Beyond English-Centric Bitexts for Better Multilingual Language Representation Learning

Barun Patra[*], **Saksham Singhal**[*], **Shaohan Huang**[*],
Zewen Chi, **Li Dong**, **Furu Wei**, **Vishrav Chaudhary**, **Xia Song**
Microsoft
{bapatra, saksingh, shaohanh, v-zewenchi, lidong1,
fuwei, vchaudhary, xiaso}@microsoft.com

Abstract

In this paper, we elaborate upon recipes for building multilingual representation models that are not only competitive with existing state-of-the-art models but are also more parameter efficient, thereby promoting better adoption in resource-constrained scenarios and practical applications. We show that going beyond English-centric bitexts, coupled with a novel sampling strategy aimed at reducing under-utilization of training data, substantially boosts performance across model sizes for both Electra and MLM pre-training objectives. We introduce **XY-LENT**: **X-Y** bitext enhanced **L**anguage **EN**codings using **T**ransformers which not only achieves state-of-the-art performance over 5 cross-lingual tasks within all model size bands, is also competitive across bands. Our XY-LENT$_{\mathrm{XL}}$ variant outperforms XLM-R$_{\mathrm{XXL}}$ and exhibits competitive performance with mT5$_{\mathrm{XXL}}$ while being 5x and 6x smaller respectively. We then show that our proposed method helps ameliorate the curse of multilinguality, with the XY-LENT$_{\mathrm{XL}}$ achieving 99.3% GLUE performance and 98.5% SQuAD 2.0 performance compared to a SoTA English only model in the same size band. We then analyze our models performance on extremely low resource languages and posit that scaling alone may not be sufficient for improving the performance in this scenario.

1 Introduction

Recent advancements in Natural Language Processing (NLP) have been a direct consequence of leveraging foundational models (Bommasani et al., 2021), pretrained on a large text corpora in a self-supervised fashion. This has also been the case for multilingual NLP where pre-trained models like multilingual BERT (mBERT) (Devlin, 2018; Devlin et al., 2019), XLM (Conneau and Lample, 2019), XLM-Roberta (Conneau et al., 2020), XLM-Electra (Chi et al., 2022) and mT5 (Xue et al., 2021) have all shown non-trivial performance gains, especially in the setup of zero-shot transfer, and have been the work-horse for a diverse number of multilingual tasks. Given their ubiquitous applicability in zero-shot downstream scenarios, improving the quality and enabling their usage in resource-constrained applications is also an important vein of research which we explore in this paper.

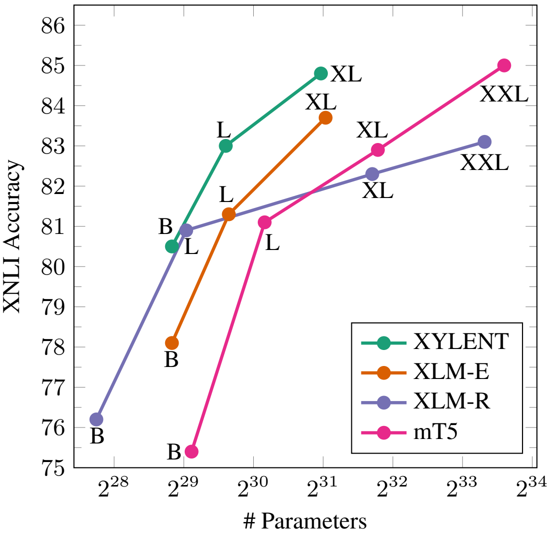


Figure 1: The proposed XY-LENT model (green line) achieves SoTA performance within all band sizes and is competitive performance across larger model-size bands. The parameter efficiency of XY-LENT$_{\mathrm{XL}}$ particularly stands out, outperforming XLM-R$_{\mathrm{XXL}}$ and being competitive with mT5$_{\mathrm{XXL}}$ while being 5x and 6x smaller than them respectively. We also present the performance of XLM-E which used as a baseline in this paper.

A source of improvement for these models has been leveraging bitext data for better representation learning (Conneau and Lample, 2019; Chi et al., 2022). Most prior work, however, has focused on leveraging English-centric (*EN-X*) bitext data. Contemporaneously, the related area of Massively Multilingual Machine Translation (a single model for translating between different pairs of languages, eg: Aharoni et al. (2019); Zhang et al. (2020); Fan

[*] Equal Contribution.

*Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers*, pages 15354–15373
July 9-14, 2023 ©2023 Association for Computational Linguistics

et al. (2021)) has shown tremendous progress, with Fan et al. (2021) showing that a crucial aspect of this improvement has been moving beyond *EN-X* parallel corpora and leveraging web-based mined *X-Y* bitexts spanning 1000s of translation directions (Schwenk et al., 2021a; El-Kishky et al., 2020; Schwenk et al., 2021b). This makes a compelling case to explore if leveraging *X-Y* bitexts can also improve multilingual representation learning.

In this work, we introduce **XY-LENT** (*pronounced as "**Excellent**"*): ***X-Y*** bitext enhanced **L**anguage **EN**codings using **T**ransformers. We first identify problems with using the commonly used sampling strategy proposed in Fan et al. (2021), showing that it induces sparse sampling distributions leading to under-utilization of data, and thus propose a novel strategy to mitigate this issue (§3.2). We then propose leveraging *X-Y* bitexts in conjunction with the improved sampling strategy, as well as a VoCAP (Zheng et al., 2021) style sentencepiece vocabulary re-construction for improving multilingual representation learning (§3.1). We show that our proposed method improves performance across all model size bands (§6). Furthermore, these performance gains hold for both Masked Language Models (MLM) and ELECTRA style models. Our approach results in an almost 12x speedup in training for MLM model training (§6.2). We systematically analyse the impact of model scaling with respect to the curse of multilinguality (Conneau et al., 2020) to observe that the gap between current English only SoTA models and multilingual models can be considerably reduced (§6.3). Our analysis reveals that XY-LENT improves performance across language families (§6.4) and helps reduce the cross-lingual transfer gap in multilingual tasks (§6.5). We then demonstrate that the training dynamics of such models can be used to better understand the underlying datasets and use it to find interesting defects in them (§6.6). Finally, we show some limitations of such multilingual representational models vis-à-vis extremely low resource languages, identifying potential shortcomings that are not addressed with scaling of such models, as well as issues around catastrophic forgetting in the way current models are used for domain adaptation.

In doing so, we establish state of the art on 5 multilingual downstream tasks (XNLI, PAWS-X, TyDIQA, XQuAD and MLQA) within a model size band, and achieve competitive performance *across* size bands, thereby showing for the first time (to the best of our knowledge) an interesting notion of parameter efficiency: XY-LENT$_{\text{XL}}$ outperforms XLM-R$_{\text{XXL}}$ (Goyal et al., 2021) and performs competitively with mT5$_{\text{XXL}}$ (Xue et al., 2021), whilst being 5x and 6x smaller respectively (Figure 1). Furthermore, our proposed model reduces the gap for English specific tasks: XY-LENT$_{\text{XL}}$ achieves 99.3% GLUE performance and 98.5% SQuAD 2.0 performance compared to a SoTA English only model in the same size band.

2 Related Work

Large scale self-supervised learning has emerged as a prominent way of building cross-lingual language models that can be adapted for numerous multilingual downstream applications. Especially for building multilingual encoder transformer (Vaswani et al., 2017) models, two popular paradigms have been Masked language modeling (MLM; Devlin et al. (2019); Conneau et al. (2020)) and pre-training encoders as discriminators (ELECTRA; Clark et al. (2020b); Chi et al. (2022)), with the latter showing considerable compute efficiency. These approaches can further be improved by leveraging parallel corpora in different ways: Conneau and Lample (2019) propose a Translation Language Modeling task (TLM) wherein the model predicts masked tokens in concatenated translation pairs, Chi et al. (2022) propose a Translation Replaced Token Detection (TRTD) task, an analogous task for Electra-style models. Other approaches include using bitexts to construct code-switched sequences as inputs during pre-training (ALM; Yang et al. (2020)) and for contrastive learning (InfoXLM; Chi et al. (2021a)), or using token-level alignments in parallel data to improve cross-lingual modeling (Hu et al., 2021; Chi et al., 2021b, *inter alia*). However, all the aforementioned works rely on English-centric bitexts.

Fan et al. (2021) show that moving beyond *EN-X* bitexts for Massively Multilingual Machine Translation affords substantial improvements over approaches that rely solely on English-centric data (Aharoni et al., 2019; Zhang et al., 2020). The primary factor responsible for this improvement has been the curation of *X-Y* aligned bitext data, constructed by mining bitexts from publicly available web data (Schwenk et al., 2021a; El-Kishky et al., 2020; Schwenk et al., 2021b). The dataset construction either follows a local mining approach (first aligning documents using heuristics, and then mining parallel bitexts from the aligned documents; used in CCAligned (El-Kishky et al., 2020)), or a global mining approach (all bitexts are embedded in a common vector space, and then aligned candidates are found by looking at the normalized nearest neighbors; used in CCMatrix (Schwenk et al., 2021b)). Fan et al. (2021) also propose a sampling strategy for leveraging the *X*-*Y* bitexts, wherein the marginals are constrained to be similar to what is used for *En*-*X* bitexts, and show their proposed method improves over uniform sampling. However, as we show in (§3.2), their proposed strategy has the undesirable artefact of inducing extremely sparse solutions, thereby resulting in data wastage.

