A Comparative Analysis of Task-Agnostic Distillation Methods for Compressing Transformer Language Models

Takuma Udagawa, Aashka Trivedi, Michele Merler, Bishwaranjan Bhattacharjee
IBM Research AI
{takuma.udagawa@, aashka.trivedi@, mimerler@us., bhatta@us.}ibm.com

Abstract

Large language models have become a vital component in modern NLP, achieving state of the art performance in a variety of tasks. However, they are often inefficient for real-world deployment due to their expensive inference costs. Knowledge distillation is a promising technique to improve their efficiency while retaining most of their effectiveness. In this paper, we reproduce, compare and analyze several representative methods for task-agnostic (general-purpose) distillation of Transformer language models. Our target of study includes Output Distribution (OD) transfer, Hidden State (HS) transfer with various layer mapping strategies, and Multi-Head Attention (MHA) transfer based on MiniLMv2. Through our extensive experiments, we study the effectiveness of each method for various student architectures in both monolingual (English) and multilingual settings. Overall, we show that MHA transfer based on MiniLMv2 is generally the best option for distillation and explain the potential reasons behind its success. Moreover, we show that HS transfer remains as a competitive baseline, especially under a sophisticated layer mapping strategy, while OD transfer consistently lags behind other approaches. Findings from this study helped us deploy efficient yet effective student models for latency-critical applications.

1 Introduction

Large language models have become a crucial component in modern NLP. They have achieved exceptional performance on various downstream tasks (Devlin et al., 2019; Liu et al., 2019; Lewis et al., 2020) and their capability shows consistent improvement with more compute, data, and model parameters (Kaplan et al., 2020; Brown et al., 2020; Touvron et al., 2023). On the downside, it is becoming increasingly difficult to deploy such models in real-world environments due to their *inefficiency*, i.e. high computation, memory, latency and storage costs (Xu and McAuley, 2023).

Knowledge distillation (Hinton et al., 2015) is a promising technique to overcome this challenge by transferring the knowledge of the original model (teacher) to a smaller, more efficient model (student). This can be conducted in either *task-specific* (Turc et al., 2019; Jiao et al., 2020) or *task-agnostic* manner (Sanh et al., 2019; Wang et al., 2020). The latter only requires distilling a single general-purpose student which can be directly finetuned on any downstream task. Due to its high convenience, we focus on this latter approach in this study.

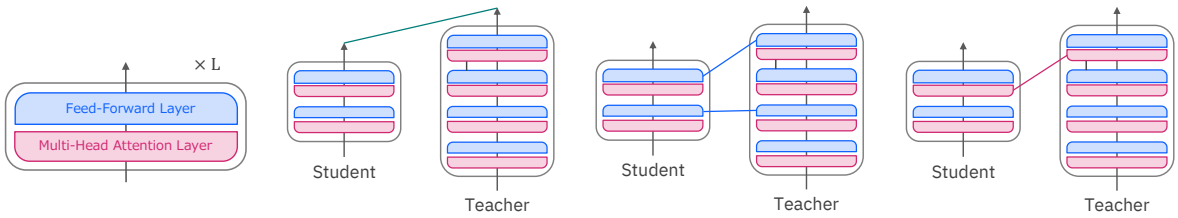
In recent years, there have been various methods proposed for task-agnostic distillation of Transformer language models. The aim of this paper is to reproduce, compare and analyze the most representative methods in this area. We generally focus on the *architecture-agnostic* distillation which imposes no or minimal restriction on the student architecture[1]: the representative methods include Output Distribution (OD) transfer (Hinton et al., 2015), Hidden State (HS) transfer based on linear mapping (Jiao et al., 2020; Mukherjee et al., 2021) and Multi-Head Attention (MHA) transfer based on MiniLMv2 (Wang et al., 2021).

For HS transfer, the layer mapping strategy between teacher and student layers plays a significant role in overall performance, however, the optimal strategy remains unknown or controversial (Sun et al., 2019; Wu et al., 2020; Ko et al., 2023). Therefore, we explore a diverse range of strategies to empirically evaluate each technique.

