mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer

Linting Xue* **Noah Constant*** **Adam Roberts***
Mihir Kale **Rami Al-Rfou** **Aditya Siddhant** **Aditya Barua** **Colin Raffel**
Google Research

Abstract

The recent "Text-to-Text Transfer Transformer" (T5) leveraged a unified text-to-text format and scale to attain state-of-the-art results on a wide variety of English-language NLP tasks. In this paper, we introduce mT5, a multilingual variant of T5 that was pre-trained on a new Common Crawl-based dataset covering 101 languages. We detail the design and modified training of mT5 and demonstrate its state-of-the-art performance on many multilingual benchmarks. We also describe a simple technique to prevent "accidental translation" in the zero-shot setting, where a generative model chooses to (partially) translate its prediction into the wrong language. All of the code and model checkpoints used in this work are publicly available.[1]

1 Introduction

Current natural language processing (NLP) pipelines often make use of transfer learning, where a model is pre-trained on a data-rich task before being fine-tuned on a downstream task of interest (Ruder et al., 2019). The success of this paradigm is partially thanks to the release of parameter checkpoints for pre-trained models. These checkpoints allow members of the NLP community to quickly attain strong performance on many tasks without needing to perform expensive pre-training themselves. As one example, the pre-trained checkpoints for the "Text-to-Text Transfer Transformer" (T5) model released by Raffel et al. (2020) have been used to achieve state-of-the-art results on many benchmarks (Khashabi et al., 2020; Roberts et al., 2020; Kale, 2020; Izacard and Grave, 2020; Nogueira et al., 2020; Narang et al., 2020, etc.).

Unfortunately, many of these language models were pre-trained solely on English-language text. This significantly limits their use given that roughly 80% of the world population does not speak English (Crystal, 2008). One way the community has addressed this English-centricity has been to release dozens of models, each pre-trained on a single non-English language (Carmo et al., 2020; de Vries et al., 2019; Le et al., 2020; Martin et al., 2020; Delobelle et al., 2020; Malmsten et al., 2020; Nguyen and Tuan Nguyen, 2020; Polignano et al., 2019, etc.). A more general solution is to produce multilingual models that have been pre-trained on a mixture of many languages. Popular models of this type are mBERT (Devlin, 2018), mBART (Liu et al., 2020a), and XLM-R (Conneau et al., 2020), which are multilingual variants of BERT (Devlin et al., 2019), BART (Lewis et al., 2020b), and RoBERTa (Liu et al., 2019), respectively.

In this paper, we continue this tradition by releasing mT5, a multilingual variant of T5. Our goal with mT5 is to produce a massively multilingual model that deviates as little as possible from the recipe used to create T5. As such, mT5 inherits all of the benefits of T5 (described in section 2), such as its general-purpose text-to-text format, its design based on insights from a large-scale empirical study, and its scale. To train mT5, we introduce a multilingual variant of the C4 dataset called mC4. mC4 comprises natural text in 101 languages drawn from the public Common Crawl web scrape. To validate the performance of mT5, we include results on several benchmark datasets, showing state-of-the-art results in many cases. Finally, we characterize a problematic behavior of pre-trained generative multilingual language models in the zero-shot setting, where they erroneously translate part of their prediction into the wrong language. To address this "accidental translation", we describe a simple procedure that involves mixing in unlabeled pre-training data during fine-tuning and demonstrate that it dramatically alleviates this issue. We release our pre-trained models and code

*Equal Contribution. Please direct correspondence to lintingx@google.com, nconstant@google.com, adarob@google.com, and craffel@google.com

[1] https://goo.gle/mt5-code

Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 483–498
June 6–11, 2021. ©2021 Association for Computational Linguistics

so that the community can leverage our work.[1]

2 Background on T5 and C4

In this section, we provide a short overview of T5 and the C4 pre-training dataset. Further details are available in Raffel et al. (2020).

T5 is a pre-trained language model whose primary distinction is its use of a unified "text-to-text" format for all text-based NLP problems. This approach is natural for generative tasks (such as machine translation or abstractive summarization) where the task format requires the model to generate text conditioned on some input. It is more unusual for classification tasks, where T5 is trained to output the literal text of the label (e.g. "positive" or "negative" for sentiment analysis) instead of a class index. The primary advantage of this approach is that it allows the use of exactly the same training objective (teacher-forced maximum-likelihood) for every task, which in practice means that a single set of hyperparameters can be used for effective fine-tuning on any downstream task. Similar unifying frameworks were proposed by Keskar et al. (2019) and McCann et al. (2018). Given the sequence-to-sequence structure of this task format, T5 uses a basic encoder-decoder Transformer architecture as originally proposed by Vaswani et al. (2017). T5 is pre-trained on a masked language modeling "span-corruption" objective, where consecutive spans of input tokens are replaced with a mask token and the model is trained to reconstruct the masked-out tokens.

An additional distinguishing factor of T5 is its scale, with pre-trained model sizes available from 60 million to 11 billion parameters. These models were pre-trained on around 1 trillion tokens of data. Unlabeled data comes from the C4 dataset, which is a collection of about 750GB of English-language text sourced from the public Common Crawl web scrape. C4 includes heuristics to extract only natural language (as opposed to boilerplate and other gibberish) in addition to extensive deduplication. The pre-training objective, model architecture, scaling strategy, and many other design choices for T5 were chosen based on a large-scale empirical study described in detail in Raffel et al. (2020).

3 mC4 and mT5

Our goal in this paper is to create a massively multilingual model that follows T5's recipe as closely as possible. Towards this end, we develop an extended version of the C4 pre-training dataset that covers 101 languages and introduce changes to T5 to better suit this multilinguality.

3.1 mC4

The C4 dataset was explicitly designed to be English only: any page that was not given a probability of at least 99% of being English by `langdetect`[2] was discarded. In contrast, for mC4 we use `cld3`[3] to identify over 100 languages. Since some of these languages are relatively scarce on the internet, we make use of all of the 71 monthly web scrapes released so far by Common Crawl. This is dramatically more source data than was used for C4, for which the April 2019 web scrape alone was enough to provide plenty of English-language data.

An important heuristic filtering step in C4 was the removal of lines that did not end in an English terminal punctuation mark. Since many languages do not use English terminal punctuation marks, we instead apply a "line length filter" that requires pages to contain at least three lines of text with 200 or more characters. Otherwise, we follow C4's filtering by deduplicating lines across documents and removing pages containing bad words.[4] Finally, we detect each page's primary language using `cld3` and remove those with a confidence below 70%.

After these filters are applied, we group the remaining pages by language and include in the corpus all languages with 10,000 or more pages. This produces text in 107 "languages" as defined by `cld3`. However, we note that six of these are just script variants of the same spoken language (e.g. `ru` is Russian in Cyrillic script and `ru-Latn` is Russian in Latin script). A histogram of the page counts for each language is shown in fig. 1. Detailed dataset statistics including per-language token counts are shown in Appendix A.

3.2 mT5

The model architecture and training procedure that we use for mT5 closely follows that of T5. Specifically, we base mT5 on the "T5.1.1" recipe,[5] which improves upon T5 by using GeGLU nonlinearities (Shazeer, 2020), scaling both d_{model} and d_{ff} instead

[2] https://pypi.org/project/langdetect/
[3] https://github.com/google/cld3
[4] https://github.com/LDNOOBW/
[5] https://github.com/google-research/text-to-text-transfer-transformer/blob/master/released_checkpoints.md#t511

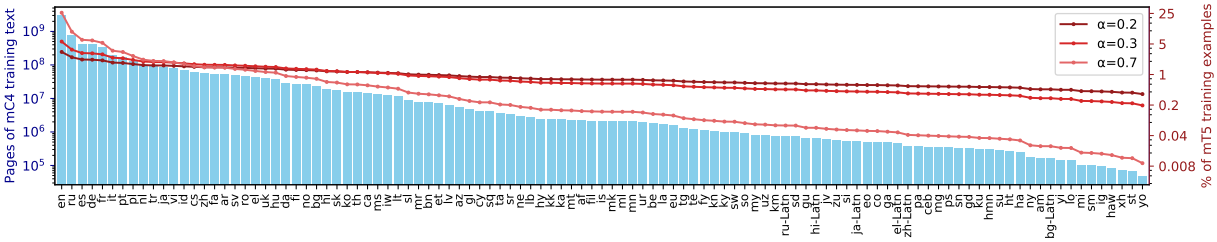


Figure 1: Page counts per language in mC4 (left axis), and percentage of mT5 training examples coming from each language, for different language sampling exponents α (right axis). Our final model uses α=0.3.

Model	Architecture	Parameters	# languages	Data source
mBERT (Devlin, 2018)	Encoder-only	180M	104	Wikipedia
XLM (Conneau and Lample, 2019)	Encoder-only	570M	100	Wikipedia
XLM-R (Conneau et al., 2020)	Encoder-only	270M – 550M	100	Common Crawl (CCNet)
mBART (Lewis et al., 2020b)	Encoder-decoder	680M	25	Common Crawl (CC25)
MARGE (Lewis et al., 2020a)	Encoder-decoder	960M	26	Wikipedia or CC-News
mT5 (ours)	Encoder-decoder	300M – 13B	101	Common Crawl (mC4)

Table 1: Comparison of mT5 to existing massively multilingual pre-trained language models. Multiple versions of XLM and mBERT exist; we refer here to the ones that cover the most languages. Note that XLM-R counts five Romanized variants as separate languages, while we ignore six Romanized variants in the mT5 language count.

of just $\mathrm{d_{ff}}$ in the larger models, and pre-training on unlabeled data only with no dropout. We refer to Raffel et al. (2020) for further details on T5.

