# Is Modularity Transferable?
# A Case Study through the Lens of Knowledge Distillation

**Mateusz Klimaszewski**[1], **Piotr Andruszkiewicz**[1], **Alexandra Birch**[2]
Institute of Computer Science, Warsaw University of Technology[1]
School of Informatics, University of Edinburgh[2]
mateusz.klimaszewski.dokt@pw.edu.pl

**Abstract**

The rise of Modular Deep Learning showcases its potential in various Natural Language Processing applications. Parameter-efficient fine-tuning (PEFT) modularity has been shown to work for various use cases, from domain adaptation to multilingual setups. However, all this work covers the case where the modular components are trained and deployed within one single Pre-trained Language Model (PLM). This model-specific setup is a substantial limitation on the very modularity that modular architectures are trying to achieve. We ask whether current modular approaches are transferable between models and whether we can transfer the modules from more robust and larger PLMs to smaller ones. In this work, we aim to fill this gap via a lens of Knowledge Distillation, commonly used for model compression, and present an extremely straightforward approach to transferring pre-trained, task-specific PEFT modules between same-family PLMs. Moreover, we propose a method that allows the transfer of modules between incompatible PLMs without any change in the inference complexity. The experiments on Named Entity Recognition, Natural Language Inference, and Paraphrase Identification tasks over multiple languages and PEFT methods showcase the initial potential of transferable modularity.

**Keywords:** Modular Deep Learning, Parameter-Efficient Fine-tuning, Pre-trained Language Models

## 1. Introduction

Modular Deep Learning has recently garnered interest as a paradigm that builds upon the idea that a model is a combination of modules with control of the information flow. This paradigm allows for the transfer of learning from one task or language to another, compositionality of the modules and parameter efficiency (Pfeiffer et al., 2023). For instance, modules allow for efficient (parameter-wise) fine-tuning of Large Language Models (Hu et al., 2022), enhance task-level generalisation (Ponti et al., 2023), improve multilingual models (Bapna and Firat, 2019), offer zero-shot capabilities (Philip et al., 2020) and enable cross-lingual (Ansell et al., 2022) or cross-domain (Klimaszewski et al., 2023) knowledge transfer. Furthermore, repositories that store pre-trained modules like AdapterHub (Pfeiffer et al., 2020a) promote the re-usability of previously trained components to new use cases.

The current modular approaches primarily focus on transferring knowledge to new languages, domains, or tasks. However, prior research assumes that the base model remains constant and overlooks the concept of *transferable modularity*, which entails the potential to transfer modules between different models. From a practical perspective, the effective utilisation of the *transferable modularity property* can reduce the computational burden, especially given the ongoing scaling of Large Language Models (Brown et al., 2020; Touvron et al., 2023), allowing for broader re-usability. Moreover,

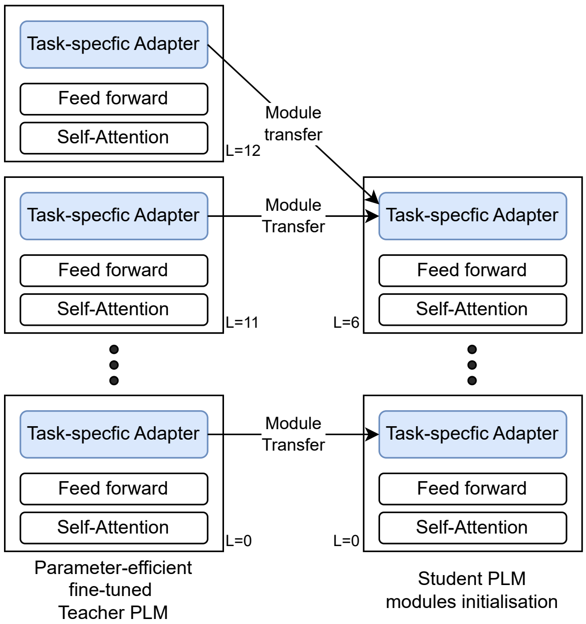


Figure 1: The most straightforward case of transferable modularity. The teacher model is first trained on a task using PEFT, e.g. Adapters, and then the student PEFT modules, prior to fine-tuning, are initialised with the teacher weights.

