When the Student Becomes the Master: Learning Better and Smaller Monolingual Models from mBERT

Pranaydeep Singh and Els Lefever
LT3, Language and Translation Technology Team
Department of Translation, Interpreting and Communication – Ghent University
Groot-Brittanniëlaan 45, 9000 Ghent, Belgium
{firstname.lastname}@ugent.be

Abstract

In this research, we present pilot experiments to distil monolingual models from a jointly trained model for 102 languages (mBERT). We demonstrate that it is possible for the target language to outperform the original model, even with a basic distillation setup. We evaluate our methodology for 6 languages with varying amounts of resources and belonging to different language families.

1 Introduction

The introduction of the Transformer architecture, which aims to solve sequence-to-sequence tasks while also handling long-range dependencies relying on self-attention (Vaswani et al., 2017), has caused a huge improvement on the state-of-the-art for a wide range of natural language processing tasks. Despite the clear gain in performance, these pre-trained language models are extremely data- and computation-hungry and their sizes keep growing at an incredible speed. For example, RoBERTa (Liu et al., 2019) is trained on a text corpus of 160 GB, and the recent collaboration of Microsoft and Nvidia resulted in a language model containing 530 billion parameters that was trained on 270 billion tokens (Smith et al., 2022). Besides the ethical questions on sustainability raised by the NLP community, these huge language models also pose a lot of operational challenges, as they require massive amounts of training data, computation power and storage capacity. More importantly, the limits on training data also introduce limits on the languages these models can work with. Consequently, English has been the default language newer transformers have been trained on. More recently, much attention has been devoted to multilingual aspects, especially with the advent of joint models like mBERT (Devlin et al., 2019) and XLM (Conneau and Lample, 2019). However, the percentage of data used to train joint models for low-resourced languages when compared to English is orders of magnitudes lesser and results in poorer representations for lower-resourced languages (Wu and Dredze, 2020). Moreover, when language representations for a specific low-resourced language are needed, there are no available monolingual models for a lot of these languages, and the entire large jointly-trained model needs to be loaded. More recently, a lot of attention is paid to sustainability for transformers, resulting in various approaches like pruning, weight sharing and distillation to reduce the size of large models. Previous research has shown good results for task-specific distillation, distillation from larger English BERT models and distillation from mBERT into a smaller multilingual model (see Section 2). In this paper, we attempt to combine these two research directions and distil smaller monolingual transformers from mBERT. Our hypothesis was that distilling would help to improve the representations for a target language, as the model can focus its prediction power on the target language, instead of attempting to accommodate 101 other languages. To the best of our knowledge, this is the first research presenting results for distilling monolingual student models from mBERT. Not only are we able to successfully distil smaller monolingual models from mBERT, we also demonstrate that these smaller models outperform mBERT for the distilled language. We experiment with student models for well-resourced (Dutch, French), middle-resourced (Hindi, Hebrew) and low-resourced (Swahili, Slovenian) languages from very diverse language families and attempt to understand how a student model can outperform a teacher model in a distillation setup.

2 Related Research

While models that generate Deep Contextualized Representations like ELMo (Peters et al., 2018), BERT (Devlin et al., 2019) and GPT (Brown

Proceedings of the 29th International Conference on Computational Linguistics, pages 4434–4441
October 12–17, 2022.

et al., 2020) have pushed the state-of-the-art for downstream tasks in English, the improvements for medium- and low-resourced languages have not been as significant. Trained models on large monolingual corpora are abundant for English and other Western European languages, but they are extremely scarce for under-resourced languages (having small Wikipedias to train language models on). Although joint models trained for multiple languages like mBERT (Devlin et al., 2019) and XLM (Conneau and Lample, 2019) have been an excellent alternative to monolingual models, and even perform better than monolingual models due to additional supervision, these large joint models also come with drawbacks. Wu et al. (2020) show that a majority of languages in mBERT are under-represented and have poor performance on downstream tasks. This can be attributed to multiple reasons, like minimal monolingual data compared to English and other Western European languages, a small percentage share when it comes to mBERT's shared vocabulary and the sub-word tokenization not being suitable for all scripts. However, mBERT still performs better than monolingual models for these languages, since the monolingual models are trained on very limited pre-training data compared to mBERT. Therefore, with monolingual transformers either not existing or performing worse than mBERT, one is forced to use mBERT for monolingual tasks for a particular low-resourced language, and thus deploying representations for 101 other languages. Abdaoui et al. (2020) propose a simple alternative by loading specific sections of mBERT's vocabulary for particular languages, causing no loss in performance while decreasing the size of the deployed model considerably.