For MHA transfer, the MiniLMv2 approach has been shown to achieve state-of-the-art performance, however, there is relatively little understanding behind its success. Therefore, we develop a novel variant named DirectMiniLM which is useful for

[1] By *architecture-agnostic*, we mean that the student and teacher can have different architectural parameters (e.g. number of layers, attention heads, hidden state size, etc).

Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track, pages 20–31
December 6-10, 2023



(a) Transformer LM (b) OD Transfer (c) HS Transfer (d) MHA Transfer

Figure 1: A high-level illustration of (a) the Transformer architecture and (b-d) representative distillation methods. (b-d) denote Output Distribution (OD), Hidden State (HS), and Multi-Head Attention (MHA) transfer, respectively. Lines between the student and teacher depict which level of information is transferred in each method.

understanding the effectiveness behind MiniLMv2 both theoretically and empirically.

In contrast to most previous studies, all methods are reproduced on a single unified codebase for fair and consistent comparison. We also conduct distillation on 4 different student architectures, reducing the model size in various dimensions to fit various parameter and latency budgets. Finally, all experiments are conducted on both monolingual and multilingual settings, distilled from open-source BERT (Devlin et al., 2019) and in-house XLM-RoBERTa (Conneau et al., 2020), respectively.

Through our extensive experiments, we critically analyze the effectiveness of each distillation method and provide practical advice for both researchers and practitioners working in this area. In summary, our key findings are:

- MHA transfer is generally the best option for various student architectures and language settings. By comparison with DirectMiniLM, we provide novel insights underlying its success.
- While the effectiveness of HS transfer depends on the layer mapping strategy, it remains as a competitive baseline. More sophisticated layer mapping strategy can provide a boost in performance, esp. in the multilingual setting.
- Methods relying on OD transfer consistently lag behind other methods. This shows that classical OD distillation can be less effective when distilling complex language models on a general-purpose objective.

2 Transformer Language Models

First, we briefly review the standard architecture of Transformer language models (Vaswani et al., 2017; Devlin et al., 2019). A Transformer consists of a stack of L Transformer layers, where each layer comprises two sub-layers: a Multi-Head Attention (MHA) layer followed by a fully connected Feed-Forward (FF) layer (Figure 1, (a)).

Formally, let x denote the input sequence, d_h the hidden state size, and $\mathbf{H}_i \in \mathbb{R}^{|x| \times d_h}$ the hidden state of the i^{th} Transformer layer ($\mathbf{H}_0$ denotes the input sequence embeddings). Given $\mathbf{H}_i$, the MHA layer first computes the query, key, and value mappings $\mathbf{Q}_{i,a}$, $\mathbf{K}_{i,a}$, $\mathbf{V}_{i,a}$ for each attention head $a \in [1, A_h]$, which are combined to obtain the attention head output $\mathbf{O}_{i,a}$:

$$\mathbf{Q}_{i,a} = \mathbf{H}_i \mathbf{W}_{Q,i,a} \quad (1)$$

$$\mathbf{K}_{i,a} = \mathbf{H}_i \mathbf{W}_{K,i,a} \quad (2)$$

$$\mathbf{V}_{i,a} = \mathbf{H}_i \mathbf{W}_{V,i,a} \quad (3)$$

$$\mathbf{O}_{i,a} = \text{softmax}(\frac{\mathbf{Q}_{i,a}\mathbf{K}_{i,a}^{\mathsf{T}}}{\sqrt{d_k}})\mathbf{V}_{i,a} \quad (4)$$

Here, d_k denotes the attention head size (typically set to $\frac{d_h}{A_h}$) and $\mathbf{W}_{Q,i,a}, \mathbf{W}_{K,i,a}, \mathbf{W}_{V,i,a} \in \mathbb{R}^{d_h \times d_k}$ are the learnt weight matrices. The output of the MHA layer is the concatenation of $\mathbf{O}_{i,a}$, namely $\text{MHA}(\mathbf{H}_i) = \bigoplus_{a=1}^{A_h} \mathbf{O}_{i,a}$.

