	mand1469	
mya_Mymr	945180	sino1245	yes	mxv_Latn	147591	otom1299		prk_Latn	32085	aust1305	
nds_Latn	944715	indo1319		hif_Latn	147261	indo1319		jam_Latn	32048	indo1319	
run_Latn	943828	atla1278		wol_Latn	146992	atla1278		twx_Latn	32028	atla1278	

Table 12: List of languages used to train Glot500-m (Part II).

Language-Script	|Sent|	Family	Head	Language-Script	|Sent|	Family	Head	Language-Script	|Sent|	Family	Head
pnb_Arab	899895	indo1319		sme_Latn	146803	ural1272		nmf_Latn	31997	sino1245	
rar_Latn	894515	aust1307		gom_Latn	143937	indo1319		caq_Latn	31903	aust1305	
fij_Latn	887134	aust1307		bum_Latn	141673	atla1278		rop_Latn	31889	indo1319	
wls_Latn	882167	aust1307		mgr_Latn	138953	atla1278		tca_Latn	31852	ticu1244	
ckb_Arab	874441	indo1319		ahk_Latn	135068	sino1245		yan_Latn	31775	misu1242	
ven_Latn	860249	atla1278		kur_Arab	134160	indo1319		xav_Latn	31765	nucl1710	
zsm_Latn	859947	aust1307	yes	bas_Latn	133436	atla1278		bih_Deva	31658		
chv_Cyrl	859863	turk1311		bin_Latn	133256	atla1278		cuk_Latn	31612	chib1249	
lua_Latn	854359	atla1278		tsz_Latn	133251	tara1323		kjb_Latn	31471	maya1287	
que_Latn	838486			sid_Latn	130406	afro1255		hne_Deva	31465	indo1319	
sag_Latn	771048	atla1278		diq_Latn	128908	indo1319		wbm_Latn	31394	aust1305	
guw_Latn	767918	atla1278		srd_Latn	127064			zlm_Latn	31345	aust1307	
bre_Latn	748954	indo1319	yes	tcf_Latn	126050	otom1299		tui_Latn	31161	atla1278	
toi_Latn	745385	atla1278		bzj_Latn	124958	indo1319		ifb_Latn	30980	aust1307	
pus_Arab	731992	indo1319	yes	udm_Cyrl	121705	ural1272		izz_Latn	30894	atla1278	
che_Cyrl	728201	nakh1245		cce_Latn	120636	atla1278		rug_Latn	30857	aust1307	
pis_Latn	714783	indo1319		meu_Latn	120273	aust1307		aka_Latn	30704	atla1278	
kon_Latn	685194			chw_Latn	119751	atla1278		pxm_Latn	30698	book1242	
oss_Cyrl	683517	indo1319		cbk_Latn	118789	indo1319		kmm_Latn	30671	sino1245	
hyw_Armn	679819	indo1319		ibg_Latn	118733	aust1307		mcn_Latn	30666	afro1255	
iso_Latn	658789	atla1278		bhw_Latn	117381	aust1307		ifa_Latn	30621	aust1307	
nan_Latn	656389	sino1245		ngu_Latn	116851	utoa1244		dln_Latn	30620	sino1245	
lub_Latn	654390	atla1278		nyy_Latn	115914	atla1278		ext_Latn	30605	indo1319	
lim_Latn	652078	indo1319		szl_Latn	112496	indo1319		ksd_Latn	30550	aust1307	
tuk_Latn	649411	turk1311		ish_Latn	111814	atla1278		mzh_Latn	30517	mata1289	
tir_Ethi	649117	afro1255		naq_Latn	109747	khoe1240		llb_Latn	30480	atla1278	
tgk_Latn	636541	indo1319		toh_Latn	107583	atla1278		hra_Latn	30472	sino1245	
yua_Latn	610052	maya1287		ttj_Latn	106925	atla1278		mwm_Latn	30432	cent2225	
min_Latn	609065	aust1307		nse_Latn	105189	atla1278		krc_Cyrl	30353	turk1311	
lue_Latn	599429	atla1278		hsb_Latn	104802	indo1319		tuc_Latn	30349	aust1307	
khm_Khmr	590429	aust1305	yes	ami_Latn	104559	aust1307		mrw_Latn	30304	aust1307	
tum_Latn	589857	atla1278		alz_Latn	104392	nilo1247		pls_Latn	30136	otom1299	
tll_Latn	586530	atla1278		apc_Arab	102392	afro1255		rap_Latn	30102	aust1307	
ekk_Latn	582595	ural1272		vls_Latn	101900	indo1319		fur_Latn	30052	indo1319	
lug_Latn	566948	atla1278		mhr_Cyrl	100474	ural1272		kaa_Latn	30031	turk1311	
niu_Latn	566715	aust1307		djk_Latn	99234	indo1319		prs_Arab	26823	indo1319	yes
tzo_Latn	540262	maya1287		wes_Latn	98492	indo1319		san_Latn	25742	indo1319	yes
mah_Latn	534614	aust1307		gkn_Latn	97041	atla1278		som_Arab	14199	afro1255	yes
tvl_Latn	521556	aust1307		grc_Grek	96986	indo1319		uig_Latn	9637	turk1311	yes
jav_Latn	516833	aust1307	yes	hbo_Hebr	96484	afro1255		hau_Arab	9593	afro1255	yes

Table 13: List of languages used to train Glot500-m (Part III).

guages (Abate et al., 2018), Phontron (Neubig, 2011), QADI (Abdelali et al., 2021), Quechua-IIC (Zevallos et al., 2022), SLI_GalWeb.1.0 (Agerri et al., 2018), Shami (Abu Kwaik et al., 2018), Stanford NLP,[23] StatMT,[24] TICO (Anastasopoulos et al., 2020), TIL (Mirzakhalov et al., 2021), Tatoeba,[25] TeDDi (Moran et al., 2022), Tilde (Rozis and Skadiņš, 2017), W2C (Majliš, 2011), WAT (Nakazawa et al., 2022), WikiMatrix (Schwenk et al., 2021), Wikipedia,[26] Workshop on NER for South and South East Asian Languages (Singh, 2008), XLSum (Hasan et al., 2021).

D Results for Each Task and Language

We report the detailed results for all tasks and languages in Table 14 (Sentence Retrieval Tatoeba), 15, 16 (Sentence Retrieval Bible), 17 (NER), and 18 (POS), 19, 20 (Text Classification), 21, 22 (Round Trip Alignment).

E Perplexity Results for all Languages

Perplexity number for all languages is presented in Table 23, Table 24, and Table 25.

[23] https://nlp.stanford.edu/
[24] https://statmt.org/
[25] https://tatoeba.org/en/
[26] https://huggingface.co/datasets/wikipedia

Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m
afr_Latn	71.9	76.5	**81.1**	heb_Hebr	76.3	**84.1**	76.0	pam_Latn	4.8	5.6	**11.0**
amh_Ethi	35.1	37.5	**44.6**	hin_Deva	73.8	**88.8**	85.6	pes_Arab	83.3	86.6	**87.6**
ara_Arab	59.2	**66.8**	64.2	hrv_Latn	79.6	85.6	**89.8**	pms_Latn	16.6	12.6	**54.5**
arz_Arab	32.5	47.8	**63.5**	hsb_Latn	21.5	23.0	**53.6**	pol_Latn	82.6	**89.6**	82.4
ast_Latn	59.8	59.8	**87.4**	hun_Latn	76.1	**81.8**	69.2	por_Latn	91.0	**92.1**	90.1
aze_Latn	62.6	78.3	**79.9**	hye_Armn	64.6	40.0	**83.2**	ron_Latn	86.0	**89.1**	82.8
bel_Cyrl	70.0	80.5	**81.4**	ido_Latn	25.7	28.8	**57.6**	rus_Cyrl	89.6	**91.6**	91.5
ben_Beng	54.1	68.2	**69.4**	ile_Latn	34.6	41.9	**75.6**	slk_Latn	73.2	**80.6**	75.9
bos_Latn	78.5	82.2	**92.4**	ina_Latn	62.7	66.2	**91.4**	slv_Latn	72.1	**78.0**	77.0
bre_Latn	10.3	10.9	**19.9**	ind_Latn	84.3	**90.2**	88.8	spa_Latn	85.5	**89.0**	88.9
bul_Cyrl	84.4	**88.3**	86.7	isl_Latn	78.7	**84.5**	84.0	sqi_Latn	72.2	81.4	**84.7**
cat_Latn	72.8	73.9	**78.7**	ita_Latn	81.3	84.7	**86.4**	srp_Latn	78.1	85.0	**90.0**
cbk_Latn	33.2	36.0	**49.4**	jpn_Jpan	74.4	**80.8**	72.6	swe_Latn	90.4	**92.4**	89.7
ceb_Latn	15.2	15.0	**41.3**	kab_Latn	3.7	3.0	**16.4**	swh_Latn	30.3	34.6	**44.1**
ces_Latn	71.1	**81.3**	75.1	kat_Geor	61.1	**79.1**	67.7	tam_Taml	46.9	42.3	**66.4**
cmn_Hani	79.5	84.8	**85.6**	kaz_Cyrl	60.3	69.9	**72.3**	tat_Cyrl	10.3	10.3	**70.3**
csb_Latn	21.3	20.2	**40.3**	khm_Khmr	41.1	45.0	**52.5**	tel_Telu	58.5	50.4	**67.9**
cym_Latn	45.7	45.7	**55.7**	kor_Hang	73.4	**84.3**	78.0	tgl_Latn	47.6	54.2	**77.1**
dan_Latn	91.9	**93.9**	91.5	kur_Latn	24.1	28.5	**54.1**	tha_Thai	56.8	39.4	**78.1**
deu_Latn	**95.9**	94.7	95.0	lat_Latn	33.6	**48.0**	42.8	tuk_Latn	16.3	14.8	**63.5**
dtp_Latn	5.6	4.7	**21.1**	lfn_Latn	32.5	35.9	**59.3**	tur_Latn	77.9	**85.4**	78.4
ell_Grek	76.2	**84.1**	80.2	lit_Latn	73.4	**76.8**	65.6	uig_Arab	38.8	58.3	**62.6**
epo_Latn	64.9	68.5	**74.3**	lvs_Latn	73.4	**78.9**	76.9	ukr_Cyrl	77.1	**88.3**	83.7
est_Latn	63.9	68.6	**69.1**	mal_Mlym	80.1	**84.4**	83.8	urd_Arab	54.4	34.3	**80.9**
eus_Latn	45.9	**54.4**	52.7	mar_Deva	63.5	**81.2**	77.9	uzb_Cyrl	25.2	32.2	**64.5**
fao_Latn	45.0	42.7	**82.4**	mhr_Cyrl	6.5	5.8	**34.9**	vie_Latn	85.4	**87.9**	87.0
fin_Latn	81.9	**85.8**	72.3	mkd_Cyrl	70.5	**83.9**	81.4	war_Latn	8.0	6.5	**26.2**
fra_Latn	85.7	85.8	**86.0**	mon_Cyrl	60.9	**77.3**	77.0	wuu_Hani	56.1	47.4	**79.7**
fry_Latn	60.1	62.4	**75.1**	nds_Latn	28.8	29.0	**77.1**	xho_Latn	28.9	31.7	**56.3**
gla_Latn	21.0	21.2	**41.9**	nld_Latn	90.3	**91.8**	**91.8**	yid_Hebr	37.3	51.8	**74.4**
gle_Latn	32.0	36.9	**50.8**	nno_Latn	70.7	77.8	**87.8**	yue_Hani	50.3	42.3	**76.3**
glg_Latn	72.6	75.8	**77.5**	nob_Latn	93.5	**96.5**	95.7	zsm_Latn	81.4	87.4	**91.8**
gsw_Latn	36.8	31.6	**69.2**	oci_Latn	22.9	23.2	**46.9**				

Table 14: Top10 accuracy of XLM-R-B, XLM-R-L, and Glot500-m on Sentence Retrieval Tatoeba.

Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m
ace_Latn	4.4	4.6	**53.4**	iba_Latn	14.4	13.6	**66.0**	pan_Guru	43.2	**59.4**	48.8
ach_Latn	4.4	3.2	**40.0**	ibo_Latn	5.0	3.0	**30.4**	pap_Latn	12.4	9.2	**72.4**
acr_Latn	2.6	3.4	**25.4**	ifa_Latn	4.4	4.4	**39.2**	pau_Latn	4.4	4.0	**29.8**
afr_Latn	76.8	**77.2**	69.4	ifb_Latn	4.8	3.6	**36.6**	pcm_Latn	13.6	10.4	**66.8**
agw_Latn	5.8	3.0	**36.0**	ikk_Latn	3.0	3.2	**50.6**	pdt_Latn	9.2	8.6	**68.6**
ahk_Latn	3.0	2.6	**3.2**	ilo_Latn	6.2	3.6	**55.0**	pes_Arab	69.4	72.2	**80.8**
aka_Latn	5.0	4.2	**57.0**	ind_Latn	**82.6**	80.4	72.2	pis_Latn	6.4	5.0	**57.2**
aln_Latn	67.8	**72.4**	67.6	isl_Latn	62.6	**73.6**	66.0	pls_Latn	5.0	4.0	**34.4**
als_Latn	51.4	48.0	**55.8**	ita_Latn	**75.4**	73.6	70.0	plt_Latn	26.6	28.0	**59.8**
alt_Cyrl	12.6	9.0	**50.8**	ium_Latn	3.2	3.0	**24.8**	poh_Latn	3.4	2.4	**15.2**
alz_Latn	4.6	3.8	**34.6**	ixl_Latn	4.0	3.0	**18.4**	pol_Latn	79.2	**79.8**	63.8
amh_Ethi	35.4	43.2	**52.8**	izz_Latn	2.8	2.8	**25.6**	pon_Latn	5.6	4.4	**21.6**
aoj_Latn	5.0	3.0	**20.4**	jam_Latn	6.6	4.4	**67.8**	por_Latn	**81.6**	79.8	76.6
arb_Arab	7.0	7.8	**14.6**	jav_Latn	25.4	33.2	**47.4**	prk_Latn	3.6	2.2	**49.8**
arn_Latn	4.8	4.0	**28.4**	jpn_Jpan	65.0	**71.8**	64.2	prs_Arab	79.4	78.6	**88.8**
ary_Arab	2.8	4.0	**15.2**	kaa_Cyrl	17.6	24.8	**73.8**	pxm_Latn	3.2	3.2	**24.0**
arz_Arab	5.4	4.8	**24.8**	kaa_Latn	9.2	9.8	**43.4**	qub_Latn	4.6	3.6	**43.4**
asm_Beng	26.2	40.6	**66.6**	kab_Latn	3.4	2.4	**20.6**	quc_Latn	3.6	2.8	**24.8**
ayr_Latn	4.8	4.8	**52.8**	kac_Latn	3.6	3.2	**26.4**	qug_Latn	4.8	3.6	**50.8**
azb_Arab	7.4	6.8	**72.4**	kal_Latn	3.4	3.6	**23.2**	quh_Latn	4.6	4.4	**56.2**
aze_Latn	71.0	**78.6**	73.0	kan_Knda	51.2	**67.6**	50.2	quw_Latn	6.2	4.6	**49.2**
bak_Cyrl	5.4	6.4	**65.2**	kat_Geor	54.2	**61.4**	51.4	quy_Latn	4.6	4.6	**61.4**
bam_Latn	3.4	3.6	**60.2**	kaz_Cyrl	61.4	**73.0**	56.8	quz_Latn	4.8	4.2	**68.0**
ban_Latn	9.0	9.8	**33.0**	kbp_Latn	2.6	2.6	**36.0**	qvi_Latn	4.4	3.4	**46.8**
bar_Latn	13.4	12.8	**40.8**	kek_Latn	5.0	3.4	**26.4**	rap_Latn	3.2	3.2	**25.6**
bba_Latn	3.8	3.4	**36.8**	khm_Khmr	28.4	42.6	**47.6**	rar_Latn	3.2	3.0	**26.6**
bbc_Latn	7.8	7.4	**57.2**	kia_Latn	4.0	5.6	**33.2**	rmy_Latn	6.8	5.8	**34.6**
bci_Latn	4.4	3.6	**13.2**	kik_Latn	3.2	2.8	**53.4**	ron_Latn	**72.2**	69.6	66.6
bcl_Latn	10.2	11.2	**79.8**	kin_Latn	5.0	5.0	**59.4**	rop_Latn	4.6	3.4	**46.0**
bel_Cyrl	67.2	**72.8**	55.8	kir_Cyrl	54.8	**70.2**	66.6	rug_Latn	3.6	3.4	**49.0**
bem_Latn	6.6	5.4	**58.2**	kjb_Latn	4.0	3.8	**29.6**	run_Latn	5.4	6.4	**54.6**
ben_Beng	46.4	52.8	**53.4**	kjh_Cyrl	11.0	7.8	**53.8**	rus_Cyrl	**75.8**	74.6	71.2
bhw_Latn	4.4	6.0	**47.8**	kmm_Latn	4.8	3.8	**42.6**	sag_Latn	6.0	4.4	**52.4**
bim_Latn	4.2	2.8	**52.2**	kmr_Cyrl	4.0	4.2	**42.4**	sah_Cyrl	6.2	4.6	**45.8**
bis_Latn	7.0	4.6	**48.6**	kmr_Latn	35.8	40.4	**63.0**	san_Deva	13.8	14.2	**27.2**
bod_Tibt	2.0	1.8	**33.2**	knv_Latn	2.8	2.2	**9.0**	san_Latn	4.6	3.8	**9.8**
bqc_Latn	3.4	3.0	**39.2**	kor_Hang	64.0	**71.6**	61.2	sba_Latn	2.8	2.8	**37.6**
bre_Latn	17.6	23.4	**32.8**	kpg_Latn	5.2	3.8	**51.8**	seh_Latn	6.4	4.8	**74.6**
bts_Latn	6.0	5.0	**56.4**	krc_Cyrl	9.2	10.2	**63.0**	sin_Sinh	44.8	**56.6**	45.0
btx_Latn	11.0	9.0	**59.6**	kri_Latn	2.8	2.8	**62.8**	slk_Latn	**75.2**	72.8	63.6
bul_Cyrl	**81.2**	78.0	76.4	ksd_Latn	7.0	5.4	**42.6**	slv_Latn	63.6	**64.6**	51.8
bum_Latn	4.8	3.6	**38.0**	kss_Latn	2.2	2.4	**6.0**	sme_Latn	6.8	6.2	**47.8**
bzj_Latn	7.8	4.0	**75.0**	ksw_Mymr	1.6	2.0	**31.8**	smo_Latn	4.4	3.4	**36.0**
cab_Latn	5.8	4.6	**17.4**	kua_Latn	4.8	5.4	**43.8**	sna_Latn	7.0	3.6	**43.0**
cac_Latn	3.6	3.0	**14.8**	lam_Latn	4.6	3.6	**27.4**	snd_Arab	52.2	64.6	**66.6**
cak_Latn	3.4	3.4	**21.4**	lao_Laoo	31.4	**52.8**	49.6	som_Latn	22.2	29.0	**33.0**
caq_Latn	3.2	4.4	**30.2**	lat_Latn	52.2	**57.8**	49.6	sop_Latn	5.2	4.2	**31.2**
cat_Latn	**86.6**	81.0	76.4	lav_Latn	74.2	**78.0**	58.8	sot_Latn	6.0	4.8	**52.2**
cbk_Latn	31.8	35.6	**54.6**	ldi_Latn	5.4	4.4	**25.2**	spa_Latn	**81.2**	78.8	80.0
cce_Latn	5.2	4.6	**51.8**	leh_Latn	5.6	4.0	**58.2**	sqi_Latn	58.2	58.2	**63.4**
ceb_Latn	14.2	12.6	**68.0**	lhu_Latn	2.0	2.0	**5.0**	srm_Latn	4.0	3.2	**32.4**
ces_Latn	75.2	**75.8**	58.0	lin_Latn	6.6	5.4	**65.4**	srn_Latn	6.8	5.2	**79.8**
cfm_Latn	4.6	4.0	**46.8**	lit_Latn	**74.4**	71.6	62.4	srp_Cyrl	83.0	**87.0**	81.2
che_Cyrl	3.4	3.4	**14.0**	loz_Latn	6.8	4.6	**49.2**	srp_Latn	85.0	**87.2**	81.2
chk_Latn	5.4	4.2	**41.2**	ltz_Latn	9.8	10.0	**73.8**	ssw_Latn	4.8	8.4	**47.0**
chv_Cyrl	4.6	4.2	**56.0**	lug_Latn	4.6	4.0	**49.4**	sun_Latn	22.4	25.4	**43.0**
ckb_Arab	4.0	4.8	**47.2**	luo_Latn	6.4	4.4	**40.8**	suz_Deva	3.6	3.4	**34.2**
cmn_Hani	39.2	40.8	**41.8**	lus_Latn	3.8	3.8	**54.4**	swe_Latn	**79.8**	**79.8**	78.0
cnh_Latn	4.8	4.2	**55.6**	lzh_Hani	25.0	31.4	**63.4**	swh_Latn	47.8	48.8	**66.4**
crh_Cyrl	8.8	11.2	**75.2**	mad_Latn	7.6	4.4	**44.4**	sxn_Latn	4.8	4.8	**25.8**
crs_Latn	7.4	5.2	**80.6**	mah_Latn	4.8	4.2	**35.6**	tam_Taml	42.8	**56.8**	52.0
csy_Latn	3.8	5.0	**50.0**	mai_Deva	6.4	9.6	**59.2**	tat_Cyrl	8.2	6.2	**67.2**
ctd_Latn	4.2	5.4	**59.4**	mal_Mlym	49.4	**62.6**	56.8	tbz_Latn	2.6	2.6	**28.0**
ctu_Latn	2.8	2.8	**21.6**	mam_Latn	3.8	3.2	**12.8**	tca_Latn	2.4	3.2	**15.4**
cuk_Latn	5.0	3.4	**22.2**	mar_Deva	66.2	69.0	**74.8**	tdt_Latn	6.2	5.0	**62.2**
cym_Latn	38.8	**46.0**	42.4	mau_Latn	2.4	2.4	**3.6**	tel_Telu	44.4	**57.2**	42.6
dan_Latn	71.6	**73.2**	63.2	mbb_Latn	3.0	3.4	**33.6**	teo_Latn	5.8	3.4	**26.0**
deu_Latn	78.8	**80.6**	66.6	mck_Latn	5.2	3.6	**57.4**	tgk_Cyrl	4.6	4.2	**71.2**
djk_Latn	4.6	4.0	**40.4**	mcn_Latn	6.0	4.2	**39.2**	tgl_Latn	61.0	60.6	**78.6**
dln_Latn	5.2	4.8	**66.4**	mco_Latn	2.6	2.6	**7.0**	tha_Thai	30.0	37.0	**45.4**

Table 15: Top10 accuracy of XLM-R-B, XLM-R-L, and Glot500-m on Sentence Retrieval Bible (Part I).

Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m
dtp_Latn	5.4	4.2	**24.2**	mdy_Ethi	2.8	2.4	**31.6**	tih_Latn	5.2	4.4	**51.6**
dyu_Latn	4.2	2.4	**50.2**	meu_Latn	5.6	4.4	**52.0**	tir_Ethi	7.4	6.2	**43.4**
dzo_Tibt	2.2	2.0	**36.4**	mfe_Latn	9.0	6.8	**78.6**	tlh_Latn	7.8	6.4	**72.4**
efi_Latn	4.4	4.2	**54.0**	mgh_Latn	5.2	3.4	**23.6**	tob_Latn	2.2	3.0	**16.8**
ell_Grek	52.6	**53.8**	48.6	mgr_Latn	4.0	4.4	**57.6**	toh_Latn	4.0	4.0	**47.2**
enm_Latn	39.8	39.2	**66.0**	mhr_Cyrl	6.6	5.4	**48.0**	toi_Latn	4.2	4.4	**47.4**
epo_Latn	**64.6**	59.8	56.2	min_Latn	9.4	6.2	**29.0**	toj_Latn	4.2	4.0	**15.6**
est_Latn	72.0	**75.6**	56.4	miq_Latn	4.4	4.4	**47.4**	ton_Latn	4.2	3.8	**22.4**
eus_Latn	26.2	**28.4**	23.0	mkd_Cyrl	**76.6**	72.6	74.8	top_Latn	3.4	3.6	**8.0**
ewe_Latn	4.6	3.0	**49.0**	mlg_Latn	29.0	28.4	**66.0**	tpi_Latn	5.8	4.4	**58.0**
fao_Latn	24.0	28.4	**73.4**	mlt_Latn	5.8	5.2	**50.4**	tpm_Latn	3.6	3.0	**39.6**
fas_Arab	78.2	80.4	**89.2**	mos_Latn	4.2	3.6	**42.8**	tsn_Latn	5.4	3.6	**41.8**
fij_Latn	3.8	3.0	**36.4**	mps_Latn	3.2	3.2	**21.6**	tso_Latn	5.6	5.0	**50.8**
fil_Latn	60.4	64.4	**72.0**	mri_Latn	4.2	3.8	**48.4**	tsz_Latn	5.6	3.2	**27.0**
fin_Latn	**75.6**	75.0	53.8	mrw_Latn	6.0	4.4	**52.2**	tuc_Latn	2.6	2.6	**31.4**
fon_Latn	2.6	2.0	**33.4**	msa_Latn	40.0	40.2	**40.6**	tui_Latn	3.6	3.2	**38.0**
fra_Latn	**88.6**	86.8	79.2	mwm_Latn	2.6	2.6	**35.8**	tuk_Cyrl	13.6	15.8	**65.0**
fry_Latn	27.8	27.4	**44.0**	mxv_Latn	3.0	3.4	**8.8**	tuk_Latn	9.6	9.6	**66.2**
gaa_Latn	3.8	3.4	**47.0**	mya_Mymr	20.2	27.8	**29.4**	tum_Latn	5.2	4.6	**66.2**
gil_Latn	5.6	3.6	**36.8**	myv_Cyrl	4.6	4.0	**35.0**	tur_Latn	74.4	**74.8**	63.2
giz_Latn	6.2	4.0	**41.0**	mzh_Latn	4.6	3.2	**36.2**	twi_Latn	3.8	3.0	**50.0**
gkn_Latn	4.0	3.4	**32.2**	nan_Latn	3.2	3.2	**13.6**	tyv_Cyrl	6.8	7.0	**46.6**
gkp_Latn	3.0	3.2	**20.4**	naq_Latn	3.0	2.2	**25.0**	tzh_Latn	6.0	5.2	**25.8**
gla_Latn	25.2	26.6	**43.0**	nav_Latn	2.4	2.8	**11.2**	tzo_Latn	3.8	3.8	**16.6**
gle_Latn	35.0	38.6	**40.0**	nbl_Latn	9.2	11.8	**53.8**	udm_Cyrl	6.0	5.0	**55.2**
glv_Latn	5.8	3.6	**47.4**	nch_Latn	4.4	3.0	**21.4**	uig_Arab	45.8	**63.6**	56.2
gom_Latn	6.0	4.6	**42.8**	ncj_Latn	4.6	3.0	**25.2**	uig_Latn	9.8	11.0	**62.8**
gor_Latn	3.8	3.0	**26.0**	ndc_Latn	5.2	4.6	**40.0**	ukr_Cyrl	**66.0**	63.4	57.0
grc_Grek	17.4	23.8	**54.8**	nde_Latn	13.0	15.2	**53.8**	urd_Arab	47.6	47.0	**65.0**
guc_Latn	3.4	2.6	**13.0**	ndo_Latn	5.2	4.0	**48.2**	uzb_Cyrl	6.2	7.4	**78.8**
gug_Latn	4.6	3.2	**36.0**	nds_Latn	9.6	8.4	**43.0**	uzb_Latn	54.8	60.8	**67.6**
guj_Gujr	53.8	71.2	**71.4**	nep_Deva	35.6	50.6	**58.6**	uzn_Cyrl	5.4	5.4	**87.0**
gur_Latn	3.8	2.8	**27.0**	ngu_Latn	4.6	3.4	**27.6**	ven_Latn	4.8	4.2	**47.2**
guw_Latn	4.0	3.4	**59.4**	nia_Latn	4.6	3.2	**29.4**	vie_Latn	**72.8**	71.0	57.8
gya_Latn	3.6	3.0	**41.0**	nld_Latn	**78.0**	75.8	71.8	wal_Latn	4.2	5.4	**51.4**
gym_Latn	3.6	3.8	**18.0**	nmf_Latn	4.6	4.6	**36.6**	war_Latn	9.8	6.6	**43.4**
hat_Latn	6.0	4.2	**68.2**	nnb_Latn	3.6	3.2	**42.0**	wbm_Latn	3.8	2.4	**46.4**
hau_Latn	28.8	36.0	**54.8**	nno_Latn	58.4	67.2	**72.6**	wol_Latn	4.6	4.4	**35.8**
haw_Latn	4.2	3.4	**38.8**	nob_Latn	82.8	**85.2**	79.2	xav_Latn	2.2	2.4	**5.0**
heb_Hebr	25.0	**26.0**	21.8	nor_Latn	81.2	84.2	**86.2**	xho_Latn	10.4	16.2	**40.8**
hif_Latn	12.2	16.4	**39.0**	npi_Deva	50.6	70.8	**76.6**	yan_Latn	4.2	3.4	**31.8**
hil_Latn	11.0	10.8	**76.2**	nse_Latn	5.2	5.0	**54.8**	yao_Latn	4.4	3.8	**55.2**
hin_Deva	67.0	72.8	**76.6**	nso_Latn	6.0	4.2	**57.0**	yap_Latn	4.0	4.0	**24.0**
hin_Latn	13.6	16.0	**43.2**	nya_Latn	4.0	4.6	**60.2**	yom_Latn	4.8	3.6	**42.2**
hmo_Latn	6.4	4.4	**48.2**	nyn_Latn	4.4	4.2	**51.8**	yor_Latn	3.4	3.6	**37.4**
hne_Deva	13.4	14.8	**75.0**	nyy_Latn	3.0	3.0	**25.6**	yua_Latn	3.8	3.4	**18.2**
hnj_Latn	2.8	2.8	**54.2**	nzi_Latn	3.2	3.0	**47.2**	yue_Hani	17.2	14.0	**24.0**
hra_Latn	5.2	4.6	**52.2**	ori_Orya	42.6	**62.0**	57.0	zai_Latn	6.2	4.2	**38.0**
hrv_Latn	79.8	**81.8**	72.6	ory_Orya	31.4	47.0	**55.2**	zho_Hani	40.4	40.2	**44.4**
hui_Latn	3.8	3.0	**28.0**	oss_Cyrl	4.2	3.6	**54.8**	zlm_Latn	83.4	78.4	**87.0**
hun_Latn	76.4	**78.2**	56.2	ote_Latn	3.6	2.4	**18.0**	zom_Latn	3.6	3.4	**50.2**
hus_Latn	3.6	3.2	**17.6**	pag_Latn	8.0	5.0	**61.2**	zsm_Latn	90.2	**91.0**	83.0
hye_Armn	30.8	33.0	**75.2**	pam_Latn	8.2	7.0	**49.8**	zul_Latn	11.0	16.0	**49.0**

Table 16: Top10 accuracy of XLM-R-B, XLM-R-L, and Glot500-m on Sentence Retrieval Bible (Part II).

Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m
ace_Latn	33.4	38.9	**44.2**	heb_Hebr	51.5	**56.5**	49.0	ori_Orya	**31.4**	27.6	31.0
afr_Latn	75.6	**78.3**	76.7	hin_Deva	67.0	**71.1**	69.4	oss_Cyrl	33.7	39.2	**52.1**
als_Latn	60.7	61.4	**80.0**	hrv_Latn	77.2	**78.9**	77.3	pan_Guru	50.0	**50.5**	48.1
amh_Ethi	42.2	40.9	**45.4**	hsb_Latn	64.0	69.0	**71.2**	pms_Latn	71.2	74.9	**75.9**
ara_Arab	44.7	48.7	**56.1**	hun_Latn	76.2	**79.8**	75.9	pnb_Arab	57.0	64.6	**65.8**
arg_Latn	73.6	74.6	**77.2**	hye_Armn	50.8	**61.7**	54.8	pol_Latn	77.5	**81.2**	78.1
arz_Arab	48.3	52.5	**57.4**	ibo_Latn	40.8	42.8	**58.6**	por_Latn	77.8	**81.2**	78.6
asm_Beng	53.2	**64.4**	64.2	ido_Latn	61.6	**78.6**	77.8	pus_Arab	37.4	39.9	**41.4**
ast_Latn	78.1	82.8	**84.5**	ilo_Latn	55.3	65.3	**77.1**	que_Latn	59.1	55.2	**66.8**
aym_Latn	40.8	38.7	**47.1**	ina_Latn	54.7	**63.4**	58.0	roh_Latn	52.6	55.7	**60.3**
aze_Latn	62.4	**69.2**	66.1	ind_Latn	49.0	54.1	**56.6**	ron_Latn	74.8	**79.9**	74.2
bak_Cyrl	35.1	49.3	**59.4**	isl_Latn	69.1	**77.2**	72.1	rus_Cyrl	63.8	**70.0**	67.6
bar_Latn	55.2	58.6	**68.4**	ita_Latn	77.3	**81.2**	78.7	sah_Cyrl	47.3	49.7	**74.2**
bel_Cyrl	74.2	**78.7**	74.3	jav_Latn	58.4	**61.2**	55.8	san_Deva	36.9	**37.3**	35.8
ben_Beng	65.3	**75.8**	71.6	jbo_Latn	18.0	26.3	**27.8**	scn_Latn	49.9	54.8	**65.8**
bih_Deva	50.7	57.1	**58.7**	jpn_Jpan	19.7	**20.6**	17.2	sco_Latn	80.9	81.8	**85.6**
bod_Tibt	2.5	3.0	**31.6**	kan_Knda	56.9	**60.8**	58.4	sgs_Latn	42.5	47.4	**62.7**
bos_Latn	74.0	**74.3**	74.2	kat_Geor	65.5	**69.5**	68.3	sin_Sinh	52.2	57.0	**57.8**
bre_Latn	59.1	**63.9**	63.3	kaz_Cyrl	43.7	**52.7**	50.0	slk_Latn	75.0	**81.7**	78.5
bul_Cyrl	76.8	**81.6**	77.2	khm_Khmr	43.3	**46.2**	40.6	slv_Latn	79.4	**82.2**	80.1
cat_Latn	82.2	**85.4**	83.7	kin_Latn	60.5	58.4	**67.1**	snd_Arab	41.2	**46.6**	41.8
cbk_Latn	**54.6**	54.0	54.1	kir_Cyrl	44.2	**46.9**	46.7	som_Latn	55.8	55.5	**58.2**
ceb_Latn	55.1	**57.8**	53.8	kor_Hang	49.1	**58.5**	50.9	spa_Latn	72.8	**73.3**	72.8
ces_Latn	77.6	**80.8**	78.3	ksh_Latn	41.3	48.3	**58.7**	sqi_Latn	74.0	74.4	**76.6**
che_Cyrl	15.4	24.6	**60.9**	kur_Latn	58.8	65.0	**69.6**	srp_Cyrl	59.7	**71.4**	66.4
chv_Cyrl	52.9	51.6	**75.9**	lat_Latn	70.7	**79.2**	73.8	sun_Latn	42.0	49.7	**57.7**
ckb_Arab	33.1	42.6	**75.5**	lav_Latn	73.4	**77.1**	74.0	swa_Latn	65.6	69.0	**69.6**
cos_Latn	54.3	**56.4**	56.0	lij_Latn	36.9	41.6	**46.6**	swe_Latn	71.8	**75.9**	69.7
crh_Latn	44.3	52.4	**54.7**	lim_Latn	59.9	64.7	**71.8**	szl_Latn	58.2	56.7	**67.6**
csb_Latn	55.1	54.2	**61.2**	lin_Latn	37.4	41.3	**54.0**	tam_Taml	55.0	**57.9**	55.2
cym_Latn	57.9	**60.1**	59.7	lit_Latn	73.4	**77.0**	73.5	tat_Cyrl	40.7	47.7	**68.0**
dan_Latn	81.5	**84.2**	81.7	lmo_Latn	68.8	68.4	**71.3**	tel_Telu	47.4	**52.5**	46.0
deu_Latn	74.3	**78.6**	75.7	ltz_Latn	47.4	55.8	**69.1**	tgk_Cyrl	24.7	38.3	**68.5**
diq_Latn	37.8	43.3	**53.1**	lzh_Hani	15.6	**21.6**	11.8	tgl_Latn	71.0	74.7	**75.1**
div_Thaa	0.0	0.0	**51.1**	mal_Mlym	61.0	**63.3**	61.3	tha_Thai	**4.2**	1.6	3.2
ell_Grek	73.7	**78.6**	72.8	mar_Deva	60.2	**63.4**	60.7	tuk_Latn	45.6	50.7	**59.7**
eml_Latn	32.9	36.1	**40.8**	mhr_Cyrl	44.3	48.3	**63.1**	tur_Latn	74.9	**79.3**	76.1
eng_Latn	82.7	**84.5**	83.3	min_Latn	42.9	**46.2**	41.8	uig_Arab	44.0	**50.9**	48.0
epo_Latn	63.8	**71.8**	68.0	mkd_Cyrl	74.5	**80.4**	73.3	ukr_Cyrl	75.2	**76.3**	74.2
est_Latn	72.2	**78.5**	73.5	mlg_Latn	54.9	54.3	**57.9**	urd_Arab	51.2	57.8	**74.5**
eus_Latn	59.0	**62.0**	58.0	mlt_Latn	43.2	48.3	**73.3**	uzb_Latn	70.6	**76.2**	75.1
ext_Latn	36.9	**47.1**	46.1	mon_Cyrl	72.4	**74.3**	66.9	vec_Latn	59.0	63.3	**66.4**
fao_Latn	61.1	70.8	**72.4**	mri_Latn	14.2	18.3	**53.5**	vep_Latn	59.8	59.3	**71.3**
fas_Arab	44.6	**58.0**	51.2	msa_Latn	62.3	**70.4**	65.8	vie_Latn	68.5	**77.8**	71.3
fin_Latn	75.5	**79.1**	75.2	mwl_Latn	42.6	**47.5**	45.3	vls_Latn	68.1	73.6	**73.7**
fra_Latn	77.2	**79.8**	76.0	mya_Mymr	51.3	53.4	**55.5**	vol_Latn	59.2	55.6	**59.2**
frr_Latn	45.4	46.8	**54.8**	mzn_Arab	36.4	43.1	**44.9**	war_Latn	61.9	61.4	**66.1**
fry_Latn	74.3	**79.0**	77.5	nan_Latn	46.2	51.4	**82.1**	wuu_Hani	29.4	**54.0**	25.1
fur_Latn	44.9	50.1	**56.4**	nap_Latn	53.0	53.9	**55.7**	xmf_Geor	40.2	40.0	**62.6**
gla_Latn	55.5	61.4	**63.5**	nds_Latn	62.4	66.7	**77.1**	yid_Hebr	47.6	**52.5**	50.3
gle_Latn	70.8	**74.6**	72.2	nep_Deva	63.2	**66.4**	62.7	yor_Latn	42.2	40.1	**63.1**
glg_Latn	80.2	**81.1**	79.4	nld_Latn	80.1	**83.6**	80.8	yue_Hani	24.8	**30.3**	22.6
grn_Latn	40.0	42.3	**54.7**	nno_Latn	76.6	**80.4**	78.0	zea_Latn	65.2	67.4	**68.6**
guj_Gujr	61.0	**61.9**	59.8	nor_Latn	76.5	**80.1**	76.7	zho_Hani	24.2	**28.8**	23.4
hbs_Latn	61.1	57.2	**61.5**	oci_Latn	65.3	67.8	**70.1**				

Table 17: F1 of XLM-R-B, XLM-R-L, and Glot500-m on NER.

Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m
afr_Latn	88.7	**89.3**	87.5	hbo_Hebr	38.9	45.7	**54.2**	pol_Latn	84.7	**85.4**	82.4
ajp_Arab	62.9	67.3	**69.7**	heb_Hebr	68.0	**69.2**	67.2	por_Latn	88.6	**89.8**	88.2
aln_Latn	53.5	**60.4**	52.3	hin_Deva	71.3	**75.3**	70.3	quc_Latn	28.9	29.3	**62.4**
amh_Ethi	64.5	**66.2**	66.1	hrv_Latn	85.9	**86.2**	85.5	ron_Latn	83.9	**85.7**	80.6
ara_Arab	68.5	**69.7**	65.4	hsb_Latn	71.5	74.4	**83.6**	rus_Cyrl	89.1	**89.7**	88.7
bam_Latn	25.4	23.5	**40.8**	hun_Latn	82.6	**82.7**	81.2	sah_Cyrl	20.3	22.8	**76.8**
bel_Cyrl	86.2	**86.2**	86.0	hye_Armn	85.2	**86.5**	84.0	san_Deva	18.3	**28.6**	26.1
ben_Beng	82.8	**83.8**	83.8	hyw_Armn	78.5	**82.5**	80.4	sin_Sinh	57.7	**60.1**	54.7
bre_Latn	61.6	**66.6**	60.7	ind_Latn	83.5	**84.1**	82.7	slk_Latn	85.6	**85.8**	84.4
bul_Cyrl	**89.1**	88.9	88.1	isl_Latn	84.2	**85.1**	82.8	slv_Latn	78.5	**79.1**	75.9
cat_Latn	86.7	**87.9**	86.3	ita_Latn	88.3	**89.6**	87.3	sme_Latn	29.8	31.5	**73.7**
ceb_Latn	49.3	49.5	**66.4**	jav_Latn	73.2	**76.7**	74.1	spa_Latn	88.5	**89.0**	88.0
ces_Latn	85.0	**85.4**	84.4	jpn_Jpan	17.3	**32.2**	31.7	sqi_Latn	81.4	**82.9**	77.9
cym_Latn	65.5	**67.0**	64.4	kaz_Cyrl	77.3	**79.1**	75.9	srp_Latn	86.1	**86.6**	85.3
dan_Latn	90.7	**91.0**	90.2	kmr_Latn	73.1	**78.2**	75.5	swe_Latn	93.5	**93.7**	92.1
deu_Latn	**88.4**	88.4	87.9	kor_Hang	**53.7**	53.4	53.1	tam_Taml	76.1	**76.9**	75.0
ell_Grek	**87.3**	87.0	85.4	lat_Latn	75.0	**80.3**	72.4	tat_Cyrl	45.0	48.8	**70.1**
eng_Latn	96.3	**96.5**	96.0	lav_Latn	86.0	**86.3**	83.5	tel_Telu	**85.0**	85.0	82.2
est_Latn	86.1	**86.4**	83.1	lij_Latn	48.1	48.6	**76.8**	tgl_Latn	72.7	**74.8**	74.7
eus_Latn	71.3	**73.7**	61.8	lit_Latn	84.1	**84.6**	81.1	tha_Thai	46.0	54.7	**56.7**
fao_Latn	77.0	80.6	**89.2**	lzh_Hani	14.1	**23.1**	23.0	tur_Latn	72.9	**74.0**	70.7
fas_Arab	71.8	**74.2**	71.5	mal_Mlym	**86.9**	86.7	84.4	uig_Arab	68.2	**70.2**	68.9
fin_Latn	85.2	**85.7**	80.8	mar_Deva	83.0	**85.2**	80.8	ukr_Cyrl	85.9	**86.3**	84.8
fra_Latn	86.7	**87.3**	85.4	mlt_Latn	21.0	21.9	**79.5**	urd_Arab	61.0	**68.2**	62.0
gla_Latn	57.4	**61.8**	60.2	myv_Cyrl	39.7	38.6	**65.7**	vie_Latn	70.9	**72.2**	67.1
gle_Latn	65.5	**68.7**	64.4	nap_Latn	52.8	17.0	**63.6**	wol_Latn	25.6	25.5	**61.6**
glg_Latn	83.7	**86.4**	82.6	nds_Latn	58.0	67.3	**77.2**	xav_Latn	8.4	5.3	**14.0**
glv_Latn	27.5	29.5	**52.7**	nld_Latn	88.5	**88.8**	88.2	yor_Latn	21.7	21.4	**63.9**
grc_Grek	62.0	68.1	**73.1**	nor_Latn	88.1	**88.9**	88.0	yue_Hani	31.5	**42.0**	40.9
grn_Latn	8.9	7.8	**19.8**	pcm_Latn	47.3	50.1	**57.1**	zho_Hani	28.6	42.4	**43.1**
gsw_Latn	48.7	55.9	**80.3**								

Table 18: F1 of XLM-R-B, XLM-R-L, and Glot500-m on POS.

Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m
ace_Latn	15	25	**60**	iba_Latn	30	35	**56**	ote_Latn	6	5	**36**
ace_Latn	15	25	**60**	iba_Latn	30	35	**56**	ote_Latn	6	5	**36**
ach_Latn	9	8	**34**	ibo_Latn	8	6	**51**	pag_Latn	22	21	**52**
acr_Latn	10	8	**46**	ifa_Latn	12	12	**47**	pam_Latn	20	18	**41**
afr_Latn	54	**64**	57	ifb_Latn	14	11	**48**	pan_Guru	53	**65**	59
agw_Latn	11	13	**54**	ikk_Latn	11	7	**47**	pap_Latn	31	36	**55**
ahk_Latn	5	5	**24**	ilo_Latn	15	13	**52**	pau_Latn	12	10	**41**
aka_Latn	11	7	**48**	ind_Latn	62	**66**	63	pcm_Latn	25	28	**46**
aln_Latn	44	**51**	49	isl_Latn	50	**60**	49	pdt_Latn	17	20	**53**
als_Latn	45	**51**	50	ita_Latn	57	**68**	61	pes_Arab	60	**70**	64
alt_Cyrl	25	23	**54**	ium_Latn	6	7	**53**	pis_Latn	13	13	**57**
alz_Latn	13	11	**34**	ixl_Latn	10	7	**33**	pls_Latn	6	7	**41**
amh_Ethi	42	**49**	43	izz_Latn	9	6	**41**	plt_Latn	30	**51**	50
aoj_Latn	12	9	**41**	jam_Latn	15	14	**55**	poh_Latn	16	8	**48**
arb_Arab	27	**55**	45	jav_Latn	44	**54**	49	pol_Latn	53	**63**	47
arn_Latn	9	8	**46**	jpn_Jpan	56	**66**	56	pon_Latn	10	8	**50**
ary_Arab	16	27	**40**	kaa_Cyrl	35	49	**59**	por_Latn	61	**67**	57
arz_Arab	28	**49**	39	kab_Latn	8	7	**30**	prk_Latn	6	6	**51**
asm_Beng	44	**53**	**53**	kac_Latn	7	8	**44**	prs_Arab	62	**67**	65
ayr_Latn	11	9	**53**	kal_Latn	9	7	**33**	pxm_Latn	9	9	**43**
azb_Arab	19	17	**55**	kan_Knda	53	**63**	59	qub_Latn	13	10	**55**
aze_Latn	56	**64**	61	kat_Geor	55	**60**	57	quc_Latn	9	7	**45**
bak_Cyrl	17	19	**57**	kaz_Cyrl	53	**64**	56	qug_Latn	13	8	**59**
bam_Latn	7	7	**46**	kbp_Latn	5	5	**35**	quh_Latn	11	10	**56**
ban_Latn	21	24	**46**	kek_Latn	6	9	**45**	quw_Latn	13	10	**48**
bar_Latn	31	42	**45**	khm_Khmr	51	**64**	59	quy_Latn	12	11	**57**
bba_Latn	6	6	**42**	kia_Latn	7	7	**39**	quz_Latn	11	8	**56**
bci_Latn	9	8	**28**	kik_Latn	7	6	**40**	qvi_Latn	9	8	**59**
bcl_Latn	28	27	**51**	kin_Latn	17	9	**50**	rap_Latn	8	7	**50**
bel_Cyrl	56	**67**	54	kir_Cyrl	55	**63**	60	rar_Latn	8	9	**48**
bem_Latn	13	14	**43**	kjb_Latn	7	9	**48**	rmy_Latn	16	12	**47**
ben_Beng	53	**65**	60	kjh_Cyrl	15	19	**50**	ron_Latn	60	**70**	60
bhw_Latn	11	11	**47**	kmm_Latn	8	6	**46**	rop_Latn	10	10	**50**
bim_Latn	7	7	**47**	kmr_Cyrl	8	8	**44**	rug_Latn	7	7	**55**
bis_Latn	13	12	**57**	knv_Latn	7	6	**44**	run_Latn	16	9	**49**
bqc_Latn	7	7	**36**	kor_Hang	59	**70**	60	rus_Cyrl	60	**66**	61
bre_Latn	30	**49**	36	kpg_Latn	9	10	**57**	sag_Latn	9	11	**42**
bts_Latn	18	17	**56**	krc_Cyrl	25	22	**56**	sah_Cyrl	10	9	**52**
btx_Latn	23	26	**53**	kri_Latn	7	9	**52**	sba_Latn	7	6	**41**
bul_Cyrl	61	**70**	57	ksd_Latn	10	11	**53**	seh_Latn	11	8	**47**
bum_Latn	9	9	**43**	kss_Latn	5	5	**23**	sin_Sinh	54	**66**	59
bzj_Latn	18	14	**56**	ksw_Mymr	5	5	**53**	slk_Latn	56	**63**	56
cab_Latn	9	8	**41**	kua_Latn	12	12	**45**	slv_Latn	59	**66**	61
cac_Latn	10	10	**47**	lam_Latn	5	8	**28**	sme_Latn	10	12	**43**
cak_Latn	7	8	**53**	lao_Laoo	56	**66**	64	smo_Latn	8	7	**51**
caq_Latn	7	7	**47**	lat_Latn	56	**64**	50	sna_Latn	13	11	**42**
cat_Latn	53	**64**	48	lav_Latn	54	**66**	55	snd_Arab	54	**64**	57
cbk_Latn	43	47	**57**	ldi_Latn	8	9	**28**	som_Latn	32	**45**	33
cce_Latn	13	9	**47**	leh_Latn	13	10	**44**	sop_Latn	12	8	**32**
ceb_Latn	28	30	**49**	lhu_Latn	6	6	**30**	sot_Latn	11	8	**45**
ces_Latn	50	**65**	53	lin_Latn	10	7	**49**	spa_Latn	61	**69**	60
cfm_Latn	8	8	**55**	lit_Latn	54	**66**	53	sqi_Latn	57	**68**	60
che_Cyrl	11	6	**20**	loz_Latn	10	10	**48**	srm_Latn	10	9	**53**
chv_Cyrl	8	7	**52**	ltz_Latn	22	30	**52**	srn_Latn	10	9	**53**
cmn_Hani	53	**62**	56	lug_Latn	16	9	**45**	srp_Latn	55	**67**	56
cnh_Latn	7	8	**56**	luo_Latn	12	10	**39**	ssw_Latn	14	17	**40**
crh_Cyrl	22	31	**57**	lus_Latn	11	7	**52**	sun_Latn	40	**47**	**47**
crs_Latn	14	17	**61**	lzh_Hani	46	**55**	**55**	suz_Deva	15	13	**53**
csy_Latn	9	7	**52**	mad_Latn	23	28	**56**	swe_Latn	60	**66**	56
ctd_Latn	9	8	**56**	mah_Latn	6	6	**42**	swh_Latn	47	**59**	56
ctu_Latn	15	14	**51**	mai_Deva	34	39	**59**	sxn_Latn	11	8	**46**
cuk_Latn	15	7	**44**	mal_Mlym	56	**64**	60	tam_Taml	56	**61**	60
cym_Latn	46	**51**	48	mam_Latn	10	6	**31**	tat_Cyrl	21	28	**64**
dan_Latn	51	**62**	50	mar_Deva	55	**63**	60	tbz_Latn	6	6	**43**
deu_Latn	56	**65**	53	mau_Latn	5	5	**6**	tca_Latn	5	5	**47**
djk_Latn	12	10	**46**	mbb_Latn	11	7	**48**	tdt_Latn	16	13	**56**
dln_Latn	10	5	**52**	mck_Latn	15	10	**41**	tel_Telu	55	**65**	60
dtp_Latn	9	8	**39**	mcn_Latn	13	9	**43**	teo_Latn	12	8	**26**
dyu_Latn	6	8	**52**	mco_Latn	6	7	**28**	tgk_Cyrl	10	7	**55**
dzo_Tibt	6	5	**55**	mdy_Ethi	6	7	**47**	tgl_Latn	48	**60**	56

Table 19: F1 of XLM-R-B, XLM-R-L, and Glot500-m on Text Classification (Part I).

Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m
efi_Latn	10	9	**50**	meu_Latn	15	11	**52**	tha_Thai	56	**67**	61
ell_Grek	37	47	**54**	mfe_Latn	16	14	**61**	tih_Latn	11	11	**56**
eng_Latn	74	**75**	68	mgh_Latn	10	6	**35**	tir_Ethi	23	27	**48**
enm_Latn	46	56	**65**	mgr_Latn	14	12	**46**	tlh_Latn	30	26	**59**
epo_Latn	53	**63**	53	mhr_Cyrl	14	10	**43**	tob_Latn	6	9	**52**
est_Latn	62	**68**	53	min_Latn	27	37	**50**	toh_Latn	11	8	**41**
eus_Latn	28	**33**	22	miq_Latn	7	7	**48**	toi_Latn	14	10	**40**
ewe_Latn	9	9	**52**	mkd_Cyrl	65	**69**	61	toj_Latn	12	11	**42**
fao_Latn	33	41	**55**	mlg_Latn	32	**51**	48	ton_Latn	6	7	**47**
fas_Arab	62	**68**	62	mlt_Latn	12	11	**49**	top_Latn	11	10	**25**
fij_Latn	8	7	**51**	mos_Latn	7	8	**41**	tpi_Latn	11	13	**55**
fil_Latn	47	**56**	53	mps_Latn	11	12	**54**	tpm_Latn	9	8	**47**
fin_Latn	57	**66**	56	mri_Latn	9	8	**47**	tsn_Latn	11	8	**45**
fon_Latn	5	6	**49**	mrw_Latn	15	18	**41**	tsz_Latn	10	10	**45**
fra_Latn	57	**66**	57	msa_Latn	43	**49**	46	tuc_Latn	7	9	**50**
fry_Latn	31	34	**37**	mwm_Latn	5	6	**50**	tui_Latn	8	8	**49**
gaa_Latn	5	6	**43**	mxv_Latn	8	8	**24**	tuk_Latn	23	26	**53**
gil_Latn	9	8	**44**	mya_Mymr	45	52	**54**	tum_Latn	12	12	**49**
giz_Latn	9	10	**49**	myv_Cyrl	11	7	**47**	tur_Latn	55	**66**	56
gkn_Latn	8	7	**40**	mzh_Latn	7	9	**45**	twi_Latn	9	6	**46**
gkp_Latn	5	6	**35**	nan_Latn	6	6	**30**	tyv_Cyrl	19	18	**54**
gla_Latn	28	**43**	42	naq_Latn	8	7	**42**	tzh_Latn	12	13	**42**
gle_Latn	37	**53**	40	nav_Latn	7	9	**25**	tzo_Latn	13	11	**41**
glv_Latn	10	12	**38**	nbl_Latn	20	26	**46**	udm_Cyrl	10	11	**51**
gom_Latn	10	13	**39**	nch_Latn	10	8	**39**	ukr_Cyrl	61	**67**	56
gor_Latn	17	15	**50**	ncj_Latn	7	9	**43**	urd_Arab	59	**65**	59
guc_Latn	8	6	**42**	ndc_Latn	13	13	**40**	uzb_Latn	49	**59**	56
gug_Latn	11	7	**44**	nde_Latn	20	26	**46**	uzn_Cyrl	13	17	**57**
guj_Gujr	57	**67**	63	ndo_Latn	13	9	**40**	ven_Latn	10	8	**43**
gur_Latn	6	6	**47**	nds_Latn	16	15	**42**	vie_Latn	57	**65**	55
guw_Latn	11	9	**49**	nep_Deva	56	**61**	**61**	wal_Latn	15	9	**41**
gya_Latn	5	5	**39**	ngu_Latn	8	10	**50**	war_Latn	19	21	**41**
gym_Latn	10	7	**47**	nia_Latn	11	9	**47**	wbm_Latn	7	6	**52**
hat_Latn	11	10	**59**	nld_Latn	50	**59**	55	wol_Latn	11	9	**40**
hau_Latn	34	40	**47**	nmf_Latn	9	7	**36**	xav_Latn	10	10	**40**
haw_Latn	8	7	**41**	nnb_Latn	11	8	**46**	xho_Latn	23	32	**48**
heb_Hebr	16	31	**41**	nno_Latn	49	56	**57**	yan_Latn	7	7	**46**
hif_Latn	22	37	**42**	nob_Latn	54	**60**	55	yao_Latn	10	8	**43**
hil_Latn	26	31	**60**	nor_Latn	53	**63**	55	yap_Latn	8	8	**46**
hin_Deva	54	**70**	57	npi_Deva	53	**62**	61	yom_Latn	13	9	**35**
hmo_Latn	14	13	**53**	nse_Latn	17	10	**45**	yor_Latn	11	7	**51**
hne_Deva	32	40	**59**	nso_Latn	11	7	**48**	yua_Latn	12	10	**39**
hnj_Latn	8	7	**55**	nya_Latn	12	10	**56**	yue_Hani	52	**61**	54
hra_Latn	10	7	**49**	nyn_Latn	16	7	**38**	zai_Latn	16	14	**40**
hrv_Latn	56	**63**	56	nyy_Latn	8	8	**34**	zho_Hani	55	**68**	55
hui_Latn	9	7	**43**	nzi_Latn	5	7	**40**	zlm_Latn	59	**70**	64
hun_Latn	62	**69**	53	ori_Orya	54	**65**	60	zom_Latn	11	9	**50**
hus_Latn	7	10	**39**	ory_Orya	55	**64**	61	zsm_Latn	61	**64**	63
hye_Armn	60	**68**	60	oss_Cyrl	6	6	**47**	zul_Latn	24	35	**52**

Table 20: F1 of XLM-R-B, XLM-R-L, and Glot500-m on Text Classification (Part II).

Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m
ace_Latn	2.50	2.83	**4.56**	hye_Armn	2.32	3.25	**4.91**	pam_Latn	2.85	3.52	**4.46**
ach_Latn	3.13	4.02	**5.60**	hye_Latn	2.34	**2.98**	2.44	pan_Guru	2.11	2.73	**4.11**
acr_Latn	2.01	2.46	**2.51**	iba_Latn	2.77	3.85	**6.01**	pap_Latn	3.12	3.85	**5.46**
afr_Latn	3.17	3.66	**5.46**	ibo_Latn	2.05	2.43	**4.33**	pau_Latn	2.67	3.09	**4.09**
agw_Latn	2.51	2.80	**4.09**	ifa_Latn	1.81	2.40	**3.45**	pcm_Latn	3.81	4.44	**6.47**
ahk_Latn	1.11	**1.23**	1.22	ifb_Latn	2.22	2.58	**3.28**	pdt_Latn	2.41	3.33	**5.11**
aka_Latn	3.38	4.50	**6.48**	ikk_Latn	1.75	2.29	**3.83**	pes_Arab	2.66	3.91	**4.81**
aln_Latn	4.06	4.92	**7.39**	ilo_Latn	3.06	3.87	**6.24**	pis_Latn	1.91	2.32	**4.42**
als_Latn	3.92	4.85	**6.32**	ind_Latn	4.06	5.00	**7.60**	pls_Latn	2.14	2.57	**4.02**
alt_Cyrl	2.91	3.36	**5.32**	isl_Latn	4.40	5.22	**7.07**	plt_Latn	3.74	3.99	**6.82**
alz_Latn	3.78	4.89	**5.94**	ita_Latn	3.55	4.02	**6.18**	poh_Latn	0.92	1.10	**1.87**
amh_Ethi	3.04	3.10	**4.87**	ium_Latn	2.00	2.27	**3.46**	pol_Latn	3.94	**5.20**	5.12
amh_Latn	1.41	**1.76**	1.70	ixl_Latn	1.62	1.94	**2.14**	pon_Latn	3.53	4.51	**5.18**
aoj_Latn	1.77	1.97	**3.22**	izz_Latn	1.65	2.06	**3.12**	por_Latn	3.61	4.35	**6.12**
arb_Arab	1.07	1.47	**2.40**	jam_Latn	2.77	3.06	**3.59**	prk_Latn	2.10	2.70	**5.40**
arn_Latn	2.40	2.79	**4.51**	jav_Latn	3.10	3.67	**5.21**	prs_Arab	3.54	4.28	**6.92**
ary_Arab	0.86	1.10	**2.43**	jpn_Jpan	3.62	**4.39**	4.07	pxm_Latn	1.76	2.15	**3.40**
arz_Arab	0.83	1.14	**2.52**	kaa_Cyrl	2.99	3.91	**5.45**	qub_Latn	2.48	2.97	**4.24**
asm_Beng	2.82	2.47	**5.21**	kaa_Latn	2.34	2.96	**3.64**	quc_Latn	1.87	2.45	**2.77**
ayr_Latn	2.61	3.09	**3.93**	kab_Latn	2.51	3.08	**3.14**	qug_Latn	2.44	2.99	**5.34**
azb_Arab	2.57	3.16	**4.96**	kac_Latn	1.66	2.17	**3.34**	quh_Latn	2.91	3.46	**5.43**
aze_Cyrl	2.76	3.26	**3.62**	kal_Latn	3.00	3.90	**4.73**	quw_Latn	2.89	3.50	**5.62**
aze_Latn	4.24	5.04	**8.00**	kan_Knda	2.58	3.18	**4.05**	quy_Latn	2.69	3.15	**5.51**
bak_Cyrl	2.20	2.38	**4.35**	kan_Latn	1.62	**2.08**	1.81	quz_Latn	3.33	3.89	**6.07**
bam_Latn	3.56	4.29	**5.73**	kat_Geor	4.06	4.99	**5.53**	qvi_Latn	2.82	3.42	**4.89**
ban_Latn	2.26	2.74	**3.37**	kaz_Cyrl	3.82	4.56	**5.31**	rap_Latn	1.31	1.61	**2.31**
bar_Latn	3.11	3.81	**3.84**	kbp_Latn	1.47	1.65	**3.32**	rar_Latn	1.83	2.22	**3.27**
bba_Latn	2.43	2.80	**4.16**	kek_Latn	1.91	2.45	**2.70**	rmy_Latn	2.85	3.68	**4.83**
bbc_Latn	3.02	3.85	**5.22**	khm_Khmr	1.57	1.70	**2.82**	ron_Latn	3.33	4.00	**4.99**
bci_Latn	2.81	3.18	**3.30**	kia_Latn	2.92	3.27	**4.69**	rop_Latn	1.60	2.08	**3.46**
bcl_Latn	3.78	4.61	**8.06**	kik_Latn	2.28	2.73	**4.38**	rug_Latn	2.56	2.95	**3.60**
bel_Cyrl	3.73	4.91	**6.46**	kin_Latn	2.67	3.26	**4.19**	run_Latn	3.33	3.98	**6.82**
bem_Latn	3.06	3.77	**5.69**	kir_Cyrl	4.54	4.35	**6.36**	rus_Cyrl	4.20	5.05	**7.38**
ben_Beng	3.29	3.07	**4.99**	kjb_Latn	2.42	3.03	**3.27**	sag_Latn	2.92	3.52	**5.17**
bhw_Latn	2.91	3.47	**5.16**	kjh_Cyrl	3.13	3.81	**5.39**	sah_Cyrl	2.31	3.01	**4.98**
bim_Latn	2.54	3.29	**4.12**	kmm_Latn	2.52	3.30	**3.73**	san_Deva	2.48	2.20	**3.64**
bis_Latn	2.59	2.96	**4.68**	kmr_Cyrl	2.31	2.76	**4.30**	san_Latn	1.54	2.23	**2.35**
bod_Tibt	0.54	**3.39**	2.43	kmr_Latn	3.75	4.19	**5.70**	sba_Latn	1.88	2.24	**3.86**
bqc_Latn	2.44	3.16	**4.61**	knv_Latn	1.27	1.53	**2.09**	seh_Latn	3.44	4.20	**4.94**
bre_Latn	3.32	**3.87**	3.79	kor_Hang	2.76	3.99	**4.89**	sin_Sinh	2.55	**3.60**	3.44
bts_Latn	4.06	4.92	**7.99**	kor_Latn	0.92	**2.40**	0.90	slk_Latn	4.65	5.06	**6.43**
btx_Latn	3.23	3.88	**5.59**	kpg_Latn	2.80	3.12	**5.77**	slv_Latn	3.11	4.32	**5.23**
bul_Cyrl	3.56	4.67	**5.88**	krc_Cyrl	2.85	3.66	**4.90**	sme_Latn	2.70	3.35	**4.40**
bum_Latn	3.22	3.73	**4.89**	kri_Latn	1.90	2.52	**5.07**	smo_Latn	2.26	2.72	**4.34**
bzj_Latn	1.65	2.43	**4.48**	ksd_Latn	2.82	3.28	**5.42**	sna_Latn	2.89	3.39	**5.32**
cab_Latn	2.16	2.63	**2.98**	kss_Latn	0.99	1.09	**1.49**	snd_Arab	3.12	3.92	**5.30**
cac_Latn	1.51	1.74	**2.86**	ksw_Mymr	0.95	1.46	**4.18**	som_Latn	3.15	3.40	**4.17**
cak_Latn	1.86	2.18	**3.24**	kua_Latn	4.25	4.92	**7.31**	sop_Latn	2.80	3.55	**4.23**
caq_Latn	2.20	2.94	**3.66**	lam_Latn	2.41	3.09	**4.03**	sot_Latn	3.49	4.31	**6.96**
cat_Latn	3.76	4.04	**5.24**	lao_Laoo	2.61	3.21	**4.39**	spa_Latn	3.71	4.21	**5.86**
cbk_Latn	3.12	3.64	**4.34**	lat_Latn	4.65	5.51	**7.44**	sqi_Latn	4.07	5.07	**6.50**
cce_Latn	2.96	3.40	**4.86**	lav_Latn	3.35	4.56	**6.45**	srm_Latn	1.75	1.96	**3.23**
ceb_Latn	3.45	4.13	**5.10**	ldi_Latn	3.41	3.94	**4.29**	srn_Latn	3.40	3.86	**5.98**
ces_Latn	4.33	5.27	**7.75**	leh_Latn	2.73	3.66	**5.28**	srp_Cyrl	6.48	6.50	**10.24**
cfm_Latn	2.69	3.18	**4.52**	lhu_Latn	1.43	**1.61**	1.36	srp_Latn	4.16	5.06	**6.31**
che_Cyrl	2.50	3.02	**3.17**	lin_Latn	1.78	2.73	**4.61**	ssw_Latn	3.27	4.02	**5.72**
chk_Hani	4.88	6.75	**7.08**	lit_Latn	4.69	5.66	**7.07**	sun_Latn	2.98	3.69	**4.61**
chk_Latn	3.20	3.94	**5.36**	loz_Latn	3.35	3.91	**6.03**	suz_Deva	1.68	1.66	**2.82**
chv_Cyrl	2.25	2.77	**4.79**	ltz_Latn	3.73	3.99	**5.16**	swe_Latn	4.77	4.76	**7.09**
ckb_Arab	2.38	3.15	**3.86**	lug_Latn	2.84	3.50	**5.59**	swh_Latn	4.05	4.99	**7.27**
ckb_Latn	2.11	2.57	**3.35**	luo_Latn	3.34	4.09	**4.90**	sxn_Latn	2.08	2.54	**3.06**
cmn_Hani	3.24	4.57	**5.22**	lus_Latn	2.43	2.99	**5.20**	tam_Latn	2.59	**3.08**	2.56
cnh_Latn	2.17	2.75	**3.62**	lzh_Hani	3.21	**5.56**	5.47	tam_Taml	3.09	3.77	**5.74**
crh_Cyrl	3.14	3.79	**6.77**	mad_Latn	2.65	3.29	**4.45**	tat_Cyrl	2.13	2.62	**4.03**
crs_Latn	2.63	3.46	**4.88**	mah_Latn	2.95	3.59	**4.92**	tbz_Latn	1.62	2.03	**4.22**
csy_Latn	2.58	3.02	**4.25**	mai_Deva	1.79	2.02	**3.86**	tca_Latn	1.29	1.56	**2.77**
ctd_Latn	2.94	3.61	**4.65**	mal_Latn	2.67	**3.36**	2.71	tdt_Latn	3.20	3.48	**5.06**
ctu_Latn	1.89	2.31	**2.40**	mal_Mlym	3.19	4.13	**4.76**	tel_Telu	2.87	3.78	**3.98**
cuk_Latn	2.20	2.87	**3.09**	mam_Latn	1.84	2.20	**2.22**	teo_Latn	3.37	4.18	**4.29**
cym_Latn	3.11	3.78	**3.85**	mar_Deva	3.87	5.13	**5.65**	tgk_Cyrl	2.63	3.29	**6.11**
dan_Latn	4.06	5.03	**6.94**	mau_Latn	1.60	**1.78**	1.12	tgl_Latn	3.22	3.35	**5.16**
deu_Latn	4.85	5.19	**7.28**	mbb_Latn	2.25	2.56	**3.51**	tha_Thai	1.50	2.72	**4.10**
djk_Latn	2.07	2.46	**3.53**	mck_Latn	3.34	4.06	**5.09**	tih_Latn	2.21	2.89	**4.57**
dln_Latn	3.89	4.89	**5.23**	mcn_Latn	3.74	4.42	**5.60**	tir_Ethi	1.90	1.93	**4.03**
dtp_Latn	2.05	2.28	**3.04**	mco_Latn	1.42	1.63	**1.69**	tlh_Latn	3.02	3.52	**5.71**
dyu_Latn	2.75	3.32	**5.29**	mdy_Ethi	1.36	1.26	**2.89**	tob_Latn	1.42	1.84	**2.00**
dzo_Tibt	0.39	**2.51**	2.03	meu_Latn	3.26	3.79	**5.10**	toh_Latn	2.17	2.90	**4.41**