A major factor in pre-training multilingual models is how to sample data from each language. Ultimately, this choice is a zero-sum game: If low-resource languages are sampled too often, the model may overfit; if high-resource languages are not trained on enough, the model will underfit. We therefore take the approach used in (Devlin, 2018; Conneau et al., 2020; Arivazhagan et al., 2019) and boost lower-resource languages by sampling examples according to the probability $p(L) \propto |L|^{\alpha}$, where $p(L)$ is the probability of sampling text from a given language during pre-training and $|L|$ is the number of examples in the language. The hyperparameter α (typically with $\alpha < 1$) allows us to control how much to "boost" the probability of training on low-resource languages. Values used by prior work include $\alpha = 0.7$ for mBERT (Devlin, 2018), $\alpha = 0.3$ for XLM-R (Conneau et al., 2020), and $\alpha = 0.2$ for MMNMT (Arivazhagan et al., 2019). We tried all three of these values (ablation results in section 4.2) and found $\alpha = 0.3$ to give a reasonable compromise between performance on high- and low-resource languages.

The fact that our model covers over 100 languages necessitates a larger vocabulary. Following XLM-R (Conneau et al., 2018), we increase the vocabulary size to 250,000 wordpieces. As in T5, we use SentencePiece (Kudo and Richardson, 2018; Kudo, 2018) models trained with the language sampling rates used during pre-training. To accommodate languages with large character sets like Chinese, we use a character coverage of 0.99999 and enable SentencePiece's "byte-fallback" feature to ensure that any string can be uniquely encoded.

3.3 Comparison to Related Models

To contextualize our new model, we provide a brief comparison with existing massively multilingual pre-trained language models. For brevity, we focus on models that support more than a few dozen languages. Table 1 gives a high-level comparison of mT5 to the most similar models.

mBERT (Devlin, 2018) is a multilingual version of BERT (Devlin et al., 2019). Similar to our approach with mT5, mBERT follows the BERT recipe as closely as possible (same architecture, objective, etc.). The primary difference is the training set: Instead of training on English Wikipedia and the Toronto Books Corpus, mBERT is trained on up to 104 languages from Wikipedia. **XLM** (Conneau and Lample, 2019) is also based on BERT but applies improved methods for pre-training multilingual language models including explicitly cross-lingual pre-training objectives. Many pre-trained versions of XLM have been released; the most massively-multilingual variant was trained on 100 languages from Wikipedia. **XLM-R** (Conneau

Model	Sentence pair		Structured	Question answering		
	XNLI	PAWS-X	WikiAnn NER	XQuAD	MLQA	TyDi QA-GoldP
Metrics	Acc.	Acc.	F1	F1 / EM	F1 / EM	F1 / EM
Cross-lingual zero-shot transfer (models fine-tuned on English data only)						
mBERT	65.4	81.9	62.2	64.5 / 49.4	61.4 / 44.2	59.7 / 43.9
XLM	69.1	80.9	61.2	59.8 / 44.3	48.5 / 32.6	43.6 / 29.1
InfoXLM	81.4	-	-	- / -	73.6 / 55.2	- / -
X-STILTs	80.4	87.7	64.7	77.2 / 61.3	72.3 / 53.5	76.0 / 59.5
XLM-R	79.2	86.4	65.4	76.6 / 60.8	71.6 / 53.2	65.1 / 45.0
VECO	79.9	88.7	65.7	77.3 / 61.8	71.7 / 53.2	67.6 / 49.1
RemBERT	80.8	87.5	**70.1**	79.6 / 64.0	73.1 / 55.0	77.0 / 63.0
mT5-Small	67.5	82.4	50.5	58.1 / 42.5	54.6 / 37.1	36.4 / 24.4
mT5-Base	75.4	86.4	55.7	67.0 / 49.0	64.6 / 45.0	59.1 / 42.4
mT5-Large	81.1	88.9	58.5	77.8 / 61.5	71.2 / 51.7	68.4 / 50.9
mT5-XL	82.9	89.6	65.5	79.5 / 63.6	73.5 / 54.5	77.8 / 61.8
mT5-XXL	**85.0**	**90.0**	69.2	**82.5 / 66.8**	**76.0 / 57.4**	**82.0 / 67.3**
Translate-train (models fine-tuned on English data plus translations in all target languages)						
XLM-R	82.6	90.4	-	80.2 / 65.9	72.8 / 54.3	66.5 / 47.7
FILTER + Self-Teaching	83.9	91.4	-	82.4 / 68.0	76.2 / 57.7	68.3 / 50.9
VECO	83.0	91.1	-	79.9 / 66.3	73.1 / 54.9	75.0 / 58.9
mT5-Small	72.0	79.9	-	64.3 / 49.5	56.6 / 38.8	49.8 / 35.6
mT5-Base	79.8	89.3	-	75.3 / 59.7	67.6 / 48.5	66.4 / 51.0
mT5-Large	84.4	91.2	-	81.2 / 65.9	73.9 / 55.2	75.7 / 60.1
mT5-XL	85.3	91.0	-	82.7 / 68.1	75.1 / 56.6	80.1 / 65.0
mT5-XXL	**87.1**	**91.5**	-	**85.2 / 71.3**	**76.9 / 58.3**	**83.3 / 69.4**
In-language multitask (models fine-tuned on gold data in all target languages)						
mBERT	-	-	89.2	-	-	77.6 / 68.0
mT5-Small	-	-	86.4	-	-	74.0 / 62.7
mT5-Base	-	-	88.2	-	-	79.7 / 68.4
mT5-Large	-	-	89.7	-	-	85.3 / 75.3
mT5-XL	-	-	91.3	-	-	87.6 / 78.4
mT5-XXL	-	-	**92.2**	-	-	**88.7 / 79.5**

Table 2: Results on XTREME sentence-pair classification, structured prediction and question answering tasks. mBERT metrics are from Hu et al. (2020). Metrics for XLM, InfoXLM, X-STILTs and XLM-R are from Fang et al. (2020), though Conneau et al. (2020) report better performance of XLM-R on XNLI (80.9). All other metrics are from the original sources: FILTER (Fang et al., 2020), VECO (Luo et al., 2020) and RemBERT (Chung et al., 2020). For the "translate-train" setting, we include English training data, so as to be comparable with Fang et al. (2020) and Luo et al. (2020). This differs from the XTREME "translate-train" setup of Hu et al. (2020). For mT5 results on TyDi QA zero-shot, we report the median across five fine-tuning runs, as we observed high variance across runs.[6] Full results for all languages in all tasks are provided in the appendix.

et al., 2020) is an improved version of XLM based on the RoBERTa model (Liu et al., 2019). XLM-R is trained with a cross-lingual masked language modeling objective on data in 100 languages from Common Crawl. To improve the pre-training data quality, pages from Common Crawl were filtered by an n-gram language model trained on Wikipedia (Wenzek et al., 2020). **mBART** (Liu et al., 2020a) is a multilingual encoder-decoder model that is based on BART (Lewis et al., 2020b). mBART is trained with a combination of span masking and sentence shuffling objectives on a subset of 25 languages from the same data as XLM-R. **MARGE** (Lewis et al., 2020a) is a multilingual encoder-decoder model that is trained to reconstruct a document in one language by retrieving documents in other languages. It uses data in 26 languages from Wikipedia and CC-News (Liu et al., 2019).

4 Experiments

To validate the performance of mT5, we evaluate our models on 6 tasks from the XTREME multilingual benchmark (Hu et al., 2020): the XNLI (Conneau et al., 2018) entailment task covering 14 languages; the XQuAD (Artetxe et al., 2020), MLQA (Lewis et al., 2019), and TyDi QA (Clark et al., 2020) reading comprehension benchmarks with 10,

[6] Standard deviations of mT5 models on TyDi QA zero-shot across five runs are: Small: 0.44, Base: 1.38, Large: 3.66, XL: 1.29, XXL: 0.20.

7, and 11 languages respectively; the Named Entity Recognition (NER) dataset of WikiAnn (Pan et al., 2017) restricted to the 40 languages from XTREME (Hu et al., 2020), and the PAWS-X (Yang et al., 2019) paraphrase identification dataset with 7 languages. We cast all tasks into the text-to-text format, i.e. generating the label text (XNLI and PAWS-X), entity tags and labels (WikiAnn NER), or answer (XQuAD, MLQA, and TyDi QA) directly in a generative fashion. For NER, if there are multiple entities, they are concatenated in the order they appear, and if there are no entities then the target text is "None". We consider three variants of these tasks: (1) "zero-shot", where the model is fine-tuned only on English data, (2) "translate-train", adding machine translations from English into each target language, and (3) "in-language multitask", training on gold data in all target languages. For brevity, we refer to Hu et al. (2020) for further details on these benchmarks.