transferring modules from larger to smaller models can significantly enhance knowledge transfer. And finally, even the term “modularity” inherently implies the transfer property, suggesting that modular ap-



*LREC-COLING 2024*, pages 9352–9360

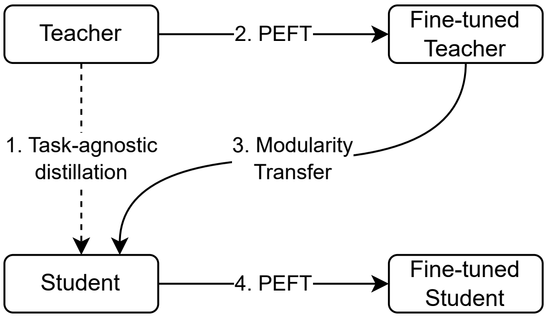


Figure 2: The schema of transferable modularity experiment. We investigate setups where the teacher-student pair result from task-agnostic distillation or are independently trained models.

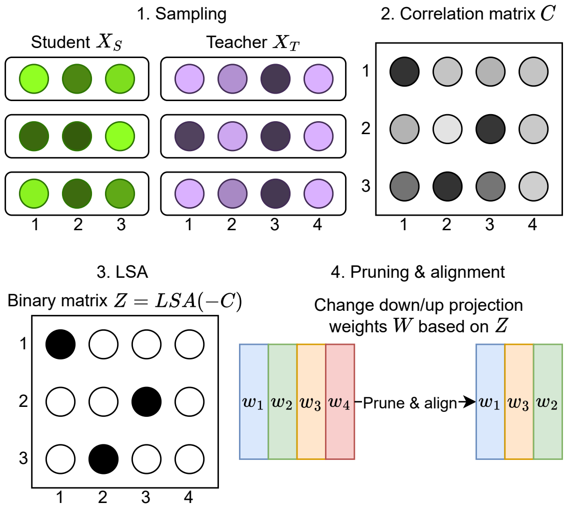


Figure 3: Toy example of adapting the PEFT modules in the case of mismatched dimensionality. Based on the sampled embeddings (1.), correlation matrix $C$ is calculated (2.) and reduced via $LSA$ to a binary matrix $Z$ (3.). In the last step (4.), the pruning and alignment mapping function (derived from $Z$) is applied to down/up projection matrices of LoRA/Adapter modules and match dimensions.

proaches should not be limited to a specific base model.

In this work, we aim to initialise the research objective of *transferable modularity*. We focus on a setup similar to Knowledge Distillation (KD) (Hinton et al., 2015), i.e. where we have two differently sized PLMs (through the paper, we adopt the KD nomenclature, where the bigger model is called a teacher and the smaller - student). Unlike KD, we do not want to use the teacher model's output directly to train a student but use exclusively its fine-tuned PEFT modules.

We show that given matching PLMs (e.g. BERT (Devlin et al., 2019) and DistilBERT (Sanh et al., 2019)), it is possible to use pre-trained modules like Adapters (Houlsby et al., 2019; Pfeiffer et al., 2021) or LoRA (Hu et al., 2022) as a better starting point for parameter-efficient (PE) fine-tuning of a smaller student PLM (see Figure 1). Moreover, we investigate a more challenging setup where the models are *incompatible*, i.e., have different internal dimensionality, and adapt modules via the proposed pruning and alignment method (without inference-time overhead).

To summarise, our contributions are as follows[1]:

- We define the property of transferable modularity.
- We investigate transferable modularity in matching and incompatible PLMs, proposing a pruning and alignment method for the latter.

## 2. Transferable Modularity

The high-level idea of our study is presented in Figure 2. Given a pair of PLMs, a teacher and a student, we aim to transfer the parameter-efficient (PE) modules from the teacher to the student. First, we use a PEFT technique to train the teacher and its PE modules. Then, we "move" the modules from the teacher and insert them into the student, followed by PEFT of the student. This approach means that PE modules of the student have non-random prior initialisation during training.

We consider two setups: (1) matching PLMs and (2) incompatible PLMs. The former uses a shallow version of a teacher with task-agnostic distillation as a student (Kim and Hassan, 2020). This case means that the models represent the same knowledge, have the same hidden dimensionality, and the only difference is the depth of the model. The latter represents a generalised version, where the models are differently parameterised (in terms of latent space size) and they are independently trained. We propose a parameter-free, sample-based pruning and alignment method to answer dimensionality mismatch.