3 Leveraging Many-to-Many Bitexts

3.1 Dataset

Prior representation learning works usually consider English-centric (*EN*-*X*) bitexts to improve model quality. Thus, given the emergence of mining based approaches for extracting parallel bitexts from large monolingual datasets that are approximate translations of each other and are multi-way aligned (the source and target languages are not restricted to be English only), in this work we explore leveraging these many-to-many (*X*-*Y*) bitext datasets for better representation learning. We consider two such publicly available datasets: CCMatrix and multiCCAligned.

3.2 Sampling Distribution

A common method used for balancing training data for the *EN*-*X* framework is using a temperature based exponential sampling approach (Aharoni et al., 2019), wherein the probability of sampling a language is chosen from a temperature smoothed distribution to downsample high resource languages, whilst upsampling low resource languages. This work was extended by Fan et al. (2021), wherein the authors propose Sinkhorn Temperature sampling: given a joint probability matrix $\mathbb{Q}$ across $L \times L$ language pairs (L being the number of unique languages), and the marginal distribution $\mathbf{p}$ of the L languages, the authors estimate a sampling distribution $\mathbb{P}^*$ as:

$$\max_{\mathbf{P}} \mathrm{Tr}(\mathbf{P}\mathbb{Q}) \mid \mathbf{P}1_L = \mathbf{p}^{\frac{1}{T}} = \mathbf{P}^\top 1_L \qquad (1)$$

where Tr is the trace operator. The primary advantage of using this is that $\mathbb{P}^*$ can be efficiently estimated with the Sinkhorn-Knopp algorithm and also allows us to set the marginal to be the temperature sampled based distribution which we know works well in practice. The authors found this to work better than uniform sampling.

However, in practice, we observed this to generate extremely sparse sampling distributions: Figure 2a show the sparsity induced by the naive application of Eq. 1.

We note that one potential way of overcoming the above issue is by modifying the optimization problem to also maximize the entropy of $\mathbf{P}$. Consequently, we propose the following modified optimization objective :

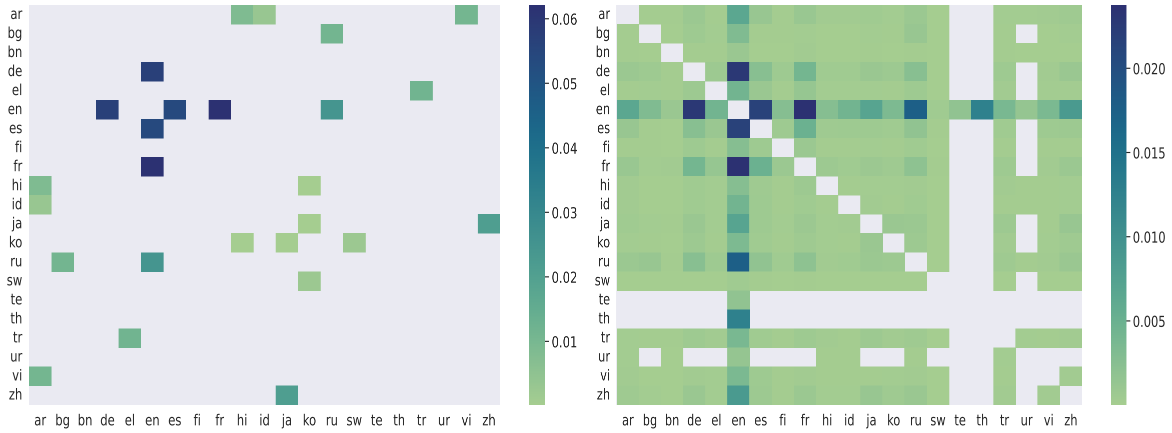
$$\begin{aligned}\mathbb{P}^* &= \mathtt{argmin}_{\mathbf{P}} \mathrm{Tr}\left(P\left(-\log \mathbb{Q}\right)\right) - \mathcal{H}\left(P\right) \\ &\mid \mathbf{P}1_L = \mathbf{p}^{\frac{1}{T}} = \mathbf{P}^\top \mathbb{Q}) \mid \mathbf{P}1_L = \mathbf{p}^{\frac{1}{T}} = \mathbf{P}^\top 1_L\end{aligned} \qquad (2)$$

where $\mathcal{H}(P)$ denotes the entropy of P and $KL(P||Q)$ denotes the Kullback-Leibler divergence between P and Q.

This can be solved by using the Sinkhorn-Knopp algorithm for the entropic regularized optimal transport problem (Cuturi, 2013), by setting the cost matrix to be $-\log(\mathbb{Q} + \epsilon)$ (in practice, since $\mathbb{Q}$ can have zero entries, ϵ is used for smoothing). Since the cost of assigning a non-zero probability value to a zero entry is extremely high ($-\log(\epsilon)$), we never observe any entry of $\mathbb{P}^*$ to be non-zero if it's corresponding entry in $\mathbb{Q}$ was zero. In addition, since Eq. 2 also maximizes the entropy of $\mathbf{P}$, it encourages its entries to be non-sparse, thereby avoiding the problem present in the solution of Eq. 1. In practice, we did not see this losing out on any data: if $\mathbb{Q}$ was non-zero, then $\mathbb{P}^*$ was also non-zero (Figure 2b).

3.3 Vocabulary Construction

We construct our vocabulary using Sentence Piece Models (SPM) (Kudo and Richardson, 2018) which cater to language specific complexities (tokenization, accent removal, etc.). We increase the vocabulary size to 500k tokens to better serve the varied scripts encountered while working in the multilingual setting. For this construction, we follow the VoCAP algorithm (Zheng et al., 2021) to quantify the vocabulary capacity for each language separately and account for varied corpora sizes across languages. Better capacity allocation leads to smaller representative sequences (especially for mid and low resource languages) which



(a) M2M 100 Sampling

(b) Proposed Sampling

Figure 2: Density plots for our probability distributions for sampling strategies for M2M 100 and our proposed sampling strategy for the 21 languages considered in downstream tasks. For similar plot for all the languages, see Figure 6b in the Appendix

in-turn improves the computational efficiency of the model. Increasing the size of the vocabulary, however, comes at the cost of inflating the model parameters which is particularly observed in the case of XY-LENT$_{\text{Base}}$ and XY-LENT$_{\text{Large}}$ where the embeddings layer constitute *80.5%* and *62.9%* of the total parameters respectively.

4 Pretraining Details

We follow the XLM-E (Chi et al., 2022) pretraining approach and only introduce a few architectural changes to improve the overall performance of the model. We use the Transformer model (Vaswani et al., 2017) trained with ELECTRA (Clark et al., 2020b) style of replace token detection (RTD) on both monolingual (MRTD) and bitext (TRTD) data. In the current setup of training, we use two Transformer encoders in conjunction: a generator G and a discriminator D, where the generator G is trained with masked language modeling objective (MLM; Devlin et al. (2019)) and the discriminator is trained on replaced token detection objective (RTD; Clark et al. (2020b) on all the tokens passing through the generator.

In addition to using the Gated Relative Position Bias introduced in Chi et al. (2022), we do not mask the [CLS] token and flip bitext language order with probability $p = 0.5$ for the TRTD task.

5 Experiments

Baselines: We compare the cross-lingual performance of our proposed model against 3 popular cross-lingual models: XLM-R, mT5 and XLM-E (across all model size variations). Note that Chi et al. (2022) use a 250k vocabulary size for XLM-E$_{\text{Base}}$ and 500k vocabulary for their large and XL variants. As a follow-up, we re-train XLM-E$_{\text{Base}}$ with the same vocabulary as used by XY-LENT for a fair comparison. Thus all references to XLM-E$_{\text{Base}}$ refer to the re-trained model variant with a 500k vocabulary size.[1] For our downstream English evaluation (§6.3), we compare against the SoTA English model METRO-LM(Bajaj et al., 2022). Note that Bajaj et al. (2022) also train the models in an ELECTRA style framework, thereby allowing for a fair comparison.

Pretraining Data: For our monolingual data, we follow Chi et al. (2022) and use the CC-100 dataset[2] (Conneau et al., 2020; Wenzek et al., 2020) which contains texts in 100 languages collected from Common Crawl. As mentioned in (§3.1), we explore the utility of the CCMatrix and the multiCCAligned *X-Y* aligned bitext data. CCMatrix consists of 1015 language pairs (97 unique languages) [3]; while the multiCCAligned dataset consists of 2959 language pairs (92 unique languages) [4]. We contrast this against only using *EN-X* bitexts (CCAligned, El-Kishky et al. (2020)).

[1] We also ablate out the impact of the vocabulary change, with Table 2 showing that this yields a 1.5 pt gain on XNLI.

[2] http://data.statmt.org/cc-100/

[3] For some language pairs that are present in CCAligned and not in CCMatrix, we combine the data from those languages. Since de-duplication is expensive, we don't merge language pairs common to both datasets.

[4] We filter out languages with less than 50k pairs

Model Size Bands: While our *base* and *large* models have more parameters when compared with XLM-R, most of the additional parameters come from the increased vocabulary size (§3.3). Concretely, our *base* model has 12 layers and 768 hidden states, while the *large* model has 24 layer and 1024 hidden states, which is identical to XLM-R_{Base} and XLM-R_{Large} respectively. However, even with the increased parameter count, the computation cost on a text classification task is roughly the same within a model size family (since mapping tokens to an embedding is a lookup operation). Finally, it is noteworthy that even with the increased vocabulary size, the number of parameters for XY-LENT_{XL} is less compared to the XL and XXL variants of both XLM-R and mT5.