Following the original T5 recipe, we consider five model sizes: *Small* ($\approx$ 300M parameters), *Base* (580M), *Large* (1.2B), *XL* (3.7B), and *XXL* (13B). The increase in parameter counts compared to the corresponding T5 model variants comes from the larger vocabulary used in mT5. Note that, because mT5 is an encoder-decoder model, it has roughly twice as many parameters as correspondingly-sized encoder-only models such as XLM-R. For example, the "Large" variant of XLM-R has 550 million parameters whereas mT5-Large has around 1 billion. However, the computational cost for text classification is roughly the same: In both cases, the model processes a length-T input sequence with an encoder of approximately equal size. In an encoder-only model like XLM-R, the encoder processes one additional "CLS" token, which is used to generate the representation for classification. In mT5, the decoder typically produces two additional tokens: the class label and an end-of-sequence token. Since the decoder has the same architecture (ignoring encoder-decoder attention) as the encoder, the computational cost of classification with mT5 typically amounts to the cost of processing $T + 2$ tokens compared to $T + 1$ for an encoder-only model. However, encoder-decoder architectures have the additional benefit of being applicable to generative tasks like abstractive summarization or dialog.

We pre-train our mT5 model variants for 1 million steps on batches of 1024 length-1024 input sequences, corresponding to roughly 1 trillion input tokens total. This is the same amount of pre-training as T5 and about $\frac{1}{6}$ as much as XLM-R.[7] Note that our pre-training dataset is large enough that we only complete a fraction of an epoch for high-resource languages (e.g. only covering 2% of the English data). While XLM-R's pre-training corpus CC-100 is 20 times smaller than mC4, XLM-R nevertheless pre-trains for more steps, and sees over 6 times more tokens in pre-training.

We use the same inverse square-root learning rate schedule used by T5 during pre-training, with the learning rate set to $1/\sqrt{\max(n, k)}$ where n is the current training iteration and $k = 10^4$ is the number of warm-up steps. Following the T5.1.1 recipe, we do not apply dropout during pre-training. We use the same self-supervised objective as T5, with 15% of tokens masked and an average noise span length of 3. We ablate some of these experimental details in section 4.2.

For fine-tuning, we use a constant learning rate of 0.001 and dropout rate of 0.1 for all tasks. We use a batch size of 2^{17} for most tasks, but decrease to 2^{16} for WikiAnn NER zero-shot, due to the small size of the training, and increase to 2^{20} tokens for XNLI, which we found gave better performance. For early stopping, we save checkpoints every 200 steps and choose the checkpoint with the highest performance on the standard validation sets specified by XTREME.

4.1 Results

Table 2 presents our main results, with per-language breakdowns for each task given in Appendix B. Our largest model mT5-XXL exceeds state-of-the-art on all classification and QA tasks and is near SOTA on NER (69.2 vs. 70.1). Note that unlike our model, InfoXLM (Chi et al., 2020) and VECO (Luo et al., 2020) benefit from parallel training data, while X-STILTs (Phang et al., 2020) leverages labeled data from tasks similar to the target task. Overall, our results highlight the importance of model capacity in cross-lingual representation learning and suggest that scaling up a simple pre-training recipe can be a viable alternative to more complex techniques relying on LM filtering, parallel data, or intermediate tasks.

In the "translate-train" setting, we exceed state-

[7] XLM-R Large sees 6.3 trillion tokens during pre-training (1.5 million batches of 8192 sequences of 512 tokens), and uses a packing mechanism similar to T5 to minimize the number of "wasted" padding tokens.

	T5	mT5
Small	87.2 / 79.1	84.7 / 76.4
Base	92.1 / 85.4	89.6 / 83.8
Large	93.8 / 86.7	93.0 / 87.0
XL	95.0 / 88.5	94.5 / 88.9
XXL	96.2 / 91.3	95.6 / 90.4

Table 3: Comparison of T5 vs. mT5 on SQuAD question answering (F1/EM).

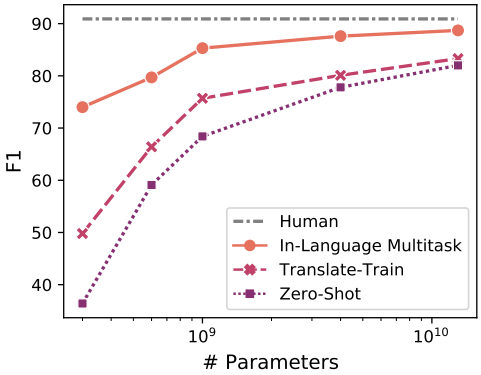


Figure 2: Average F1 on the TyDi QA GoldP task across languages. Performance improves with increasing model capacity. The importance of in-language training data (whether gold In-Language Multitask or synthetic Translate-Train) decreases with model scale, as seen by Zero-Shot closing the quality gap.

of-the-art on all XTREME classification and QA tasks. For these tasks, we fine-tune on the combination of the labeled English data and machine translations thereof.[8] This allows direct comparison with both FILTER (Fang et al., 2020) as well as the XLM-R baseline of Fang et al. (2020). Note that this setup differs from XTREME "translate-train" (Hu et al., 2020), which excludes English.

Figure 2 shows that model capacity is key to improving performance on variants of the TyDi QA GoldP task in the absence of "gold" multilingual data: For the smallest model, training on gold datasets (in-language multitask) achieves dramatically better performance than using weakly supervised data (translate-train) or English-only data (zero-shot), whereas the gap between these three settings is much smaller for the largest model. For our two largest models, zero-shot and translate-train performance is nearly the same, showing that machine translations of the monolingual dataset bring diminishing returns as model capacity increases. Overall, these trends point to the possibility of avoiding the costly step of annotating data in more than one language when using large models.

Massively multilingual models have been observed to underperform on a given language when compared to a similarly-sized "dedicated" model trained specifically for that language (Arivazhagan et al., 2019). To quantify this effect, we compare the performance of mT5 and T5 when fine-tuned on the SQuAD reading comprehension benchmark (Rajpurkar et al., 2016). The results are shown in table 3, with results for T5 reproduced from Raffel et al. (2020). While the *Small* and *Base* mT5 models fall short of their English T5 counterparts, we find that the larger models close the gap. This suggests there may be a turning point past which the model has enough capacity to effectively learn 101 languages without significant interference effects.

Looking at the per-language breakdowns in Appendix B, we find that mT5 performs well on both high- and low-resource languages. For example, in table 7, we see mT5-XXL outperforms XLM-R by between +3 (English) and +9 (Swahili) points on each individual language on XNLI zero-shot. In table 12 we see similarly strong performance across languages on TyDi QA GoldP (including lower-resource languages like Swahili and Telugu), with mT5-XXL surpassing human performance in four of nine languages on the "in-language" setting.

4.2 Ablation

We run six ablations, modifying various settings, using our *Large* model as a baseline: (i) increase dropout to 0.1 in hopes of mitigating overfitting on low-resource languages, (ii) decrease sequence length to 512 (as was used in T5), (iii) increase the average noise span length in the pre-training objective to 10 since we observe fewer characters per token than T5, (iv) adjust the language sampling exponent α to $\{0.2, 0.7\}$ as used in MMNMT (Arivazhagan et al., 2019) and mBERT (Devlin, 2018), respectively, (v) turn off the "line length filter" in the mC4 data pipeline, and (vi) supplement mC4 with Wikipedia data[9] from 103 languages.

The effect of these ablations on XNLI zero-shot accuracy is shown in table 4. In each case, the average XNLI score is lower than the mT5-Large baseline, justifying our chosen settings. The line

[8] We use the translation data provided by Hu et al. (2020) throughout. On the PAWS-X task, FILTER used translation data from the original task instead. Switching to this data would improve our scores slightly (mT5-XXL 91.5 → 92.0).

[9] We use the 2020 Wikipedia data from TensorFlow Datasets, selecting the same languages as mBERT. https://www.tensorflow.org/datasets/catalog/wikipedia

Model	Accuracy
Baseline (mT5-Large)	**81.1**
Dropout 0.1	77.6
Sequence length 512	80.5
Span length 10	78.6
$\alpha = 0.7$	80.7
$\alpha = 0.2$	80.7
No line length filter	79.1
Add Wikipedia data	80.3

Table 4: Average XNLI zero-shot accuracy of various ablations on our mT5-Large model. Per-language metrics are shown in Appendix C.

length filter provides a +2 point boost, corroborating the findings of Conneau et al. (2020) and Raffel et al. (2020) that filtering low-quality pages from Common Crawl is valuable. Increasing the language sampling exponent α to 0.7 has the expected effect of improving performance in high-resource languages (e.g. Russian 81.5 $\rightarrow$ 82.8), while hurting low-resource languages (e.g. Swahili 75.4 $\rightarrow$ 70.6), with the average effect being negative. Conversely, lowering α to 0.2 boosts one tail language slightly (Urdu 73.5 $\rightarrow$ 73.9) but is harmful elsewhere. Detailed per-language metrics on XNLI and the results of our ablations on zero-shot XQuAD are provided in Appendix C, showing similar trends.