Various methodologies have been proposed to improve the sustainability of transformers. Distillation methods have shown to be very successful, but also low-rank approximation (Chen et al., 2021), weight sharing (Reid et al., 2021), pruning (Lagunas et al., 2021) and quantization (Bondarenko et al., 2021) propose excellent alternatives. While the central idea behind distillation is to reduce the model size as much as feasible while keeping the performance as close to the original model as possible, task-specific distillation (Tang et al., 2019) is a variant which focuses on a certain subset of the model's capabilities. The broader concept behind both methodologies is similar: a student model is initialised with significantly lesser parameters than the original teacher model. The student is trained with the loss for the objective at hand (masked language modeling (MLM) for generic distillation and specific tasks for task-based distillation), while also forcing the predictions to be identical to the teacher model. The underlying hypothesis being that once a larger model has learned the nuances of the data, a much smaller model can simply mimic the predictions with significantly lesser parameters. Hinton et al. (2015) further introduced the softmax temperature to emphasize learning from the entire distribution. A seminal work in the distilation of BERT-like transformers was performed by Sanh et al. (2019), who report a considerable model size reduction (40%) while retaining 97% performance on downstream tasks.

In this paper, we present pilot experiments for a new type of knowledge distillation that aims to extract representations for a particular language from a multilingual model. While there have been more advanced distillation approaches proposed, such as Patient Knowledge Distillation (Sun et al., 2019), this research is a first attempt at language-based distillation, where the basic distillation setup is validated as a proof of concept. We demonstrate that even when using basic distillation, it is possible to obtain monolingual models 3 times smaller than mBERT, while also performing consistently better on the distilled language. In addition, we show promising results for languages with a large variety in terms of available resources and belonging to different language families.

3 Distillation Methodology and Experimental Setup

We use the basic, proven distillation technique by Sanh et al. (2019), which uses mBERT as a teacher for all setups, and a 6-layer BERT architecture without the pooler and token-type embeddings as the student. We also use the 3 loss functions proposed, which are formally represented in the equations below. The distillation loss (1) ensures the similarity between the teacher output distribution t_i and the student output distribution s_i using negative log-likelihood. The cosine embedding loss (2) ensures the distributions not only have similar magnitudes but a similar directional alignment as well by penalising cosine distance. The final loss (3) is the standard cross-entropy used for most modern MLM systems, where s_i is once again the

[1]https://github.com/NirantK/hindi2vec

	Wikipedia	Downstream Task 1	Downstream Task 2	Monolingual Model
French	12M	Sentiment (Le et al., 2019)	UD POS (GSD)	CamemBERT-base
Dutch	4.4M	Sentiment (van der Burgh (2019))	UD POS (Lassy-small)	BERTje
Hindi	1.1M	News (hindi2vec: *github NirantK/hindi2vec*)	UD POS (HDTB)	indicBERT
Hebrew	1.3M	Sentiment (Amram et al., 2018)	UD POS (HTB)	AlephBERT
Swahili	0.1M	News (SNCD: *huggingface datasets/swahili_news*)	NER (Adelani et al., 2021)	SwahBERT
Slovene	0.4M	NER (Rahimi et al., 2019)	UD POS (SSJ)	SloBERTa

Table 1: Overview of the used resources for each of the 6 languages, including the monolingual Wikipedia (in million pages) used for the distillation process, as well as the two tasks used to evaluate the distilled model. For all PoS-tagging tasks (*UD POS*), datasets were retrieved from the Universal Dependencies dataset (*https://universaldependencies.org*). The last column lists the monolingual model that was used to present an upperbound score for the given task and language.

student's output distribution while y_i is the ground truth output distribution. The three losses are then aggregated with a weighted sum (4). An important factor here is the unlabelled monolingual data used for distillation. While it is often the case for distillation systems to pre-train the student model on the entire corpus to ensure optimal transfer, we want to focus on the capabilities of a single target language, and therefore only distil for $i \epsilon L$ where L is the target language in question.

$$L_{distillation} = \sum_{i \epsilon L} t_i * log(s_i) \quad (1)$$

$$L_{cosine} = \sum_{i \epsilon L} 1 - cos(t_i, s_i) \quad (2)$$

$$L_{mlm} = \sum_{i \epsilon L} y_i * log(s_i) \quad (3)$$

$$L = \alpha_1 L_{distillation} + \alpha_2 L_{mlm} + \alpha_3 L_{cosine} \quad (4)$$

A vital thing to note here is that L_{mlm} and $L_{distillation}$ can be slightly contradictory. While both enforce the same obejctive, L_{mlm} uses the ground truth while $L_{distillation}$ uses the teacher's predictions. Therefore, if the teacher predictions are often dissimilar to the ground-truth, the two losses might interfere with each other's progress. We explore this further in Section 4.