Next, the MHA layer output is followed by a position-wise FF layer with an intermediate size of d_f and a non-linear activation (we use GELU (Hendrycks and Gimpel, 2016) in all models). The hidden state of the next Transformer layer is computed as $\mathbf{H}_{i+1} = \text{FF}(\text{MHA}(\mathbf{H}_i))$.[2]

Finally, to predict the output distribution over the entire vocabulary V, a linear layer $\mathbf{W}_O \in \mathbb{R}^{d_h \times |V|}$ is applied on top of the last hidden state to compute the logits $\mathbf{z} = \mathbf{H}_L \mathbf{W}_O \in \mathbb{R}^{|x| \times |V|}$. The output distribution can be obtained by applying the softmax function over $\mathbf{z}$, denoted as $\text{softmax}(\mathbf{z})$.

Throughout this paper, we assume that both the student and teacher are Transformer language models with L^S and L^T layers, respectively.

[2]Both MHA and FF sub-layers have a residual connection (He et al., 2016) and are followed by layer normalization (Ba et al., 2016), which are omitted for brevity.

3 Distillation Methods

Next, we introduce the representative task-agnostic distillation methods illustrated in Figure 1, (b-d). For Multi-Head Attention (MHA) transfer, we consider two approaches: MiniLMv2 and its novel variant DirectMiniLM. For a survey of advanced methods and topics we could not cover in this study, please refer to Appendix A.

Output Distribution (OD) Transfer The output distribution of the teacher contains useful information on the relative probabilities of plausible (even if incorrect) predictions (Hinton et al., 2015). In OD transfer, the student is trained to replicate the teacher's output distribution. This is achieved by optimizing the following loss function, where $\mathbf{z}^S, \mathbf{z}^T$ denote the student/teacher logits, $\mathrm{CE}(.)$ the cross entropy loss and $\mathcal{T}$ the output temperature:

$$\mathcal{L}_{\mathrm{OD}} = \mathcal{T}^2 \cdot \mathrm{CE}\Big(\mathrm{softmax}(\frac{\mathbf{z}^T}{\mathcal{T}}), \mathrm{softmax}(\frac{\mathbf{z}^S}{\mathcal{T}})\Big) \tag{5}$$

Hidden State (HS) Transfer Transformer language models progressively learn useful and generalizable features layer by layer. In HS transfer, the student is trained to predict such useful features represented in the teacher's hidden states.

Formally, each student layer is mapped to a set of teacher layers to be predicted. Let $\phi(i)$ denote the set mapped from the i^{th} student layer, where $\emptyset \subseteq \phi(i) \subseteq [1, L^T]$. For each $j \in \phi(i)$, the hidden state of the i^{th} student layer $\mathbf{H}_i^S \in \mathbb{R}^{|x| \times d_h^S}$ is linearly transformed to predict the hidden state of the j^{th} teacher layer $\mathbf{H}_j^T \in \mathbb{R}^{|x| \times d_h^T}$.[3] This is represented by the following loss function, where $\mathbf{W}_i^j \in \mathbb{R}^{d_h^S \times d_h^T}$ denotes the linear transformation weight and $\mathrm{MSE}(.)$ the mean squared error loss:

$$\mathcal{L}_{\mathrm{HS}} = \sum_{i=1}^{L^S} \sum_{j \in \phi(i)} \mathrm{MSE}\Big(\mathbf{H}_i^S \mathbf{W}_i^j, \mathbf{H}_j^T\Big) \tag{6}$$

One open problem in this approach is the choice of layer mapping strategy ϕ. We conduct extensive experiments to compare a diverse range of strategies, which will be discussed in §4.

MiniLMv2 The MHA layer is a key component in Transformer language models which controls the long-range dependencies and interactions within input texts. MiniLMv2 (Wang et al., 2021) is an effective method to deeply transfer this module while allowing different number of attention heads A_h^S and A_h^T for the student and teacher. Their main idea is to distil the attention *relation* matrices (Q-Q, K-K and V-V) obtained by first concatenating the query (Q), key (K), and value (V) mappings from all attention heads and re-splitting them into the same number of attention *relation* heads A_r.