5 Zero-Shot Generation

Since mT5 is a generative model, it can output arbitrary text predictions in a free form fashion. This is in contrast to "encoder-only" models like mBERT and XLM(-R) that make a prediction by either extracting it from the input or producing a class label. We found that the lack of constraints during prediction caused mT5 to sometimes have trouble generating a well-formed prediction in a language unseen during fine-tuning. Focusing on XQuAD zero-shot, we find that many of these errors are due to "accidental translation" into the fine-tuning language (English). In this section, we characterize this behavior and demonstrate that it can be counteracted by mixing a small amount of our multilingual pre-training task into the fine-tuning stage.

5.1 Illegal Predictions

In using a generative model for span selection (as in extractive QA tasks), we hope the model learns to generate **"legal" spans** that are substrings of the provided context. However, unlike encoder-based models like BERT, this is not a hard constraint of the model. Notably, T5 learns to always output legal spans on SQuAD, suggesting this is not a major issue for generative models in simple cases.

Target	Prediction	Explanation
จำนวนเฉพาะ	จำนวนเฉพาะ	Decomposed Thai ำ into ◌ + า
लोथर डे माइज़ियर	लोथर डे माइज़ियर	Decomposed Hindi ज़ into ज + ◌़
27 - 30 %	27 - 30 %	Replaced full-width percent sign
12 . ª	12 . a	Removed superscript
للبكتريا اللاهوائية	البكتريا اللاهوائية	Arabic "for anaerobic bacteria" ⇒ "anaerobic bacteria"
строками битов	строки битов	Russian "bit strings (instrumental)" ⇒ "bit strings (nominative)"
seis años	six years	Translated from Spanish
Zweiten Weltkrieg	the Second World War	Translated from German
新英格兰爱国者队	New英格兰爱国者队	Partially translated Chinese "New England Patriots"
хлоропласт	chloroпласт	Partially translated Russian "chloroplast"

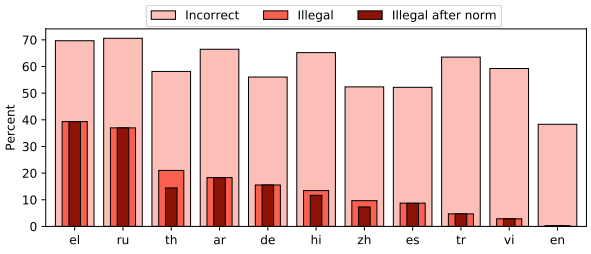
Table 5: Illegal mT5-XXL predictions on XQuAD zero-shot, illustrating normalization (top), grammatical adjustment (middle) and translation (bottom).

A more challenging case for generative models is zero-shot cross-lingual span selection. Here, a pre-trained multilingual model is fine-tuned on English but tested on other languages. We want the model to generate legal non-English predictions despite having only seen English targets in fine-tuning.

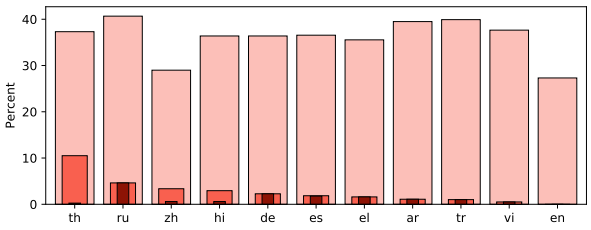
In practice, while mT5 achieves SOTA on the zero-shot variants of XQuAD, MLQA and TyDi QA, illegal predictions are still a problem. For example, on zero-shot XQuAD, a non-trivial portion of mT5 mistakes are in fact illegal spans, for all model sizes (cf. fig. 4 "Baseline"). Through inspection, we find these illegal predictions mainly fall into three categories: (i) normalization, (ii) grammatical adjustment, and (iii) accidental translation. Table 5 provides examples of each type.

Normalization indicates predictions that would be legal, except that "equivalent" Unicode characters have been substituted, so a legal span may be recovered through Unicode NFKC normalization. This is particularly common in Thai, Chinese and Hindi, where most mT5-XXL illegal predictions are resolved by normalization, as seen in fig. 3b.

Grammatical adjustment involves minor morphological changes to the original text. We frequently observe these adjustments when the target span cannot stand as a well-formed answer on its own. For example, mT5-XXL's Arabic and Russian predictions in the middle rows of table 5 are judged by native speakers as correct and grammatical answers to the posed XQuAD questions, while the gold targets are judged as ungrammatical answers. This type of illegal prediction is most common in



(a) mT5-Small



(b) mT5-XXL

Figure 3: Per-language error rates on XQuAD zero-shot, sorted by illegal rate. **Incorrect**: Not matching the target span. **Illegal**: Missing from the input context. **Illegal after norm**: Illegal even after Unicode NFKC normalization is applied to the prediction and context.

languages with extensive grammatical case marking, such as Russian, Turkish and German.

Accidental translation involves the model translating part or all of a contextual span into English (the language of all fine-tuning data). On the one hand, it is remarkable that mT5 performs "spontaneous" translation despite never seeing parallel training data. On the other, as practitioners we would ideally be able to control this behavior.

We observe accidental translation across all model sizes and all XQuAD languages. The problem is most prevalent in mT5-Small and mT5-Base, where from manual inspection, half or more of the illegal predictions within each language exhibit accidental translation, with many of the illegal predictions coming from Greek and Russian, as shown in fig. 3a. While we do observe full phrase translations, a more common occurrence is *partial* translation, where the model outputs a token or two of English before reverting to the correct target language. The transition may even occur mid-word, as in the prediction "chloропласт", where the first half of the target "хлоропласт" (Russian: chloroplast) has been translated to English.

5.2 Preventing Accidental Translation

The most direct solution to avoiding accidental translation on span selection tasks would be to modify our inference procedure. As is common practice with encoder-based models, we could devise a task-specific fine-tuning mechanism that restricts the model to perform ranking over legal spans, removing the possibility of illegal predictions entirely. While this would likely improve our zero-shot metrics, it is unsatisfying for two reasons: First, it implies taking a step backward from the general text-to-text interface, as different tasks would demand different types of inference. Second, this solution won't extend to more "open-ended" zero-shot generative tasks like summarization, where the legal output space can't be easily delimited.

For these reasons, we consider a more general solution that remains within the text-to-text framework and can apply to all zero-shot generation tasks. Our motivating intuition is that the reason the model outputs English when given a non-English test input is that it has never observed a non-English target during fine-tuning. As English-only fine-tuning proceeds, the model's assigned likelihood of non-English tokens presumably decreases, eventually reaching the point where English becomes the most likely answer to any question.

To prevent the model from "forgetting" how to generate other languages, we use a strategy inspired by domain/task-adaptive pre-training (Howard and Ruder, 2018; Gururangan et al., 2020): We simply mix in our unsupervised multilingual pre-training task during fine-tuning. A similar approach was explored by Liu et al. (2020b). We use the same mC4 task definition as in pre-training, with two adjustments: First, we remove all "sentinel" tokens (corresponding to non-masked spans in the input text) from the target sequence, as otherwise we observe occasional sentinels in downstream predictions. Second, we reduce the language sampling parameter α from 0.3 to 0.1. This produces a near-uniform distribution of languages, encouraging the model to treat all languages as equally likely.[10]

With these changes, we mix a small amount of our unsupervised task (covering 101 languages) into XQuAD fine-tuning, at a ratio of just 1:100. Figure 4 shows the results on XQuAD zero-shot error rates. The addition of even this small amount of multilingual data has a marked effect on the mT5-Small and mT5-Base models (where accidental translation was most rampant), reducing the illegal prediction rates by more than 70% (relative), and contributing to an overall reduction in errors.

[10] Alternatively, one could mix in unlabeled data only for a single language at a time. However, we believe this is contrary

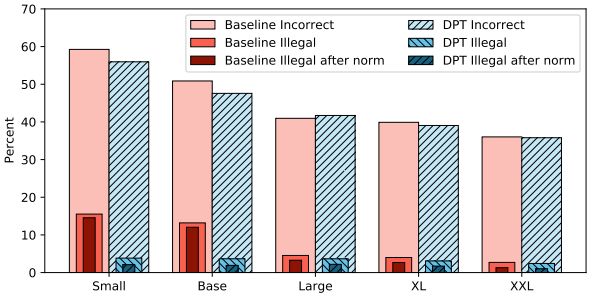


Figure 4: Error rates of mT5 on XQuAD zero-shot. **Baseline**: Fine-tuning on XQuAD alone. **Domain Preserving Training (DPT)**: Mixing in the unsupervised mC4 task with fine-tuning.

6 Conclusion

In this paper, we introduced mT5 and mC4: massively multilingual variants of the T5 model and C4 dataset. We demonstrated that the T5 recipe is straightforwardly applicable to the multilingual setting, and achieved strong performance on a diverse set of benchmarks. We also characterized illegal predictions that can occur in zero-shot evaluation of multilingual pre-trained generative models, and described a simple technique to avoid this issue. We release all code and pre-trained datasets used in this paper to facilitate future work on multilingual language understanding.[11]

Acknowledgements

We thank Melvin Johnson for tips on the translate-train procedure for XTREME and Itai Rolnick for help with infrastructure.