We also follow in the footsteps of the previous work for the student initialisation. It was noted by Sun et al. (2019) that initialising the student model from the teacher model by skipping alternate layers greatly speeds up the learning process and improves the student model considerably. Our final step was to reduce the vocabulary of the student model to only the target language L. Since the student vocabulary was initialized from the teacher mBERT, which contains sub-words from over a 100 languages, this majorly contributes to the large vocabulary, and therefore directly to the model size and inference speed. We use the approach suggested by Abdaoui et al. (2020), which selects a subset of the vocabulary of a large model and eliminates the unnecessary parameters from the embedding layer and tokenizer, thus significantly bringing down model parameters.

We replicate the distillation experiments for a set of 6 languages, using the entire available Wikipedia for each language. Since the data size varies considerably (ranging between 0.1 million pages for Swahili to 12 million pages for French), we train for 20,000 steps to ensure all models are trained to a similar extent. The student models for Dutch and French might therefore be improved by further training since they have more available data, but our objective here was to focus on the viability of the approach for low-resourced languages. Experiments with the high-resourced languages (Dutch, French) are only added for the sake of comparison. We use the values of 5.0 for α_1, 2.0 for α_2 and 1.0 for α_3, respectively. A starting learning rate of $5e-4$ was used, with a batch size of 8 per device, for 4 Tesla A100 GPUs, and the distillation takes approximately 48 hrs per language. Post-distillation, we reduce the model's vocabulary as described above. Next, we test the monolingual student models on two different downstream tasks for each language. For the fine-tuning, we add a

classification layer to the distilled model and train with an LR of $5e-5$ for 5 epochs with 500 warm up steps before a linear LR decay. We include tasks requiring both semantic understanding (sentiment analysis, news classification) as well as syntactic knowledge (PoS-tagging, NER).

An overview of the evaluated tasks and datasets per language is presented in Table 1. The last column refers to the models used for the upper-bound scores in Table 2. While for some languages we could use the standard monolingual BERT models as the upper bound (BERTje[1], AlephBERT[2], SwahBERT[3]), for other languages we had to use more advanced architectures like RoBERTa (CamemBERT[4], SloBERTa[5]), while for Hindi we used IndicBERT[6], which is an ALBERT model for 12 Indic Languages.

4 Results and Discussion

The results for all experiments are summarized in Table 2. The distilled language models (*Eliquare*) almost consistently outperform mBERT, even when mBERT has 3 times as many parameters. A more fair comparison is made with distilmBERT (Sanh et al., 2019), which has fewer parameters than mBERT and a similar model structure to the *Eliquare* models, but uses the entire Wikipedia for all languages for the distillation process (i.e. 237 million pages). In addition, we notice that DistilmBERT is also consistently outperformed by *Eliquare*. This outcome may seem counter-intuitive and unusual at first, since a student model should in principle not be outperforming a teacher model based on evidence from a number of distillation methodologies presented over the years. It can be explained, however, by the loss setup discussed in Section 3. If mBERT consistently makes mistakes for MLM, $L_{distillation}$ and L_{mlm} continue to contradict each other. While the MLM loss will encourage the model to improve, since the distillation loss is weighted (α_1) so heavily, it ensures the model cannot be very different from the teacher. As a result, each *Eliquare* model, even though better, is only marginally improved due to the distillation loss halting the progress. This is illustrated by Figure 1, which shows the progress of the two losses for Dutch distillation. While $L_{distillation}$ converges much faster, and maintains a much lower absolute value, L_{mlm} encounters a lot of fluctuation and has an almost four times higher mean value, even though both loss functions are near identical (cross-entropy). The distillation and MLM loss plots for the other 5 languages are provided in appendix A. The results for the upper-bound models are only

