MERGEDISTILL: Merging Pre-trained Language Models using Distillation

Simran Khanuja
Google Research
India
simrankh@google.com

Melvin Johnson
Google Research
USA
melvinp@google.com

Partha Talukdar
Google Research
India
partha@google.com

Abstract

Pre-trained multilingual language models (LMs) have achieved state-of-the-art results in cross-lingual transfer, but they often lead to an inequitable representation of languages due to limited capacity, skewed pre-training data, and sub-optimal vocabularies. This has prompted the creation of an ever-growing pre-trained model universe, where each model is trained on large amounts of language or domain specific data with a carefully curated, linguistically informed vocabulary. However, doing so brings us back full circle and prevents one from leveraging the benefits of multilinguality. To address the gaps at both ends of the spectrum, we propose MERGEDISTILL, a framework to merge pre-trained LMs in a way that can best leverage their assets with minimal dependencies, using *task-agnostic* knowledge distillation. We demonstrate the applicability of our framework in a practical setting by leveraging pre-existing teacher LMs and training student LMs that perform competitively with or even outperform teacher LMs trained on several orders of magnitude more data and with a fixed model capacity. We also highlight the importance of teacher selection and its impact on student model performance.

1 Introduction

While current state-of-the-art multilingual language models (LMs) (Devlin et al., 2019; Conneau et al., 2020) aim to represent 100+ languages in a single model, efforts towards building monolingual (Martin et al., 2019; Kuratov and Arkhipov, 2019) or language-family based (Khanuja et al., 2021) models are only increasing with time (Rust et al., 2020). A single model is often incapable of effectively representing a diverse set of languages, evidence of which has been provided by works highlighting the importance of vocabulary curation and size (Chung et al., 2020; Artetxe et al., 2020),

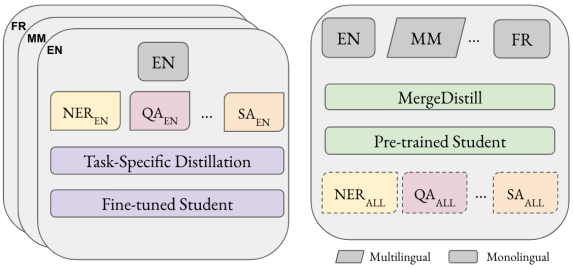

Figure 1: Previous works (left) typically focus on combining *fine-tuned* models derived from a single pre-trained model using distillation. We propose MERGEDISTILL to combine *pre-trained* teacher LMs from multiple monolingual/multilingual LMs into a single *multilingual task-agnostic* student LM.

pre-training data volume (Liu et al., 2019a; Conneau et al., 2020), and the curse of multilinguality (Conneau et al., 2020). Language specific models alleviate these issues with a custom vocabulary which captures language subtleties[1] and large magnitudes of pre-training data scraped from several domains (Virtanen et al., 2019; Antoun et al., 2020). However, building language specific LMs brings us back to where we started, preventing us from leveraging the benefits of multilinguality like zero-shot task transfer (Hu et al., 2020), positive transfer between related languages (Pires et al., 2019; Lauscher et al., 2020) and an ability to handle code-mixed text (Pires et al., 2019; Tsai et al., 2019). We need an approach that encompasses the best of both worlds, i.e., leverage the capabilities of the powerful language-specific LMs while still being multilingual and enabling positive language trans-

[1]For example, in Arabic, (Antoun et al., 2020) argue that while the definite article "Al", which is equivalent to "the" in English, is always prefixed to other words, it is not an intrinsic part of that word. While with a BERT-compatible tokenization tokens will appear twice, once with "Al-" and once without it, AraBERT first segments the words using Farasa (Abdelali et al., 2016) and then learns the vocabulary, thereby alleviating the problem.

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 2874–2887

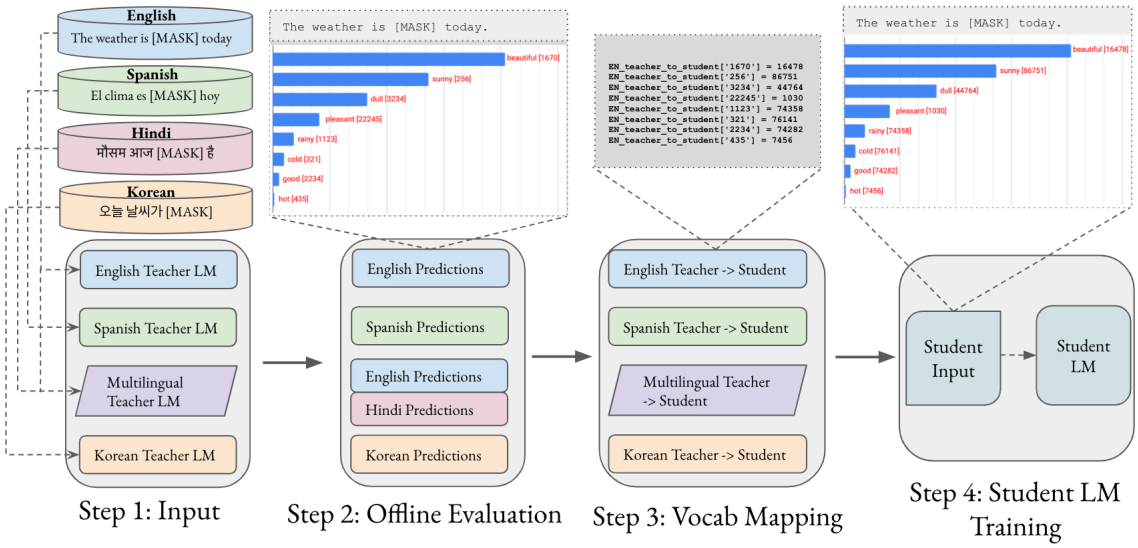

Figure 2: *Overview of* MERGEDISTILL: The input to MERGEDISTILL is a set of pre-trained teacher LMs and pre-training transfer corpora for all the languages we wish to train our student LM on. Here, we combine four teacher LMs comprising of three monolingual (trained on English, Spanish and Korean respectively) and one multilingual LM (trained on English and Hindi). The student LM is trained on English, Spanish, Hindi and Korean. Pre-training transfer corpora for each language is tokenized and masked using their respective teacher LMs vocabulary. We then obtain predictions for each masked word in each language, by evaluating *all* of their respective teacher LMs. For example, we evaluate English masked examples on both the monolingual and multilingual LM as shown. The student's vocabulary is a union of *all* teacher vocabularies. Hence, the *input, prediction and label indices* obtained from teacher evaluation are now mapped to the student vocabulary, and input to the student LM for training. Please refer to Section 3.1 for details.

fer.

In this paper, we use knowledge distillation (KD) (Hinton et al., 2015) to achieve this. In the context of language modeling, KD methods can be broadly classified into two categories: task-specific and task-agnostic. In task-specific distillation, the teacher LM is first fine-tuned for a specific task and is then distilled into a student model which can solve that task. Task-agnostic methods perform distillation on the pre-training objective like masked language modeling (MLM) in order to obtain a task-agnostic student model. Prior work has either used task-agnostic distillation to compress *single*-language teachers (Sanh et al., 2019; Sun et al., 2020) or used *task-specific* distillation to combine multiple fine-tuned teachers into a multi-task student (Liu et al., 2019b; Clark et al., 2019). The former prevents positive language transfer while the latter restricts the student's capabilities to the tasks and languages in the fine-tuned teacher LMs (as shown in Figure 1).