### 2.1. Pruning and Alignment

In the case of incompatible PLMs, a dimensionality mismatch problem causes two main issues for transferable modularity. First, the module expects different (higher) dimensionality. Additionally, there exists an alignment discrepancy between the latent spaces of the two models, i.e. if the models have learned the same features, we do not have any guarantee of their placement in the latent space - their indices.

[1] Code available at `https://github.com/mklimasz/transferable-modularity`

A crucial element of a successful Knowledge Distillation framework is the computational overhead; therefore, we propose an offline, parameter-free solution that does not change the final student model. The method presented in Figure 3 consists of four phrases:

- sampling
- calculating correlation
- solving linear sum assignment (LSA) problem
- pruning & alignment

At first, we sample matching embeddings that would be an input to a PEFT module (we denote the set of embeddings $X_s$ for student and $X_t$ for teacher with $x_s \in X_s$ and $x_t \in X_t$). We store embeddings per layer $l$ (for clarity, we omit the notation of the layer).

In the next step, we establish a correlation matrix between latent spaces. We calculate Pearson's correlation coefficient matrix $C$. $C_{ij}$ is a correlation between the $i$ dimension of a $x_s$ and the $j$ dimension of a $x_t$ embedding.

Given the correlation matrix, we attempt to find the best possible alignment. We define the problem as a linear sum assignment (LSA) (Crouse, 2016) to establish the optimal mapping. As LSA calculates the minimum cost assignment, we use $-C$ as an input to the LSA algorithm. The algorithm produces a binary matrix $Z$ where $Z_{ij} = 1$ means that the $i$ index of $X_s$ is mapped to $j$ of $X_t$.

$$min \sum_i \sum_j (-C_{ij}) Z_{ij}$$

Finally, using the calculated assignment indices, we remove not-mapped weights from both down/up projection weights $W$ of PEFT modules.

# 3. Experiments

## 3.1. Datasets

To evaluate our method, we benchmark it on three tasks: Named Entity Recognition (NER), Paraphrase Identification (PI) and Natural Language Inference (NLI) using multilingual datasets: WikiNeural (Tedeschi et al., 2021), PAWS-X (Yang et al., 2019) and XNLI (Conneau et al., 2018) covering jointly a set of over 20 languages[2].

[2]Arabic, Bulgarian, Chinese, Dutch, English, French, German, Hindi, Italian, Japanese, Korean, Greek, Polish, Portuguese, Russian, Spanish, Swahili, Thai, Turkish, Urdu, Vietnamese

| Model | Params | Layers | Hidden dim |
|---|---|---|---|
| D'mBERT | $135M$ | 6 | 768 |
| mBERT | $178M$ | 12 | 768 |
| XLM-R$_{\text{BASE}}$ | $278M$ | 12 | 768 |
| XLM-R$_{\text{LARGE}}$ | $560M$ | 24 | 1024 |

Table 1: Parameters, layer count and hidden dimension size of the evaluated models.

## 3.2. Training Setup

We fine-tune multilingual models for each language/task pair using two PEFT methods: Adapter (architecture of Pfeiffer et al. (2021), bottleneck size of 96) and LoRA (rank 8). We provide the training setup details for each dataset in Appendix A.

For teacher-student pairs, we define two configurations:

- *matching*: multilingual BERT (mBERT[3], teacher) – multilingual DistilBERT (D'mBERT[4], student)
- *incompatible*: XLM-RoBERTa Large (XLM-R$_{\text{LARGE}}$[5], teacher) – XLM-RoBERTa Base (XLM-R$_{\text{BASE}}$[6], student) (Conneau et al., 2020)

We report the relevant hyper-parameters of the models in Table 1. As the models have mismatched layer counts, we test two approaches: skip modules (denoted SKIP, e.g., transfer every second module) or average them (denoted AVG, e.g., average the first and second layer's teacher module and transfer to the first module of a student).

## 3.3. Baselines and Metrics

For both *matching* and *incompatible* experiments, we define the following structure. As an upper bound of our evaluation, we provide the teacher results after PEFT (Step 2 in Figure 2). The baseline is a parameter-efficient fine-tuned student with default modules initialisation (i.e. omitting Step 3 in Figure 2).