	Task 1	Task 2
French		
Upper-bound $*$	0.9338	0.9818
mBERT	0.8923	**0.9795**
distilmBERT	0.8773	0.9790
Eliquare	**0.8952**	0.9792
Dutch		
Upper-bound	0.9300	0.9630
mBERT	**0.9033**	0.9623
distilmBERT	0.8812	0.9607
Eliquare	0.8970	**0.9625**
Hindi		
Upper-bound $\star *$	0.2553	0.9208
mBERT	0.4744	0.9666
distilmBERT	0.4555	0.9597
Eliquare	**0.5066**	**0.9683**
Hebrew		
Upper-bound	0.8871	0.9620
mBERT	0.8512	0.9681
distilmBERT	0.8391	0.9597
Eliquare	**0.8567**	**0.9705**
Swahili		
Upper-bound	0.9090	0.8850
mBERT	0.8689	0.8490
distilmBERT	0.8666	0.8452
Eliquare	**0.8701**	**0.8632**
Slovene		
Upper-bound $*$	0.9410	0.9902
mBERT	0.9326	0.9791
distilmBERT	0.9268	0.9790
Eliquare	**0.9365**	**0.9822**

Table 2: Experimental results (macro-F1) for multilingual BERT (*mBERT*), distilled mBert (*distil*) and our language-specific distillation approach (*Eliquare*) for various end-tasks for 2 high-resourced (French, Dutch), 2 middle-resourced (Hindi, Hebrew) and 2 low-resourced languages (Swahili, Slovene). $\star$ Refers to non-monolingual models trained with additional similar languages, while $*$ refers to more advanced architectures (like RoBERTa) expected to perform better than BERT-based models.

[1]https://huggingface.co/GroNLP/bert-base-dutch-cased
[2]https://huggingface.co/onlplab/alephbert-base
[3]https://github.com/gatimartin/SwahBERT
[4]https://huggingface.co/camembert-base
[5]https://huggingface.co/EMBEDDIA/sloberta
[6]https://huggingface.co/ai4bharat/indic-bert

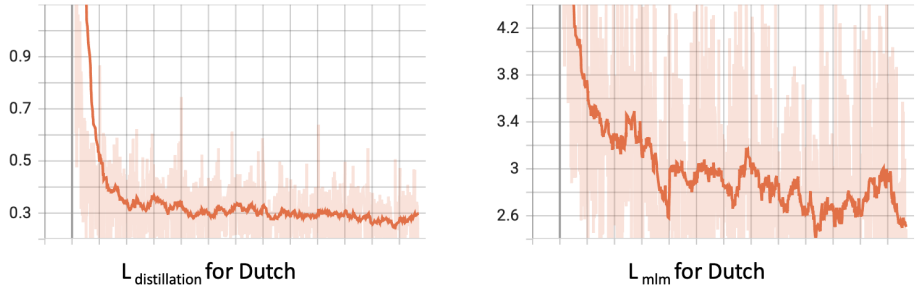

Figure 1: Distillation loss (left) and MLM loss (right) for the Dutch language distillation.

Model	# of Parameters	Inference Speed	Inference Memory
mBERT	167M	0.384s	10880 MB
distilmBERT	134M	0.165s	8798 MB
Eliquare	66M	0.066s	2944 MB

Table 3: Sustainability comparisons of *Eliquare* with standard mBERT and distilled mBERT versions. All numbers were computed on a Tesla V100 with a Batch Size of 32 and Sequence Length of 512 for inference.

provided for reference and do not serve as a fair comparison since these are specialized models for the language trained on significantly more monolingual data compared to mBERT.

While the performances of all the models maybe very close together, where the *Eliquare* set of models really shine is in a practical deployment setting. As shown in Table 3, *Eliquare* models are about 6 times faster than mBERT for inference (for a Batch Size of 32 and Sequence Length of 512 on a single Tesla V100), while being approximately 2.5 times faster than the distilled mBERT model. Moreover, they occupy around 1/3rd of the memory of mBERT with the same inference setting.

To conclude, we demonstrate that it is possible to distil better, smaller and faster monolingual models from mBERT, using a very basic distillation setup. The results show that the *Eliquare* monolingual models consistently outperform mBERT which has 3 times more parameters, and distilmBERT which has almost 2 times more parameters and uses orders of magnitude more data for the distillation process. While the *Eliquare* models may not be useful for the high-resourced languages due to the availability of large monolingual models, they present a major step towards having small monolingual models for a number of low-resourced languages.