References

Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Dmitry Lepikhin, Melvin Johnson, Maxim Krikun, Mia Xu Chen, Yuan Cao, George Foster, Colin Cherry, et al. 2019. Massively multilingual neural machine translation in the wild: Findings and challenges. *arXiv preprint arXiv:1907.05019*.

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

Diedre Carmo, Marcos Piau, Israel Campiotti, Rodrigo Nogueira, and Roberto Lotufo. 2020. PTT5: Pretraining and validating the t5 model on brazilian portuguese data. *arXiv preprint arXiv:2008.09144*.

to the spirit of multilingual models and zero-shot evaluation.

[11] https://goo.gle/mt5-code

Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. 2020. InfoXLM: An information-theoretic framework for cross-lingual language model pre-training. *arXiv preprint arXiv:2007.07834*.

Hyung Won Chung, Thibault Févry, Henry Tsai, Melvin Johnson, and Sebastian Ruder. 2020. Rethinking embedding coupling in pre-trained language models. *arXiv preprint arXiv:2010.12821*.

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

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

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, volume 32, pages 7059–7069.

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.

David Crystal. 2008. Two thousand million? *English today*, 24(1):3–6.

Wietse de Vries, Andreas van Cranenburgh, Arianna Bisazza, Tommaso Caselli, Gertjan van Noord, and Malvina Nissim. 2019. BERTje: A Dutch BERT model. *arXiv preprint arXiv:1912.09582*.

Pieter Delobelle, Thomas Winters, and Bettina Berendt. 2020. RobBERT: a dutch RoBERTa-based language model. *arXiv preprint arXiv:2001.06286*.

Jacob Devlin. 2018. Multilingual BERT README. https://github.com/google-research/bert/blob/master/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.

Yuwei Fang, Shuohang Wang, Zhe Gan, Siqi Sun, and Jingjing Liu. 2020. FILTER: An enhanced fusion method for cross-lingual language understanding. *arXiv preprint arXiv:2009.05166*.

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.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. 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 generalization. *arXiv preprint arXiv:2003.11080*.

Gautier Izacard and Edouard Grave. 2020. Leveraging passage retrieval with generative models for open domain question answering. *arXiv preprint arXiv:2007.01282*.

Mihir Kale. 2020. Text-to-text pre-training for data-to-text tasks. *arXiv preprint arXiv:2005.10433*.

Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Unifying question answering and text classification via span extraction. *arXiv preprint arXiv:1904.09286*.

Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. 2020. UnifiedQA: Crossing format boundaries with a single QA system. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1896–1907, Online. Association for Computational Linguistics.

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.

Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoit Crabbé, Laurent Besacier, and Didier Schwab. 2020. FlauBERT: Unsupervised language model pre-training for French. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2479–2490, Marseille, France. European Language Resources Association.

Mike Lewis, Marjan Ghazvininejad, Gargi Ghosh, Armen Aghajanyan, Sida Wang, and Luke Zettlemoyer. 2020a. Pre-training via paraphrasing. *arXiv preprint arXiv:2006.15020*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020b. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. 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*.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020a. Multilingual denoising pre-training for neural machine translation. *arXiv preprint arXiv:2001.08210*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

Zihan Liu, Genta Indra Winata, Andrea Madotto, and Pascale Fung. 2020b. Exploring fine-tuning techniques for pre-trained cross-lingual models via continual learning. *arXiv preprint arXiv:2004.14218*.

Fuli Luo, Wei Wang, Jiahao Liu, Yijia Liu, Bin Bi, Songfang Huang, Fei Huang, and Luo Si. 2020. Veco: Variable encoder-decoder pre-training for cross-lingual understanding and generation. *arXiv preprint arXiv:2010.16046*.

Martin Malmsten, Love Börjeson, and Chris Haffenden. 2020. Playing with words at the national library of sweden–making a swedish BERT. *arXiv preprint arXiv:2007.01658*.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. 2020. CamemBERT: a tasty French language model. In *Proceedings of the 58th Annual Meeting of the*

Association for Computational Linguistics, pages 7203–7219, Online. Association for Computational Linguistics.

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730*.

Sharan Narang, Colin Raffel, Katherine Lee, Adam Roberts, Noah Fiedel, and Karishma Malkan. 2020. WT5?! Training text-to-text models to explain their predictions. *arXiv preprint arXiv:2004.14546*.

Dat Quoc Nguyen and Anh Tuan Nguyen. 2020. PhoBERT: Pre-trained language models for Vietnamese. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1037–1042, Online. Association for Computational Linguistics.

Rodrigo Nogueira, Zhiying Jiang, Ronak Pradeep, and Jimmy Lin. 2020. Document ranking with a pre-trained sequence-to-sequence model. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 708–718, Online. Association for Computational Linguistics.

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

Jason Phang, Phu Mon Htut, Yada Pruksachatkun, Haokun Liu, Clara Vania, Katharina Kann, Iacer Calixto, and Samuel R Bowman. 2020. English intermediate-task training improves zero-shot cross-lingual transfer too. *arXiv preprint arXiv:2005.13013*.

Marco Polignano, Pierpaolo Basile, Marco de Gemmis, Giovanni Semeraro, and Valerio Basile. 2019. AlBERTo: Italian BERT language understanding model for NLP challenging tasks based on tweets. In *CLiC-it*.

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. *Journal of Machine Learning Research*, 21(140):1–67.

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.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426, Online. Association for Computational Linguistics.

Sebastian Ruder, Matthew E. Peters, Swabha Swayamdipta, and Thomas Wolf. 2019. Transfer learning in natural language processing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials*, pages 15–18, Minneapolis, Minnesota. Association for Computational Linguistics.

Noam Shazeer. 2020. GLU variants improve transformer. *arXiv preprint arXiv:2002.05202*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pages 5998–6008.

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.

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.

A mC4 Corpus Language Distribution

ISO Code	Language	Tokens (B)	Pages (M)	mT5 (%)	ISO Code	Language	Tokens (B)	Pages (M)	mT5 (%)
en	English	2,733	3,067	5.67	mk	Macedonian	1.8	2.1	0.62
ru	Russian	713	756	3.71	ml	Malayalam	1.8	2.1	0.62
es	Spanish	433	416	3.09	mn	Mongolian	2.7	2.1	0.62
de	German	347	397	3.05	ur	Urdu	2.4	1.9	0.61
fr	French	318	333	2.89	be	Belarusian	2.0	1.7	0.59
it	Italian	162	186	2.43	la	Latin	1.3	1.7	0.58
pt	Portuguese	146	169	2.36	eu	Basque	1.4	1.6	0.57
pl	Polish	130	126	2.15	tg	Tajik	1.4	1.3	0.54
nl	Dutch	73	96	1.98	te	Telugu	1.3	1.2	0.52
tr	Turkish	71	88	1.93	fy	West Frisian	0.4	1.1	0.51
ja	Japanese	164	87	1.92	kn	Kannada	1.1	1.1	0.51
vi	Vietnamese	116	79	1.87	ky	Kyrgyz	1.0	1.0	0.50
id	Indonesian	69	70	1.80	sw	Swahili	1.0	1.0	0.50
cs	Czech	63	60	1.72	so	Somali	1.4	0.9	0.48
zh	Chinese	39	55	1.67	my	Burmese	0.9	0.8	0.47
fa	Persian	52	54	1.67	uz	Uzbek	0.9	0.8	0.46
ar	Arabic	57	53	1.66	km	Khmer	0.6	0.8	0.46
sv	Swedish	45	49	1.61	-	Russian (Latin)	0.9	0.7	0.46
ro	Romanian	52	46	1.58	sd	Sindhi	1.6	0.7	0.45
el	Greek	43	42	1.54	gu	Gujarati	0.8	0.6	0.43
uk	Ukrainian	41	39	1.51	-	Hindi (Latin)	0.6	0.6	0.43
hu	Hungarian	39	37	1.48	jv	Javanese	0.3	0.6	0.42
da	Danish	29	29	1.38	zu	Zulu	0.2	0.6	0.42
fi	Finnish	25	27	1.35	si	Sinhala	0.8	0.5	0.41
no	Norwegian	27	25	1.33	-	Japanese (Latin)	0.3	0.5	0.41
bg	Bulgarian	22	23	1.29	eo	Esperanto	0.7	0.5	0.40
hi	Hindi	24	19	1.21	co	Corsican	0.2	0.5	0.40
sk	Slovak	18	18	1.19	ga	Irish	0.5	0.5	0.40
ko	Korean	26	16	1.14	-	Greek (Latin)	0.4	0.4	0.39
th	Thai	11	15	1.14	-	Chinese (Latin)	0.2	0.4	0.37
ca	Catalan	13	14	1.12	pa	Punjabi	0.6	0.4	0.37
ms	Malay	13	13	1.09	ceb	Cebuano	0.2	0.4	0.36
iw	Hebrew	17	12	1.06	mg	Malagasy	0.2	0.3	0.36
lt	Lithuanian	11	11	1.04	ps	Pashto	0.4	0.3	0.36
sl	Slovenian	8.8	8.5	0.95	sn	Shona	0.2	0.3	0.35
mr	Marathi	14	7.8	0.93	gd	Scottish Gaelic	0.4	0.3	0.35
bn	Bengali	7.3	7.4	0.91	ku	Kurdish	0.4	0.3	0.34
et	Estonian	6.9	6.9	0.89	hmn	Hmong	0.2	0.3	0.34
lv	Latvian	7.0	6.4	0.87	su	Sundanese	0.1	0.3	0.34
az	Azerbaijani	4.4	5.3	0.82	ht	Haitian Creole	0.2	0.3	0.33
gl	Galician	2.4	4.6	0.79	ha	Hausa	0.2	0.2	0.33
cy	Welsh	4.9	4.1	0.76	ny	Chichewa	0.1	0.2	0.29
sq	Albanian	4.0	4.1	0.76	am	Amharic	0.3	0.2	0.29
ta	Tamil	3.4	3.5	0.73	-	Bulgarian (Latin)	0.09	0.2	0.29
sr	Serbian	4.3	3.4	0.72	yi	Yiddish	0.3	0.1	0.28
ne	Nepali	3.2	2.9	0.69	lo	Lao	0.1	0.1	0.28
lb	Luxembourgish	1.0	2.7	0.68	mi	Maori	0.1	0.1	0.25
hy	Armenian	2.4	2.4	0.65	sm	Samoan	0.09	0.1	0.25
kk	Kazakh	3.1	2.4	0.65	ig	Igbo	0.09	0.09	0.24
ka	Georgian	2.5	2.3	0.64	haw	Hawaiian	0.09	0.08	0.24
mt	Maltese	5.2	2.3	0.64	xh	Xhosa	0.06	0.07	0.22
af	Afrikaans	1.7	2.2	0.63	st	Sotho	0.08	0.07	0.22
fil	Filipino	2.1	2.1	0.62	yo	Yoruba	0.05	0.05	0.20
is	Icelandic	2.6	2.1	0.62					