Formally, let $\mathbf{A}_{Q,i,a}^S, \mathbf{A}_{K,i,a}^S, \mathbf{A}_{V,i,a}^S \in \mathbb{R}^{|x| \times d_r^S}$ denote the concatenated and re-split queries, keys, and values for the i^{th} student layer, where $a \in [1, A_r]$ and $d_r^S = \frac{d_h^S}{A_r}$. For instance, $\bigoplus_{a=1}^{A_h^S} \mathbf{Q}_{i,a}^S = \bigoplus_{a=1}^{A_r} \mathbf{A}_{Q,i,a}^S$, i.e. original queries from A_h^S attention heads are simply concatenated and then re-split into A_r matrices. We use the same notation for the j^{th} teacher layer, $\mathbf{A}_{Q,j,a}^T, \mathbf{A}_{K,j,a}^T, \mathbf{A}_{V,j,a}^T \in \mathbb{R}^{|x| \times d_r^T}$, where $d_r^T = \frac{d_h^T}{A_r}$. Then, the loss function of MiniLMv2 can be defined as follows:

$$\mathcal{L}_{\mathrm{MHA}} = \sum_{\alpha \in \{Q,K,V\}} \sum_{a=1}^{A_r} \mathrm{CE}\Big(\mathbf{R}_{\alpha,j,a}^T, \mathbf{R}_{\alpha,i,a}^S\Big) \tag{7}$$

$$\mathbf{R}_{\alpha,j,a}^T = \mathrm{softmax}\Big(\frac{\mathbf{A}_{\alpha,j,a}^T \mathbf{A}_{\alpha,j,a}^{T\intercal}}{\sqrt{d_r^T}}\Big) \tag{8}$$

$$\mathbf{R}_{\alpha,i,a}^S = \mathrm{softmax}\Big(\frac{\mathbf{A}_{\alpha,i,a}^S \mathbf{A}_{\alpha,i,a}^{S\intercal}}{\sqrt{d_r^S}}\Big) \tag{9}$$

Here, $\mathbf{R}_{\alpha,j,a}^T, \mathbf{R}_{\alpha,i,a}^S \in \mathbb{R}^{|x| \times |x|}$ denote the attention *relation* matrices which are computed based on the matrix products of $\mathbf{A}_{\alpha,i,a}^T, \mathbf{A}_{\alpha,i,a}^S$ in eq. (8), (9), respectively. Intuitively, this aims to transfer the teacher's queries (Q), keys (K) and values (V) in a somewhat indirect way through their matrix products (Q-Q, K-K and V-V).

However, there is minimal justification for why this method works effectively. It is also difficult to directly compare the method against HS transfer since the losses are computed differently. To better understand MiniLMv2, we propose its novel variant named DirectMiniLM for our analysis.

DirectMiniLM In DirectMiniLM, we aim to transfer the teacher's Q/K/V mappings more directly through the linear transformation of the student's ones, just as we did in HS transfer. Specifically, we use the following loss function with the linear transformation $\mathbf{W}_{\alpha,a} \in \mathbb{R}^{d_r^S \times d_r^T}$:

$$\mathcal{L}_{\mathrm{MHA}}^{\mathrm{Direct}} = \sum_{\substack{\alpha \in \\ \{Q,K,V\}}} \sum_{a=1}^{A_r} \mathrm{MSE}\Big(\mathbf{A}_{\alpha,i,a}^S \mathbf{W}_{\alpha,a}, \mathbf{A}_{\alpha,j,a}^T\Big) \tag{10}$$

[3] Note that d_h^S and d_h^T are the student and teacher hidden state sizes which can take different values.

DirectMiniLM is important in two aspects. First, this approach is directly comparable to HS transfer based on eq. (6) with the only difference in which information you transfer: the hidden states $\mathbf{H}_i^T \rightarrow \mathbf{H}_j^S$ or the Q/K/V mappings $\mathbf{A}_{\alpha,i,a}^T \rightarrow \mathbf{A}_{\alpha,j,a}^S$. From this comparison, we can quantify the precise advantage of transferring each knowledge in an apples-to-apples manner.