In future research, we will investigate how to further improve the student from the teacher by diminishing the impact of the distillation loss. In addition, we will run the same set of experiments with XLM-R to investigate whether the same distillation approach can be applied to other joint multilingual models as well. Finally, we will also explore the impact of distilling multiple typologically similar languages from mBERT in parallel.

References

Amine Abdaoui, Camille Pradel, and Grégoire Sigel. 2020. Load what you need: Smaller versions of multilingual bert. In *SustaiNLP / EMNLP*.

D. Adelani, Jade Abbott, Graham Neubig, Daniel D'Souza, Julia Kreutzer, Constantine Lignos, Chester Palen-Michel, Happy Buzaaba, Shruti Rijhwani, Sebastian Ruder, Stephen Mayhew, Israel Abebe Azime, S. Muhammad, Chris C. Emezue, Joyce Nakatumba-Nabende, Perez Ogayo, Anuoluwapo Aremu, Catherine Gitau, Derguene Mbaye, J. Alabi, Seid Muhie Yimam, Tajuddeen R. Gwadabe, Ignatius Ezeani, Rubungo Andre Niyongabo, Jonathan Mukiibi, V. Otiende, Iroro Orife, Davis David, Samba Ngom, Tosin P. Adewumi, Paul Rayson, Mofetoluwa Adeyemi, Gerald Muriuki, Emmanuel Anebi, C. Chukwuneke, N. Odu, Eric Peter Wairagala, S. Oyerinde, Clemencia Siro, Tobius Saul Bateesa, Temilola Oloyede, Yvonne Wambui, Victor Akinode, Deborah Nabagereka, Maurice Katusiime, Ayodele Awokoya, Mouhamadane Mboup, D. Gebreyohannes, Henok Tilaye, Kelechi Nwaike, Degaga Wolde, Abdoulaye Faye, Blessing Sibanda, Orevaoghene Ahia, Bonaventure F. P. Dossou, Kelechi Ogueji, Thierno Ibrahima Diop, A. Diallo, Adewale Akinfaderin, T. Marengereke, and Salomey Osei. 2021. Masakhaner: Named entity recognition for african languages. *ArXiv*, abs/2103.11811.

Adam Amram, Anat Ben David, and Reut Tsarfaty. 2018. Representations and architectures in neu-

ral sentiment analysis for morphologically rich languages: A case study from Modern Hebrew. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2242–2252, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Yelysei Bondarenko, Markus Nagel, and Tijmen Blankevoort. 2021. Understanding and overcoming the challenges of efficient transformer quantization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7947–7969, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Beidi Chen, Tri Dao, Eric Winsor, Zhao Song, Atri Rudra, and Christopher Ré. 2021. Scatterbrain: Unifying sparse and low-rank attention. In *Advances in Neural Information Processing Systems (NeurIPS)*.

A. Conneau and G. Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

J. Devlin, M.-W. Chang, K. Lee, and K. 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.

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

François Lagunas, Ella Charlaix, Victor Sanh, and Alexander M. Rush. 2021. Block pruning for faster transformers. *CoRR*, abs/2109.04838.

Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoît Crabbé, Laurent Besacier, and Didier Schwab. 2019. Flaubert: Unsupervised language model pre-training for french.

Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. 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.

Machel Reid, Edison Marrese-Taylor, and Yutaka Matsuo. 2021. Subformer: Exploring Weight Sharing for Parameter Efficiency in Generative Transformers. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. In *NeurIPS EMC^{2}Workshop*.

Shaden Smith, Mostofa Patwary, Brandon Norick, Patrick LeGresley, Samyam Rajbhandari, Jared Casper, Zhun Liu, Shrimai Prabhumoye, George Zerveas, Vijay Korthikanti, Elton Zhang, Rewon Child, Reza Yazdani Aminabadi, Julie Bernauer, Xia Song, Mohammad Shoeybi, Yuxiong He, Michael Houston, Saurabh Tiwary, and Bryan Catanzaro. 2022. Using deepspeed and megatron to train megatron-turing nlg 530b, a large-scale generative language model.

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

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

Benjamin van der Burgh and Suzan Verberne. 2019. The merits of universal language model fine-tuning for small datasets - a case with dutch book reviews. *CoRR*, abs/1910.00896.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. volume abs/1706.03762.

Shijie Wu and Mark Dredze. 2020. Are all languages created equal in multilingual bert? *CoRR*, abs/2005.09093.