We focus on the problem of merging *multiple* pre-trained LMs into a single multilingual student LM in the *task-agnostic* setting. To the best of our knowledge, this is the first effort of its kind, and makes the following contributions:

- We propose MERGEDISTILL, a task-agnostic distillation approach to merge *multiple* teacher LMs at the *pre-training* stage, to train a strong multilingual student LM that can then be fine-tuned for *any* task on all languages in the student LM. Our approach is more maintainable (fewer models), compute efficient and teacher-architecture agnostic (since we obtain offline predictions).

- We use MERGEDISTILL to **i)** combine *monolingual* teacher LMs into a single *multilingual* student LM that is competitive with or outperforms individual teachers, **ii)** combine *multilingual* teacher LMs, such that the overlapping languages can learn from *multiple* teachers.

- Through extensive experiments and analysis, we study the importance of typological similarity in building multilingual models, and the impact of strong teacher LM vocabularies and predictions in our framework.

2 Related Work

Language Model pre-training has evolved from learning pre-trained word embeddings (Mikolov et al., 2013) to contextualized word representations (McCann et al., 2017; Peters et al., 2018; Eriguchi et al., 2018) and to the most recent Transformer-based (Vaswani et al., 2017) LMs (Devlin et al., 2019; Liu et al., 2019a) with state-of-the-art results on various downstream NLP tasks. Most commonly, these LMs are pre-trained with the MLM objective (Taylor, 1953) on large unsupervised corpora and then fine-tuned on labeled data for the task at hand. Concurrently, multilingual LMs (Lample and Conneau, 2019; Siddhant et al., 2020; Conneau et al., 2020; Chung et al., 2021), trained on massive amounts of multilingual data, have surpassed cross-lingual word embedding spaces (Glavaš et al., 2019; Ruder et al., 2019) to achieve state-of-the-art in cross-lingual transfer. While Pires et al. (2019); Wu and Dredze (2019) highlight their cross-lingual ability, several limitations have been studied. Conneau et al. (2020) highlight the curse of multilinguality. Hu et al. (2020) highlight that even the best multilingual models do not yield satisfactory transfer performance on the XTREME bechmark covering 9 tasks and 40 languages. Importantly, Wu and Dredze (2020) and Lauscher et al. (2020) observe that these models significantly under-perform for low-resource languages as representation of these languages in the vocabulary and pre-training corpora are severely limited.

Language-specific LMs are becoming increasingly popular as issues with multilingual language models persist. As language identification systems are extended to 1000+ languages (Caswell et al., 2020), increasing capacity for a single model to uniformly represent all languages is prohibitive. Often, practitioners prefer to have a model performing well on a subset of languages that their application calls for. To address this, the community continues its efforts in building strong multi-domain language models using linguistic expertise. A few examples of these are AraBERT (Antoun et al., 2020), CamemBERT (Martin et al., 2020), and FinBERT (Virtanen et al., 2019).[2]

Knowledge Distillation in pre-trained LMs has most commonly been used for task-specific model compression of a teacher into a single-task student (Tang et al., 2019; Kaliamoorthi et al., 2021). This has been extended to perform task-specific distillation of multiple single-task teachers into one multi-task student (Clark et al., 2019; Liu et al., 2020; Turc et al., 2019). In the task-agnostic scenario, prior work has focused on distilling a single large teacher model into a student model leveraging teacher predictions (Sanh et al., 2019) or internal teacher representations (Sun et al., 2020, 2019; Wang et al., 2020) with the goal of model compression. To the best of our knowledge, this is the first attempt to perform task-agnostic distillation from *multiple teachers* into a *single task-agnostic student*. In the context of neural machine translation, Tan et al. (2019) come close to our work where they attempt to combine multiple single language-pair teacher models to train a multilingual student. However, our work differs from theirs in three key aspects: 1) our students are task-agnostic while theirs are task-specific, 2) we can leverage pre-existing teachers while they cannot, and 3) we support teachers with overlapping sets of languages while they only consider single language-pairs teachers.

3 MERGEDISTILL

Notations: Let K denote the set of languages we train our student LM on and T denote the set of teacher LMs input to MERGEDISTILL[3]. Consequently, $\mathrm{T_k}$ denotes the set of teacher LMs trained on language k, where $|\mathrm{T_k}| \geq 1 \; \forall \; \mathrm{k} \in \mathrm{K}$.

3.1 Workflow

An overview of MERGEDISTILL is presented in Figure 2. Here we detail each step involved in training the student LM from multiple teacher LMs.

Step 1: Input

The input to MERGEDISTILL is a set of pre-trained teacher LMs and pre-training transfer corpora for all the languages we wish to train our student LM on. With reference to Figure 2, the student LM is trained on K ={English (en), Spanish (es), Hindi (hi), Korean (ko)}. We combine four teacher LMs comprising of three monolingual and one multilingual LM. The monolingual LMs are trained on English ($\mathrm{M_{en}}$), Spanish ($\mathrm{M_{es}}$), and Korean ($\mathrm{M_{ko}}$) while the multilingual LM is trained on English

[2](Nozza et al., 2020) maintain an ever-growing list of BERT models here

[3]Note that T can comprise of monolingual or multilingual models

and Hindi ($M_{en,hi}$). Therefore, for each language, the corresponding set of teacher LMs (T_k) can be defined as: [$T_{en} = \{M_{en}, M_{en,hi}\}$, $T_{es} = \{M_{es}\}$, $T_{hi} = \{M_{en,hi}\}$, $T_{ko} = \{M_{ko}\}$]. First, the pre-training transfer corpora is tokenized and masked for each language using their respective teacher LM's tokenizer. For the language with two teachers, English, we tokenize each example using both the teacher LMs.

Step 2: Offline Teacher LM Evaluation

We now obtain predictions and logits for each masked, tokenized example in each language, by evaluating their respective teacher LMs. For English, we obtain predictions from both M_{en} and $M_{en,hi}$ on their respective copies of each training example. In an ideal situation, we believe that multiple strong teachers can present a multi-view generalisation to the student as each teacher learns different features in training. Let x denote a sequence of tokens where $x_m = \{x_1, x_2, x_3 ... x_n\}$ denote the masked tokens, and x_{-m} denote the non-masked tokens. Let v be the vocabulary of student LM θ_s. In the conventional case of learning from gold labels, we minimize the cross-entropy of student logit distribution for a masked word x_{m_i}, with the *one-hot label* v_j, given by:

$$P(x_{m_i}, v_j) = \mathbf{1}(x_{m_i} = v_j) \times \log p(x_{m_i} = v_j | x_{-m}; \theta_s) \quad (1)$$

With the teacher evaluations, we obtain predictions (and corresponding logits) of the teacher for the masked tokens. Let us denote the teacher output probability distribution (softmax over logits) for token x_{m_i} by $Q(x_{m_i}|x_{-m}; \theta_t)$. Therefore, in addition to the loss from gold labels, we minimize the entropy between the student logits and the *teacher distribution*, given by :

$$\hat{P}(x_{m_i}, v_j) = Q(x_{m_i} = v_j | x_{-m}; \theta_t) \times \log p(x_{m_i} = v_j | x_{-m}; \theta_s) \quad (2)$$

It is extremely burdensome (both memory and time) to load multiple teacher LMs and obtain predictions during training. Hence, we first store the *top-k* logits for each masked word offline, loading and normalizing them during student LM training, similar to (Tan et al., 2019). Additionally, obtaining offline predictions gives one the freedom to use expensive teacher LMs without increasing the student model training costs and makes our framework teacher-architecture agnostic.