Second, DirectMiniLM is also closely relevant to MiniLMv2: if we constrain $\mathbf{W}_{\alpha,a}$ to be orthogonal (i.e. $\mathbf{W}_{\alpha,a}\mathbf{W}_{\alpha,a}^\intercal = \mathbf{I}$) and take the matrix product for each term within the MSE loss in eq. (10), we obtain the following loss function:

$$\sum_{\substack{\alpha \in \\ \{Q,K,V\}}} \sum_{a=1}^{A_r} \text{MSE}\Big(\mathbf{A}_{\alpha,i,a}^S \mathbf{A}_{\alpha,i,a}^{S\intercal}, \mathbf{A}_{\alpha,j,a}^T \mathbf{A}_{\alpha,i,a}^{T\intercal}\Big) \tag{11}$$

This loss closely resembles MiniLMv2 from eq. (7) with a minor difference of using MSE loss instead of CE loss with softmax. Therefore, DirectMiniLM with certain constraints naturally corresponds to MiniLMv2. The major difference is in whether $\mathbf{A}_{\alpha,i,a}^T$ is transferred directly (with linear mappings) or indirectly (with relation matrices): by comparing these two approaches, we can precisely quantify the advantage of each optimization technique.

4 Experimental Setup

We explore the task-agnostic knowledge distillation methods under two settings:[4]

1. Monolingual Distillation: We train English students using the open-source BERT (Devlin et al., 2019) as the teacher. These models are distilled on the same corpus used for pretraining BERT, i.e., English Wikipedia (Devlin et al., 2019) and BookCorpus (Zhu et al., 2015).
2. Multilingual Distillation: We train multilingual students using our in-house XLM-RoBERTa (Conneau et al., 2020) as the teacher, and distill on the CC100 dataset (Conneau et al., 2020), which consists of data in more than 100 languages. We only use a small subset of the corpus to conduct our experiments within a reasonable computation budget while maintaining the language-wise distribution.

In both settings, we use the Base (12 layer) architecture for the teacher, as shown in Table 1. For more details on each distillation setup (e.g. hyperparameters), please refer to Appendix B.

Student Models To conduct a strong comparison of the representative knowledge distillation methods, we train 4 students of varying architectures and latency/parameter budgets. A summary of the student architectures, with their parameters and latency of inference, are shown in Table 1.

Our largest student is a 6 layer model that follows the same architecture as DistilBERT (Sanh et al., 2019). We also use the 6 layer model used in Mukherjee et al. (2021), which has a smaller hidden size than the teacher. Our smaller 4 and 3 layer students were obtained as recommendations from a Neural Architecture Search process (Trivedi et al., 2023) to find good student architectures for distillation from the XLM-RoBERTa teacher, conditioned to minimize the latency on CPU. Please refer to Appendix C for more details.

Layer Mapping Strategies The layer mapping strategy ϕ is a parameter that needs to be considered for both HS and MHA transfer. For HS transfer, we explore the following three settings:

1. Single Mapping: We only distil the last ($L^{T^{\text{th}}}$) teacher layer into the last student layer, which has been shown to be a simple yet competitive baseline (Ko et al., 2023).
2. 1-to-1 Mapping: Prior work shows that mapping not only the last layer but also the intermediate layers improves distillation (Sun et al., 2019). In 1-to-1 mapping, we distil one teacher layer into each student layer by choosing:
   - *Last* L^S teacher layers, i.e. $\phi(i) = \{L^T - L^S + i\}$ $(i \in [1, L^S])$. Empirically, last teacher layers capture more high-level (e.g. semantic) knowledge in their representations (Tenney et al., 2019; Jawahar et al., 2019).
   - A *Uniform* selection of teacher layers which chooses every k^{th} teacher layer, i.e. $\phi(i) = \{ki\}$, where $k = \lceil L^T/L^S \rceil$.[5] This method can also transfer the lower teacher layers, which empirically captures local (e.g. syntactic) knowledge (Tenney et al., 2019).
3. 1-to-N Mapping: Some works even show that mapping each student layer to multiple teacher layers can avoid the loss of information and facilitate student learning (Wu et al., 2020; Passban et al., 2021). For 1-to-N Mapping, we ex-

[4] Note that we limit our study to encoder-only models and leave the distillation of decoder-only (Radford et al., 2019) or encoder-decoder (Lewis et al., 2020) models as future work.

[5] This strategy is used in DistilBERT (Sanh et al., 2019) and also known as the "skip" strategy (Sun et al., 2019).

