# Extracting General-use Transformers for Low-resource Languages via Knowledge Distillation

**Jan Christian Blaise Cruz** and **Alham Fikri Aji**
MBZUAI
`{jan.cruz,alham.fikri}@mbzuai.ac.ae`

## Abstract

In this paper, we propose the use of simple knowledge distillation to produce smaller and more efficient single-language transformers from Massively Multilingual Transformers (MMTs) to alleviate tradeoffs associated with the use of such in low-resource settings. Using Tagalog as a case study, we show that these smaller single-language models perform on-par with strong baselines in a variety of benchmark tasks in a much more efficient manner. Furthermore, we investigate additional steps during the distillation process that improves the soft-supervision of the target language, and provide a number of analyses and ablations to show the efficacy of the proposed method[1].

## 1 Introduction

To curb the detrimental effects of pretraining with very little pretraining data in a low-resource language, most works opt to use pretrained *Massively Multilingual Transformers* (MMTs) such as mBERT (Devlin et al., 2019) and mDeBERTa (He et al., 2021b,a) instead.

However, this comes with a number of tradeoffs. Finetuning in only one language causes negative interference in a model that compresses many languages within a limited parameter budget (Berend, 2022; Lee and Hwang, 2023). This would mean that an MMT, in theory, would perform worse than using a transformer pretrained in one specific language (Cruz and Cheng, 2022; Pfeiffer et al., 2022). Additionally, MMTs are unnecessarily costly as most researchers who use them are only interested in one language among many – this is most especially the case in low-resource language research communities that also suffer from a lack of computational resources (Alabi et al., 2022; Ansell et al., 2023).

In this work, we propose the use of simple knowledge distillation to extract robust and efficient single-language pretrained transformers from an MMT. We study a number of intermediate steps that improve the distillation method, such as target-language conditioning and student initialization. We then compare the performance of our extracted models on strong baselines on a variety of benchmark tasks and perform ablations and analyses to pinpoint the sources of strong performance from our simple method.

## 2 Methodology

### 2.1 Distillation

To simplify the study, we limit ourselves to one type of MMT – mBERT (`bert-base-multilingual-cased`) (Devlin et al., 2019) – and one language (Tagalog) for both distillation and task finetuning.

In the interest of resource-scarce research settings, the proposed method is *very* simple and computationally cheap: we take a pretrained mBERT and freeze its weights. We then construct a blank student transformer with a modified architecture and use teacher-student model distillation (Hinton et al., 2015) using masked language modeling (MLM) as the main objective. No further tricks, post-processing, or augmentations are done after distillation. We use OSCAR's Tagalog split (Ortiz Suárez et al., 2019) as the training corpus for knowledge distillation.

Mathematically, we optimize our distillation loss as a mix of the weighted sum of the Kullback-Leibler (KL) divergence and the MLM loss between the student and teacher's output logits:

$$\begin{aligned} L_{\text{distil}} &= \alpha_{\text{KL}}\text{KL}(out_{student}||out_{teacher}) + \\ &= \alpha_{\text{MLM}}L_{\text{MLM}}(out_{student}, out_{teacher}) \end{aligned} \tag{1}$$

where $\alpha_{kl}$ and $\alpha_{mlm}$ represent the weights of the

[1] Code can be found in the following repository: `https://github.com/jcblaisecruz02/nlp805-distillation`

*Proceedings of the First Workshop on Language Models for Low-Resource Languages (LoResLM 2025)*, pages 219–224
January 20, 2025. ©2025 Association for Computational Linguistics

|  | Teacher | Base | Tiny |
|---|---|---|---|
| Hidden Dim | 768 | 768 | 312 |
| Intermediate Size | 3072 | 3072 | 1200 |
| Layers | 12 | 6 | 4 |
| Attention Heads | 12 | 12 | 12 |
| Max Positions | 512 | 512 | 512 |

Table 1: Student vs Teacher hyperparameters. We reduce the hidden dimensionality, feedforward intermediate size, and the number of layers. The number of attention heads and max number of positions (tokens) are kept the same.

divergence and the MLM loss respectively to the final distillation loss. For our experiments, we use cross entropy as our MLM loss. Note that we also apply a temperature parameter to cool down the logits of the student and teacher and encourage diversity in outputs.

This gives us a distilled version of the pretrained mBERT but without the risk of negative interference caused by parameter sharing between multiple languages in the model during downstream finetuning. We produce two distilled models this way which we refer to as `dBERT Base` and `dBERT Tiny`, depending on the hyperparameters used. Hyperparameter choices used for distillation are listed on Table 1. We run distillation for a total of three epochs on the training dataset.