Step 3: Vocab Mapping

A deterrent in attempting to distill from multiple pre-trained teacher LMs is that each LM has its own vocabulary. This makes it non-trivial to uniformly process an input example for consumption by both the teacher and student LMs. Our student model's vocabulary is the union of *all* teacher LM vocabularies. In the vocab mapping step, the *input indices*, *prediction indices*, and the *gold label indices*, obtained after evaluation from each teacher LM are processed using a teacher→student vocab map. This converts each teacher token index to its corresponding student token index, ready for consumption by the student model. For simplicity, each teacher and student LM uses WordPiece tokenization (Schuster and Nakajima, 2012; Wu et al., 2016) in all our experiments.

Step 4: Student LM Training

The processed *input indices*, *prediction indices*, and *gold label indices* can now be used to train the multilingual student LM. In training, examples from different languages are shuffled together, even within a batch. We train the student LM with the MLM objective. Let L_{MLM} denote the MLM loss from gold labels. Therefore, with reference to Equation 1 :

$$L_{MLM}(x_m|x_{-m}) = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{|v|} P(x_{m_i}, v_j)$$

In addition to learning from gold labels, we use teacher predictions as soft labels and minimize the cross entropy between student and teacher distributions. Let L_{KD} denote the KD loss from a single teacher LM. With reference to Equation 2:

$$L_{KD}(x_m|x_{-m}) = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{|v|} \hat{P}(x_{m_i}, v_j);$$

The total loss across all languages is minimized, as shown below:

$$L_{ALL} = \sum_{k=1}^{K} \lambda(L_{KD}^{T_k}) + (1-\lambda)L_{MLM}^{k}$$

In the case of multiple teacher LMs, we have n tokenized instances for a given example (where n denotes the number of teachers for a particular language). In this case, each example in English has

two copies – one tokenized using M_{en} and another using $M_{en,hi}$. Thus, we explore two possibilities of training in this multi-teacher scenario :

- Include all the copies in training. Here the model is exposed to n different teacher LM predictions, each presenting a multi-view generalisation to the student LM.
- Include the best copy in training. The best copy is the one having minimum teacher LM loss for a given example. Here the model is only exposed to the best teacher LM predictions for each example.

4 Experiments

In this section, we aim to answer the following questions :

1) How effective is MERGEDISTILL in combining *monolingual* teacher LMs, to train a *multilingual* student LM that leverages the benefits of multilinguality while performing competitively with individual teacher LMs? (Section 4.2)

2) How effective is MERGEDISTILL in combining *multilingual* teacher LMs, trained on an overlapping set of languages, such that each language can benefit from *multiple* teachers? (Section 4.3)

3) How important are the teacher LM vocabulary and predictions in MERGEDISTILL? Further, can MERGEDISTILL enable pre-trained zero-shot transfer? (Section 4.4)

4.1 Setup

Data: For all our experiments, we use Wikipedia data as pre-training transfer corpora to train the student model, irrespective of the data used in training individual teacher LMs. We use $\alpha = 0.7$ for exponential smoothing of data across languages, similar to mBERT (Devlin et al., 2019).

Model Size: Since transformer-based models perform better as capacity increases (Conneau et al., 2020; Arivazhagan et al., 2019), we keep the number of parameters close to mBERT (∼178M) by appropriately modifying the vocabulary embedding size (like Lan et al. (2019)) to isolate the positive effects of learning from teacher LMs.

Student	Language	Language Family	Model
$\text{Student}_{\text{similar}}$	English	Indo-European	BERT(Devlin et al., 2019)
	German	Indo-European	DeepSet(Chan et al., 2020)
	Italian	Indo-European	ItalianBERT(Schweter, 2020b)
	Spanish	Indo-European	BETO(Cañete et al., 2020)
$\text{Student}_{\text{dissimilar}}$	Arabic	Afroasiatic	AraBERT(Antoun et al., 2020)
	English	Indo-European	BERT(Devlin et al., 2019)
	Finnish	Uralic	FinBERT(Virtanen et al., 2019)
	Turkish	Turkic	BERTurk(Schweter, 2020a)
	Chinese	Sino-Tibetan	ChineseBERT(Devlin et al., 2019)

Table 1: *Monolingual BERT Models* used as teacher LMs. Please refer to Section 4.2 for details.

Distillation Parameters: We have two hyperparameter choices here: 1) k in top-k logits - as it increases, we observe that while performances remain similar, storing k>8 number of predictions for each masked word offline significantly increases resource requirements[4]. Hence, we set k=8 in all our experiments. 2) the value of λ in the loss function, which decides the proportion of teacher loss, is annealed through training similar to Clark et al. (2019).

Evaluation Metrics: We report F1 scores for structured prediction tasks (NER, POS), accuracy (Acc.) scores for sentence classification tasks (XNLI, PAWS-X), and F1/Exact Match (F1/EM) scores for question answering tasks (XQuAD, MLQA, TyDiQA). We also report a task-specific *relative deviation from teachers* (**RDT**) (in %) averaged across all languages (n). For each task, **RDT** is calculated as:

$$\text{RDT}(\text{S}, \{\text{T}_1, ..., \text{T}_\text{n}\}) = \frac{100}{\text{n}} \sum_{\text{i}=1}^{\text{n}} \frac{(\text{P}_{\text{T}_\text{i}} - \text{P}_\text{S})}{\text{P}_{\text{T}_\text{i}}} \tag{3}$$

where $\text{P}_{\text{T}_\text{i}}$ and P_S are performances of the i^{th} teacher and student LMs, respectively.

4.2 Monolingual Teacher LMs

Pre-training: In this experiment, we use pre-existing monolingual teacher LMs, as shown in Table 1, to train a multilingual student LM on the union of all teacher languages. In this setup, $|\text{T}_\text{k}| = 1 \ \forall \ \text{k} \in \text{K}$, i.e., each language can learn from its respective monolingual teacher LM only.

Our teacher selection and setup follows a two-step process. First, we aim to select languages having pre-trained monolingual LMs available, and evaluation sets across a number of downstream tasks. This makes us choose teacher LMs for : Arabic (*ar*), Chinese (*zh*), English (*en*), Finnish (*fi*),

[4]More details in Appendix A.4

Language	Model	NER F1	UDPOS F1	QA F1/EM
English	BERT	89.5	96.6	87.1/78.6
	$\text{Student}_{\text{similar}}$	89.8	96.3	89.8/82.1
German	DeepsetBERT	93.0	98.3	-
	$\text{Student}_{\text{similar}}$	93.9	98.3	-
Italian	ItalianBERT	94.5	98.6	73.5/61.6
	$\text{Student}_{\text{similar}}$	95.2	98.6	75.8/63.8
Spanish	BETO	94.2	99.0	74.9/56.6
	$\text{Student}_{\text{similar}}$	94.7	98.9	76.5/58.4
	RDT(%)	**+0.6**	**-0.1**	**+2.8/+3.7**
Arabic	AraBERT	94.3	96.3	83.1/68.6
	$\text{Student}_{\text{dissimilar}}$	93.7	96.4	81.3/66.6
Chinese	ChineseBERT	83.0	96.9	81.8/81.8
	$\text{Student}_{\text{dissimilar}}$	82.6	96.8	80.8/80.8
English	BERT	89.5	96.6	87.1/78.6
	$\text{Student}_{\text{dissimilar}}$	89.5	96.3	88.6/80.7
Finnish	FinBERT	94.4	97.9	81.0/68.8
	$\text{Student}_{\text{dissimilar}}$	94.4	95.5	77.7/65.9
Turkish	BERTurk	95.2	95.6	76.7/59.8
	$\text{Student}_{\text{dissimilar}}$	95.4	92.9	76.2/59.1
	RDT(%)	**-0.2**	**-1.1**	**-1.3/-1.4**