Model	Architecture	Monolingual Params	Multilingual Params	Monolingual Latency GPU	Monolingual Latency CPU	Multilingual Latency GPU	Multilingual Latency CPU
6L-DistilBERT	6, 12, 768, 3072	66	234	5.98 (0.03)	33.28 (0.09)	6.01 (0.06)	34.02(0.06)
6L	6, 12, 384, 1536	23	106	5.69 (0.02)	11.98 (0.07)	5.99 (0.07)	12.52 (0.06)
4L	4, 12, 576, 768	27	153	3.66 (0.01)	9.53 (0.04)	3.98 (0.02)	9.66 (0.05)
3L	3, 12, 384, 1024	16	100	3.02 (0.01)	5.41 (0.08)	3.25 (0.01)	6.01 (0.06)
Teacher	12, 12, 768, 3072	110	277	8.69 (0.08)	64.91 (0.61)	9.47 (0.01)	66.31 (0.57)

Table 1: Model Architectures displayed as [L, A_h, d_h, d_f]. All parameters are in millions, with the difference in the monolingual and multilingual parameters due to the vocabulary sizes (30K for monolingual and 252K for multilingual). All latencies are in milliseconds, measured over 5 runs, with standard deviation in parenthesis.

Distillation Method	Layer Mapping Strategies
HS Transfer	Single: $L^{T^{\text{th}}}$ 1-to-1: Last, Uniform 1-to-N: Uniform-Cons., Uniform+Last
MHA Transfer	Single: $L^{T^{\text{th}}}$, $(L^T-1)^{\text{th}}$, $(L^T-2)^{\text{th}}$

Table 2: Layer mapping strategies explored in each distillation method. The same strategies are explored for MiniLMv2 and DirectMiniLM in MHA Transfer.

plore the following choices of teacher layers:

- A uniform selection of k consecutive layers (*Uniform-Cons.*), i.e. $\phi(i) = [k(i-1), ki]$, where $k = \lceil L^T/L^S \rceil$. This avoids the loss of information since all teacher layers are mapped to at least one student layer.
- Combining the *Uniform* and *Last* strategies from the 1-to-1 mapping (*Uniform+Last*). This selects 2 teacher layers per student layer based on each 1-to-1 strategy, expecting to take the best out of both approaches.

For MHA transfer, we always take the single mapping strategy and distill a single teacher layer into the last student layer, following Wang et al. (2021). Specifically, we experiment with the last three teacher layers as a choice for distillation for both MiniLMv2 and DirectMiniLM. Table 2 summarizes our layer selection options.

While OD transfer can be conducted from scratch, we found this converges slowly and does not perform competitively.[6] Therefore, we take the style of *multi-stage* distillation (Mukherjee et al., 2021) and conduct OD transfer after HS transfer, using the distilled checkpoint from HS transfer. This approach converges much faster with better final performance, hence we take this approach as the representative OD transfer method.

[6]Our 6L monolingual student takes 49 hours on 30 V100 GPUs to reach acceptable performance, while the same model achieves better scores in only 10.5 hours when initialized from the HS transferred checkpoint.

5 Evaluation and Results

For both our monolingual and multilingual models, we measure performance on the English GLUE Benchmark (Wang et al., 2019) and report the average score of all tasks (without CoLA[7]). For multilingual models, we provide evaluations on the XNLI dataset (Conneau et al., 2018), a set of inference tasks which evaluates the model's performance on 15 languages after being finetuned on only English training data. We report the average score of all languages for XNLI.

Table 3 summarizes the performance of each distillation method on 4 student architectures. For detailed evaluations of each method based on the best configuration, please refer to Appendix D. We also provide a comparison against DistilBERT (Sanh et al., 2019), a representative *architecture-constrained* method, in Appendix E.

HS Transfer From Table 3, we can verify that the performance of HS transfer varies with different layer mapping strategies, and no strategy dominates the others in all settings. In the monolingual setting, we found that the single mapping strategy performs competitively, which is in line with the findings of Ko et al. (2023). However, in the multilingual setting, more sophisticated 1-to-N strategies generally show superiority over the simpler baselines. This indicates that more supervision from the teacher can be helpful (and at worst harmless), hence we advocate for the adoption 1-to-N strategies, esp. in the challenging multilingual distillation.