Table 6: Statistics of the mC4 corpus, totaling 6.6B pages and 6.3T tokens. The "mT5" column indicates the percentage of mT5 training data coming from a given language, using the default exponential smoothing value of α=0.3. We list 107 "languages" as detected by `cld3`, but note six of these (marked "Latin") are just Romanized variants of existing languages.

B Per-Language Results on All Tasks

Model	en	ar	bg	de	el	es	fr	hi	ru	sw	th	tr	ur	vi	zh	avg
Cross-lingual zero-shot transfer (models fine-tune on English data only)																
mBERT	80.8	64.3	68.0	70.0	65.3	73.5	73.4	58.9	67.8	49.7	54.1	60.9	57.2	69.3	67.8	65.4
XLM	82.8	66.0	71.9	72.7	70.4	75.5	74.3	62.5	69.9	58.1	65.5	66.4	59.8	70.7	70.2	69.1
XLM-R	88.7	77.2	83.0	82.5	80.8	83.7	82.2	75.6	79.1	71.2	77.4	78.0	71.7	79.3	78.2	79.2
mT5-Small	79.6	65.2	71.3	69.2	68.6	72.7	70.7	62.5	70.1	59.7	66.3	64.4	59.9	66.3	65.8	67.5
mT5-Base	84.7	73.3	78.6	77.4	77.1	80.3	79.1	70.8	77.1	69.4	73.2	72.8	68.3	74.2	74.1	75.4
mT5-Large	89.4	79.8	84.1	83.4	83.2	84.2	84.1	77.6	81.5	75.4	79.4	80.1	73.5	81.0	80.3	81.1
mT5-XL	90.6	82.2	85.4	85.8	85.4	81.3	85.3	80.4	83.7	78.6	80.9	82.0	77.0	81.8	82.7	82.9
mT5-XXL	**91.6**	**84.5**	**87.7**	**87.3**	**87.3**	**87.8**	**86.9**	**83.2**	**85.1**	**80.3**	**81.7**	**83.8**	**79.8**	**84.6**	**83.6**	**85.0**
Translate-train (models fine-tune on English training data plus translations in all target languages)																
mT5-Small	78.3	70.3	74.8	73.6	73.6	74.9	74.1	68.3	73.6	67.6	72.0	70.8	65.1	70.2	73.2	72.0
mT5-Base	85.8	78.8	82.2	81.6	81.4	83.0	82.1	77.0	81.1	74.8	78.6	78.4	73.3	78.9	80.2	79.8
mT5-Large	90.1	83.3	86.8	85.9	85.8	87.2	86.1	82.6	84.7	79.7	82.9	83.8	78.8	84.0	84.4	84.4
mT5-XL	91.0	84.0	87.5	87.2	86.7	88.5	87.4	83.1	85.3	80.9	83.2	84.7	80.3	84.8	85.0	85.3
mT5-XXL	**92.4**	**87.1**	**88.7**	**89.2**	**88.7**	**89.4**	**88.7**	**85.3**	**86.4**	**83.4**	**84.5**	**86.4**	**82.9**	**86.6**	**86.2**	**87.1**

Table 7: XNLI accuracy scores for each language.

Model	en	de	es	fr	ja	ko	zh	avg
Cross-lingual zero-shot transfer (models fine-tune on English data only)								
mBERT	94.0	85.7	87.4	87.0	73.0	69.6	77.0	81.9
XLM	94.0	85.9	88.3	87.4	69.3	64.8	76.5	80.9
XLM-R	94.7	89.7	90.1	90.4	78.7	79.0	82.3	86.4
mT5-Small	92.2	86.2	86.1	86.6	74.7	73.5	77.9	82.4
mT5-Base	95.4	89.4	89.6	91.2	79.8	78.5	81.1	86.4
mT5-Large	96.1	91.3	92.0	**92.7**	82.5	82.7	84.7	88.9
mT5-XL	96.0	92.8	**92.7**	92.4	83.6	83.1	86.5	89.6
mT5-XXL	**96.3**	**92.9**	92.6	**92.7**	**84.5**	**83.9**	**87.2**	**90.0**
Translate-train (models fine-tune on English training data plus translations in all target languages)								
mT5-Small	87.9	81.4	83.1	84.1	74.2	71.7	76.7	79.9
mT5-Base	95.5	90.9	91.4	92.5	83.6	84.8	86.4	89.3
mT5-Large	96.4	92.7	93.3	93.6	86.5	87.4	88.4	91.2
mT5-XL	96.4	92.5	93.1	93.6	85.5	86.9	**89.0**	91.0
mT5-XXL	**96.1**	**92.9**	**93.6**	**94.2**	**87.0**	**87.9**	**89.0**	**91.5**

Table 8: PAWS-X accuracy scores for each language.