Table 2: Results for *monolingual teacher LMs and multilingual students* on downstream tasks as described in Section 4.2. Relative deviations of 5% or less from teacher (i.e., RDT $\geq -5\%$) are marked in bold. We find that $\text{Student}_{\text{similar}}$ outperforms individual teacher LMs, with a maximum gain of upto +2.8/+3.7% for QA, while $\text{Student}_{\text{dissimilar}}$ is competitive with teacher LMs, with a maximum drop of -1.3/-1.4% for QA. Please refer to Section 4.2 for details.

German (*de*), Italian (*it*), Spanish (*es*), and Turkish (*tr*). Second, as previous work has evidenced positive transfer between related languages in a multilingual setup (Pires et al., 2019; Wu and Dredze, 2020), we further group the chosen teacher LMs based on language families as shown in Table 1, where:

i) $\textbf{Student}_{\textbf{similar}}$ is trained on four closely related languages from the Indo-European family – *de*, *en*, *es* and *it*.

ii) $\textbf{Student}_{\textbf{dissimilar}}$ is trained on languages from different language families – *ar*, *en*, *fi*, *tr* and *zh*.

Both student LMs have a BERT-base architecture. $\text{Student}_{\text{similar}}$ has a vocabulary size of 99,112 with a total of 162M parameters, while $\text{Student}_{\text{dissimilar}}$ has a vocabulary size of 180,996 with a total of 225M parameters. We keep a batch size of 4096 and train for 250,000 steps with a maximum sequence length of 512.

Fine-tuning: We evaluate both the teacher and student LMs on three downstream tasks with in-language fine-tuning for each task[5] :

Student	Language	Teacher LM Tokens	Student LM Tokens	% of Data
$\text{Student}_{\text{similar}}$	English	3300M	2285M	69.25%
	German	23723M	847M	3.57%
	Italian	13139M	506M	3.85%
	Spanish	3000M	639M	21.31%
	Total	**43162M**	**4277M**	**9.9%**
$\text{Student}_{\text{dissimilar}}$	Arabic	8600M	135M	1.58%
	English	3300M	2285M	69.25%
	Finnish	3000M	83M	2.77%
	Turkish	4405M	60M	1.36%
	Chinese	71M	71M	100.00%
	Total	**19376M**	**2634M**	**13.6%**

Table 3: *Number of Tokens (in Millions)* in the teacher (Table 1) and student LMs as described in Section 4.2

i) Named Entity Recognition *(NER)*: We use the WikiAnn (Pan et al., 2017; Rahimi et al., 2019) dataset for all languages.

ii) Part-of-Speech Tagging *(UDPOS)*: We use the Universal Dependencies v2.6 (Zeman et al., 2020) dataset for all languages.

iii) Question Answering *(QA)*: We use DRCD for *zh* (Shao et al., 2018), TQuAD[6] for *tr*, SQuADv1.1 (Rajpurkar et al., 2016) for *en*, SQuADv1.1-translated for *it* (Croce et al., 2018) and *es* (Carrino et al., 2020) and the TyDiQA-GoldP dataset (Clark et al., 2020) for *ar* and *fi*.

Results: We report results of our teacher and student LMs in Table 2. Overall, we find that $\text{Student}_{\text{similar}}$ outperforms individual teacher models on NER (+0.6%) and QA (+2.8/3.7%) while performing competitively on UDPOS (-0.1%). $\text{Student}_{\text{dissimilar}}$ is competitive with the teacher LMs with only small differences of up to 1.3/1.4% (QA), as shown in Table 2. For each language, we find $\text{Student}_{\text{similar}}$ is either competitive or outperforms its respective teacher LM. Our results provide evidence for positive transfer across languages in two ways. First, we observe that $\text{Student}_{\text{similar}}$ outperforms $\text{Student}_{\text{dissimilar}}$ for the common language - English. Given that the English teacher (BERT) and the pre-training transfer corpora[7] is common for both student LMs,

[5] More details in Appendix A.3

[6] https://tquad.github.io/turkish-nlp-qa-dataset

[7] In fact, we can hypothesize that $\text{Student}_{\text{dissimilar}}$ sees more English tokens as compared to $\text{Student}_{\text{similar}}$ because the Non-English languages in $\text{Student}_{\text{dissimilar}}$ are relatively low resourced (a sum total of 349M unique tokens) in comparison to $\text{Student}_{\text{similar}}$ (a sum total of 1992M unique tokens)

Languages	Model	Teacher	PANX F1	UDPOS F1	PAWSX Acc.	XNLI Acc.	XQUAD F1/EM	MLQA F1/EM	TyDiQA F1/EM	Avg.
MuRIL Languages	mBERT	-	58.8	68.5	93.4	66.2	70.3/57.5	65.0/50.8	62.5/52.	69.2
	MuRIL	-	76.9	74.5	95.0	74.4	77.7/64.2	73.6/58.6	76.1/60.2	78.3
	$\text{Student}_{\text{MuRIL}}$	MuRIL	69.3	72.3	95.4	71.9	75.7/62.1	72.0/56.3	70.7/59.2	75.3
	$\text{Student}_{\text{mBERT}}$	mBERT	38.1	52.1	93.5	64.8	56.9/44.8	51.1/39.7	41.6/33.9	56.9
	$\text{Student}_{\text{Both\_all}}$	mBERT + MuRIL	67.9	72.3	94.5	71.1	76.1/62.9	70.4/55.5	70.8/55.3	74.7
	$\text{Student}_{\text{Both\_best}}$	mBERT + MuRIL	68.5	71.5	93.9	70.7	77.7/64.3	70.8/55.6	70.6/58.4	74.8
	$\text{RDT}(\text{Student}_{\text{MuRIL}}, \text{mBERT})$ (%)		**+17.9**	**+5.6**	**+2.1**	**+8.6**	**+7.7/+8**	**+10.8/+10.8**	**+13.1/+12.3**	**+8.8**
	$\text{RDT}(\text{Student}_{\text{MuRIL}}, \text{MuRIL})$ (%)		-9.9	**-3**	**+0.4**	**-3.4**	**-2.6/-3.3**	**-2.2/-3.9**	-7.1/**-1.7**	**-3.8**
Non MuRIL Languages	mBERT	-	63.5	71.1	80.2	65.9	62.2/47.1	59.7/41.4	60.4/46.1	66.1
	$\text{Student}_{\text{MuRIL}}$	mBERT	63.9	72.8	83.3	68.7	66.5/51.2	63.1/44.4	61.7/45.0	68.6
	$\text{Student}_{\text{mBERT}}$	mBERT	64.6	72.1	84.0	68.8	64.5/49.0	61.1/42.7	58.9/44.1	67.7
	$\text{Student}_{\text{Both\_all}}$	mBERT	64.1	72.6	83.9	68.1	61.3/47.1	60.5/42.2	59.7/44.0	67.2
	$\text{Student}_{\text{Both\_best}}$	mBERT	63.3	72.6	83.2	67.2	66.0/50.6	61.4/43.2	62.4/46.5	68.0
	$\text{RDT}(\text{Student}_{\text{MuRIL}}, \text{mBERT})$ (%)		**+0.6**	**+2.4**	**+3.9**	**+4.3**	**+6.9/+8.7**	**+5.7/+7.2**	**+2.2/-2.4**	**+3.8**