We report F1 for NER and Accuracy for PI and NLI tasks with an average score over all languages in Section 4. The detailed per-language results are provided in Appendix B.

# 4. Results and Discussion

## 4.1. Matching Models

Table 2 presents the results of the *matching* experiments. The prefix TM denotes the transfer modu-

[3]`bert-base-multilingual-cased`
[4]`distilbert-base-multilingual-cased`
[5]`xlm-roberta-large`
[6]`xlm-roberta-base`

| | NER (F1) | | PI (Acc) | | NLI (Acc) | |
|---|---|---|---|---|---|---|
| | AVG | REL | AVG | REL | AVG | REL |
| | | | Adapter | | | |
| Teacher | $95,35$ | | $82,60$ | | $67,98$ | |
| Student | $92,94$ | $-2,41$ | $71,32$ | $-11,28$ | $62,12$ | $-5,86$ |
| TM-Student$_{AVG}$ | $93,02$ | $-2,32$ | $72,96$ | $-9,64$ | $62,33$ | $-5,65$ |
| TM-Student$_{SKIP}$ | $93,45$ | $-1,90$ | $75,11$ | $-7,49$ | $63,01$ | $-4,97$ |
| | | | LoRA | | | |
| Teacher | $93,27$ | | $74,68$ | | $63,00$ | |
| Student | $90,09$ | $-3,18$ | $65,80$ | $-8,88$ | $60,56$ | $-2,43$ |
| TM-Student$_{AVG}$ | $90,63$ | $-2,64$ | $68,52$ | $-6,16$ | $60,53$ | $-2,47$ |
| TM-Student$_{SKIP}$ | $90,80$ | $-2,47$ | $70,69$ | $-3,99$ | $60,52$ | $-2,47$ |

Table 2: Results of the *matching* PLMs experiment. We report an average score (F1 or Accuracy) over all the datasets' languages and a relative performance gap to the teacher model.

| | NER (F1) | | PI (Acc) | |
|---|---|---|---|---|
| | AVG | REL | AVG | REL |
| | | Adapter | | |
| Teacher | $95,34$ | | $88,81$ | |
| Student | $93,30$ | $-2,04$ | $84,12$ | $-4,69$ |
| TM-Student$_{SKIP}$ | $93,34$ | $-2,00$ | $84,27$ | $-4,54$ |
| | | LoRA | | |
| Teacher | $93,64$ | | $87,03$ | |
| Student | $90,83$ | $-2,82$ | $78,72$ | $-8,31$ |
| TM-Student$_{SKIP}$ | $90,84$ | $-2,80$ | $78,64$ | $-8,39$ |

Table 3: Results of the *incompatible* PLMs experiment.

larity experiments. The initialisation of the modules transferred from the teacher PLM improved over a default initialisation on average in all the evaluated tasks. Moreover, the SKIP method presents consistency; the difference compared to the baseline was positive across most tasks and languages ($88,7\%$ cases). While at times the improvement was marginal ($+0.02$ gain in Swahili in NLI task), in most cases, as averages indicate, our approach significantly closes the gap to the teacher model (e.g. $+4$ point improvement in Korean on PAWS-X datasets using Adapter or over $+2$ in Spanish LoRA on XNLI). SKIP struggles to outperform the baseline exclusively on XNLI when using LoRA. The results are on par; however, even the teacher models struggle with the task, and the knowledge that can be transferred is relatively limited.

The SKIP outperforms AVG across all the experiments. Considering the results and the findings of van Aken et al. (2019) indicating that the Transformer-based models have internal modularity and each layer has its own defined task, we hypothesise that the averaging might not reflect these phenomena. Therefore, in the *incompatible* experiment, we evaluated just the SKIP method.

## 4.2. Incompatible Models

We present the results of the evaluation in Table 3. In the case of non-distilled PLMs, the TM method does not significantly outperform the baseline. The changes are uneven; while the transfer shows improvement up to almost $+2$ points in Korean PAWS-X, it can also decrease the performance as in French PAWS-X, losing $-1.05$.