OD Transfer As mentioned in §4, we initialize the model from the HS transferred checkpoints with each layer mapping strategy. Interestingly, we see a slight *degradation* in performance on downstream tasks compared to only HS transfer, with a signifi-

[7]Distilled models often perform poorly on CoLA: We hypothesize this is because CoLA is the only *syntactic* task in the benchmark as opposed to the other *semantic* tasks (Xu et al., 2022). We include the results of CoLA in Appendix D.

Distillation Method	Layer Mapping Strategy	Avg. GLUE (Monolingual) 6L-DistilBERT	6L	4L	3L	Avg. GLUE (Multilingual) 6L-DistilBERT	6L	4L	3L	Avg. XNLI (Multilingual) 6L-DistilBERT	6L	4L	3L
HS Transfer	$L^{T\text{th}}$	84.1	79.4	80.2	**78.9**	80.8	77.1	78.0	74.7	56.2	55.1	51.6	50.6
	Last	83.2	80.4	79.3	77.7	81.7	77.0	78.3	72.6	63.1	61.0	60.3	54.4
	Uniform	82.9	80.6	79.6	76.6	81.6	78.2	78.3	73.5	59.9	59.9	59.7	59.9
	Uniform-Cons.	83.9	80.6	80.6	77.7	82.4	78.8	78.0	**75.9**	65.5	62.2	60.4	58.6
	Uniform+Last	84.1	80.4	80.4	77.7	83.1	78.7	79.2	75.0	67.0	62.7	62.5	57.9
OD Transfer (init. from HS Transfer)	$L^{T\text{th}}$	84.1	78.1	79.4	76.6	78.5	75.1	75.2	67.9	50.5	48.2	51.6	43.8
	Last	83.1	80.4	79.3	76.4	80.7	76.9	76.1	69.8	62.6	57.0	54.1	42.7
	Uniform	83.4	79.8	79.8	77.1	79.9	78.0	77.9	65.4	60.4	54.1	52.0	42.8
	Uniform-Cons.	83.7	80.3	79.5	76.7	81.7	78.7	76.4	70.1	63.1	61.0	56.5	48.2
	Uniform+Last	84.1	80.5	79.9	77.1	82.1	78.4	76.4	72.3	66.0	60.9	60.0	48.6
MiniLMv2	$L^{T\text{th}}$	84.2	81.9	79.9	77.6	82.3	80.1	79.3	74.4	67.0	66.7	63.1	59.3
	$(L^T-1)^{\text{th}}$	84.2	**82.5**	80.0	78.2	83.1	81.0	80.2	75.8	**69.1**	**67.5**	**65.6**	**62.0**
	$(L^T-2)^{\text{th}}$	**84.4**	82.2	**80.7**	78.3	82.9	80.5	78.3	73.4	67.5	66.9	63.5	61.5
DirectMiniLM	$L^{T\text{th}}$	84.0	81.3	79.7	78.2	83.2	80.8	79.0	75.1	66.3	66.1	64.7	60.7
	$(L^T-1)^{\text{th}}$	**84.4**	81.7	79.6	78.0	81.9	**81.1**	**80.3**	73.8	66.9	65.9	64.8	61.0
	$(L^T-2)^{\text{th}}$	84.3	81.7	80.4	78.3	**83.4**	80.9	79.7	75.6	66.3	64.8	65.4	60.5
Teacher		85.5				84.8				70.9			

Table 3: Performance of the representative distillation methods evaluated on avg. GLUE and XNLI. Results based on the best layer mapping strategy for each method is underlined, and the best overall result is shown in bold.

cant loss observed for smaller students. This indicates that learning effective representations from the output distribution signals is difficult, especially for students with lower capacity. Moreover, given how computationally expensive OD transfer can be, HS transfer is a cheaper and more effective alternative for knowledge transfer.