A Loss Variation

In this appendix, we further illustrate with Figure 2 the differences in convergence for the MLM loss

and the distillation loss for the other 5 languages, and consistently observe the findings presented in Section 4. The MLM loss continues to have a much larger mean value than the distillation loss, while also having issues converging with various spikes in the loss, while the training for the distillation loss is much more stable. This is line with our hypothesis that the large α_1 values prioritize the distillation over the language modelling objective, thus not allowing the model to further improve from the teacher.

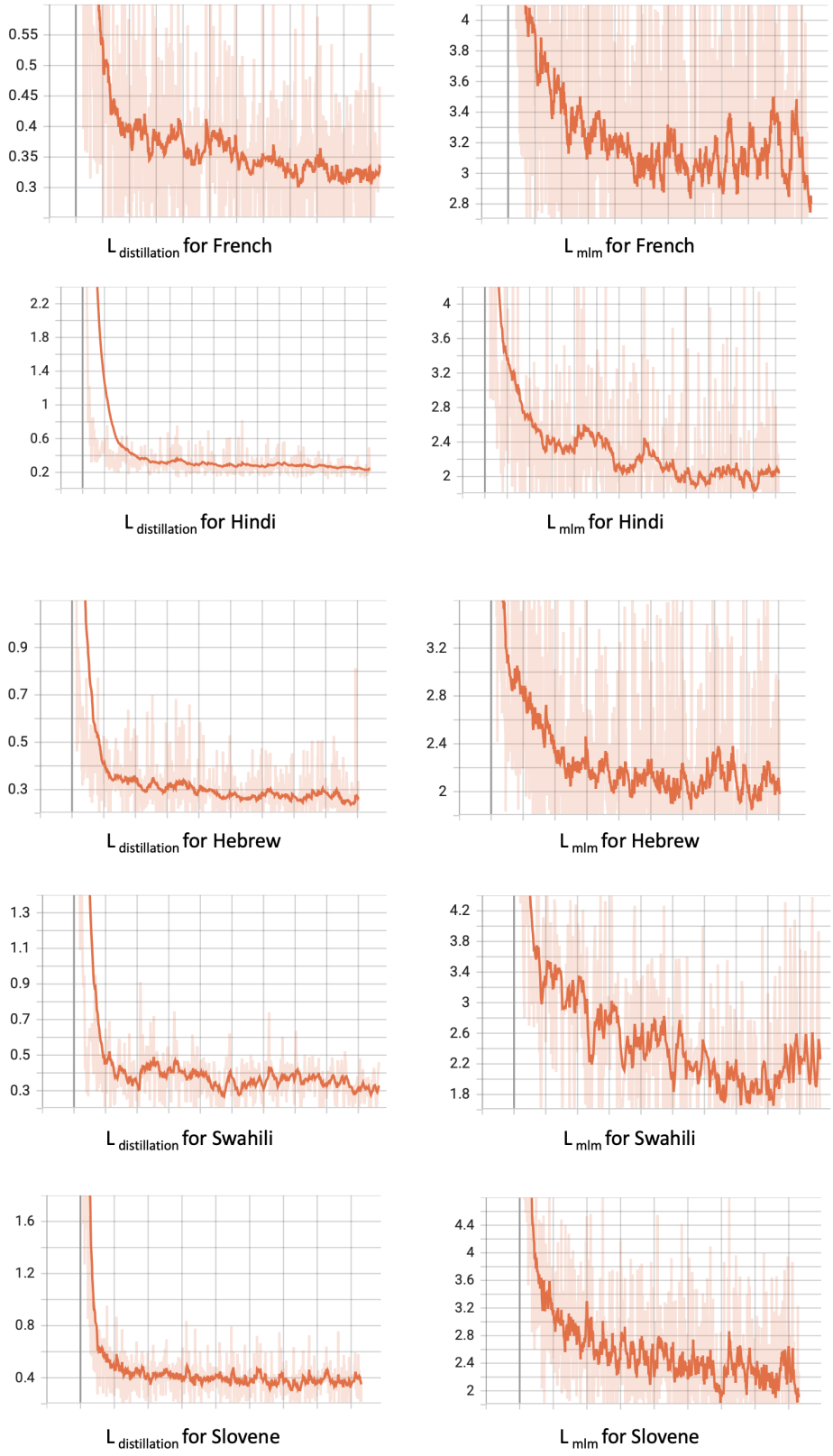

Figure 2: Distillation loss (left) and MLM loss (right) for the other 5 languages: French, Hindi, Hebrew, Swahili and Slovene.