The disparity between *matching* and *incompatible* experiments can be attributed to alignment challenges. Models subjected to distillation exhibit reliable alignment, thanks to the inclusion of an auxiliary loss term such as the cosine embedding loss (Sanh et al., 2019) in the task-agnostic distillation process. In contrast, the correlation-based method encounters difficulties when dealing with models of greater depth. Notably, the LSA algorithm yields lower scores for deeper layers. Considering the different representations required for each language and task pair, this outcome implies that independently trained models require more robust alignment techniques to ensure consistent modularity transfer across all encoded features.

## 5. Conclusions

In this work, we present a case study of transferable modularity property. We evaluate current modular techniques in two scenarios: (1) *matching*, where a student is a shallow, task-agnostic distillation of the teacher and (2) *incompatible*, where a student is independently trained, a shallower model with mismatched internal dimensionality.

The results show that the current modular approach can be transferable as the modules from a matching teacher improve the PEFT of a student model. However, when a student is not distilled from the teacher, the evaluated techniques are inconsistent under the transfer condition, showing the limitation of the current modular methods. We hope this study will inspire future work on modular techniques to consider the transferable modularity property under a more challenging incompatible models scenario.

## 6. Acknowledgements

Part of this work was funded from the European Union's Horizon Europe Research and Innovation program under Grant Agreement No 101070631 and from the UK Research and Innovation (UKRI) under the UK government's Horizon Europe funding guarantee (Grant No 10039436).

The computations in this work were performed on Poznań Supercomputing and Networking Center and Baskerville. The Baskerville Tier 2 HPC was funded by the EPSRC and UKRI through the World Class Labs scheme (EP/T022221/1) and the Digital Research Infrastructure programme (EP/W032244/1) and is operated by Advanced Research Computing at the University of Birmingham.

## 7. Bibliographical References

Alan Ansell, Edoardo Ponti, Anna Korhonen, and Ivan Vulić. 2022. Composable sparse fine-tuning for cross-lingual transfer. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1778–1796, Dublin, Ireland. Association for Computational Linguistics.

Ankur Bapna and Orhan Firat. 2019. Simple, scalable adaptation for neural machine translation. 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 1538–1548, Hong Kong, China. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D 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 Ziegler, Jeffrey Wu, Clemens Winter, Chris 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. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

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

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

David F. Crouse. 2016. On implementing 2d rectangular assignment algorithms. *IEEE Transactions on Aerospace and Electronic Systems*, 52(4):1679–1696.

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.

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. *ArXiv*, abs/1503.02531.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Young Jin Kim and Hany Hassan. 2020. FastFormers: Highly efficient transformer models for natural language understanding. In *Proceedings of SustaiNLP: Workshop on Simple and Efficient Natural Language Processing*, pages 149–158, Online. Association for Computational Linguistics.

Mateusz Klimaszewski, Zeno Belligoli, Satendra Kumar, and Emmanouil Stergiadis. 2023. Gated adapters for multi-domain neural machine translation. In *ECAI 2023 - 26th European Conference on Artificial Intelligence*, volume 372 of *Frontiers in Artificial Intelligence and Applications*, pages 1264–1271. IOS Press.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2021. AdapterFusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 487–503, Online. Association for Computational Linguistics.

Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulić, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. 2020a. Adapterhub: A framework for adapting transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020): Systems Demonstrations*, pages 46–54, Online. Association for Computational Linguistics.

Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulić, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. 2020b. AdapterHub: A framework for adapting transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 46–54, Online. Association for Computational Linguistics.

Jonas Pfeiffer, Sebastian Ruder, Ivan Vulić, and Edoardo Ponti. 2023. Modular deep learning. *Transactions on Machine Learning Research*. Survey Certification.

Jerin Philip, Alexandre Berard, Matthias Gallé, and Laurent Besacier. 2020. Monolingual adapters for zero-shot neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4465–4470, Online. Association for Computational Linguistics.

Edoardo Maria Ponti, Alessandro Sordoni, Yoshua Bengio, and Siva Reddy. 2023. Combining parameter-efficient modules for task-level generalisation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 687–702, Dubrovnik, Croatia. 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. *CoRR*, abs/1910.01108.

Simone Tedeschi, Valentino Maiorca, Niccolò Campolungo, Francesco Cecconi, and Roberto Navigli. 2021. WikiNEuRal: Combined neural and knowledge-based silver data creation for multilingual NER. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2521–2533, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin R. Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Daniel M. Bikel, Lukas Blecher, Cristian Cantón Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony S. Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel M. Kloumann, A. V. Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, R. Subramanian, Xia Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zhengxu Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *ArXiv*, abs/2307.09288.