### 2.2 Downstream Finetuning

To measure the performance of the distilled model on downstream tasks, we finetune on several benchmarks in Tagalog:

- TLUnified NER (Miranda, 2023) – NER classification dataset developed using the TLUnified (Cruz and Cheng, 2022) corpus.
- Hatespeech Filipino (Cabasag et al., 2019) – a text classification dataset on hatespeech mined from election tweets in Tagalog.
- NewsPH NLI (Cruz et al., 2021) – an entailment dataset created using news articles in Tagalog.

We measure accuracy for the hate speech classification and NLI tasks and measure F1 for the NER task. We compare the performance of our models with mBERT (as the teacher), Tagalog-RoBERTa (Cruz and Cheng, 2022) (to compare against a full model trained on Tagalog), DistilmBERT (Sanh et al., 2020) (a full distilled version of mBERT retaining all the languages supported), and from-scratch training (where a blank model is directly tuned on the downstream task).

## 3 Results and Discussion

A summary of the results can be found on Table 2.

We can see that our models perform strongly across the three benchmark tasks. For the hate speech classification and NLI tasks, our dBERT Base model outperforms its teacher mBERT as well as the distilled DistilmBERT version with an almost 2x speedup in terms of training time. This shows that the method, albeit simple, works well to produce general-use transformers for these tasks. Performance lags slightly behind on NER, which we assume is a harder task for an extracted model as there are a lot of named entities in the vocabulary from other languages that are not completely removed and present a significant amount of negative interference. We investigate these behaviors further in ablations.

The dBERT Tiny variant showed strong results that came close to the baselines on hate speech classification but lags behind the other models in all other tasks. We hypothesize that this is due to the size of the model not having enough capacity to fully capture the teacher's representation of the target language given that the source representation space is extremely large due to the presence of other languages.

Unsurprisingly, RoBERTa Tagalog performs the best in all three tasks given that it is a full-sized BERT-type model that is trained solely in Tagalog. The mBERT and DistilmBERT models are likewise strong performers but are much slower during training than the dBERT models which has a significant impact on research in low-resource languages where computing is often scarce.

Overall, this provides empirical evidence that distilling a general-purpose transformer from a larger MMT yields robust results despite the method's relative simplicity.

### 3.1 Can we outperform the teacher with less training data?

One surprising result from the benchmarking is the fact that the student model `dBERT Base` outperforms its teacher `mBERT` on hate speech classification by 1.86% in accuracy. This suggests that a smaller dataset may be as-effective for isolating

| | TLUnified NER | | Hatespeech | | NewsPH NLI | | Avg. |
|---|---|---|---|---|---|---|---|
| | F1 | Runtime | Accuracy | Runtime | Accuracy | Runtime | Speedup |
| From Scratch | 0.4818 | 71s | 0.7382 | 617s | 0.5392 | 25819s | |
| Tagalog RoBERTa | 0.8939 | 66s | 0.7767 | 606s | 0.9406 | 25798 | |
| mBERT | 0.8925 | 70s | 0.7543 | 618s | 0.9318 | 25811s | |
| DistilmBERT | 0.8818 | 44s | 0.7372 | 366s | 0.9172 | 15316s | 1.68x |
| dBERT Base (Ours) | 0.8074 | 44s | 0.7729 | 309s | 0.9188 | 13006s | 1.97x |
| dBERT Tiny (Ours) | 0.6085 | 31s | 0.7261 | 107s | 0.8328 | 4917s | 5.23x |

Table 2: Main Results. Accuracy refers to evaluation accuracy on the test set. Runtime refers to the total amount of time (in seconds) that it takes to finetune on the task dataset (rounded down). Avg. Speed refers to the factor by which the distilled models are faster compared to mBERT (averaged across the three tasks).

| Model | Accuracy | Perf. Diff. |
|---|---|---|
| dBERT @100% | 0.7729 | +0.0186 |
| dBERT @80% | 0.7200 | -0.0343 |
| dBERT @50% | 0.7108 | -0.0435 |
| mBERT | 0.7543 | |

Table 3: Ablation on the amount of training data used for distillation. Data size refers to how much training data is retained. Accuracy represents accuracy on the test set of Hatespeech Filipino. Perf. Diff. refers to the difference in the performance of the finetuned distilled model against mBERT's finetuned performance on Hatespeech Filipino.

| Model | F1 | Perf. Diff. |
|---|---|---|
| dBERT | 0.8074 | -0.0851 |
| dBERT Conditioned | 0.7587 | -0.1338 |
| mBERT | 0.8925 | |
| mBERT Conditioned | 0.8900 | -0.0025 |

Table 4: Ablation on teacher conditioning. Perf. Diff. refers to the difference in the performance of the finetuned distilled models against mBERT's finetuned performance on TLUnified NER.

performance for one language in an MMT as opposed to using a larger one. To further investigate this, we distill more versions of `dBERT Base` using 80% and 50% of the original training data and rerun the experiments for Hatespeech classification. A summary of the results can be found on Table 3.