Table 21: Accuracy of XLM-R-B, XLM-R-L, and Glot500-m on Round Trip Alignment (Part I).

Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m
efi_Latn	2.55	3.25	**6.23**	mfe_Latn	3.61	4.19	**6.26**	toi_Latn	3.19	4.10	**4.31**
ell_Grek	2.79	3.38	**4.77**	mgh_Latn	2.78	3.28	**3.48**	toj_Latn	1.43	1.84	**2.25**
eng_Latn	4.02	4.49	**6.39**	mgr_Latn	3.32	4.06	**6.39**	ton_Latn	2.01	2.64	**3.63**
enm_Latn	3.77	4.60	**7.19**	mhr_Cyrl	2.75	3.28	**5.32**	top_Latn	1.56	2.16	**2.19**
epo_Latn	4.01	4.83	**5.88**	min_Latn	2.62	3.05	**3.78**	tpi_Latn	2.44	2.71	**5.96**
est_Latn	4.34	5.24	**8.21**	miq_Latn	2.23	3.13	**4.12**	tpm_Latn	2.79	3.39	**4.67**
eus_Latn	3.12	3.80	**4.19**	mkd_Cyrl	3.99	4.54	**7.37**	tsn_Latn	2.82	3.12	**4.63**
ewe_Latn	2.22	2.67	**4.74**	mlg_Latn	3.34	3.81	**6.33**	tso_Latn	2.40	3.05	**5.00**
fao_Latn	3.85	4.62	**5.75**	mlt_Latn	2.94	3.57	**4.87**	tsz_Latn	2.68	3.14	**4.20**
fas_Arab	4.54	4.48	**7.00**	mos_Latn	2.71	3.24	**4.25**	tuc_Latn	1.43	1.83	**2.36**
fij_Latn	2.81	3.17	**4.94**	mps_Latn	1.50	1.65	**3.05**	tui_Latn	2.47	2.83	**4.53**
fil_Latn	3.26	3.92	**4.80**	mri_Latn	2.81	3.44	**5.49**	tuk_Cyrl	2.74	3.68	**4.33**
fin_Latn	4.06	5.19	**6.03**	mrw_Latn	2.69	3.24	**4.58**	tuk_Latn	2.43	3.23	**4.74**
fon_Latn	1.63	1.89	**3.70**	msa_Latn	3.17	3.50	**5.38**	tum_Latn	3.41	4.13	**6.15**
fra_Latn	3.19	3.97	**5.08**	mwm_Latn	1.74	1.99	**3.20**	tur_Latn	5.18	4.86	**7.45**
fry_Latn	3.36	3.99	**4.52**	mxv_Latn	1.75	2.11	**2.31**	twi_Latn	3.05	4.06	**6.70**
gaa_Latn	2.74	3.26	**6.01**	mya_Mymr	1.54	1.53	**2.46**	tyv_Cyrl	2.31	2.83	**3.33**
gil_Latn	2.76	3.20	**4.50**	myv_Cyrl	2.90	3.42	**4.46**	tzh_Latn	2.16	2.50	**3.08**
giz_Latn	3.00	3.43	**5.40**	mzh_Latn	2.62	3.02	**4.10**	tzo_Latn	2.01	2.29	**2.77**
gkn_Latn	1.93	2.07	**3.31**	nan_Latn	1.99	2.51	**2.56**	udm_Cyrl	2.90	3.48	**4.72**
gkp_Latn	1.88	2.25	**3.40**	naq_Latn	2.42	3.15	**4.41**	uig_Arab	2.58	3.11	**3.61**
gla_Latn	2.90	3.48	**3.61**	nav_Latn	1.75	2.10	**2.71**	uig_Latn	2.26	2.76	**3.79**
gle_Latn	3.52	4.24	**4.49**	nbl_Latn	3.09	3.87	**4.85**	ukr_Cyrl	5.71	5.96	**7.47**
glv_Latn	2.76	3.38	**4.45**	nch_Latn	2.18	2.74	**3.32**	urd_Arab	1.88	2.88	**3.96**
gom_Latn	3.05	3.59	**4.40**	ncj_Latn	2.64	3.40	**3.69**	urd_Latn	2.29	2.97	**3.03**
gor_Latn	2.26	2.73	**3.71**	ndc_Latn	3.32	3.85	**6.67**	uzb_Cyrl	2.73	3.26	**7.24**
grc_Grek	1.11	2.00	**2.93**	nde_Latn	4.00	4.60	**6.05**	uzb_Latn	3.32	3.98	**5.91**
guc_Latn	1.46	1.80	**2.23**	ndo_Latn	3.21	3.85	**5.61**	uzn_Cyrl	2.61	3.06	**5.86**
gug_Latn	2.60	3.23	**4.70**	nds_Latn	2.98	3.69	**4.70**	ven_Latn	2.96	3.64	**5.34**
guj_Gujr	3.18	4.15	**4.38**	nep_Deva	3.02	2.97	**6.31**	vie_Latn	3.99	4.48	**6.69**
gur_Latn	2.14	2.59	**3.22**	ngu_Latn	1.86	2.34	**3.39**	wal_Latn	2.87	3.65	**4.24**
guw_Latn	2.18	2.54	**4.56**	nia_Latn	2.75	**3.47**	3.24	war_Latn	3.04	3.74	**5.43**
gya_Latn	1.94	2.25	**4.63**	nld_Latn	2.81	3.63	**4.90**	wbm_Latn	2.44	2.86	**6.53**
gym_Latn	1.44	1.78	**2.63**	nmf_Latn	3.30	4.27	**5.05**	wol_Latn	3.47	4.48	**6.10**
hat_Latn	3.21	3.64	**6.39**	nnb_Latn	2.46	3.14	**4.08**	xav_Latn	0.87	1.03	**1.12**
hau_Latn	3.69	4.24	**6.31**	nno_Latn	3.90	4.61	**7.41**	xho_Latn	3.61	4.27	**5.90**
haw_Latn	2.25	2.63	**3.55**	nob_Latn	3.88	4.81	**5.83**	yan_Latn	2.95	3.35	**5.59**
heb_Hebr	1.85	2.41	**3.92**	nor_Latn	3.31	4.14	**5.82**	yao_Latn	2.01	2.66	**3.87**
hif_Latn	2.90	3.43	**3.60**	npi_Deva	3.29	3.30	**5.93**	yap_Latn	2.86	3.41	**3.45**
hil_Latn	2.92	3.48	**4.88**	nse_Latn	3.29	4.06	**5.74**	yom_Latn	3.25	4.00	**5.17**
hin_Deva	3.39	3.80	**5.13**	nso_Latn	3.06	3.92	**5.51**	yor_Latn	2.24	2.68	**3.88**
hin_Latn	2.94	3.20	**4.77**	nya_Latn	2.76	**3.19**	**5.96**	yua_Latn	2.04	2.26	**2.86**
hmo_Latn	2.43	2.70	**6.12**	nyn_Latn	2.77	3.50	**5.59**	yue_Hani	2.37	**3.19**	2.95
hne_Deva	2.48	2.53	**4.95**	nyy_Latn	2.21	2.74	**2.95**	zai_Latn	3.22	3.76	**5.21**
hnj_Latn	2.14	2.53	**4.28**	nzi_Latn	2.09	2.70	**4.20**	zho_Hani	2.77	4.38	**5.03**
hra_Latn	3.32	3.86	**5.19**	ori_Orya	2.73	2.77	**3.92**	zlm_Latn	4.39	5.15	**7.54**
hrv_Latn	4.14	5.24	**7.02**	ory_Orya	3.27	3.20	**4.39**	zom_Latn	3.65	4.45	**5.36**
hui_Latn	1.84	2.10	**3.47**	oss_Cyrl	2.20	2.52	**5.85**	zsm_Latn	4.49	5.07	**8.83**
hun_Latn	4.54	4.10	**5.62**	ote_Latn	1.89	2.23	**2.66**	zul_Latn	3.67	4.39	**5.44**
hus_Latn	1.70	2.00	**2.42**	pag_Latn	2.93	3.44	**4.56**				

Table 22: Accuracy of XLM-R-B, XLM-R-L, and Glot500-m on Round Trip Alignment (Part II).

Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m
srd_Latn	87.2	66.6	**5.4**	aka_Latn	86.7	74.1	**14.2**	dyu_Latn	68.5	27.4	**10.2**
ben_Beng	5.2	**3.7**	7.2	mon_Latn	288	282.4	**33.7**	nyy_Latn	628.5	198.3	**18.0**
ajp_Arab	74.6	**34.0**	44.8	gor_Latn	89.8	140.7	**8.8**	tzh_Latn	320.3	82.8	**4.7**
tdx_Latn	688.4	716.4	**16.0**	kjb_Latn	110.8	81.1	**16.2**	hne_Deva	80.1	60.3	**9.1**
tpm_Latn	99.9	90.2	**17.9**	lhu_Latn	44.7	12.3	**2.0**	bel_Cyrl	3.4	**2.5**	5.3
grc_Grek	10.1	10.4	**3.4**	bos_Latn	6.1	**3.4**	7.9	szl_Latn	46.4	30.2	**3.1**
sxn_Latn	469.2	148.3	**14.5**	lmo_Latn	48.4	25.9	**6.1**	ksh_Latn	340.3	227.6	**19.9**
cos_Latn	52.1	22.8	**13.3**	mwn_Latn	697.8	543.8	**30.7**	pcd_Latn	61.2	40.8	**13.2**
tlh_Latn	53.6	46.3	**11.1**	aym_Latn	1084.6	727.8	**14.5**	ada_Latn	100	78.5	**9.5**
sid_Latn	1003.6	782.3	**34.5**	aoj_Latn	95.1	53.7	**7.4**	pxm_Latn	101.3	120.7	**2.7**
jam_Latn	213.3	195.2	**15.8**	est_Latn	7.7	**4.0**	22.1	xho_Latn	32.5	**9.4**	16.7
ban_Latn	40.8	76.1	**16.1**	bre_Latn	12.9	**3.7**	12.3	kaa_Cyrl	72.9	29.2	**8.8**
kin_Latn	544.1	203.2	**6.6**	bsb_Latn	74.5	45.1	**7.6**	kea_Latn	754.2	525.3	**13.4**
rop_Latn	150.7	93.4	**8.4**	yua_Latn	246.8	55.1	**4.6**	teo_Latn	587.1	271.7	**62.0**
alz_Latn	511.9	145.6	**47.7**	hrv_Latn	7.4	**4.9**	9.7	tsc_Latn	726.3	501.1	**17.0**
kwy_Latn	598.8	514.4	**30.5**	jav_Latn	20.2	**4.4**	22	hin_Deva	7.4	**3.1**	10
yor_Latn	109.1	55.9	**11.0**	mai_Deva	42.9	48.8	**6.0**	ekk_Latn	7	**3.8**	11.8
lao_Laoo	4.2	4.4	**3.8**	tyv_Cyrl	104.1	104.4	**7.3**	umb_Latn	920	838.8	**17.4**
aze_Latn	5.6	**3.6**	5.4	afb_Arab	68.7	**44.4**	55.9	tam_Taml	7.2	**2.3**	9.8
mya_Mymr	6.9	**2.7**	6.3	twi_Latn	178.9	66.7	**17.9**	toi_Latn	988.7	246.5	**20.9**
ssw_Latn	345.7	108.4	**20.2**	sme_Latn	293	368.2	**6.5**	kon_Latn	463.7	418.9	**16.3**
lus_Latn	493.5	131.2	**16.4**	yom_Latn	468	240.7	**43.1**	che_Cyrl	266.4	127.6	**5.7**
krc_Cyrl	120.1	63.2	**9.3**	tob_Latn	115	78.8	**7.2**	gaa_Latn	109.3	33.3	**13.5**
hbo_Hebr	6.3	**3.6**	5.6	mxv_Latn	69.8	29.7	**5.0**	tzo_Latn	246.5	54.3	**7.0**
mgr_Latn	737.8	254.2	**33.0**	ron_Latn	4.4	**2.9**	10.4	mon_Cyrl	5.8	**3.4**	8.6
crh_Cyrl	138.6	86.3	**5.2**	ile_Latn	67.9	40.1	**5.7**	cuk_Latn	211.5	72.1	**32.0**
ara_Arab	10.1	**6.3**	18.8	cce_Latn	468.3	123.5	**22.5**	ces_Latn	4.4	**3.1**	11.6
mar_Deva	7.5	**4.6**	11.2	uzn_Cyrl	402.4	138.7	**5.2**	rmy_Latn	288.2	349.8	**25.0**
nba_Latn	638.8	675.1	**14.6**	ibg_Latn	897.3	807.3	**21.8**	phm_Latn	914.5	678.5	**11.6**
mny_Latn	568.9	492.5	**38.7**	hat_Latn	228	113.3	**14.0**	glv_Latn	240.2	182.3	**9.4**
run_Latn	817.5	218.5	**16.9**	fij_Latn	377.3	96	**12.8**	diq_Latn	256.6	120.5	**13.4**
rus_Cyrl	3.3	**2.3**	4.5	kbp_Latn	34.6	24.5	**7.1**	poh_Latn	62.8	68.9	**3.8**
hbs_Latn	4.5	**2.6**	6	mlt_Latn	223	162.2	**10.3**	oss_Cyrl	121.8	58.7	**5.1**
lug_Latn	489	197.5	**13.1**	kjh_Cyrl	209.8	88.8	**16.4**	san_Deva	20.5	**12.4**	15.5
pls_Latn	91.7	98.9	**6.9**	ndo_Latn	892.3	178.1	**21.1**	ote_Latn	127.8	71.2	**8.0**
hif_Latn	21.6	46.7	**13.5**	rar_Latn	458.1	50.2	**12.1**	her_Latn	776	707.3	**31.6**
tll_Latn	244.6	161	**24.3**	ell_Grek	3.4	**2.6**	5.9	efi_Latn	256.8	47	**11.5**
crs_Latn	782.2	146.5	**7.4**	tvl_Latn	634.1	378.5	**7.1**	idu_Latn	117.7	90.9	**12.0**
rng_Latn	656.6	606.8	**11.7**	toj_Latn	287.1	113.6	**9.6**	hye_Armn	**3.6**	4.4	3.8
cjk_Latn	530.8	419.6	**24.0**	ikk_Latn	67.8	49.5	**8.6**	gcf_Latn	450.8	292.4	**5.5**
seh_Latn	917.8	230	**11.2**	ory_Orya	6.1	**2.8**	6.3	pus_Arab	12.9	**7.5**	12.7
rug_Latn	260.9	214.2	**5.4**	nor_Latn	5	**2.8**	8.5	sgs_Latn	119.2	124.7	**10.5**
hau_Latn	14.5	**7.1**	17.2	enm_Latn	43.1	**31.0**	36.6	mbb_Latn	177.1	138	**4.2**
uzb_Latn	5.6	**3.6**	5.8	arz_Arab	17.5	**1.5**	6.8	som_Arab	7.2	**3.1**	9.3
bim_Latn	142.2	97.3	**11.3**	bem_Latn	706.9	219.9	**27.1**	hsb_Latn	109.6	103.6	**5.2**
vep_Latn	218.1	111.5	**6.1**	gkp_Latn	33.1	30.2	**12.7**	ary_Arab	32.7	**4.6**	26
slv_Latn	7.8	**4.9**	26.9	guj_Gujr	6.2	**3.6**	6.5	hmo_Latn	509.3	77.7	**10.9**
azj_Latn	5.3	**3.3**	5.1	tbz_Latn	39.2	40.4	**8.4**	quw_Latn	177.8	157.7	**26.1**
cac_Latn	51.4	39.3	**7.0**	ven_Latn	268.3	62	**9.4**	pag_Latn	923.5	232.4	**25.8**
npi_Deva	8.6	**4.9**	7.3	crh_Latn	151	70.9	**6.5**	ber_Latn	639.1	981.4	**21.3**
lin_Latn	377.3	96.6	**15.3**	xmv_Latn	593.2	491.4	**19.4**	chk_Latn	766.9	151.6	**19.1**
zom_Latn	238.7	176.2	**22.8**	slk_Latn	4	**2.9**	11.2	kan_Knda	7.2	**2.8**	8.9
kmr_Cyrl	140.6	56.7	**4.1**	zne_Latn	854.7	658.4	**48.8**	loz_Latn	895	113.7	**27.8**
acm_Arab	113.6	**74.0**	81	cgg_Latn	565.7	454.4	**12.4**	tih_Latn	247.6	151.3	**4.9**
fin_Latn	4.2	**3.1**	21.7	vie_Latn	7.6	**3.1**	16.4	mfe_Latn	767.9	255.4	**10.1**
rmn_Grek	108.9	76.8	**3.3**	amh_Ethi	8.9	**5.3**	7.5	tel_Telu	6.5	**4.0**	7.9
wls_Latn	334.9	207.9	**4.0**	nyu_Latn	926.2	479.2	**9.3**	ina_Latn	26.9	17.1	**7.2**
hun_Latn	5.1	**3.3**	25.1	suz_Deva	63.4	76.4	**2.5**	isl_Latn	7.9	**4.9**	16.7
lij_Latn	98.8	55.1	**5.9**	tuc_Latn	108.9	80.8	**7.6**	tsz_Latn	990.6	199.7	**14.2**
quh_Latn	279	176.6	**16.5**	lub_Latn	670.8	577.5	**23.8**	ori_Orya	5.2	**3.0**	4.7
yap_Latn	507.3	195.9	**10.6**	epo_Latn	10.8	**5.2**	21	tat_Latn	168.4	65.5	**6.9**
abk_Cyrl	122.6	89.5	**20.1**	ksw_Mymr	16.6	7.5	**4.6**	arg_Latn	29.2	13.6	**7.2**
cmn_Hani	10.4	**5.0**	9.8	mwl_Latn	69.1	35.6	**4.9**	kia_Latn	132.4	126.8	**18.5**
csb_Latn	112.8	59.4	**6.1**	cak_Latn	101.7	46.1	**5.4**	afr_Latn	12.2	**7.8**	19.2
nbl_Latn	137.7	19.6	**13.9**	bar_Latn	124.7	108.9	**14.4**	myv_Cyrl	97.7	153.3	**8.5**
ndc_Latn	1188.5	374.6	**19.4**	asm_Beng	6	**3.8**	5	bik_Latn	170.4	60.3	**13.7**
oci_Latn	41.2	24.4	**8.3**	grn_Latn	199.3	141.6	**10.3**	ltz_Latn	39.7	165.1	**10.9**
fao_Latn	84.2	35.6	**5.5**	tso_Latn	506.1	115.2	**13.2**	iso_Latn	236.2	222.4	**8.7**
tui_Latn	126.1	127	**20.6**	nso_Latn	656.3	153.4	**9.1**	ewe_Latn	198	54.6	**20.0**
xav_Latn	21.4	15.9	**5.7**	bum_Latn	282.8	91.5	**22.1**	als_Latn	7.6	**2.5**	6.4

Table 23: Perplexity of all languages covered by Glot500-m (Part I).

Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m
swc_Latn	39.2	22.5	**13.2**	top_Latn	589.2	89.6	**23.5**	hin_Latn	**11.1**	22.1	11.9
deu_Latn	4.4	**3.6**	10.2	bin_Latn	278.1	169.8	**13.3**	eng_Latn	5.7	**4.0**	7.5
caq_Latn	185.9	129	**21.6**	chw_Latn	778.9	645.8	**33.9**	hus_Latn	134.6	68.2	**5.3**
ceb_Latn	63.1	53.1	**2.1**	hyw_Cyrl	268.5	233.5	**6.3**	urh_Latn	236.8	211.5	**11.4**
nia_Latn	280.3	85.5	**7.5**	kor_Hang	7.2	**2.6**	11	mkd_Cyrl	4.3	**3.1**	6.2
urd_Arab	8.3	**5.3**	8.7	btx_Latn	463	163.1	**19.3**	wbm_Latn	58.9	47.3	**13.6**
niu_Latn	600.1	437.5	**10.1**	srn_Latn	609.3	137.2	**12.6**	kwn_Latn	1053.6	753.2	**32.0**
mrw_Latn	320.8	174.9	**7.6**	llb_Latn	555.6	589.8	**41.1**	guc_Latn	432.6	117.8	**9.4**
bul_Cyrl	3.9	**3.6**	6.8	cbk_Latn	129.5	60.4	**11.6**	quc_Latn	270.7	83.9	**5.6**
pau_Latn	333.7	147.3	**7.2**	bcl_Latn	270	60.1	**12.5**	nds_Latn	112.5	161.1	**7.4**
tha_Thai	10.8	**2.9**	14.6	csy_Latn	198.3	152.5	**21.7**	ind_Latn	8.5	**5.4**	17.1
ilo_Latn	786.7	184.4	**13.8**	ctd_Latn	249.2	166.1	**11.6**	nde_Latn	56.7	21.5	**12.1**
kss_Latn	90.4	13.2	**11.2**	plt_Latn	10.8	**3.6**	5.7	kua_Latn	1104.8	191.2	**13.4**
zai_Latn	719.4	212.5	**10.4**	smo_Latn	235.7	55.6	**7.0**	nch_Latn	705.1	166.4	**11.2**
guw_Latn	267.7	65.5	**6.9**	kab_Latn	744.5	203.5	**24.3**	por_Latn	5.1	**3.9**	9.3
kbd_Cyrl	175.7	94.4	**9.1**	gom_Deva	82.8	48.4	**9.0**	jpn_Jpan	7.9	**3.9**	10
dln_Latn	238.8	207.8	**7.5**	ukr_Cyrl	3.1	**2.9**	5.9	spa_Latn	4.6	**3.5**	7.8
war_Latn	200.9	110.7	**2.3**	ast_Latn	27.5	18.6	**4.8**	knv_Latn	129	78.3	**5.8**
tca_Latn	70.4	49	**6.0**	lvs_Latn	4.8	**2.7**	5.7	agw_Latn	150.1	73.4	**16.3**
iku_Cans	2.2	**1.9**	5.8	rmn_Cyrl	624.3	513.1	**8.7**	ige_Latn	181.1	105.2	**11.9**
bjn_Latn	41.3	17.6	**11.4**	kir_Cyrl	7.7	**2.9**	11.9	dua_Latn	232.8	152.2	**19.1**
ngu_Latn	918	110.9	**13.4**	pfl_Latn	152	101.3	**11.3**	ogo_Latn	131.3	129.7	**31.1**
kmr_Latn	68	**4.6**	10.6	bqc_Latn	102.7	71.1	**26.5**	bas_Latn	410.4	437.7	**16.7**
tgl_Latn	7.9	**4.4**	8.9	yid_Hebr	7.6	**4.8**	5.1	bpy_Beng	20	21.4	**2.9**
eus_Latn	10.7	**6.2**	37.3	fil_Latn	9.2	**2.3**	9.9	lfn_Latn	60.4	51	**6.9**
hra_Latn	212.1	177.7	**54.3**	nap_Latn	81.7	39.6	**10.5**	ton_Latn	116	65.2	**2.8**
lue_Latn	839.2	627.4	**19.8**	heb_Hebr	6.7	**4.9**	13.5	lim_Latn	66.8	43.5	**11.4**
pol_Latn	4.5	**2.7**	10.6	sba_Latn	75.7	81.8	**6.0**	lav_Latn	4.2	**2.2**	6.6
leh_Latn	476.5	253.9	**26.2**	ifa_Latn	371.9	266.1	**6.0**	bih_Deva	27.6	16.1	**5.0**
lat_Latn	15.3	**3.7**	24.5	ami_Latn	1070.7	710.2	**29.2**	gym_Latn	509.6	66.3	**17.0**
div_Thaa	1.6	**1.5**	3.5	gil_Latn	763.5	161.3	**15.7**	ish_Latn	144.9	134	**11.6**
min_Latn	105	39.7	**3.9**	djk_Latn	360.4	93.4	**13.4**	zea_Latn	69.6	27.5	**8.7**
ctu_Latn	177.4	37.9	**4.5**	new_Deva	36.1	29.8	**4.5**	aln_Latn	3.9	**2.3**	12.7
tur_Latn	9.1	**4.1**	29.5	bam_Latn	74.5	**23.7**	46.8	gcr_Latn	352.9	314.7	**7.5**
dhv_Latn	509	435.8	**11.8**	wol_Latn	236.4	158.3	**32.0**	kal_Latn	377.2	370.9	**8.3**
lua_Latn	706	784.5	**21.7**	alt_Cyrl	140.7	50.9	**9.3**	dan_Latn	6	**3.6**	13.1
rmy_Cyrl	488.1	389.3	**9.3**	kri_Latn	87.6	35.8	**8.6**	tah_Latn	363	330.9	**4.8**
zpa_Latn	476.1	550.1	**13.6**	kom_Cyrl	93.4	57	**4.9**	kik_Latn	205.8	55.5	**12.1**
gom_Latn	405.7	282.9	**27.9**	sah_Cyrl	99.9	91.1	**4.5**	vmw_Latn	828.8	434.8	**17.8**
dtp_Latn	166.4	78.7	**5.5**	mzh_Latn	132.8	133.4	**9.6**	eml_Latn	283.4	144.9	**6.6**
fra_Latn	4.1	**2.8**	6.9	sna_Latn	316.6	331.1	**16.4**	sco_Latn	28.1	15.5	**9.8**
cat_Latn	4.1	**2.2**	7.3	bzj_Latn	264.7	75.8	**10.9**	kac_Latn	189.9	76.3	**17.9**
xmf_Geor	71.2	72.3	**3.8**	nld_Latn	5.7	**4.5**	12	ttj_Latn	865.2	509.5	**15.5**
ixl_Latn	53	29.6	**4.2**	gug_Latn	626.9	141.6	**8.4**	lun_Latn	720.1	565.6	**31.9**
ckb_Arab	72.2	80.6	**6.0**	yue_Hani	17.8	**10.6**	10.8	sot_Latn	269.1	122.4	**8.1**
ahk_Latn	44.8	9.1	**2.1**	fry_Latn	16.1	**15.4**	17.2	mau_Latn	199.7	13.6	**8.4**
sag_Latn	491.4	68.7	**11.1**	jbo_Latn	132.3	187.1	**9.0**	yan_Latn	134.4	108.4	**31.4**
qug_Latn	505	135.2	**13.7**	iba_Latn	529.3	87	**16.6**	ido_Latn	79.8	24.2	**7.1**
nyn_Latn	834.8	236.9	**16.8**	nya_Latn	319.6	256.8	**12.7**	rmn_Latn	968.8	1062.8	**22.9**
koo_Latn	481.3	321.6	**13.8**	tat_Cyrl	99.8	116	**4.1**	sat_Olck	1.4	**1.2**	4.6
uig_Arab	8.1	**2.4**	5.5	nzi_Latn	113.7	47.4	**12.5**	mad_Latn	132.7	90.2	**7.9**
kam_Latn	225.9	155.7	**10.3**	wal_Latn	492.7	120.3	**18.1**	hil_Latn	366	38.7	**9.6**
gkn_Latn	248	74.6	**9.4**	pdt_Latn	417.7	143	**13.3**	khm_Khmr	4.8	**3.2**	4.5
twx_Latn	1209.8	978.2	**15.5**	apc_Arab	74.8	42.2	**37.2**	fon_Latn	71.8	27	**10.4**
skg_Latn	665.4	624.1	**15.8**	mdy_Ethi	65.7	68.4	**5.4**	ngl_Latn	664.9	518.3	**15.9**
arb_Arab	4.1	**2.1**	6	rue_Cyrl	18.7	11.4	**4.5**	tcf_Latn	224.5	225.4	**6.9**
mco_Latn	295	37.6	**4.6**	azb_Arab	194.1	141.8	**4.8**	gur_Latn	86.2	39	**17.9**
sqi_Latn	6.2	**2.1**	8.4	bci_Latn	129.6	95.6	**8.7**	qvi_Latn	863.4	91.5	**12.3**
cnh_Latn	496	154.4	**16.3**	kmm_Latn	193.3	164.9	**20.2**	izz_Latn	95.5	78.5	**5.5**
sin_Sinh	7.5	**5.4**	9.8	bak_Cyrl	99	79	**5.3**	kur_Arab	90.3	76.3	**5.7**
kmb_Latn	564.8	465.8	**15.6**	miq_Latn	347.4	198.9	**23.6**	hbs_Cyrl	3.7	**2.3**	4.3
vol_Latn	78.4	67.7	**2.4**	kaa_Latn	94.2	100.6	**7.3**	ach_Latn	488.8	114.6	**77.3**
msa_Latn	**8.2**	26.1	15	bod_Tibt	8.8	**4.0**	6.3	wuu_Hani	35.9	16.8	**11.7**
bba_Latn	75.5	65.5	**16.3**	glg_Latn	5.9	**4.6**	9.2	quz_Latn	804.5	269.4	**12.2**
tgk_Latn	11.9	11.7	**7.5**	tum_Latn	516.4	168.3	**10.2**	tok_Latn	592.4	423	**94.5**
tiv_Latn	912.3	716.3	**29.3**	bbc_Latn	787.9	203.7	**13.6**	bis_Latn	727.1	47.7	**10.7**
hmn_Latn	60.9	52.5	**8.8**	kek_Latn	126.4	40.6	**4.3**	fur_Latn	196.5	142.8	**7.7**
swh_Latn	12.6	**5.8**	24.4	ace_Latn	81.5	54	**6.4**	ium_Latn	36.6	33.1	**7.2**
pis_Latn	563.2	64.7	**9.7**	pam_Latn	59.6	276.7	**28.2**	nse_Latn	771.7	292.3	**13.7**
mzn_Arab	50	34.3	**6.3**	fas_Arab	8	**4.1**	14.1	zul_Latn	36.3	**10.1**	21.7

Table 24: Perplexity of all languages covered by Glot500-m (Part II).

Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m	Language-Script	XLM-R-B	XLM-R-L	Glot500-m
bts_Latn	205.7	204.5	**8.8**	tsn_Latn	264.7	137.8	**12.5**	orm_Latn	23.4	**8.6**	16
gla_Latn	11.5	12.7	**7.2**	pon_Latn	928.4	181.9	**19.2**	luo_Latn	699.4	258.5	**85.1**
kat_Latn	36.4	24.8	**18.3**	nmf_Latn	297.6	310.6	**44.9**	pcm_Latn	38.3	169.6	**3.6**
uig_Latn	188.8	173.9	**15.2**	ajg_Latn	147.1	149.5	**22.6**	nnb_Latn	364.1	95	**28.6**
kat_Geor	6	**3.9**	6.4	tir_Ethi	28.3	15.7	**4.4**	kaz_Cyrl	**4.3**	5.4	9.6
mlg_Latn	10.9	**4.4**	7.6	bhw_Latn	411.2	126.2	**21.6**	dzo_Tibt	8.5	**3.3**	5.7
arn_Latn	382.7	96.7	**17.6**	mhr_Cyrl	122.9	168.4	**5.8**	sun_Latn	23.6	**11.9**	17
tuk_Latn	456.7	197.8	**5.8**	swe_Latn	4.8	**3.5**	12.7	vec_Latn	40.6	21.1	**9.2**
vls_Latn	97.7	39.6	**9.7**	scn_Latn	117	64.9	**7.8**	ayr_Latn	261.1	237.6	**27.7**
hyw_Armn	15.8	9.1	**4.3**	udm_Cyrl	356.7	224.9	**6.7**	oke_Latn	209.2	220.1	**13.0**
que_Latn	447.9	536.1	**11.9**	ifb_Latn	246.3	177.9	**5.1**	kur_Latn	14.2	**6.8**	10.3
snd_Arab	13.2	**4.1**	19.5	naq_Latn	136.8	60.2	**15.7**	mgh_Latn	680	272.8	**23.7**
giz_Latn	81.9	82.9	**37.7**	zlm_Latn	5.6	**3.3**	4.6	tgk_Cyrl	181.3	153	**4.5**
ita_Latn	4.5	**3.3**	7.2	hrx_Latn	478.1	679.1	**14.9**	sop_Latn	607.5	228.2	**29.5**
qub_Latn	283.2	312.7	**9.4**	lzh_Hani	70	58	**21.8**	mos_Latn	272.6	118.3	**13.2**
nav_Latn	228.5	126.5	**5.2**	pap_Latn	674.4	149.3	**18.1**	rap_Latn	36.1	31.1	**2.8**
kqn_Latn	825.9	686.6	**17.5**	cfm_Latn	235.1	155	**14.0**	prk_Latn	69.4	45.9	**7.1**
toh_Latn	758.3	216.6	**19.6**	chv_Cyrl	122.5	73.8	**5.4**	uzb_Cyrl	236.2	138.4	**4.9**
mah_Latn	314.7	81.8	**17.3**	tdt_Latn	641.9	78.6	**9.7**	tog_Latn	821.1	777.7	**13.4**
wes_Latn	144.6	103.9	**14.3**	pan_Guru	4.4	**2.5**	4.3	mal_Mlym	5	**3.7**	6.2
nob_Latn	6.8	**4.0**	9.5	pms_Latn	83.6	46.2	**3.6**	nyk_Latn	1182.6	914.2	**16.5**
ext_Latn	68.3	38.2	**8.1**	roh_Latn	243.5	170	**7.0**	quy_Latn	949.7	320.2	**14.5**
lam_Latn	233.7	160.8	**21.6**	prs_Arab	6.8	**3.5**	4.8	abn_Latn	245.2	272.5	**8.7**
mwm_Latn	44.8	53.1	**7.1**	tuk_Cyrl	277.4	86.3	**6.7**	mcn_Latn	120.7	129.7	**43.6**
kpg_Latn	165.9	122.6	**15.1**	srm_Latn	257.5	74.5	**12.3**	nep_Deva	8.8	**6.3**	10
hau_Arab	5.3	**3.0**	8.1	gsw_Latn	288.2	181.2	**22.3**	gle_Latn	10.5	**3.7**	9.8
ksd_Latn	150	154.9	**7.7**	fat_Latn	192.3	149	**17.6**	cab_Latn	1216.7	155.6	**15.4**
zsm_Latn	12.2	**2.9**	22.7	ldi_Latn	394.8	107.1	**38.2**	mps_Latn	75.2	55.2	**17.4**
hui_Latn	209.9	177	**10.0**	kos_Latn	470.7	485.7	**27.0**	pnb_Arab	51.8	30.8	**7.1**
cym_Latn	8.2	**4.8**	11.2	acr_Latn	155.7	90.7	**5.8**	swa_Latn	11.4	**6.4**	20
srp_Latn	10.9	**7.9**	13.3	mri_Latn	63	59.5	**8.7**	hnj_Latn	88.3	92.5	**11.3**
bak_Latn	347.1	211	**7.5**	frr_Latn	117.6	101	**9.5**	haw_Latn	63.5	66.7	**7.4**
zho_Hani	20.7	**5.9**	31.3	mck_Latn	369.3	164.8	**24.7**	tpi_Latn	891.8	67.8	**8.8**
nno_Latn	**9.9**	12.7	10.4	pes_Arab	5.5	**3.1**	5.3	ncj_Latn	1019	136.2	**13.7**
gya_Latn	31	24.3	**16.5**	san_Latn	94.4	96.8	**12.0**	som_Latn	14.1	**6.9**	22.2
ibo_Latn	77.1	90.1	**8.5**	yao_Latn	738.9	162.4	**13.8**	mam_Latn	132.7	62.4	**6.1**
meu_Latn	380.2	158.5	**26.7**	srp_Cyrl	7.4	**4.5**	8.4	lit_Latn	4.4	**2.5**	10.6
ncx_Latn	1084.7	948.5	**14.6**	ful_Latn	104	105.6	**13.1**				

Table 25: Perplexity of all languages covered by Glot500-m (Part III).

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
section 'Limitation'

☑ A2. Did you discuss any potential risks of your work?
section 'Ethics Statement'

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract and section 1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

section 3.3, section 4, appendix c

☑ B1. Did you cite the creators of artifacts you used?
section 3.3, section 4, appendix c

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
section 'Ethics Statement'

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
section 'Ethics Statement'

☒ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Since our work deals with millions of sentences in hundreds of languages, it was impossible for us to check the content. We leave it as a future work

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
section 3.1, appendix a, appendix c

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
section 5

C ☑ Did you run computational experiments?

section 4.2

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
section 4.2

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
section 5

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
section 5. For continued pretraining, it is a single run due to computational resource limitation. For downstream task evaluation, it is multilple runs across 5 seeds.

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
section 3.3

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.