Betty van Aken, Benjamin Winter, Alexander Löser, and Felix A. Gers. 2019. How does bert answer questions? a layer-wise analysis of transformer representations. In *Proceedings of the 28th ACM International Conference on Information* *and Knowledge Management*, CIKM '19, page 1823–1832, New York, NY, USA. Association for Computing Machinery.

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

## A. Experimental Setup

We use the AdapterHub library (Pfeiffer et al., 2020b) for all our experiments. We train all our models using a single GPU with a batch size of $64$ and a learning rate of 1e–5 for $10$ epochs for NER & NLI tasks and $30$ epochs for the PI task. We choose the final checkpoint based on validation dataset performance.

For PEFT hyper-parameters, we set the bottleneck size to $96$ for Adapter modules and a rank of $8$ for LoRA. We apply LoRA to the query and value self-attention modules.

## B. Per Language Results

In Tables 4, 5 and 6, we expand the results reported in Tables 2 and 3 and provide the scores for each evaluated language.

| Model | de | en | es | fr | it | nl | pl | pt | ru |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | *Matching PLMs* | | | | |
| | | | | | Adapter | | | | |
| Teacher | 97,57 | 92,79 | 98,08 | 95,49 | 94,85 | 97,74 | 95,54 | 95,91 | 90,15 |
| Student | 95,34 | 90,23 | 95,81 | 93,23 | 92,67 | 95,27 | 93,73 | 94,21 | 85,97 |
| TM-Student$_{\text{AVG}}$ | 95,53 | 90,21 | 95,87 | 93,26 | 92,74 | 95,39 | 93,88 | 94,40 | 85,92 |
| TM-Student$_{\text{SKIP}}$ | 95,88 | 90,65 | 96,24 | 93,77 | 93,13 | 95,87 | 94,20 | 94,72 | 86,57 |
| | | | | | Lora | | | | |
| Teacher | 95,78 | 90,49 | 96,45 | 93,26 | 93,13 | 95,94 | 93,97 | 94,41 | 85,97 |
| Student | 92,45 | 87,25 | 93,55 | 90,06 | 89,95 | 92,85 | 91,55 | 92,15 | 80,96 |
| TM-Student$_{\text{AVG}}$ | 92,92 | 87,74 | 93,94 | 90,57 | 90,75 | 93,24 | 91,97 | 92,52 | 82,03 |
| TM-Student$_{\text{SKIP}}$ | 93,03 | 87,99 | 93,87 | 90,88 | 91,04 | 93,44 | 92,04 | 92,61 | 82,27 |
| | | | | | *Incompatible PLMs* | | | | |
| | | | | | Adapter | | | | |
| Teacher | 97,36 | 92,30 | 97,95 | 95,61 | 94,99 | 97,79 | 96,15 | 96,12 | 89,74 |
| Student | 95,20 | 89,92 | 96,19 | 93,34 | 93,06 | 96,29 | 94,14 | 94,56 | 86,99 |
| TM-Student$_{\text{SKIP}}$ | 95,30 | 90,03 | 96,21 | 93,40 | 93,00 | 96,09 | 94,19 | 94,78 | 87,01 |
| | | | | | Lora | | | | |
| Teacher | 94,68 | 89,94 | 96,19 | 92,35 | 92,85 | 95,67 | 93,82 | 94,11 | 85,09 |
| Student | 92,21 | 86,65 | 92,74 | 89,40 | 90,05 | 93,14 | 91,61 | 92,25 | 81,62 |
| TM-Student$_{\text{SKIP}}$ | 92,10 | 86,65 | 93,00 | 89,61 | 90,01 | 93,01 | 91,51 | 92,32 | 81,69 |

Table 4: Named Entity Recognition results per language.