Pretraining Setup: For the base model, we train for 125k steps with a batch size of 8192 for MRTD task and for the large model, we train the model for 500k steps with a batch size of 2048. Finally for the XL model, we train for 150k steps with a batch size of 8192. We use a dynamic batch size for TRTD task which is based on original length of the translated bi-text pair. Please refer Appendix A for additional details. We adopt the standard practice of using a linear warmup schedule for the learning rate and use the Adam (Kingma and Ba, 2014) optimizer for all the models. Following Meng et al. (2021), we do not apply any dropout to the generator.

Cross-lingual Downstream Evaluation: For evaluating the cross-lingual understanding of the model, we consider 5 multilingual evaluation benchmarks. We consider 2 classification tasks and 3 question answering tasks. For classification, we evaluate on the cross-lingual Natural Language Inference dataset (XNLI; Conneau et al. (2018)) and the cross-lingual paraphrase adversaries from word scrambling dataset (PAWS-X; Yang et al. (2019)). For cross-lingual question answering, we consider MLQA (Lewis et al., 2019), XQuAD (Artetxe et al., 2019) and TyDiQA-GoldP (Clark et al., 2020a). For all the aforementioned tasks, we perform the evaluation in zero-shot setting, i.e. only using the English data for fine-tuning. To further assess the model's performance when translated data is available, we evaluate the model on the translate-train setup for the classification tasks.

English Downstream Evaluation: To further assess XY-LENT's performance on English and see how the curse of multilinguality impacts the model, we also assess the model's performance on the commonly used GLUE benchmark (Wang et al., 2018), comprising of 8 tasks: MNLI (Williams et al., 2017), SST-2 (Socher et al., 2013), QNLI (Rajpurkar et al., 2018a), MRPC (Dolan and Brockett, 2005), CoLA (Warstadt et al., 2018), QQP , STS-B (Cer et al., 2017) and RTE. Additionally, we also evaluate the English performance of our model on a question answering task, using the SQuAD 2.0 dataset (Rajpurkar et al., 2018b).

Please refer to Appendix B for additional details on the datasets.

6 Results and Analysis

6.1 Main Results

Table 1 presents our proposed model's performance across different model sizes for zero-shot transfer on sentence classification as well as question answering tasks (detailed results for all languages and all tasks can be found in Appendix D). We see that XY-LENT outperforms the baselines of XLM-E, XLM-R and mT5 across all model sizes, establishing (to the best of our knowledge) the state-of-the-art (SoTA) for all the 5 considered multilingual datasets within the model size bands: with XY-$\text{LENT}_{\text{Base}}$ outperforming XLM-E_{Base} by 3.1 pts, XY-$\text{LENT}_{\text{Large}}$ outperforming XLM-E_{Large} by 1.8 pts and XY-LENT_{XL} outperforming XLM-E_{XL} by 0.9 pts (averaged across all 5 datasets). Another interesting observation is that XY-LENT is competitive across model size families: the XY-$\text{LENT}_{\text{Base}}$ model out-performs XLM-R_{Large} and $\text{mT5}_{\text{Large}}$ variants on 4 out of 5 datasets, similarly the XY-$\text{LENT}_{\text{Large}}$ outperforms the mT5_{XL} model on 4 out of 5 datasets. Furthermore, the XY-LENT_{XL} model outperforms XLM-R_{XXL} and is competitive with mT5_{XXL} while being 5x and 6x smaller respectively. A practical implication of these better performing smaller models is their easy usage in downstream tasks.

This behaviour is also consistent in the *Translate-Train* setting where the translated version of the training data is present across all languages for training. Table 1 presents XY-LENT's performance on this setup for sentence classification tasks. We see that even in this setting, XY-LENT outperforms other models with the same size band, and is competitive across model size bands.

Model	Zero-Shot					Translate-Train	
	Question Answering			Sentence Classification		Sentence Classification	
	XQuAD	MLQA	TyDiQA	XNLI	PAWSX	XNLI	PAWSX
Metrics	F1/EM	F1/EM	F1/EM	Acc.	Acc.	Acc.	Acc.
XLM-R$_{\text{Base}}$	-	-	-	76.2	-	79.1	-
mT5$_{\text{Base}}$	67.0 / 49.0	64.4 / 45.0	58.1 / 42.8	75.4	86.4	75.9	89.3
XLM-E$_{\text{Base}}$	74.3 / 59.2	68.7 / 50.5	62.7 / 46.2	78.1	87.0	81.7	91.1
XY-LENT$_{\text{Base}}$	**76.8 / 62.1**	**71.3 / 53.2**	**67.1 / 51.5**	**80.5**	**89.7**	**84.9**	**92.4**
XLM-R$_{\text{Large}}$	76.6 / 60.8	65.1 / 45.0	71.6 / 53.2	80.9	86.4	83.6	-
mT5$_{\text{Large}}$	77.8 / 61.5	71.2 / 51.7	57.8 / 41.2	81.1	88.9	81.8	91.2
XLM-E$_{\text{Large}}$	78.7 / 63.1	72.8 / 54.4	71.8 / 54.7	81.3	89.0	84.1	91.9
XY-LENT$_{\text{Large}}$	**79.7 / 64.9**	**74.3 / 55.7**	**74.0 / 57.5**	**83.0**	**90.4**	**84.9**	**92.4**
XLM-R$_{\text{XL}}$	80.0 / 64.9	73.4 / 55.3	-	82.3	-	85.4	-
mT5$_{\text{XL}}$	79.5 / 63.6	73.5 / 54.4	77.4 / 61.5	82.9	89.6	84.8	91.0
XLM-E$_{\text{XL}}$	80.4 / 66.0	74.3 / 55.8	76.7 / 60.6	83.7	90.3	85.5	92.2
XY-LENT$_{\text{XL}}$	**81.3 / 66.3**	**75.4 / 56.7**	**78.0 / 62.1**	**84.8**	**91.0**	**87.1**	**92.6**
XLM-R$_{\text{XXL}}$	81.1 / 66.3	74.8 / 56.6	-	83.1	-	86.0	-
mT5$_{\text{XXL}}$	**82.5 / 66.8**	**76.0 / 57.4**	**81.0 / 65.6**	**85.0**	**90.0**	**87.8**	**91.5**

Table 1: Results on sentence-pair classification and question answering tasks. XLM-R metrics and mT5 metrics are reported from Goyal et al. (2021) and Xue et al. (2021) respectively. Metrics for XLM-E and XY-LENT are reported based on median across five fine-tuning runs. Scores of best performing models within a model size band have been highlighted for each task. Full results for all the languages across all tasks can be referred in Appendix B.

Parameter	Choice	XNLI (Avg)
Vocabulary Size	250K	76.6
	500K	78.1
Bitext Data	CCAligned	78.1
	multiCCAligned	79.5
	CCMatrix	80.5
Training Objective	Masked LM	78.4
	ELECTRA	80.5

Table 2: Ablation studies for XY-LENT. We study the effects of changing few parameters in the pre-training setup while keeping others same.

6.2 Ablations

Different Many-to-Many Datasets Table 2 shows the impact of moving from English-centric bitexts to *X-Y* bitext data. Using multiCCAligned dataset gives a +1.4 pt improvement on average XNLI performance over the baseline which uses only the CCAligned data, thereby showing that the utility of leveraging multi-way bitext data is not limited to CCMatrix dataset. However, we still see an additional improvement of 1.0 pt with usage of CCMatrix data and we hypothesize this gain to more diversity present in it which in-turn helps in improving the multilingual representations.

Different Pretraining Objectives While the gains are more substantial with ELECTRA training objective, Table 2 shows that the benefits of having a better quality bitext data is not just restricted to the ELECTRA paradigm of training and can also be observed with the Masked LM objective. For the ablation experiment, we train a base model model with the MLM objective for 125k steps with a batch size of 8192. Comparing this with XLM-R$_{\text{Base}}$'s setup, which uses only monolingual data with MLM objective and trains for 1.5M steps (*i.e. 12 times longer*), finally achieving an XNLI (Avg) of 76.2, we observe that introduction of *X-Y* data not only brings performance gains but also significantly improves the training efficiency of these models.

6.3 On English Performance of Multi-Lingual Models

Given the strong performance of multilingual models on the English subset of XNLI, one interesting question that arises is how does model scaling impact the performance on English centric downstream tasks. In order to evaluate that, we measure the performance of XY-LENT on the commonly used GLUE benchmark (Wang et al., 2018) and the SQuAD 2.0 benchmark. To compare the multilingual model performance on English, we also consider English specific encoder models trained in an Electra pre-training paradigm. Specifically, we consider the Base, Large, XL and XXL models