Table 4: *Results for multilingual teacher and student LMs* on the XTREME benchmark. We compare performances of three student LM variants as described in Section 4.3 to the two teachers mBERT and MuRIL. Relative deviations of 5% or less from teacher (i.e., $\text{RDT} \geq -5\%$) are marked in bold. Overall, we find that $\text{Student}_{\text{MuRIL}}$ performs the best among all student variants and report its RDT (in %) (Equation 3) from the two teachers. Please refer to Section 4.3 for a detailed analysis.

we can attribute this gain to the fact that English is trained with linguistically and typologically similar languages in $\text{Student}_{\text{similar}}$. Second, $\text{Student}_{\text{similar}}$ outperforms its teacher LMs while $\text{Student}_{\text{dissimilar}}$ is competitive for *all* languages. These two results across *all* languages point towards $\text{Student}_{\text{similar}}$ benefiting from a positive transfer across similar languages. In Table 3, we observe that $\text{Student}_{\text{similar}}$ is trained on 9.9% of the total unique tokens seen by its respective teacher LMs and $\text{Student}_{\text{dissimilar}}$ lies close with 13.6%. Despite this huge disparity in pre-training corpora, student LMs are competitive with their teachers. This encouraging result proves that even with very limited data, MERGEDISTILL enables one to combine strong monolingual teacher LMs to train competitive student LMs that can leverage the benefits of multilinguality.

4.3 Multilingual Teacher LMs

Pre-training: In this experiment, we make use of pre-existing multilingual models: mBERT and MuRIL. mBERT is trained on 104 languages and MuRIL covers 12 of these (11 Indian languages + English): Bengali (*bn*), English (*en*), Gujarati (*gu*), Hindi (*hi*), Kannada (*kn*), Malayalam (*ml*), Marathi (*mr*), Nepali (*ne*), Punjabi (*pa*), Tamil (*ta*), Telugu (*te*), and Urdu (*ur*), with higher performance for these languages on the XTREME benchmark. We train the student model on all 104 languages. In this case, the *MuRIL Languages* (MuL) have two teachers (mBERT and MuRIL) and the *Non-MuRIL Languages* (Non-MuL) can learn from mBERT only. Therefore, while we only use mBERT as the teacher LM for Non-MuL across all experiments, we consider three possibilities for MuL :

i) $\mathbf{Student_{MuRIL}}$: We only use MuRIL as the teacher LM and each input training example is tokenized using MuRIL.
ii) $\mathbf{Student_{mBERT}}$: We only use mBERT as the teacher LM and each input training example is tokenized using mBERT.
iii) $\mathbf{Student_{Both}}$: As highlighted in Section 3, we consider two possibilities to incorporate both teacher LM predictions in training:

- $\mathbf{Student_{Both\_all}}$: Tokenize each input example using mBERT and MuRIL separately and include both copies in training.
- $\mathbf{Student_{Both\_best}}$: Tokenize each input example using mBERT and MuRIL separately and include only the best copy in training. The best copy is the one having minimum teacher LM loss for the example.

Note, it is non-trivial to tokenize each example in a way that is compatible with all teacher LMs. One must resort to tokenization using an intersection of vocabularies which is sub-optimal.

All the student LMs use a BERT-base architecture and have a vocabulary size of 288,973. We reduce our embedding dimension to 256 as opposed to

as shown in Table 3

Model	Vocabulary	Labels	PANX	UDPOS	PAWSX	XNLI	XQUAD	MLQA	TyDiQA	Avg.
SM1	mBERT	Gold	63.2	73.0	94.8	71.2	70.2/57.9	65.1/51.3	60.8/48.7	71.2
SM2	mBERT∪MuRIL	Gold	**69.3**	**73.9**	95.3	71.2	**76.2/63.1**	71.1/56.0	70.9/56.0	**75.4**
SM3	mBERT∪MuRIL	Gold+Teacher	**69.3**	72.3	**95.4**	**71.9**	75.7/62.1	**72.0/56.3**	**70.7/59.2**	75.3
SM2_100k	mBERT∪MuRIL	Gold	65.5	72.3	94.3	67.5	72.3/58.2	66.9/51.5	62.5/51.9	71.6
SM3_100k	mBERT∪MuRIL	Gold+Teacher	71.2	73.5	93.1	69.6	76.4/62.9	69.1/53.9	68.6/54.9	74.5

Table 5: *Importance of teacher vocabulary and predictions in* MERGEDISTILL. We observe maximum performance gains, by changing the vocabulary from mBERT in SM1 to (mBERT∪MuRIL) vocabulary in SM2. Here, SM3 is the standard $\text{Student}_{\text{MuRIL}}$. We also observe that SM3_100k, trained for 20% of the total training steps, is competitive to SM3 and significantly outperforms SM2_100k, highlighting the importance of teacher LM predictions in a limited data scenario. Please see Section 4.4 for details.

768 to bring down the model size to be around 160M, comparable to mBERT (178M). We keep a batch size of 4096 and train for 500,000 steps with a maximum sequence length of 512.

Finetuning: We report zero-shot performance for all languages in the XTREME (Hu et al., 2020) benchmark[8].

Results: We report results of our teacher and student LMs in Table 4. Overall, we find that $\text{Student}_{\text{MuRIL}}$ performs the best among all student variants. For Non-MuL, $\text{Student}_{\text{MuRIL}}$ beats the teacher (mBERT) by an average relative score of 3.8%. For MuL, $\text{Student}_{\text{MuRIL}}$ beats one teacher (mBERT) by 8.8%, but underperforms the other teacher (MuRIL) by 3.8%. There can be two factors at play here. MuRIL is trained on monolingual and parallel data [9] while the student LMs only see ∼22% of unique tokens in comparison. MuRIL also has different language sampling strategies ($\alpha = 0.3$ as opposed to 0.7 in our setting, where a lower α value upsamples more rigorously from the tail languages), which have a significant role to play in multilingual model performances (Conneau et al., 2020). We also observe a significant drop in $\text{Student}_{\text{mBERT}}$'s performance for MuL when compared to the other student LM variants. This might be because the input is tokenized using the mBERT tokenizer which prevents learning from MuRIL tokens in the student vocabulary. For $\text{Student}_{\text{Both}}$, we do not observe much of a difference between $\text{Student}_{\text{Both\_all}}$ and $\text{Student}_{\text{Both\_best}}$. This observation may differ with one's choice of teacher LMs depending on how well it performs for a particular language. In our case, we don't observe much of a difference in incorporating mBERT predictions for MuL.