Model	en	af	ar	bg	bn	de	el	es	et	eu	fa	fi	fr	he	hi	hu	id	it	ja	jv	
Cross-lingual zero-shot transfer (models fine-tune on English data only)																					
mBERT	85.2	77.4	41.1	77.0	70.0	78.0	72.5	77.4	**75.4**	**66.3**	46.2	**77.2**	79.6	56.6	65.0	**76.4**	53.5	81.5	29.0	66.4	
mT5-Small	80.6	67.0	36.2	59.8	60.0	66.1	54.0	63.6	58.4	42.3	25.3	64.5	74.6	39.6	57.4	61.5	46.6	73.2	28.8	49.6	
mT5-Base	83.2	73.8	45.4	62.1	67.1	72.5	57.0	70.3	67.3	49.2	30.4	68.6	78.6	46.1	67.6	64.7	49.7	78.9	35.0	56.9	
mT5-Large	84.2	74.7	55.0	60.6	64.5	75.2	68.2	74.2	66.4	48.4	51.4	65.8	82.4	55.8	69.0	67.3	51.1	80.6	43.0	57.1	
mT5-XL	86.3	79.3	60.2	80.3	78.1	80.4	78.3	74.5	71.8	52.2	**61.5**	70.1	85.9	65.3	76.3	71.9	56.8	83.2	47.7	63.2	
mT5-XXL	**86.6**	**81.1**	**66.5**	**85.1**	**78.8**	**82.0**	**79.1**	**85.8**	74.1	55.1	59.6	70.5	**86.8**	**66.1**	**78.4**	74.2	**75.0**	**86.3**	**51.0**	**69.1**	
	ka	kk	ko	ml	mr	ms	my	nl	pt	ru	sw	ta	te	th	tl	tr	ur	vi	yo	zh	avg
mBERT	64.6	**45.8**	**59.6**	52.3	58.2	72.7	**45.2**	81.8	80.8	64.0	67.5	50.7	48.5	3.6	71.7	**71.8**	36.9	71.8	44.9	42.7	62.2
mT5-Small	53.2	22.6	26.6	38.2	38.1	68.8	28.9	75.0	70.5	46.5	54.8	37.5	32.5	7.0	68.7	56.0	24.8	63.8	58.8	37.7	50.5
mT5-Base	49.9	22.1	33.9	45.5	43.8	68.9	36.4	80.1	76.0	53.2	62.4	40.8	41.8	8.5	74.1	58.4	38.4	72.1	56.5	41.0	55.7
mT5-Large	58.2	23.3	36.2	46.3	46.5	65.8	32.2	82.7	79.6	50.2	72.3	46.4	44.5	9.1	79.0	65.1	44.2	77.1	47.2	44.0	58.5
mT5-XL	66.0	31.6	38.1	54.1	57.6	74.5	42.6	85.5	85.2	66.9	72.8	49.0	54.7	9.6	84.1	67.4	64.7	79.6	59.0	53.9	65.5
mT5-XXL	**66.1**	39.2	43.2	**54.1**	**62.8**	**77.4**	44.1	**87.6**	**86.8**	**71.4**	**73.1**	**56.5**	**59.4**	**10.2**	**85.1**	71.6	**81.2**	**84.6**	**66.4**	**56.9**	**69.2**
Model	en	af	ar	bg	bn	de	el	es	et	eu	fa	fi	fr	he	hi	hu	id	it	ja	jv	
In-language multitask (models fine-tuned on gold data in all target languages)																					
mBERT	85.4	92.0	89.6	93.5	95.3	90.1	**91.1**	93.3	92.4	92.5	**79.6**	92.4	91.6	86.5	88.9	93.5	93.8	92.6	74.6	91.5	
mT5-Small	80.8	92.1	87.8	91.9	92.8	87.2	85.5	91.6	91.4	90.2	73.7	89.2	88.8	83.5	87.9	90.9	93.1	90.1	73.0	89.4	
mT5-Base	84.2	92.1	89.6	93.4	94.2	89.4	87.1	93.1	92.9	92.3	74.8	91.5	91.2	86.2	90.6	92.7	93.8	92.2	73.5	89.4	
mT5-Large	86.0	93.6	91.3	94.4	94.0	91.1	88.6	93.9	94.3	94.1	76.1	93.1	92.4	88.9	92.3	94.4	95.0	93.6	75.2	92.0	
mT5-XL	87.7	94.4	93.0	95.2	94.4	92.6	89.7	94.7	95.4	95.1	77.0	94.4	93.4	91.2	93.2	95.2	95.5	94.7	78.6	**94.9**	
mT5-XXL	**88.5**	**95.2**	**94.1**	**96.0**	**95.4**	**93.3**	90.5	**95.4**	**96.0**	**95.8**	77.5	**95.2**	**94.0**	**92.8**	**94.3**	**96.0**	**96.1**	**95.6**	**80.6**	92.8	
	ka	kk	ko	ml	mr	ms	my	nl	pt	ru	sw	ta	te	th	tl	tr	ur	vi	yo	zh	avg
mBERT	88.0	88.2	89.0	84.3	88.5	94.8	78.1	93.0	93.5	89.6	**91.8**	86.0	82.3	75.3	94.9	93.1	94.4	92.9	84.8	82.5	89.2
mT5-Small	87.1	85.8	84.2	79.9	85.0	93.8	64.8	90.4	90.9	86.2	77.9	87.1	75.0	76.4	95.4	91.9	94.8	91.3	86.4	78.7	86.4
mT5-Base	88.8	88.7	86.1	81.8	87.2	94.7	72.2	92.3	92.4	88.2	79.4	88.4	78.1	73.9	96.4	93.1	96.0	92.3	91.9	80.6	88.2
mT5-Large	91.1	89.8	89.2	84.1	89.3	96.0	74.4	93.9	93.8	90.4	80.7	91.1	81.4	73.9	97.3	94.8	96.1	93.8	91.5	82.4	89.7
mT5-XL	92.6	91.7	91.1	85.3	91.2	95.9	**84.7**	94.8	94.4	91.6	80.9	92.6	84.3	78.5	98.0	95.6	97.4	94.9	**93.7**	85.0	91.3
mT5-XXL	**93.8**	**94.3**	**92.7**	**86.6**	**93.1**	**97.3**	83.3	**95.5**	**95.4**	**92.7**	83.0	**93.2**	**86.1**	**79.7**	**98.0**	**96.2**	**97.4**	**95.5**	93.3	**86.2**	**92.2**

Table 9: WikiAnn NER F1 scores for each language.

Model	en	ar	de	el	es	hi	ru	th	tr	vi	zh	avg
Cross-lingual zero-shot transfer (models fine-tune on English data only)												
mBERT	83.5 / 72.2	61.5 / 45.1	70.6 / 54.0	62.6 / 44.9	75.5 / 56.9	59.2 / 46.0	71.3 / 53.3	42.7 / 33.5	55.4 / 40.1	69.5 / 49.6	58.0 / 48.3	64.5 / 49.4
XLM	74.2 / 62.1	61.4 / 44.7	66.0 / 49.7	57.5 / 39.1	68.2 / 49.8	56.6 / 40.3	65.3 / 48.2	35.4 / 24.5	57.9 / 41.2	65.8 / 47.6	49.7 / 39.7	59.8 / 44.3
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-Small	78.5 / 66.1	51.4 / 34.0	63.8 / 45.9	53.8 / 33.4	67.0 / 50.3	47.8 / 34.5	50.5 / 30.1	54.0 / 44.5	55.7 / 38.9	58.1 / 41.3	58.9 / 48.7	58.1 / 42.5
mT5-Base	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
mT5-Large	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
mT5-XL	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
mT5-XXL	**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**
Translate-train (models fine-tune on English training data plus translations in all target languages)												
mT5-Small	74.0 / 61.2	61.0 / 45.0	66.0 / 50.2	64.1 / 47.2	67.5 / 50.8	60.2 / 43.7	64.4 / 46.7	58.9 / 52.9	59.0 / 39.4	63.5 / 46.0	68.2 / 61.2	64.3 / 49.5
mT5-Base	83.1 / 70.3	72.4 / 55.2	76.9 / 59.7	76.8 / 58.8	79.0 / 61.2	71.4 / 53.4	76.1 / 58.5	67.9 / 62.0	72.5 / 51.4	75.9 / 56.3	76.9 / 69.7	75.3 / 59.7
mT5-Large	87.3 / 75.5	79.4 / 62.7	82.7 / 66.0	81.8 / 63.5	83.8 / 66.1	78.0 / 59.8	81.9 / 66.3	74.7 / 68.2	80.2 / 59.2	80.4 / 60.8	83.2 / 76.9	81.2 / 65.9
mT5-XL	88.5 / 77.1	80.9 / 65.4	83.4 / 66.7	83.6 / 64.9	84.9 / 68.2	79.6 / 63.1	82.7 / 67.1	78.5 / 72.9	82.4 / 63.8	82.4 / 64.1	83.2 / 75.9	82.7 / 68.1
mT5-XXL	**91.3 / 80.3**	**83.4 / 68.2**	**85.0 / 68.2**	**85.9 / 68.9**	**87.4 / 70.8**	**83.7 / 68.2**	**85.2 / 70.4**	**80.2 / 74.5**	**84.4 / 67.7**	**85.3 / 67.1**	**85.7 / 80.0**	**85.2 / 71.3**

Table 10: XQuAD results (F1/EM) for each language.

Model	en	ar	de	es	hi	vi	zh	avg
Cross-lingual zero-shot transfer (models fine-tune on English data only)								
mBERT	80.2 / 67.0	52.3 / 34.6	59.0 / 43.8	67.4 / 49.2	50.2 / 35.3	61.2 / 40.7	59.6 / 38.6	61.4 / 44.2
XLM	68.6 / 55.2	42.5 / 25.2	50.8 / 37.2	54.7 / 37.9	34.4 / 21.1	48.3 / 30.2	40.5 / 21.9	48.5 / 32.6
XLM-R	83.5 / 70.6	66.6 / 47.1	70.1 / 54.9	74.1 / 56.6	70.6 / 53.1	74.0 / 52.9	62.1 / 37.0	71.6 / 53.2
mT5-Small	77.2 / 63.0	44.7 / 27.3	53.3 / 35.7	60.1 / 41.5	43.0 / 29.2	52.9 / 33.2	51.3 / 29.7	54.6 / 37.1
mT5-Base	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
mT5-Large	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
mT5-XL	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
mT5-XXL	**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**
Translate-train (models fine-tune on English training data plus translations in all target languages)								
mT5-Small	70.5 / 56.2	49.3 / 31.0	55.6 / 40.6	60.5 / 43.0	50.4 / 32.9	55.2 / 36.3	54.4 / 31.6	56.6 / 38.8
mT5-Base	80.7 / 66.3	61.1 / 40.7	65.5 / 49.2	70.7 / 52.1	63.6 / 44.3	68.0 / 47.6	63.5 / 39.4	67.6 / 48.5
mT5-Large	85.3 / 72.0	68.5 / 47.7	71.6 / 55.8	75.7 / 57.1	71.8 / 52.6	74.3 / 54.0	70.1 / 47.1	73.9 / 55.2
mT5-XL	86.0 / 73.0	70.0 / 49.8	72.7 / 56.8	76.9 / 58.3	73.4 / 55.0	75.4 / 55.0	71.4 / 48.4	75.1 / 56.6
mT5-XXL	**86.5 / 73.5**	**71.7 / 51.4**	**74.9 / 58.7**	**78.8 / 60.3**	**76.6 / 58.5**	**77.1 / 56.3**	**72.5 / 49.8**	**76.9 / 58.3**

Table 11: MLQA results (F1/EM) for each language.