Model	GLUE DEV Single Task									SQuAD 2.0	
	MNLI-(m/mm) (Acc)	QQP-(Acc/F1)	QNLI (Acc)	SST-2 (Acc)	CoLA (MCC)	RTE (Acc)	MRPC (Acc)	STS-B (SCC)	AVG	EM	F1
METRO-LM$_{\text{Base}}$	90.3 / 90.2	92.4/-	94.4	95.9	71.8	88.1	91.4	92.0	89.5	85.9	88.5
XLM-E$_{\text{Base}}$	86.1 / 86.3	91.5/88.7	92.8	94.0	67.4	77.8	90.2	91.4	86.4	82.3	85.3
XY-LENT$_{\text{Base}}$	87.3 / 87.6	91.9/89.2	93.3	94.4	66.6	85.2	90.7	91.7	87.6	83.5	86.3
Δ	3.0 / 2.6	1.3/-	1.1	1.5	5.2	2.9	0.7	0.7	1.9	2.4	2.2
METRO-LM$_{\text{Large}}$	91.7 / 91.7	92.9	95.8	96.3	75.2	93.1	92.2	92.8	91.4	88.5	91.1
XLM-R$_{\text{Large}}$	88.9 / 89.0	92.3	93.8	95.0	-	-	89.5	91.2	-	-	-
XLM-E$_{\text{Large}}$	89.9 / 90.1	92.9/90.4	94.5	96.8	73.9	85.7	92.1	92.5	89.8	85.6	88.7
XY-LENT$_{\text{Large}}$	89.7 / 89.9	92.7/90.3	94.7	95.8	71.1	88.4	91.4	92.6	89.6	85.8	88.7
Δ	2.0 / 1.8	0.2/-	1.1	0.5	4.1	4.7	0.8	0.2	1.8	2.8	2.4
METRO-LM$_{\text{XL}}$	92.2 / 92.0	93.2	96.3	97.3	76.0	93.5	91.7	93.0	91.8	89.4	92.1
XLM-R$_{\text{XL}}$	90.4/-	92.5	94.9	96.6	-	-	90.4	-	-	-	-
XLM-E$_{\text{XL}}$	91.1 / 91.2	92.5/89.9	94.0	97.2	74.7	91.4	92.1	93.2	90.8	87.8	90.7
XY-LENT$_{\text{XL}}$	91.2 / 91.1	93.0/90.7	95.8	96.4	74.9	92.8	91.9	93.2	91.2	88.1	90.9
Δ	1.0 / 0.9	0.2/-	0.5	0.9	1.1	0.7	-0.2	-0.2	0.6	1.3	1.2

Table 3: Results for models on GLUE dev set and SQuAD 2.0 dev set. Δ represents the performance difference between METRO-LM and XY-LENT which keeps shrinking we scale up.

presented in (Bajaj et al., 2022).

Table 3 shows the performance of our proposed method against the SoTA monolingual as well as other multilingual baselines. As observed in the results, with an increase in the number of parameters, we see that the gap in the performance of an English centric model and a multilingual model decreases, with the XL model being just 0.6 points behind on GLUE and 1.3 points on SQuAD 2.0. We hypothesize that an increase in model capacity alleviates the issues caused by the curse of multilinguality (Conneau et al., 2020); and when that is the case, English performance actually benefits from the presence of other languages in the training data.

It is noteworthy that the even for the English language performance, having an *X-Y* centric data is more beneficial compared to an *EN-X* data (XLM-E vs XY-LENT). Furthermore, our proposed method outperforms XLM-R on large and XL sizes.

6.4 Performance Across Language Families

Figure 3 shows the performance the delta of performance between XLM-E and XY-LENT across different language families. Following Hu et al. (2020), we use the number of Wikipedia articles as a proxy for a language family being high or low resource. As can be seen, leveraging X-Y bitexts helps improves performance consistently across language families.

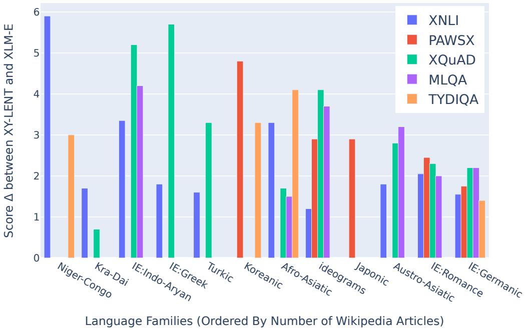


Figure 3: Performance Δ between XLM-E and XY-LENT across language families

Model	XQuAD	MLQA	TyDiQA	XNLI	PAWS-X
MBERT	25.0	27.5	22.2	16.5	14.1
XLM-R	15.9	20.3	15.2	10.4	11.4
XLM-E	**14.9**	**19.2**	13.1	11.2	8.8
XY-LENT	15.3	19.9	**8.6**	**7.8**	**6.8**

Table 4: Crosslingual Transfer Gap scores on 5 multilingual benchmark tasks. A lower score indicates better cross-lingual transfer. For QnA datasets, this is computed using the EM scores.

6.5 Crosslingual Transfer Gap

In order to further evaluate the cross-lingual transferrability of our model, we follow Hu et al. (2020) and evaluate the cross-lingual transfer gap (the difference between the performance on the English test set and the average test set performance for other languages) for XY-LENT$_{\text{Base}}$. This score indicates how much end task knowledge is not transferred to other languages post fine-tuning, with a smaller gap indicating better transferrability. As seen in Table 4, XY-LENT achieves lower scores on 3 out of 5 tasks, thereby demonstrating strong transferrability.

6.6 Using Training Dynamics to Explore Dataset Quality

So far we have seen that leveraging *X-Y* aligned bitexts improves model quality. In this section, we consider the inverse direction: whether training dynamics of representation learning models can be used to identify dataset artifacts. Given these bitext datasets span over 1000 language pairs, a manual inspection of these datasets is extremely hard. Thus an automated method for spot-checking the dataset quality is quite valuable.

To do so, we first train a model in line with the methodology presented by Zhou et al. (2021) for Distributionally Robust Multilingual Machine Translation. Specifically, we train XY-LENT with the following modified objective:

$$\min_{\theta_D,\theta_G} \sup_{\mathbf{P}:\chi^2(\mathbf{P},\mathbb{Q})\leq\rho} \sum_i p_i(\mathcal{L}_D(\mathbf{x};\theta_D)+ \lambda\mathcal{L}_G(\mathbf{x};\theta_G)) \tag{3}$$

Here $\mathcal{L}_G$ and $\mathcal{L}_D$ refer to the generator and discriminator losses respectively (§4), $\mathbf{P}$ is the joint distribution over the bitext language pairs that we want to estimate (i.e $\mathbb{P} = p_i \mid 1 \leq i \leq L \times L; \sum_i p_i = 1$); and $\mathbb{Q}$ is the original training distribution (i.e the probability distribution over the bitexts when the training starts, equal to $\mathbb{P}^*$ as estimated in §3.2). At a high level, the objective minimizes the training loss over a χ^2 ball around the original training distribution, with the supremum up-weighing language pairs with higher loss values, and down-weighing languages with lower loss values [5]. We train a model with the Distributional Robustness Optimization objective (DRO) using Iterated Best Response strategy, as proposed by Zhou et al. (2021) and resample 10 times throughout the training. We hypothesize that the two extremities (i.e language pairs that are highly upsampled as well as those that are downsampled) would be bitext datasets of interest for spot-checking.

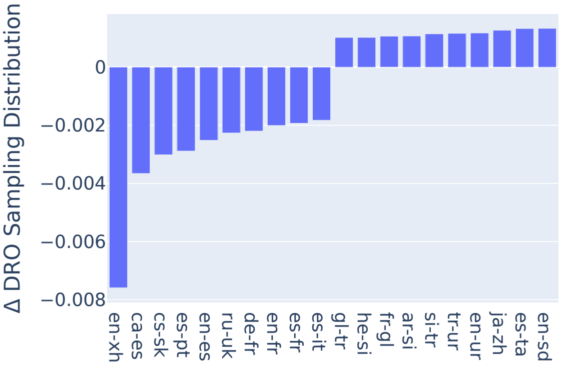


Figure 4: The 10 upsampled and downsampled languages when the model is trained with the DRO objective

Figure 4 presents the top 10 upsampled and 10 downsampled languages between the initial and final language distributions. Manual inspection of these language pairs shows that our hypothesis indeed holds true: we observe that the translations for English and Xhosa (en - xh) are extremely noisy and aligned with non-sensical text, with multiple different English sentences being aligned to the same Xhosa sentence. This can potentially be a manifestation of the hubness issue for Nearest Neighbor lookups in high dimensional spaces (Radovanović et al., 2010; Dinu and Baroni, 2014). Bitexts for Catalan and Spanish (ca - es) and Czech and Slovak (cs - sk) are near duplicates, since the language pairs are very similar. Both of these issues can cause the TRD task to be trivial, explaining the downsampling. Similarly, looking at languages that are up-sampled, we observe a lot of translation quality noise in bitexts for Spanish and Tamil (es - ta), Turkish and Urdu (tr - ur) and Sinhala and Turkish (si - tr).

7 Conclusion

In this work, we introduced a family of models which achieve SoTA performance over 5 multilingual benchmarks compared to other models belonging to similar model size bands and are competitive across the bands. Our XY-LENT$_{\text{XL}}$ model outperforms XLM-R$_{\text{XXL}}$ and is competitive with mT5$_{\text{XXL}}$ being 5x and 6x smaller respectively. Furthermore, the XL model variant also achieves 99.3% and 98.5% of the current best performant models on GLUE and SQuAD 2.0 respectively, thereby aiding in reducing the curse of multilinguality. The

[5] Table 11 in the Appendix shows that such an approach achieves reasonable performance on XNLI.

performance gains are consistent across language families.

8 Limitations

Even though XY-LENT paves the way towards better general-purpose multilingual representation foundation models, in this section, we highlight the limitations associated with this work. We first expound upon the limitations associated with self-supervised learning on large web extracted corpora. Then we show that while XY-LENT achieves strong performance on multiple multilingual benchmarks, when the downstream task involves unseen (during pretraining) languages, the performance drops by a substantial margin. Finally, we show the potential limitation associated with a common methodology used for domain adaptation associated with leveraging these multilingual foundation models, illustrating how catastrophic forgetting exacerbates certail issues pertaining to low resource language performance.