[8]More details in Appendix A.3
[9]More details in Appendix A.2

4.4 Further Analysis

The importance of vocabulary and teacher LM preditions: In Table 4, we see that $\text{Student}_{\text{MuRIL}}$ significantly outperforms mBERT for MuL, despite both being trained on Wikipedia corpora, and having comparable model sizes. With regard to MuL, $\text{Student}_{\text{MuRIL}}$ differs from mBERT in two main aspects – **i)** $\text{Student}_{\text{MuRIL}}$'s vocabulary is a union of mBERT and MuRIL vocabularies. **ii)** $\text{Student}_{\text{MuRIL}}$ is trained with additional MuRIL predictions as soft labels. To disentangle the role both these factors play in $\text{Student}_{\text{MuRIL}}$'s improved performance, we train two models :
i) SM1 is trained exactly like $\text{Student}_{\text{MuRIL}}$, but with mBERT vocabulary and on gold labels.
ii) SM2 is trained using $\text{Student}_{\text{MuRIL}}$'s vocabulary (mBERT ∪ MuRIL) but on gold labels only, without teacher predictions.

The results are summarized in Table 5. Note, we refer to $\text{Student}_{\text{MuRIL}}$ as SM3. Overall, we observe a ∼**4.2%** gain in average performance for SM2 over SM1. This clearly highlights that given fixed data and model capacity, LM training significantly benefits by incorporating a strong teacher's vocabulary.

Furthermore, we also observe that SM2 and SM3 achieve competitive performances despite SM3 being additionally trained on teacher LM labels. To motivate the need for teacher predictions, Hinton et al. (2015) argue that when soft targets have high entropy, they provide much more information per training case than hard targets and can be trained on *much less data* than the original cumbersome model. In our case, we hypothesize that training on 500,000 steps exposes the model to sufficient data for it to generalize well enough and mask the benefits of teacher LM predictions. To validate this, we evaluate the performances of SM2 and SM3,

20% into training (i.e. 100,000 steps / 500,000 total steps) as shown in Table 5. We observe a **∼2.9%** gain in average performance for SM3 over SM2, clearly highlighting the importance of teacher LM predictions in a limited data scenario. This is especially important when one has access to very limited monolingual data and a strong teacher LM for a particular language.

Pre-trained zero-shot transfer: Interestingly, $\text{Student}_{\text{MuRIL}}$ performs the best on almost all tasks for *Non-MuL*. This hints at positive transfer from strong teachers to languages that the teacher does not cover at all, due to the shared multilingual representations.[10] This would mean that learning from strong teachers can improve the student model's performance in a zero-shot manner on related languages not covered by the teacher. This would make MERGEDISTILL highly beneficial for low-resource languages that do not have a strong teacher or limited gold data. We leave this exploration to future work.

5 Conclusion

In this paper we address the problem of merging multiple pre-trained teacher LMs into a single multilingual student LM by proposing MERGEDISTILL, a task-agnostic distillation method. To the best of our knowledge, this is the first attempt of its kind. The student LM learned by MERGEDISTILL may be further fine-tuned for any task across all of the languages covered by the teacher LMs. Our approach results in better maintainability (fewer models) and is compute efficient (due to offline predictions). We use MERGEDISTILL to **i)** combine *monolingual* teacher LMs into one student multilingual LM which is competitive with the teachers, thereby demonstrating positive cross-lingual transfer, and **ii)** combine multilingual LMs to train student LMs that learn from *multiple* teachers. Through experiments on multiple benchmark datasets, we show that student LMs learned by MERGEDISTILL perform competitively or even outperform teacher LMs trained on orders of magnitude more data. We disentangle the positive impact of incorporating strong teacher LM vocabularies and learning from teacher LM predictions, highlighting the importance of the latter in a limited data scenario. We also find that MERGEDISTILL enables positive transfer from strong teachers to languages not covered by them (i.e. zero-shot transfer). Our work bridges the gap between the universe of language-specific models and massively multilingual LMs, incorporating benefits of both into one framework.

[10] For example, if you want to train a multilingual model covering English and a closely related low-resource language for which there exists no strong teacher, it may be possible to improve performance for the low resource language using teacher predictions for English only, due to a shared embedding space and possibly shared sub-words.

6 Acknowledgements

We would like to thank the anonymous reviewers for their insightful and constructive feedback. We thank Iulia Turc, Ming-Wei Chang, and Slav Petrov for valuable comments on an earlier version of this paper.

References

Ahmed Abdelali, Kareem Darwish, Nadir Durrani, and Hamdy Mubarak. 2016. Farasa: A fast and furious segmenter for arabic. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Demonstrations*, pages 11–16.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. AraBERT: Transformer-based model for Arabic language understanding. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 9–15, Marseille, France. European Language Resource Association.

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.

Casimiro Pio Carrino, Marta R. Costa-jussà, and José A. R. Fonollosa. 2020. Automatic Spanish translation of SQuAD dataset for multi-lingual question answering. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 5515–5523, Marseille, France. European Language Resources Association.

Isaac Caswell, Theresa Breiner, Daan van Esch, and Ankur Bapna. 2020. Language ID in the wild: Unexpected challenges on the path to a thousand-language web text corpus. In *Proceedings of*

the 28th International Conference on Computational Linguistics, pages 6588–6608, Barcelona, Spain (Online). International Committee on Computational Linguistics.

José Cañete, Gabriel Chaperon, Rodrigo Fuentes, Jou-Hui Ho, Hojin Kang, and Jorge Pérez. 2020. Spanish pre-trained bert model and evaluation data. In *PML4DC at ICLR 2020*.

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model.

Hyung Won Chung, Thibault Fevry, Henry Tsai, Melvin Johnson, and Sebastian Ruder. 2021. Rethinking embedding coupling in pre-trained language models. In *International Conference on Learning Representations*.

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

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.

Kevin Clark, Minh-Thang Luong, Urvashi Khandelwal, Christopher D. Manning, and Quoc V. Le. 2019. BAM! born-again multi-task networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5931–5937, Florence, Italy. Association for Computational Linguistics.

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

Danilo Croce, Alexandra Zelenanska, and Roberto Basili. 2018. Neural learning for question answering in italian. In *AI*IA 2018 – Advances in Artificial Intelligence*, pages 389–402, Cham. Springer International Publishing.

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.

Akiko Eriguchi, Melvin Johnson, Orhan Firat, Hideto Kazawa, and Wolfgang Macherey. 2018. Zero-shot cross-lingual classification using multilingual neural machine translation. *arXiv preprint arXiv:1809.04686*.

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*.

Goran Glavaš, Robert Litschko, Sebastian Ruder, and Ivan Vulić. 2019. How to (properly) evaluate cross-lingual word embeddings: On strong baselines, comparative analyses, and some misconceptions. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 710–721, Florence, Italy. Association for Computational Linguistics.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network.

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*.

Prabhu Kaliamoorthi, Aditya Siddhant, Edward Li, and Melvin Johnson. 2021. Distilling large language models into tiny and effective students using pqrnn.

Simran Khanuja, Diksha Bansal, Sarvesh Mehtani, Savya Khosla, Atreyee Dey, Balaji Gopalan, Dilip Kumar Margam, Pooja Aggarwal, Rajiv Teja Nagipogu, Shachi Dave, Shruti Gupta, Subhash Chandra Bose Gali, Vish Subramanian, and Partha Talukdar. 2021. Muril: Multilingual representations for indian languages.