We see that when reducing the training data used for distillation, the performance starts to be impacted but not by a significant margin. The original mBERT model only outperforms dBERT @80% training data by around 3.43% accuracy on hate speech classification. Once we go down to half the training data, the original only outperforms the student model by 4.35% – a sub 1% degradation in performance! We hypothesize that this is connected to the amount of pretraining data used for the target language in the original MMT. The more robust the MMT's performance is in the target language, the less data might be needed to retain that performance post-distillation.

### 3.2 Can we improve the student by properly conditioning the teacher?

In our experiments, the NER results are lackluster when compared against `DistilmBERT`, which was a distilled version of the original `mBERT`. We assume that this is because the teacher model is not conditioned properly on the target language and experiences some form of negative transfer during the distillation process as the source representation space is very large. To curb this effect, we experiment with first conditioning the teacher on the training dataset by finetuning using masked language modeling *before* performing distillation. We then finetune on the NER downstream task and evaluate after to compare performance. A summary of the results can be found in Table 4.

In the initial results, a conditioned mBERT model experiences very minimal performance degradation when finetuned on MLM prior to distillation by a factor of 0.0025 F1. Once we distill, we find that a student distilled from a conditioned teacher performs significantly worse than without teacher conditioning. We hypothesize that the downstream performance suffers because there is some negative interference occurring in the teacher model during conditioning – a consequence of having a majority of its parameters being dedicated for languages other than the target language we want – and this creates further instability during distillation to the student.

This suggests that further conditioning of the teacher to the target language may not be necessary

| Model | F1 | Perf. Diff. |
|---|---|---|
| dBERT | 0.8074 | -0.0851 |
| dBERT Init | 0.7597 | -0.1330 |
| dBERT Init+Freeze | 0.7659 | -0.1266 |
| mBERT | 0.8925 | |

Table 5: Ablation on weight initialization. Perf. Diff. refers to the difference in the performance of the finetuned distilled models against mBERT's finetuned performance on TLUnified NER.

for extracting a language-specific model.

### 3.3 What if we initialize the student weights from the teacher?

In this work, we aim to extract general-use language-specific models from large MMTs in the most straightforward way possible, which is why we originally opted to not do any weight initialization and layer copying tricks commonly found in most knowledge distillation works (Jiao et al., 2020). However, it will be useful to see how much of a contribution weight initialization is in comparison to our method. For this ablation, we perform the simplest initialization commonly used – copying the embedding weights of the teacher – and then freezing them before beginning distillation. Like the previous ablation, we evaluate on the NER downstream task to compare performance with our baselines. A summary of the results can be found in Table 5.

We see that interestingly, the student model performs worse when the embedding layer is initialized from the teacher weights by a factor of -0.1330 F1 score. Freezing the embedding layer while performing distillation does not inhibit the performance loss significantly – the model now performs 0.1266 F1 worse than the original dBERT model without initialization.

While embedding layer initialization is often useful for retaining teacher knowledge when distilling multilingual models (Sanh et al., 2020), we can see some empirical evidence that it might not be as useful in cases where we do not want to recapture the entirety of the original embedding space. For extracting single-language models from multilingual models, it may be useful to not copy the embeddings at all.

## 4 Related Work

Knowledge distillation is an established tool in modern NLP research, especially after the release of BERT in 2018. Most works such as DistilBERT and TinyBERT (Jiao et al., 2020) aim to distill the full model while retaining all languages that may be incorporated in the original training data. These models perform well across a number of cross-lingual benchmarks such as XNLI (Conneau et al., 2018), but represent a challenge in real-world use especially for low-resource languages.

Recent works have begun to use knowledge distillation for smaller, targeted use-case models. Wibowo et al. (2024) explores student initializations to improve task-based performance with minimal training needed, and Ansell et al. (2023) distills smaller models for the goal of efficiently producing stronger task-based models via further distillation. However, most of these works focus directly on the end task, instead of creating a general-use case student model that is targeted for one language specifically.

## 5 Future Work

The current method provides a strong way to distill a language-specific general-use model from a much larger MMT, while being flexible enough to function as the base for more targeted tasks. For future work, the following may be explored as an augmentation to the current method:

**Extrapolating to an Unseen Language** – Much like in BLOOM+1 (Yong et al., 2023), we could explore teacher conditioning to add an unseen language to an existing language model.

**General Purpose LLMs** – Moving beyond small pretrained models, we can explore the use of the same method for general purpose multilingual LLMs such as Aya (Üstün et al., 2024) and BLOOMZ (Muennighoff et al., 2022) to see if we can transfer learned instruction-following performance on a language-specific student model.