Training Data

XY-LENT uses CC-100 which a static multilingual corpus extracted from Common Crawl for 100 languages. As noted by Wenzek et al. (2020), several data filtering strategies have been applied to remove duplicated documents, paragraphs with high ratio of punctuations, digits and profanities, the resultant data may still result in many potential biases requiring further analysis. Additionally, these issues might be aggravated for models that leverage bitext data, since the bitexts themselves are mined from web crawls, and thus potentially have all the associated biases, stereotypes and other associated harms. Furthermore, the raw data was compiled from static Common Crawl snapshots from January, 2020 to December, 2020 and hence may not include information about some of the recent events such as COVID-19.

Performance on Unseen Languages

Given the performance improvements observed with scaling, we investigate how it impacts extremely low resource languages which are not present in the pre-training data. In order to do so, we consider our model's performance on the AmericasNLI dataset (Ebrahimi et al., 2022) which extends the XNLI dataset to 10 Indigenous languages of the Americas.

Table 5 presents the results on the AmericasNLI dataset. As can be seen, XY-LENT does outperform XLM-R, indicating that better representation learning also benefits these extremely low resource languages. However, we do not see an increase in performance while scaling our models. Specifically, the performance of XY-LENT$_{\text{Base}}$ and XY-LENT$_{\text{XL}}$ model is nearly the same, and substantially worse that the performance observed on the XNLI dataset. This indicates that, while parameter scaling can help improve performance on languages that the model has seen during pre-training, it does not automatically improve performance in the extremely low-resource regime [6]. Thus, while model scaling allows for improvements across numerous dimensions, it is far from a panacea, especially if not done in conjunction with data scaling efforts. To be able to improve performance for unseen languages, an intervention would need to be made at the data collection efforts during pretraining, which we aim to assess in future works.

Continued Training for Domain Adaptation in Pre-Trained Encoders

In recent years, continued training on domain specific corpora has been considered a viable approach for domain adaptation of MLM style pre-trained models (Gururangan et al., 2020; Yao et al., 2021) where the core idea is to continue train the pretrained model on domain specific corpora with the goal of improving in-domain downstream evaluation.

We first show that this phenomenon can be extended to models pretrained with an ELECTRA style training objective. Concretely, we apply domain adaptation in the biomedical domain where we continue to train our XY-LENT$_{\text{Base}}$ as well as XY-LENT$_{\text{MLM + TLM}}$ model on the PubMed data presented in Yao et al. (2021), and evaluate it on the ChemProt task (which aims at extracting relations between chemicals and proteins) presented in Gururangan et al. (2020) as the in-domain downstream task.

We observe that the continued training approach presented in Gururangan et al. (2020) for the ELECTRA style models, using the same peak learning rate as used during pre-training, results in divergence. Interestingly, this neither happens for the generator of the ELECTRA model nor for the

[6] Note that since the tokenizer is a sentencepiece tokenzier, there are extremely few UNK words in the low-resource languages. Consequently, the poor performance is not explained by excessive occurrences of UNK tokens

Model	Avg	Avg w/o en	en	aym	bzd	cni	gn	hch	nah	oto	quy	shp	tar
XLM-R$_{\text{Base}}$	39.4	38.5	85.8	36.1	39.7	37.9	39.5	37.2	42.6	37.8	37.2	40.5	36.4
XLM-E$_{\text{Base}}$	44.8	40.6	87.5	40	38.8	41.7	43.6	38	43.8	39.8	41.7	42.3	35.9
XY-LENT$_{\text{Base}}$	45.5	41.6	84.4	40.7	40.7	42.9	42.5	38.9	45.5	40.9	42.1	43.9	37.6
XY-LENT$_{\text{XL}}$	47.2	42.8	90.8	42.1	42.5	45.6	42.9	41.2	45.0	41.3	42.7	46.9	37.9

Table 5: Performance of models on the AmericasNLI dataset. Note that model scaling does not seem to improve performance as much for these unseen languages.

Model	CT	Avg	en	fr	es	de	el	bg	ru	tr	ar	vi	th	zh	hi	sw	ur
XY-LENT MLM + TLM	✗	78.4	86.2	81.5	82.9	81.5	80.6	81.8	79.8	77.4	77.9	78.3	75.2	78.3	73.3	72.8	68.0
	✓	67.5	85.7	73.9	74.5	71.4	71.7	70.9	71.2	57.8	65.1	66.7	68.5	71.8	60.6	45.8	57.1
	Relative Δ(%)	13.9	0.6	9.3	10.1	12.4	11.0	13.3	10.8	25.3	16.4	14.8	8.9	8.3	17.3	37.1	16.0
	✓ w/ low LR	73.5	85.9	78.2	78.6	76.2	76.8	77	76	68	72.2	73.7	73.8	76.2	68.7	57.4	64.4
	Relative Δ(%)	6.3	0.3	4.0	5.2	6.5	4.7	5.9	4.8	12.1	7.3	5.9	1.9	2.7	6.3	21.2	5.3
XY-LENT$_{\text{Base}}$	✗	80.3	87.9	83.4	84.4	82.9	82.6	83.1	81.1	79.5	79.5	80.0	77.7	80.1	76.4	75.3	71.3
	✓	75.6	87.5	80.5	81.6	78.2	79.3	79.4	76.8	72.4	74.2	76.4	75.3	78.7	70.3	58.8	65.1
	Relative Δ(%)	5.9	0.5	3.5	3.3	5.7	4.0	4.5	5.3	8.9	6.7	4.5	3.1	1.7	8.0	21.9	8.7

Table 6: Drop in cross-lingual zero-shot performance before and after continued training (CT). For MLM, we show with original LR and lower LR. Δ measured as a relative (%) drop compared to no CT

Model	Acc. (w/o Contd. Train)	Acc. (Contd. Train)
XY-LENT MLM + TLM	82.0	86.0
XY-LENT$_{\text{Base}}$	81.6	86.2

Table 7: Domain Specific Downstream task: Accuracy on Chemprot dataset

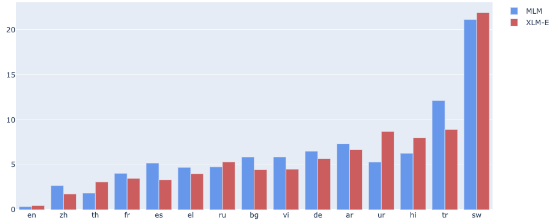


Figure 5: Relative Zero-Shot performance drop with continued training for MLM and ELECTRA style models

MLM style pre-trained model. Thus, for an ELECTRA style continued training setup, we posit reducing the peak learning rate to be a crucial change. Table 7 shows the performance on the downstream task post the continued training approach and unsurprisingly it helps with improving in-domain performance.

However, given the multilingual nature of such models, we test the multilinguality of these models before and after continued training; using cross-lingual zero-shot XNLI as a proxy for multilingual model quality. Table 6 shows the drop in performance across all languages pre and post continued training. We first note that this drop in performance is present for both MLM and ELECTRA style of models, and thus is not an artifact of the pre-training objective. We observe that the drop in performance is not uniform across all languages and the drop is worse for MLM style models (with using the same peak learning rate suffering more from this issue; Table 7). While we expect the drop in English performance to be relatively less, we do see that the drop is substantially more for the mid and low resource languages (especially Hindi, Turkish, Urdu and Swahili; see Fig. 5). While this can potentially be ameliorated by using techniques like Adapters (Houlsby et al., 2019) etc., we would like to draw attention towards the fact that general purpose continued training does suffer from this issue.

References

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. 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 3874–3884, Minneapolis, Minnesota. Association for Computational Linguistics.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2019. On the cross-lingual transferability of monolingual representations. *arXiv preprint arXiv:1910.11856*.

Payal Bajaj, Chenyan Xiong, Guolin Ke, Xiaodong Liu, Di He, Saurabh Tiwary, Tie-Yan Liu, Paul Bennett, Xia Song, and Jianfeng Gao. 2022. Metro: Efficient denoising pretraining of large scale autoencoding language models with model generated signals. *arXiv preprint arXiv:2204.06644*.

Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. 2021. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.

Daniel Cer, Mona Diab, Eneko Agirre, Inigo Lopez-Gazpio, and Lucia Specia. 2017. Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation. *arXiv preprint arXiv:1708.00055*.

Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. 2021a. 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, Li Dong, Bo Zheng, Shaohan Huang, Xian-Ling Mao, Heyan Huang, and Furu Wei. 2021b. Improving pretrained cross-lingual language models via self-labeled word alignment. 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 3418–3430, 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.

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

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020b. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. *Advances in neural information processing systems*, 32.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485, Brussels, Belgium. Association for Computational Linguistics.

Marco Cuturi. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in Neural Information Processing Systems*, 26:2292–2300.

Jacob Devlin. 2018. Multilingual BERT README document. `https://github.com/google-research/bert/blob/a9ba4b8d7704c1ae18d1b28c56c0430d41407eb1/multilingual.md`.

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.

Georgiana Dinu and Marco Baroni. 2014. Improving zero-shot learning by mitigating the hubness problem. volume abs/1412.6568.

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*.

Abteen Ebrahimi, Manuel Mager, Arturo Oncevay, Vishrav Chaudhary, Luis Chiruzzo, Angela Fan, John Ortega, Ricardo Ramos, Annette Rios, Ivan Vladimir Meza Ruiz, Gustavo Giménez-Lugo, Elisabeth Mager, Graham Neubig, Alexis Palmer, Rolando Coto-Solano, Thang Vu, and Katharina Kann. 2022. AmericasNLI: Evaluating zero-shot natural language understanding of pretrained multilingual models in truly low-resource languages. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6279–6299, Dublin, Ireland. Association for Computational Linguistics.