| Model | de | en | es | fr | ja | ko | zh |
|---|---|---|---|---|---|---|---|
| | | | | *Matching PLMs* | | | |
| | | | | Adapter | | | |
| Teacher | 83,60 | 91,60 | 85,20 | 86,90 | 76,05 | 75,95 | 78,90 |
| Student | 73,30 | 75,85 | 72,90 | 74,65 | 67,25 | 65,25 | 70,05 |
| TM-Student$_{\text{AVG}}$ | 74,15 | 82,35 | 73,35 | 75,10 | 67,10 | 67,40 | 71,25 |
| TM-Student$_{\text{SKIP}}$ | 74,50 | 85,05 | 77,85 | 78,15 | 69,10 | 69,25 | 71,85 |
| | | | | Lora | | | |
| Teacher | 75,10 | 83,30 | 78,70 | 77,75 | 67,85 | 67,95 | 72,10 |
| Student | 70,20 | 63,45 | 67,30 | 70,10 | 62,80 | 61,85 | 64,90 |
| TM-Student$_{\text{AVG}}$ | 71,95 | 69,35 | 69,95 | 71,85 | 64,75 | 64,05 | 67,75 |
| TM-Student$_{\text{SKIP}}$ | 72,25 | 74,50 | 72,30 | 74,85 | 66,70 | 64,90 | 69,30 |
| | | | | *Incompatible PLMs* | | | |
| | | | | Adapter | | | |
| Teacher | 90,45 | 94,70 | 91,20 | 92,15 | 82,35 | 85,00 | 85,80 |
| Student | 85,75 | 92,55 | 87,25 | 89,25 | 77,10 | 75,65 | 81,30 |
| TM-Student$_{\text{SKIP}}$ | 86,15 | 92,05 | 88,50 | 88,20 | 76,80 | 77,45 | 80,75 |
| | | | | Lora | | | |
| Teacher | 89,40 | 93,80 | 89,90 | 89,65 | 80,95 | 81,45 | 84,05 |
| Student | 80,00 | 88,05 | 82,95 | 83,55 | 72,55 | 68,60 | 75,35 |
| TM-Student$_{\text{SKIP}}$ | 80,95 | 88,05 | 82,10 | 82,70 | 71,65 | 69,80 | 75,20 |

Table 5: Paraphrase Identification results per language.

| Model | ar | bg | de | el | en | es | fr | hi | ru | sw | th | tr | ur | vi | zh |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | Adapter | | | | | | | |
| Teacher | 65,53 | 70,18 | 70,12 | 68,08 | 77,03 | 73,01 | 72,00 | 63,39 | 69,58 | 60,12 | 60,40 | 67,49 | 60,40 | 71,28 | 71,10 |
| Student | 60,12 | 63,47 | 65,07 | 63,35 | 69,62 | 66,11 | 65,89 | 57,33 | 62,14 | 56,99 | 56,43 | 61,86 | 55,71 | 63,67 | 64,01 |
| TM-Student$_{AVG}$ | 60,54 | 63,57 | 65,19 | 63,27 | 70,38 | 66,69 | 66,13 | 57,54 | 62,53 | 56,17 | 56,39 | 62,00 | 56,59 | 63,33 | 64,59 |
| TM-Student$_{SKIP}$ | 61,16 | 64,23 | 65,27 | 63,49 | 70,58 | 68,16 | 66,53 | 58,48 | 63,45 | 57,01 | 57,05 | 62,61 | 57,47 | 63,89 | 65,73 |
| | | | | | | | | Lora | | | | | | | |
| Teacher | 61,46 | 63,79 | 66,83 | 63,77 | 70,12 | 67,84 | 66,99 | 59,90 | 64,47 | 54,57 | 55,37 | 61,60 | 56,47 | 65,01 | 66,77 |
| Student | 59,20 | 62,40 | 63,43 | 61,96 | 67,15 | 64,09 | 63,87 | 57,56 | 60,62 | 54,87 | 53,67 | 59,84 | 55,97 | 61,74 | 62,08 |
| TM-Student$_{AVG}$ | 59,20 | 62,40 | 63,43 | 61,96 | 67,15 | 64,09 | 63,87 | 57,56 | 60,62 | 54,87 | 53,67 | 59,72 | 55,77 | 61,74 | 61,88 |
| TM-Student$_{SKIP}$ | 59,10 | 62,36 | 63,75 | 61,66 | 67,19 | 64,11 | 64,19 | 57,56 | 60,94 | 54,37 | 53,19 | 59,76 | 56,11 | 61,30 | 62,26 |

Table 6: Natural Language Inference results per language for the *matching* PLMs experiment.