Yuri Kuratov and Mikhail Arkhipov. 2019. Adaptation of deep bidirectional multilingual transformers for russian language. *arXiv preprint arXiv:1905.07213*.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual Language Model Pretraining. In *Proceedings of NeurIPS 2019*.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

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

Hairong Liu, Mingbo Ma, Liang Huang, Hao Xiong, and Zhongjun He. 2019a. Robust neural machine translation with joint textual and phonetic embedding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3044–3049, Florence, Italy. Association for Computational Linguistics.

Linqing Liu, Huan Wang, Jimmy Lin, Richard Socher, and Caiming Xiong. 2020. Mkd: a multi-task knowledge distillation approach for pretrained language models.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019b. Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4487–4496, Florence, Italy. Association for Computational Linguistics.

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.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric Villemonte de la Clergerie, Djamé Seddah, and Benoît Sagot. 2019. Camembert: a tasty french language model. *arXiv preprint arXiv:1911.03894*.

Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. In *Proceedings of NIPS 2017*, pages 6294–6305.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26:3111–3119.

Debora Nozza, Federico Bianchi, and Dirk Hovy. 2020. What the [mask]? making sense of language-specific bert models. *arXiv preprint arXiv:2003.02912*.

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.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Afshin Rahimi, Yuan Li, and Trevor Cohn. 2019. Massively multilingual transfer for NER. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 151–164, Florence, Italy. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Sebastian Ruder, Ivan Vulić, and Anders Søgaard. 2019. A survey of cross-lingual word embedding models. *Journal of Artificial Intelligence Research*, 65:569–631.

Phillip Rust, Jonas Pfeiffer, Ivan Vulić, Sebastian Ruder, and Iryna Gurevych. 2020. How good is your tokenizer? on the monolingual performance of multilingual language models. *arXiv preprint arXiv:2012.15613*.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Mike Schuster and Kaisuke Nakajima. 2012. Japanese and korean voice search. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5149–5152. IEEE.

Stefan Schweter. 2020a. Berturk - bert models for turkish.

Stefan Schweter. 2020b. Italian bert and electra models.

Chih Chieh Shao, Trois Liu, Yuting Lai, Yiying Tseng, and Sam Tsai. 2018. Drcd: a chinese machine reading comprehension dataset. *arXiv preprint arXiv:1806.00920*.

Aditya Siddhant, Melvin Johnson, Henry Tsai, Naveen Arivazhagan, Jason Riesa, Ankur Bapna, Orhan Firat, and Karthik Raman. 2020. Evaluating the Cross-Lingual Effectiveness of Massively Multilingual Neural Machine Translation. In *Proceedings of AAAI 2020*.

Siqi Sun, Yu Cheng, Zhe Gan, and Jingjing Liu. 2019. Patient knowledge distillation for bert model compression. *arXiv preprint arXiv:1908.09355*.

Zhiqing Sun, Hongkun Yu, Xiaodan Song, Renjie Liu, Yiming Yang, and Denny Zhou. 2020. MobileBERT: a compact task-agnostic BERT for resource-limited

devices. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2158–2170, Online. Association for Computational Linguistics.

Xu Tan, Yi Ren, Di He, Tao Qin, Zhou Zhao, and Tie-Yan Liu. 2019. Multilingual neural machine translation with knowledge distillation. *arXiv preprint arXiv:1902.10461*.

Raphael Tang, Yao Lu, Linqing Liu, Lili Mou, Olga Vechtomova, and Jimmy Lin. 2019. Distilling task-specific knowledge from bert into simple neural networks. *arXiv preprint arXiv:1903.12136*.

Wilson L Taylor. 1953. "cloze procedure": A new tool for measuring readability. *Journalism quarterly*, 30(4):415–433.

Henry Tsai, Jason Riesa, Melvin Johnson, Naveen Arivazhagan, Xin Li, and Amelia Archer. 2019. Small and practical bert models for sequence labeling. *arXiv preprint arXiv:1909.00100*.

Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Well-read students learn better: On the importance of pre-training compact models. *arXiv preprint arXiv:1908.08962*.

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*, pages 5998–6008.

Antti Virtanen, Jenna Kanerva, Rami Ilo, Jouni Luoma, Juhani Luotolahti, Tapio Salakoski, Filip Ginter, and Sampo Pyysalo. 2019. Multilingual is not enough: Bert for finnish. *arXiv preprint arXiv:1912.07076*.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers.

Shijie Wu and Mark Dredze. 2019. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 833–844, Hong Kong, China. Association for Computational Linguistics.

Shijie Wu and Mark Dredze. 2020. Are all languages created equal in multilingual BERT? In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 120–130, Online. Association for Computational Linguistics.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation.

Daniel Zeman, Joakim Nivre, and Mitchell Abrams et al. 2020. Universal dependencies 2.6. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

A Appendix

A.1 Knowledge Distillation

We train our LMs with the MLM objective. Let x denote a sequence of tokens where $x_m = \{x_1, x_2, x_3 ... x_n\}$ denote the masked tokens, and x_{-m} denote the non-masked tokens. Let v be the vocabulary of LM θ. The log-likelihood loss (cross-entropy with one-hot label) can be formulated as follows:

$$L_{MLM}(x_m|x_{-m}) = -\frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{|v|} P(x_{m_i}, k);$$

$$P(x_{m_i}, k) = \mathbf{1}(x_{m_i} = k) \log p(x_{m_i} = k|x_{-m}; \theta)$$

In a distillation setup, the student is trained to not only match the one-hot labels for masked words, but also the probability output distribution of the teacher t. Let us denote the teacher output probability distribution for token x_{m_i} by $Q(x_{m_i}|x_{-m}; \theta_t)$. The cross entropy between the teacher and student distributions then serves as the distillation loss :

$$L_{KD}(x_m|x_{-m}) = -\frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{|v|} \hat{P}(x_{m_i}, k);$$

$$\hat{P}(x_{m_i}, k) = Q(x_{m_i} = k|x_{-m}; \theta_t)$$
$$\log p(x_{m_i} = k|x_{-m}; \theta)$$

The total loss is then defined as :

$$L_{ALL} = \lambda L_{KD} + (1 - \lambda) L_{MLM}$$

With the addition of the teacher, the target distribution is no longer a single one-hot label, but a smoother distribution with multiple words having non-zero probabilities which yields in a smaller variance in gradients (Hinton et al., 2015). Intuitively, a single masked word can have several valid predictions, which appropriately fit the context.

Teacher	Language	Teacher LM Tokens	Student LM Tokens	% of Data
MuRIL	Bengali	1181M	27M	2.30%
	English	6986M	2816M	40.30%
	Gujarati	173M	7M	3.90%
	Hindi	2368M	38M	1.61%
	Kannada	196M	15M	7.64%
	Malayalam	337M	14M	4.17%
	Marathi	274M	8M	3.02%
	Nepali	231M	5M	2.16%
	Punjabi	141M	9M	6.45%
	Tamil	769M	26M	3.34%
	Telugu	331M	30M	8.99%
	Urdu	722M	23M	3.21%
	Total	**13709M**	**3018M**	**22%**

Table 6: *Number of Tokens (in Millions)* in the teacher (MuRIL) and student LMs as described in Section 4.3. Note, we only show the *MuRIL Languages* here because for *Non-MuRIL Languages*, the teacher (mBERT) and student variants are trained on the same data.