Ahmed El-Kishky, Vishrav Chaudhary, Francisco Guzmán, and Philipp Koehn. 2020. CCAligned: A massive collection of cross-lingual web-document pairs. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020)*, pages 5960–5969. Association for Computational Linguistics.

Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, et al. 2021. Beyond english-centric multilingual machine translation. *Journal of Machine Learning Research*, 22(107):1–48.

Naman Goyal, Jingfei Du, Myle Ott, Giri Anantharaman, and Alexis Conneau. 2021. Larger-scale transformers for multilingual masked language modeling. *arXiv preprint arXiv:2105.00572*.

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

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799. PMLR.

Junjie Hu, Melvin Johnson, Orhan Firat, Aditya Siddhant, and Graham Neubig. 2021. Explicit alignment objectives for multilingual bidirectional encoders. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3633–3643, Online. 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 *International Conference on Machine Learning*, pages 4411–4421. PMLR.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

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.

Patrick Lewis, Barlas Oğuz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. 2019. Mlqa: Evaluating cross-lingual extractive question answering. *arXiv preprint arXiv:1910.07475*.

Yu Meng, Chenyan Xiong, Payal Bajaj, Paul Bennett, Jiawei Han, Xia Song, et al. 2021. Coco-lm: Correcting and contrasting text sequences for language model pretraining. *Advances in Neural Information Processing Systems*, 34:23102–23114.

Miloš Radovanović, Alexandros Nanopoulos, and Mirjana Ivanović. 2010. Hubs in space: Popular nearest neighbors in high-dimensional data. volume 11, pages 2487–2531.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018a. Know what you don't know: Unanswerable questions for squad. *ACL*.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018b. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

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

Holger Schwenk, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. 2021a. 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.

Holger Schwenk, Guillaume Wenzek, Sergey Edunov, Edouard Grave, Armand Joulin, and Angela Fan. 2021b. CCMatrix: Mining billions of high-quality parallel sentences on the web. 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 6490–6500, Online. Association for Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *EMNLP*, pages 1631–1642.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Alex Warstadt, Amanpreet Singh, and Samuel R Bowman. 2018. Neural network acceptability judgments. *arXiv preprint arXiv:1805.12471*.

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

Adina Williams, Nikita Nangia, and Samuel R Bowman. 2017. A broad-coverage challenge corpus for sentence understanding through inference. *Proceedings of the 2nd Workshop on Evaluating Vector-Space Representations for NLP*.

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.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. PAWS-X: A cross-lingual adversarial dataset for paraphrase identification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3687–3692, Hong Kong, China. Association for Computational Linguistics.

Yunzhi Yao, Shaohan Huang, Wenhui Wang, Li Dong, and Furu Wei. 2021. Adapt-and-distill: Developing small, fast and effective pretrained language models for domains. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 460–470, Online. Association for Computational Linguistics.

Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. 2020. Improving massively multilingual neural machine translation and zero-shot translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1628–1639, Online. Association for Computational Linguistics.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. PAWS: Paraphrase adversaries from word scrambling. 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 1298–1308, Minneapolis, Minnesota. Association for Computational Linguistics.

Bo Zheng, Li Dong, Shaohan Huang, Saksham Singhal, Wanxiang Che, Ting Liu, Xia Song, and Furu Wei. 2021. Allocating large vocabulary capacity for cross-lingual language model pre-training. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3203–3215, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Chunting Zhou, Daniel Levy, Xian Li, Marjan Ghazvininejad, and Graham Neubig. 2021. Distributionally robust multilingual machine translation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5664–5674, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Appendix

A Pre-Training and Model Hyperparameters

Hyperparameters	Base	Large	XL
Layers	12	24	48
Hidden size	768	1,024	1,536
FFN inner hidden size	3,072	4,096	6,144
Attention heads	12	16	24

Table 8: Model hyperparameters of XY-LENT discriminators across different sizes.

Hyperparameters	Base	Large	XL
Training steps	125K	500K	150K
Batch tokens per task	4M	1M	4M
Adam ϵ	1e-6	1e-6	1e-6
Adam β	(0.9, 0.98)	(0.9, 0.98)	(0.9, 0.98)
Learning rate	8e-4	2e-4	1e-4
Learning rate schedule	Linear	Linear	Linear
Warmup steps	10,000	10,000	10,000
Gradient clipping	2.0	1.0	1.0
Weight decay	0.01	0.01	0.01

Table 9: Hyperparameters used for pre-training XY-LENT.

Table 8 shows the hyper-parameters of XY-LENT across various model sizes. All the models are trained with a vocabulary size of 500K and we use batch size of 1M or 4M tokens based on model size as mentioned in Table 9. For multilingual replace token detection task we work with a fixed input sequence length of 512 and hence maintains a constant batch size. For translation replace token detection task, the input sequence length is dynamically set as the length of original translation pair and the max one is chosen across the batch. For the base and large models, we train on 128 Nvidia A100-40GB GPU cards, and for the XL model, we use 512 Nvidia A100-80GB GPU cards.

B Downstream Performance

For evaluating cross lingual understanding, we consider five multilingual evaluation benchmarks. We use XNLI (Cross-Lingual Natural Language Inference) and PAWS-X for classification and XQuAD, MLQA and TyDiQA-GP for question answering. Additionally, we use GLUE benchmark and SQuAD2.0 to evaluate the English performance of our model.

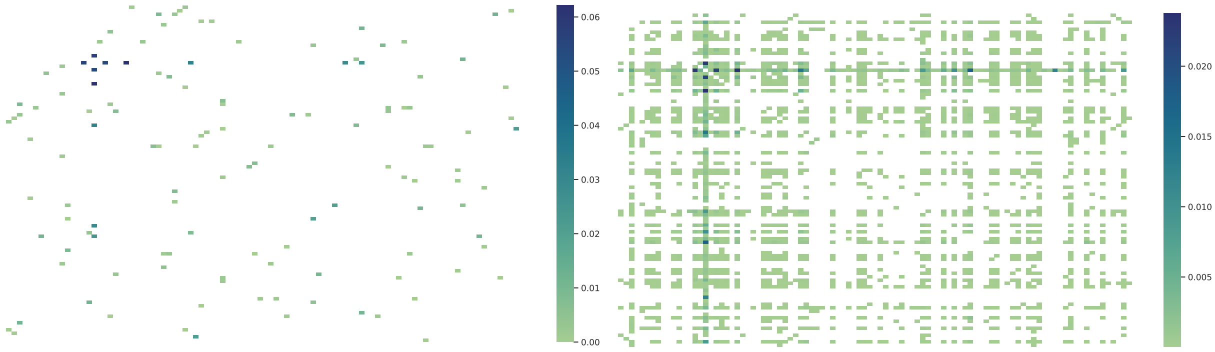
XNLI The XNLI dataset (Conneau et al., 2018) comes with ground-truth dev and test sets in 15 languages, and a ground-truth English training set. The training set has been machine-translated to the remaining 14 languages, providing synthetic training data for these languages as well. We evaluate our model on cross-lingual transfer from English to other languages in two modes: (i)*zero-shot*: the model is fine-tuned only using the English training data and (ii) *translate-train-all*: the English training set is machine-translated to each language and we fine-tune a multilingual model on all training sets. For translations, we use the original XNLI data for consistency.

PAWS-X The PAWS (Paraphrase Adversaries from Word Scrambling) dataset (Zhang et al., 2019) requires to determine whether two sentences are paraphrases. We use the subset of the PAWS dev and test sets translated to six other languages by professional translators, dubbed as PAWS-X (Yang et al., 2019) for evaluation, while using the PAWS set for training.

XQuAD The English SQuAD v1.1(Rajpurkar et al., 2016) requires identifying the answer to a question as a span in the corresponding paragraph. In XQuAD(Artetxe et al., 2019), a subset of the English dev set was translated into ten other languages by professional translators which is then used for evaluation.

MLQA The Multilingual Question Answering(Lewis et al., 2019) dataset is another cross-lingual question answering dataset. In this dataset, the evaluation data for English and six other languages was obtained by automatically mining target language sentences that are parallel to sentences in English from Wikipedia, crowd-sourcing annotations in English, and translating the question and aligning the answer spans in the target languages. We use the SQuAD v1.1(Rajpurkar et al., 2016) training data for training and evaluate on the test data of the corresponding task.

TyDiQA-GP We use the gold passage version of the Typologically Diverse Question Answering(Clark et al., 2020a) dataset, a benchmark for information-seeking question answering, which covers nine languages. The gold passage version is a simplified version of the primary task, which uses only the gold passage as context and excludes unanswerable questions. It is thus similar to XQuAD and MLQA, while being more challenging as questions have been written without seeing the answers,



(a) M2M 100 Sampling

(b) Proposed Sampling

Figure 6: Density plots for our probability distributions obtained for sampling strategies for M2M 100 vs our proposed strategy for all languages in the training set.

leading to 3× and 2× less lexical overlap compared to XQuAD and MLQA respectively. We use the English training data for training and evaluate on the test sets of the target languages.

GLUE and SQuAD 2.0 We evaluate English performance of our model on the GLUE benchmark (Wang et al., 2018) which is a benchmark of eight diverse NLU tasks spanning over single-sentence tasks (CoLA, SST-2), similarity and paraphrase tasks (MRPC, STS-B, QQP) and inference tasks (RTE, MNLI, QNLI). The benchmark is also varied in terms of the training data sizes across tasks which makes it an effective benchmark for testing NLU capabilities of a pretrained model in a robust fashion. We also evaluate the English performance on SQuAD 2.0 (Rajpurkar et al., 2018b) task which is a collection of 100k crowdsourced question/answer pairs collected from Wikipedia where given a passage and a question, the task is to predict the answer span in the passage. The task also has the possibility that no answer exists, making the problem more grounded.