Model	en	ar	bn	fi	id	ko	ru	sw	te	avg
Cross-lingual zero-shot transfer (models fine-tune on English data only)										
mBERT	75.3 / 63.6	62.2 / 42.8	49.3 / 32.7	59.7 / 45.3	64.8 / 45.8	58.8 / 50.0	60.0 / 38.8	57.5 / 37.9	49.6 / 38.4	59.7 / 43.9
XLM	66.9 / 53.9	59.4 / 41.2	27.2 / 15.0	58.2 / 41.4	62.5 / 45.8	14.2 / 5.1	49.2 / 30.7	39.4 / 21.6	15.5 / 6.9	43.6 / 29.1
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-small	58.9 / 48.2	44.3 / 28.2	18.2 / 9.7	42.0 / 25.8	46.4 / 32.0	27.5 / 18.5	43.7 / 27.5	35.4 / 22.8	16.1 / 10.5	36.9 / 24.8
mT5-Base	72.8 / 60.2	68.9 / 50.3	44.9 / 28.3	67.9 / 53.1	73.3 / 55.2	48.6 / 34.4	58.0 / 35.7	59.9 / 42.5	46.2 / 34.1	60.0 / 43.8
mT5-Large	75.1 / 63.0	67.2 / 45.9	51.9 / 31.9	69.6 / 53.1	72.5 / 55.9	57.4 / 44.2	62.8 / 37.9	71.2 / 51.7	65.0 / 46.3	65.8 / 47.8
mT5-XL	79.6 / 68.9	82.4 / 65.9	72.8 / 54.9	79.9 / 65.7	82.6 / 68.5	68.3 / 57.6	73.9 / 49.5	77.3 / 59.7	79.1 / **60.2**	77.3 / 61.2
mT5-XXL	**84.5 / 73.2**	**84.6 / 68.8**	**82.5 / 70.8**	**82.8 / 70.1**	**85.8 / 73.3**	**77.2 / 66.3**	**77.4 / 57.0**	**83.8 / 69.7**	**79.6** / 59.8	**82.0 / 67.7**
Translate-train (models fine-tune on English training data plus translations in all target languages)										
mT5-Small	58.2 / 47.3	55.9 / 39.3	40.3 / 23.0	51.7 / 37.9	62.2 / 46.0	41.5 / 30.8	51.6 / 35.0	51.8 / 37.1	34.8 / 24.2	49.8 / 35.6
mT5-Base	71.0 / 59.1	71.8 / 55.5	56.8 / 36.3	71.5 / 58.8	76.8 / 60.5	61.5 / 49.3	66.1 / 47.5	67.0 / 50.7	55.3 / 41.7	66.4 / 51.0
mT5-Large	77.2 / 65.7	80.3 / 64.1	71.8 / 54.9	75.9 / 61.3	81.7 / 68.0	69.7 / 56.9	75.0 / 56.8	76.9 / 60.3	73.1 / 53.1	75.7 / 60.1
mT5-XL	81.7 / 68.9	82.1 / 66.0	79.0 / 64.6	79.5 / 65.3	84.9 / 71.2	71.8 / 57.6	78.7 / 60.6	82.4 / 67.3	80.8 / 63.8	80.1 / 65.0
mT5-XXL	**83.3 / 72.3**	**83.9 / 66.6**	**83.3 / 71.7**	**83.0 / 69.1**	**85.9 / 71.5**	**77.6 / 63.4**	**81.1 / 64.4**	**86.0 / 75.4**	**85.2 / 70.7**	**83.3 / 69.4**
In-language multitask (models fine-tuned on gold data in all target languages)										
mT5-Small	67.8 / 57.0	79.5 / 67.2	73.1 / 59.3	72.3 / 59.5	78.7 / 67.8	59.1 / 51.1	71.2 / 58.0	79.1 / 70.9	84.1 / 72.5	74.0 / 62.7
mT5-Base	74.6 / 63.2	82.8 / 69.7	79.7 / 67.3	78.5 / 66.4	84.9 / 73.5	70.7 / 62.7	76.1 / 62.3	81.7 / 72.5	87.2 / 77.4	79.7 / 68.4
mT5-Large	81.9 / 71.1	87.3 / 75.6	86.7 / 79.6	85.1 / 73.5	87.3 / 77.5	79.2 / 70.3	83.5 / 70.2	85.8 / 78.0	90.6 / 81.9	85.3 / 75.3
mT5-XL	83.8 / 74.3	88.4 / 76.7	88.7 / 83.2	86.7 / 75.6	90.1 / 81.4	82.9 / 74.6	85.3 / 73.2	90.1 / 82.8	**92.4 / 84.0**	87.6 / 78.4
mT5-XXL	**85.4 / 75.2**	**89.4 / 77.6**	**90.3 / 85.0**	**87.7 / 77.1**	**90.7 / 82.8**	**84.2 / 75.0**	**86.9 / 75.5**	**90.8 / 83.6**	**92.4** / 83.7	**88.7 / 79.5**
(Human)	84.2 / -	85.8 / -	94.8 / -	87.0 / -	92.0 / -	82.0 / -	96.3 / -	92.0 / -	97.1 / -	90.1 / -

Table 12: TyDi QA GoldP results (F1/EM) for each language.

C Per-Language Results of Ablation Models

Model	ar	bg	de	el	en	es	fr	hi	ru	sw	th	tr	ur	vi	zh	avg
Baseline (mT5-Large)	**79.8**	**84.1**	83.4	**83.2**	**89.4**	84.2	84.1	**77.6**	81.5	**75.4**	**79.4**	**80.1**	73.5	**81.0**	**80.3**	**81.1**
Dropout 0.1	76.4	82.1	81.7	81.0	88.0	70.8	80.3	74.4	79.0	72.3	75.8	75.9	70.6	78.6	76.5	77.6
Sequence length 512	78.1	83.4	83.1	82.1	88.8	84.5	82.8	77.3	81.2	**75.4**	78.2	79.6	73.8	80.0	78.9	80.5
Span length 10	77.6	81.5	80.5	81.2	87.2	83.0	81.2	74.7	79.8	73.6	76.7	75.9	71.3	78.6	76.5	78.6
$\alpha = 0.7$	79.3	**84.1**	**84.5**	83.1	**89.4**	**85.3**	**84.4**	76.4	**82.8**	70.6	78.7	79.8	71.7	80.3	79.9	80.7
$\alpha = 0.2$	78.7	83.8	83.3	82.5	89.3	83.4	83.6	77.3	81.2	**75.4**	78.6	79.4	**73.9**	79.9	79.7	80.7
No line length filter	78.4	83.3	81.5	81.4	88.9	83.8	82.5	74.4	80.5	69.4	77.6	76.9	71.3	78.8	78.3	79.1
Add Wikipedia data	79.3	83.1	83.1	82.7	88.6	80.1	83.2	77.3	81.4	75.0	78.9	79.3	73.5	80.2	79.2	80.3

Table 13: XNLI zero-shot accuracy of various ablations on our mT5-Large model.

Model	en	ar	de	el	es	hi	ru	th	tr	vi	zh	avg
Baseline (mT5-Large)	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
Span length 10	88.1 / 76.3	70.0 / 50.6	78.1 / 60.2	68.8 / 44.0	79.0 / 60.8	67.3 / 48.4	65.4 / 43.3	68.1 / 57.2	74.4 / 53.6	77.9 / 57.7	76.6 / 66.4	74.0 / 56.2
Dropout 0.1	87.3 / 76.0	54.9 / 33.9	77.6 / 60.2	64.4 / 40.1	79.2 / 60.6	59.1 / 40.4	59.5 / 38.4	65.7 / 51.0	73.6 / 52.8	75.8 / 55.8	77.0 / 64.5	70.4 / 52.1
Sequence length 512	88.0 / 76.9	**77.0** / 59.6	80.2 / 62.4	**79.8 / 60.0**	81.7 / **64.4**	**75.1 / 57.5**	**77.4 / 58.5**	72.7 / 59.8	75.3 / 53.9	79.4 / 58.9	78.5 / 67.2	**78.6 / 61.7**
$\alpha = 0.7$	88.4 / 77.1	76.5 / 58.8	78.5 / 59.8	77.2 / 55.5	78.7 / 59.5	74.6 / 56.8	73.1 / 54.5	72.5 / 60.2	75.7 / 55.0	79.2 / 58.3	**78.6** / 66.2	77.5 / 60.2
$\alpha = 0.2$	87.9 / 76.8	75.5 / 57.3	80.2 / 62.4	76.2 / 54.0	81.6 / 63.7	73.7 / 57.0	70.7 / 50.8	72.2 / 60.4	75.5 / 55.7	**79.7** / 59.7	78.3 / **67.5**	77.4 / 60.5
No line length filter	88.9 / 77.4	73.8 / 54.0	**80.8** / 62.7	74.2 / 51.8	80.9 / 62.8	74.1 / 56.6	75.0 / 56.4	71.7 / 60.3	**76.7** / 56.0	78.8 / 58.6	78.5 / 67.1	77.6 / 60.3
Add Wikipedia data	**89.3 / 78.4**	69.6 / 48.9	79.6 / 61.1	59.5 / 36.0	80.6 / 61.0	73.6 / 55.0	68.7 / 47.0	70.5 / 58.1	**76.7 / 56.9**	78.6 / 56.4	77.5 / 66.3	74.9 / 56.8

Table 14: XQuAD zero-shot F1/EM of various ablations on our mT5-Large model.