A.2 Pre-training Details

A.2.1 Monolingual Teacher LMs

We pre-train our student models using the BERT base architecture. $\text{Student}_{\text{similar}}$ has a vocabulary size of 99112 and a model size of 162M parameters. $\text{Student}_{\text{different}}$ has a vocabulary size of 180996 and a model size of 225M parameters. We keep a batch size of 4096 and train for 250k steps with a maximum sequence length of 512. We use TPUs, and it takes around 1.5 days to pre-train each student LM.

A.2.2 Multilingual Teacher LMs

We pre-train our student models using the BERT base architecture. All student LMs have a vocabulary size of 288973. Hence, we reduce our embedding dimension to 256 as opposed to 768 to bring down the model size to be around 160M, comparable to mBERT (178M). We keep a batch size of 4096 and train for 500k steps with a maximum sequence length of 512. We use TPUs, and it takes around 3 days to pre-train each student LM.

We present pre-training data statistics for MuRIL and the student LMs in Table 6. Here we only include the monolingual data statistics, but MuRIL is additionally trained on parallel translated and transliterated data.

Task	Batch	Learning Rate	No. of Epochs	Warmup Ratio	Max. seq. Length
NER	32	3e-5	10	0.1	256
POS	32	3e-5	10	0.1	256
QA	32	3e-5	10	0.1	384

Table 7: Hyperparameter Details for each fine-tuning task in Section 4.2

A.3 Fine-tuning Details

A.3.1 Monolingual Teacher LMs

Data Statistics We evaluate our monolingual teacher LMs and multilingual student LMs, as described in Section 4.2, on three tasks as follows:

i) Named Entity Recognition (NER): We use the WikiAnn (Pan et al., 2017; Rahimi et al., 2019) dataset for all languages. Each language comprises of a train/dev/test split of 20000/10000/10000 tokens. Specifically, we use the huggingface re-packaged implementation of the dataset[11].

ii) Part-of-Speech tagging (POS): We use the Universal Dependencies v2.6 (Zeman et al., 2020) dataset for all languages. Detailed statistics for each language can be found in Table 9. Specifically, we use the huggingface re-packaged implementation of the dataset[12].

iii) Question Answering (QA): We use the TyDiQA dataset (Clark et al., 2020) for *ar* and *fi*, SQuADv1.1 (Rajpurkar et al., 2016) for *en*, SQuAD-translated for *it* (Croce et al., 2018) and *es* (Carrino et al., 2020), DRCD for *zh* (Shao et al., 2018) and TQuAD[13] for *tr*. Detailed statistics for each language can be found in Table 10. Note, we use the dev set as our test sets, since most datasets only have a train/dev split. We use 10% of randomly shuffled training examples as our dev sets.

Hyperparameter Details: We use the same hyperparameters for fine-tuning all teacher and student LMs, as shown in Table 7. We report results on the best-performing checkpoint for the validation set. The performance of the best checkpoint on validation sets are shown in Table **??**

[11] https://huggingface.co/datasets/wikiann
[12] https://huggingface.co/datasets/universal_dependencies
[13] https://tquad.github.io/turkish-nlp-qa-dataset

Languages	Model	PANX	UDPOS	PAWSX	XNLI	XQUAD	MLQA	TyDiQA
MuRIL Languages	mBERT	58.8	68.5	93.4	66.2	70.3/57.5	65.0/50.8	62.5/52.7
	MuRIL	**76.9**	74.5	95.0	**74.4**	**77.7/64.2**	**73.6/58.6**	**76.1/60.2**
	k=8	69.3	72.3	**95.4**	71.9	75.7/62.1	72.0/56.3	70.7/59.2
	k=128	67.5	72.8	94.4	70.7	75.5/61.9	71.1/56.1	70.2/55.4
	k=512	69.2	**77.2**	94.7	71.3	75.6/61.8	72.3/56.9	68.5/53.9
Non MuRIL Languages	mBERT	63.5	71.1	80.2	65.9	62.2/47.1	59.7/41.4	60.4/46.1
	k=8	63.9	72.8	83.3	**68.7**	**66.5/51.2**	63.1/44.4	**61.7/45.0**
	k=128	63.7	72.8	**83.4**	67.9	66.1/51.1	61.4/43.4	62.6/46.7
	k=512	**64.8**	**73.3**	82.7	67.4	65.7/50.7	**63.6/44.9**	58.7/44.8
All Languages	mBERT	62.5	70.6	82.0	65.9	63.7/49.0	61.2/44.1	61.1/48.3
	k=8	65.0	72.7	**85.0**	**69.3**	**68.2/53.2**	65.6/47.8	**64.7/49.7**
	k=128	64.5	72.8	**85.0**	68.4	67.9/53.0	64.2/47.1	65.2/49.6
	k=512	**65.7**	**74.0**	84.4	68.2	67.5/52.8	**66.1/48.3**	62.0/47.8

Table 8: *Results of the best performing student model* $\text{Student}_{\text{MuRIL}}$ for different top-k values

Language	Dataset	Examples (Train/Dev/Test)
Arabic	AR_PADT	6075/909/680
Chinese	ZH_GSD	3997/500/500
English	EN_EWT	12543/2002/2077
German	DE_HDT	15305/18434/18459
Finnish	FI_FTB	14981/1875/1867
Italian	IT_ISDT	13121/564/482
Spanish	ES_ANCORA	14305/1654/1721
Turkish	TR_IMST	3664/988/983

Table 9: *Universal Dependencies v2.6* overview for each language, used in Section 4.2

Language	Dataset	Examples (Train/Test)
Arabic	TyDiQA-GoldP	14805/921
Chinese	DRCD	26936/3524
English	SQuADv1.1	87599/10570
German	-	-
Finnish	TyDiQA-GoldP	6855/782
Italian	SQuADv1.1-translated	87599/10570
Spanish	SQuADv1.1-translated	87595/10570
Turkish	TQuAD	8308/892

Table 10: *Question Answering datasets*, used in Section 4.2

A.3.2 Multilingual Teacher LMs

Data Statistics We evaluate all the teacher (mBERT and MuRIL) and student ($\text{Student}_{\text{MuRIL}}$, $\text{Student}_{\text{mBERT}}$ and $\text{Student}_{\text{Both}}$) LMs on the XTREME (Hu et al., 2020) benchmark. We fine-tune the pre-trained models on English training data for the particular task, except TyDiQA, where we use additional SQuAD v1.1 English training data, similar to (Fang et al., 2020). All results are computed in a zero-shot setting.

Hyperparameter Details We use the same hyperparameters for fine-tuning all teacher and student LMs, as shown in Table 11. We report results on the best-performing checkpoint for the eval set.

Task	Batch	Learning Rate	No. of Epochs	Warmup Ratio	Max. seq. Length
PANX	32	2e-5	10	0.1	128
UDPOS	64	5e-6	10	0.1	128
PAWSX	32	2e-5	5	0.1	128
XNLI	32	2e-5	3	0.1	128
XQuAD	32	3e-5	2	0.1	384
MLQA	32	3e-5	2	0.1	384
TyDiQA	32	3e-5	2	0.1	384

Table 11: Hyperparameter Details for each task in XTREME

A.4 Different top-k values

We present results for $\text{Student}_{\text{MuRIL}}$ trained with different top-k values from teacher predictions in Table 8. We observe that while performances remain similar for higher values of k, storage becomes increasingly expensive. Hence, we stick to a value of k=8 in all our experiments.