C Sampling Sparsity Across All Language Pairs

Figure 6 shows the sampling distribution as induced by the M2M sampling method and by our proposed method for all language pair directions. Our proposed method induces a much less sparse distribution, resulting in less data wastage.

D Detailed Performance on All Tasks and Languages

We present the detailed results associated with all tasks and languages in this section.

Model	**Avg**	**en**	**de**	**es**	**fr**	**ja**	**ko**	**zh**
Zero-shot Crosslingual Transfer								
Base								
mT5	86.4	95.4	89.4	89.6	91.2	79.8	78.5	81.1
XLM-E	87.0	94.9	89.4	90.3	90.5	81.1	78.9	83.8
XY-LENT	89.7	95.5	92.3	92.5	93.2	84	83.7	86.7
Large								
mT5	88.9	96.1	91.3	92	92.7	82.5	82.7	84.7
XLM-R	86.4	94.7	89.7	90.1	90.4	78.7	79	82.3
XLM-E	89.0	95.9	91.3	91.7	92.4	82.9	82.5	86.4
XY-LENT	90.4	96.5	92.7	93.2	93.6	84.6	84.6	87.4
XL								
mT5	89.6	96	92.8	92.7	92.4	83.6	83.1	86.5
XLM-E	90.3	95.9	93.2	93.1	92.9	84.8	84.7	87.4
XY-LENT	91.0	95.9	92.7	93.2	93.7	86.9	87.0	87.8
XXL								
mT5	90	96.3	92.9	92.6	92.7	84.5	83.9	87.2
Translate-Train								
Base								
mT5	89.3	95.5	90.9	91.4	92.5	83.6	84.8	86.4
XLM-E	91.1	95.7	93.1	92.8	93.3	86.6	87.8	88.7
XY-LENT	91.8	96.2	93.6	93.6	94.2	87.2	89	89.1
Large								
mT5	91.2	96.4	92.7	93.3	93.6	86.5	87.4	88.4
XLM-E	91.9	96.0	93.6	93.4	94.2	87.8	89.2	89.0
XY-LENT	92.4	96.7	94.9	94.1	94.3	87.3	89.7	89.5
XL								
mT5	91.0	96.4	92.5	93.1	93.6	85.5	86.9	89.0
XLM-E	92.2	96.1	93.9	93.6	94.9	88.1	89.4	89.3
XY-LENT	92.6	97.1	94.2	94.6	95.3	88.4	88.8	89.8
XXL								
mT5	91.5	96.1	92.9	93.6	94.2	87	87.9	89.0

Table 10: PAWS-X accuracy scores for each language

Model	**# Params**	**Avg**	**en**	**fr**	**es**	**de**	**el**	**bg**	**ru**	**tr**	**ar**	**vi**	**th**	**zh**	**hi**	**sw**	**ur**
Cross-lingual zero-shot transfer (models fine-tune on English data only)																	
Base																	
mT5	580M	75.4	84.7	79.1	80.3	77.4	77.1	78.6	77.1	72.8	73.3	74.2	73.2	74.1	70.8	69.4	68.3
XLM-R	225M	76.2	85.8	79.7	80.7	78.7	77.5	79.6	78.1	74.2	73.8	76.5	74.6	76.7	72.4	66.5	68.3
XLM-E	477M	78.1	87.3	81.9	82.4	81	80.2	81.1	79.7	77.7	76.4	78.5	76.2	79.0	72.7	69.6	68.3
XY-LENT mCCA	477M	79.5	87.8	82.9	83.8	81.5	81.7	81.8	80.8	79.1	79.1	79.8	77.7	79.3	74.6	72.7	69.9
XY-LENT DRO + CCM	477M	79.7	87.3	82.2	83.7	82.6	82.0	82.5	80.2	78.5	79.0	80.1	77.0	80.3	74.4	74.9	70.2
XY-LENT CCM	477M	80.5	87.7	83.7	84.7	83.7	82	83	81.5	79.3	79.7	80.3	77.9	80.2	76.1	75.5	71.6
Large																	
XLM-R$_{\text{Large}}$	550M	80.9	89.1	84.1	85.1	83.9	82.9	84	81.2	79.6	79.8	80.8	78.1	80.2	76.9	73.9	73.8
mT5$_{\text{Large}}$	1.2B	81.1	89.4	84.1	84.2	83.4	83.2	84.1	81.5	80.1	79.8	81	79.4	80.3	77.6	75.4	73.5
XLM-E$_{\text{Large}}$	840M	81.3	89.4	84.7	85.5	84.4	83.5	84.1	81.9	81.3	80.7	81.2	79.2	81.5	76.5	74.1	72.4
XY-LENT$_{\text{Large}}$	814M	83	90.1	86	86.7	85.4	85.7	85.3	83.2	82.6	83.4	82.8	81.0	82.5	78.3	78.1	74.3
XL																	
XLM-R$_{\text{XL}}$	3.5B	82.3	90.7	85.5	86.5	84.6	84	85.2	82.7	81.7	81.6	82.4	79.4	81.7	78.5	75.3	74.3
mT5$_{\text{XL}}$	3.7B	82.9	90.6	85.3	81.3	85.8	85.4	85.4	83.7	82	82.2	81.8	80.9	82.7	80.4	78.6	77.0
XLM-E$_{\text{XL}}$	2.2B	83.7	91.3	86.8	87.4	86.7	85.8	85.9	84.2	83.4	82.7	83.4	80.9	83.1	80.2	77.6	75.7
XY-LENT$_{\text{XL}}$	2.1B	84.8	92.2	87.4	88.7	87.3	87.2	87.3	83.8	84	84.6	85.1	81.9	83.9	81.6	80.5	77.0
XXL																	
XLM-R$_{\text{XXL}}$	10.7B	83.1	91.6	86.2	87.3	87	85.1	85.7	82.5	82	82.5	83	79.5	82.6	79.8	76.2	74.9
mT5$_{\text{XXL}}$	13B	85.0	91.6	86.9	87.8	87.3	87.3	87.7	85.1	83.8	84.5	79.8	81.7	83.6	83.2	80.3	84.6
Translate-train (models fine-tune on English training data plus translations in all target languages)																	
Base																	
mT5$_{\text{Base}}$	300M	75.9	82	77.9	79.1	77.7	78.1	78.5	76.5	74.8	74.4	74.5	75	76	72.2	71.5	70.4
XLM-R$_{\text{Base}}$	225M	79.1	85.4	81.4	82.2	80.3	80.4	81.3	79.7	78.6	77.3	79.7	77.9	80.2	76.1	73.1	73.0
XLM-E$_{\text{Base}}$	477M	81.7	88.2	83.8	84.7	83.9	83.5	84.1	82.6	81.6	81.1	82.6	81.0	82.5	77.8	75.2	73.7
XY-LENT mCCA$_{\text{Base}}$	477M	82.4	88.0	84.7	85.6	84.2	83.8	84.4	83.3	82.1	82.2	82.7	81.4	82.9	79.4	77.3	73.3
XY-LENT CCM$_{\text{Base}}$	477M	82.9	88.7	85.6	86.1	85.3	85.2	85.8	83.1	83.1	82.9	83.3	81.0	83.7	79.6	78.1	72.7
Large																	
mT5$_{\text{Large}}$	1.2B	81.8	88.3	83.8	84.9	84.0	83.7	84.1	82.0	81.0	80.3	81.3	79.9	81.7	79.8	76.4	75.9
XLM-R$_{\text{Large}}$	550M	83.6	89.1	85.1	86.6	85.7	85.3	85.9	83.5	83.2	83.1	83.7	81.5	83.7	81.6	78	78.1
XLM-E$_{\text{Large}}$	840M	84.1	90.1	86.8	87.1	86.0	86.1	86.4	84.8	83.5	83.7	84.4	81.9	84.9	81.2	78.5	76.4
XY-LENT$_{\text{Large}}$	814M	84.9	90.2	87.4	87.9	86.7	87.0	87.4	85.0	84.7	84.8	85.0	83.4	85.0	82.0	80.9	75.9
XL																	
mT5$_{\text{XL}}$	3.7B	84.8	90.9	86.8	87.4	86.8	86.4	86.8	84.9	84.4	84.2	83.9	82.3	84	83.1	81.3	79.4
XLM-R$_{\text{XL}}$	3.5B	85.4	91.1	87.2	88.1	87	87.4	87.8	85.3	85.2	85.3	86.2	83.8	85.3	83.1	79.8	78.2
XLM-E$_{\text{XL}}$	2.2B	85.5	90.9	87.4	88.3	87.4	87.2	87.6	85.1	85.1	85.1	86.1	83.7	85.4	82.5	81.3	78.9
XY-LENT$_{\text{XL}}$	2.1B	87.1	92.2	88.9	89.7	89.1	89.1	89.1	86.2	86.8	87.0	87.3	85.2	86.7	84.5	83.2	80.8
XXL																	
XLM-R$_{\text{XXL}}$	10.7B	86.0	91.5	87.6	88.7	87.8	87.4	88.2	85.6	85.1	85.8	86.3	83.9	85.6	84.6	81.7	80.6
mT5$_{\text{XXL}}$	13B	87.8	92.7	89.1	90	89.8	89.5	89.4	87.6	87.1	87.2	87.5	85.6	86.5	86.5	84.3	83.8