## 6 Conclusion

In this work, we present an extremely simple method of extracting general-use language-specific transformers from pretrained MMTs that retain the robust performance of the original teacher models. These models and the process of obtaining them are both ideal for research in low-resource languages as both the compute resources and the data available for researchers in these areas are often very scarce. For future work, we present a number of augmentations that can be explored from this relatively simple method, such as unseen language extrapolation, and extension to large language models.

## Limitations

While we provide good empirical results, we acknowledge a number of limitations in our work, mostly due to a lack of compute resources. We study only one MMT – mBERT – to simplify the study. In future work, we aim to have a more diverse set of MMTs to test the method on. We also only limit the study to Tagalog as a case study. For future work, we aim to test the method on a wider variety of low-resource languages, as well as using a benchmark high-resource langauge to compare ablations against. Additionally, our distillation step is quick (three epochs) due to the size of the training dataset and limitations in compute. For future work, we aim to identify the relationship between the size of the training dataset, size of the target language in the pretraining dataset, and the length of distillation.

## References

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

Alan Ansell, Edoardo Maria Ponti, Anna Korhonen, and Ivan Vulić. 2023. Distilling efficient language-specific models for cross-lingual transfer. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8147–8165, Toronto, Canada. Association for Computational Linguistics.

Gábor Berend. 2022. Combating the curse of multilinguality in cross-lingual WSD by aligning sparse contextualized word representations. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2459–2471, Seattle, United States. Association for Computational Linguistics.

Neil Vicente Cabasag, Vicente Raphael Chan, Sean Christian Lim, Mark Edward Gonzales, and Charibeth Cheng. 2019. Hate speech in philippine election-related tweets: Automatic detection and classification using natural language processing. *Philippine Computing Journal, XIV No*, 1.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel R. 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*. Association for Computational Linguistics.

Jan Christian Blaise Cruz and Charibeth Cheng. 2022. Improving large-scale language models and resources for Filipino. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 6548–6555, Marseille, France. European Language Resources Association.

Jan Christian Blaise Cruz, Jose Kristian Resabal, James Lin, Dan John Velasco, and Charibeth Cheng. 2021. Exploiting news article structure for automatic corpus generation of entailment datasets. In *PRICAI 2021: Trends in Artificial Intelligence: 18th Pacific Rim International Conference on Artificial Intelligence, PRICAI 2021, Hanoi, Vietnam, November 8–12, 2021, Proceedings, Part II 18*, pages 86–99. Springer.

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.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021a. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *Preprint*, arXiv:2111.09543.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021b. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *Preprint*, arXiv:1503.02531.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2020. Tinybert: Distilling bert for natural language understanding. *Preprint*, arXiv:1909.10351.

Jaeseong Lee and Seung-won Hwang. 2023. Multilingual lottery tickets to pretrain language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9387–9398, Singapore. Association for Computational Linguistics.

Lester James V. Miranda. 2023. Developing a named entity recognition dataset for tagalog. *Preprint*, arXiv:2311.07161.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey

Schoelkopf, et al. 2022. Crosslingual generalization through multitask finetuning. *arXiv preprint arXiv:2211.01786*.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous pipelines for processing huge corpora on medium to low resource infrastructures. Proceedings of the Workshop on Challenges in the Management of Large Corpora (CMLC-7) 2019. Cardiff, 22nd July 2019, pages 9 – 16, Mannheim. Leibniz-Institut für Deutsche Sprache.

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

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *Preprint*, arXiv:1910.01108.

Haryo Akbarianto Wibowo, Thamar Solorio, and Alham Fikri Aji. 2024. The privileged students: On the value of initialization in multilingual knowledge distillation. *arXiv preprint arXiv:2406.16524*.

Zheng Xin Yong, Hailey Schoelkopf, Niklas Muennighoff, Alham Fikri Aji, David Ifeoluwa Adelani, Khalid Almubarak, M Saiful Bari, Lintang Sutawika, Jungo Kasai, Ahmed Baruwa, Genta Winata, Stella Biderman, Edward Raff, Dragomir Radev, and Vassilina Nikoulina. 2023. BLOOM+1: Adding language support to BLOOM for zero-shot prompting. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11682–11703, Toronto, Canada. Association for Computational Linguistics.

Ahmet Üstün, Viraat Aryabumi, Zheng-Xin Yong, Wei-Yin Ko, Daniel D'souza, Gbemileke Onilude, Neel Bhandari, Shivalika Singh, Hui-Lee Ooi, Amr Kayid, Freddie Vargus, Phil Blunsom, Shayne Longpre, Niklas Muennighoff, Marzieh Fadaee, Julia Kreutzer, and Sara Hooker. 2024. Aya model: An instruction finetuned open-access multilingual language model. *arXiv preprint arXiv:2402.07827*.