Table 11: XNLI accuracy scores for each language

Model	**en**	**ar**	**de**	**es**	**hi**	**vi**	**zh**	**Avg**
				Base				
mT5	81.7/66.9	57.1/36.9	62.1/43.2	67.1/47.2	55.4/37.9	65.9/44.1	61.6/38.6	64.4/45.0
XLM-E	82.1/69.2	62.4/42.4	65.7/50.7	71.2/53.1	65.12/47.5	69.8/48.8	64.6/41.5	68.7/50.5
XY-LENT	83.1/70.3	63.9/43.9	68.9/54.0	73.3/55.1	69.0/51.7	72.7/52.0	68.0/45.2	71.3/53.2
				Large				
XLM-R	80.6/67.8	63.1/43.5	68.5/53.6	74.1/56.0	69.2/51.6	71.3/50.9	68.0/45.4	70.7/52.7
mT5	84.9/70.7	65.3/44.6	68.9/51.8	73.5/54.1	66.9/47.7	72.5/50.7	66.2/42.0	71.2/51.7
XLM-E	84.1/71.2	66.6/46.3	70.0/54.8	74.7/56.8	71.0/53.3	74.6/53.6	68.8/44.9	72.8/54.4
XY-LENT	85.0/72.3	68.0/47.6	72.1/56.9	75.4/57.1	72.9/54.7	75.4/54.0	71.2/47.6	74.3/55.7
				XL				
mT5	85.5/71.9	68.0/47.4	70.5/54.4	75.2/56.3	70.5/51.0	74.2/52.8	70.5/47.2	73.5/54.4
XLM-R	85.1/72.6	66.7/46.2	70.5/55.5	74.3/56.9	72.2/54.7	74.4/52.9	70.9/48.5	73.4/55.3
XLM-E	85.2/72.6	68.1/47.6	71.1/56.4	75.7/57.4	73.1/55.2	75.4/53.9	71.3/47.7	74.3/55.8
XY-LENT	85.4/72.4	69.0/48.5	73.0/57.7	76.8/58.6	75.0/56.5	76.2/54.7	72.1/48.6	75.4/56.7
				XXL				
XLM-R	85.5/72.4	68.6/48.4	72.7/57.8	75.4/57.6	73.7/55.8	76.0/55.0	71.7/48.9	74.8/56.6
mT5	86.7/73.5	70.7/50.4	74.0/57.8	76.8/58.4	75.6/57.3	76.4/56.0	71.8/48.8	76.0/57.4

Table 12: MLQA results (F1/EM) for each language.

Model	**en**	**ar**	**bn**	**fi**	**id**	**ko**	**ru**	**sw**	**te**	**Avg**
					Base					
mT5	71.8/60.9	67.1/50.4	40.7/22.1	67.0/52.2	71.3/54.5	49.5/37.7	54.9/32.6	60.4/43.9	40.6/31.1	58.1/42.8
XLM-E	71.8/57.7	68.3/50.0	60.6/45.1	68.0/52.6	73.2/56.1	53.2/40.2	63.4/38.4	64.4/48.1	41.3/27.2	62.7/46.2
XY-LENT	73.4/59.1	71.6/54.1	63.7/51.3	66.5/52.3	77.0/63.4	57.2/43.5	68.0/49.0	67.3/51.1	59.4/39.3	67.1/51.5
					Large					
XLM-R	71.5/56.8	67.6/40.4	64.0/47.8	70.5/53.2	77.4/61.9	31.9/10.9	67.0/42.1	66.1/48.1	70.1/43.6	65.1/45.0
mT5	71.6/58.9	60.5/40.4	42.0/23.9	64.6/48.8	67.0/49.2	47.6/37.3	58.9/36.8	65.7/45.3	41.9/29.7	57.8/41.2
XLM-E	74.7/62.0	75.2/57.1	72.9/56.6	69.9/54.9	78.9/66.7	61.4/47.8	68.0/44.9	72.2/56.7	72.8/45.6	71.8/54.7
XY-LENT	75.6/62.0	77.0/59.9	74.6/62.8	74.0/57.5	80.7/67.1	66.4/52.2	69.5/46.3	76.0/61.3	72.2/48.4	74.0/57.5
					XL					
mT5	80.3/70.9	81.7/65.5	74.5/57.5	79.4/65.3	83.5/70.4	70.0/60.5	71.6/47.8	77.3/59.7	77.9/55.8	77.4/61.5
XLM-E	79.1/64.3	78.2/60.3	76.9/64.1	75.0/60.2	84.4/70.3	66.7/54.8	76.4/56.3	78.3/63.7	75.6/51.1	76.7/60.6
XY-LENT	78.2/64.1	79.3/60.8	78.8/67.3	77.7/63.2	84.9/70.6	68.5/56.2	77.0/57.5	79.9/66.3	77.7/53.2	78.0/62.1
					XXL					
mT5	83.7/72.5	82.8/66.0	80.2/63.7	83.3/70.2	85.3/73.3	76.2/64.1	76.6/55.8	81.9/66.1	79.2/58.7	81.0/65.6

Table 13: TYDi QA GP results (F1/EM) for each language.

Model	**en**	**ar**	**de**	**el**	**es**	**hi**	**ru**	**th**	**tr**	**vi**	**zh**	**Avg**
Base												
mT5	84.6/71.7	63.8/44.3	73.8/54.5	59.6/35.6	74.8/56.1	60.3/43.4	57.8/34.7	57.6/45.7	67.9/48.2	70.7/50.3	66.1/54.1	67.0/49.0
XLM-E	84.9/72.9	70.5/54.3	78.9/63.2	75.6/57.8	78.4/60.8	71.2/54.5	75.9/59.7	68.7/58.8	71.6/55.4	75.9/56.4	65.5/56.9	74.3/59.2
XY-LENT	87.2/76.0	72.9/56.0	80.0/64.5	79.6/63.5	81.2/63.1	75.3/59.7	77.7/61.5	70.9/59.5	74.0/58.7	77.4/59.2	69.0/61.	76.8/62.1
Large												
XLM-R	86.5/75.7	68.6/49.0	80.4/63.4	79.8/61.7	82.0/63.9	76.7/59.7	80.1/64.3	74.2/62.8	75.9/59.3	79.1/59.0	59.3/50.0	76.6/60.8
mT5	88.4/77.3	75.2/56.7	80.0/62.9	77.5/57.6	81.8/64.2	73.4/56.6	74.7/56.9	73.4/62.0	76.5/56.3	79.4/60.3	75.9/65.5	77.8/61.5
XLM-E	87.1/75.5	75.1/58.1	82.1/66.0	80.9/64.0	82.5/64.3	77.5/61.3	80.3/63.7	73.4/59.4	76.8/60.8	79.2/59.0	70.5/61.6	78.7/63.1
XY-LENT	88.1/77.4	76.3/59.6	82.6/67.1	82.5/65.1	83.9/66.6	77.9/61.3	80.2/63.6	74.3/63.8	78.5/62.9	80.6/61.6	71.4/64.6	79.7/64.9
XL												
mT5	88.8/78.1	77.4/60.8	80.4/63.5	80.4/61.2	82.7/64.5	76.1/60.3	76.2/58.8	74.2/62.5	77.7/58.4	80.5/60.8	80.5/71.0	79.5/63.6
XLM-R	89.5/79.0	78.4/61.6	81.3/64.1	82.3/63.9	84.6/66.2	78.8/63.2	81.5/65.0	76.0/65.5	73.9/57.9	81.7/61.8	72.3/66.1	80.0/64.9
XLM-E	89.1/79.0	78.5/62.0	82.4/66.9	81.8/65.5	84.3/67.1	79.3/63.4	82.2/66.9	75.4/65.1	78.3/62.5	81.5/62.9	71.6/65.1	80.4/66.0
XY-LENT	89.4/79.2	79.2/62.0	84.1/68.3	83.5/66.1	84.9/66.6	80.4/64.5	82.9/67.1	75.0/61.7	79.5/64.5	83.2/64.1	72.7/65.0	81.3/66.3
XXL												
XLM-R	89.3/79.4	80.1/63.7	82.7/65.8	83.4/65.5	83.8/66.0	80.7/65.4	82.4/65.4	76.6/65.6	76.8/61.7	82.2/63.0	74.1/67.4	81.1/66.3
mT5	90.9/80.1	80.3/62.6	83.1/65.5	83.3/65.5	85.1/68.1	81.7/65.9	79.3/63.6	77.8/66.1	80.2/60.9	83.1/63.6	83.1/73.4	82.5/66.8

Table 14: XQuAD results (F1/EM) for each language.

E Hyperparameters for Fine-Tuning

In Table 15, we report the hyperparameters for fine-tuning XY-LENT on the downstream tasks.

	XQuAD	MLQA	TyDiQA	XNLI	PAWS-X
Batch size	32	32	32	32	32
Learning rate	{2,3,4}e-5	{2,3,4}e-5	{2,3,4}e-5	{5,...,8}e-6	{8,9,10,20}e-6
LR schedule	Linear	Linear	Linear	Linear	Linear
Warmup	10%	10%	10%	12,500 steps	10%
Weight decay	0	0	0	0	0
Epochs	4	{2,3,4}	{10,20,40}	10	10

Table 15: Hyperparameters used for fine-tuning on the downstream tasks.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
8

☑ A2. Did you discuss any potential risks of your work?
8

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

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

B ☒ Did you use or create scientific artifacts?

Left blank.

□ B1. Did you cite the creators of artifacts you used?
No response.

□ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
No response.

□ 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)?
No response.

□ 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?
No response.

□ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
No response.

□ 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.
No response.

C ☑ Did you run computational experiments?

5

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

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?
Appendix A and E

☑ 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?
6

☑ 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.)?
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.