MHA Transfer For both MiniLMv2 and DirectMiniLM, we found distilling the upper-middle teacher layer, i.e. $(L^T-1)^{\text{th}}$ or $(L^T-2)^{\text{th}}$ strategy, led to the best performance, in line with the original findings of Wang et al. (2021). Importantly, we found that both MHA transfer methods generally outperform HS transfer, which points to the benefit of transferring the Q/K/V knowledge over the hidden state knowledge. This is consistent with the latest comparative study by Wang et al. (2023), although they only evaluate on the 6L-DistilBERT architecture in the monolingual setting.

We also note that MiniLMv2 and DirectMiniLM perform equivalently, with the notable exception on XNLI. We attribute this to two factors:

1. MiniLMv2 transfers relational representations conditioned on the whole input, while DirectMiniLM transfers absolute position-wise representations. The former may be more semantically informative, as the contextual representations often exhibit rich relational structures (Park et al., 2021; Liu et al., 2022a).
2. DirectMiniLM requires learning the linear transformation weight $\mathbf{W}_{\alpha,a}$, while MiniLMv2 does not incur any additional parameters.

From these observations, we generally expect MiniLMv2 to be the best distillation method and have adopted it in our latency-critical applications.[8] However, DirectMiniLM performs comparably and provides meaningful insights on the benefit of each optimization technique, which can be useful for debugging and analyzing MiniLMv2. Therefore, we recommend its comparison for both reseachers and practitioners in future studies.

6 Conclusion

This study critically analyzes the representative methods for task-agnostic distillation of language models. Specifically, we compare Output Distribution (OD), Hidden State (HS), and Multi-Head Attention (MHA) transfer for different student architectures, language settings, and layer mapping strategies. Through our extensive experiments, we show that MHA transfer based on MiniLMv2 is the best option across many settings, followed by HS transfer with sophisticated 1-to-N mapping strategies. Meanwhile, we did not find OD transfer to be an effective alternative. Finally, we propose DirectMiniLM to demistify the precise advantage of the indirect (i.e. relation matrix based) optimization technique proposed in MiniLMv2. Overall, we hope this study will be a useful guide for both researchers and practitioners working in this area.

[8] Specifically, the 4L monolingual and multilingual students with 7x speedup on CPU have been deployed for various NLP applications, such as entity extraction, document classification and relation detection, while maintaining 93% of the teacher's performance on average (Trivedi et al., 2023).

References

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

Haoli Bai, Wei Zhang, Lu Hou, Lifeng Shang, Jin Jin, Xin Jiang, Qun Liu, Michael Lyu, and Irwin King. 2021. BinaryBERT: Pushing the limit of BERT quantization. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4334–4348, Online. Association for Computational Linguistics.

Matan Ben Noach and Yoav Goldberg. 2020. Compressing pre-trained language models by matrix decomposition. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 884–889, Suzhou, 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.

Patrick Chen, Hsiang-Fu Yu, Inderjit Dhillon, and Cho-Jui Hsieh. 2021. Drone: Data-aware low-rank compression for large nlp models. In *Advances in Neural Information Processing Systems*, volume 34, pages 29321–29334. 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.

Alexis Conneau, Guillaume Lample, Ruty Rinott, Adina Williams, Samuel R. Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. Xnli: Evaluating cross-lingual sentence representations.

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.

Angela Fan, Edouard Grave, and Armand Joulin. 2020. Reducing transformer depth on demand with structured dropout. In *International Conference on Learning Representations*.

Md Akmal Haidar, Nithin Anchuri, Mehdi Rezagholizadeh, Abbas Ghaddar, Philippe Langlais, and Pascal Poupart. 2022. RAIL-KD: RAndom intermediate layer mapping for knowledge distillation. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1389–1400, Seattle, United States. Association for Computational Linguistics.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Dan Hendrycks and Kevin Gimpel. 2016. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*.

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

Aref Jafari, Mehdi Rezagholizadeh, Pranav Sharma, and Ali Ghodsi. 2021. Annealing knowledge distillation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2493–2504, Online. Association for Computational Linguistics.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.

Xiaoqi Jiao, Huating Chang, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2021. Improving task-agnostic bert distillation with layer mapping search. *Neurocomputing*, 461:194–203.

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. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4163–4174, Online. Association for Computational Linguistics.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Sehoon Kim, Amir Gholami, Zhewei Yao, Michael W Mahoney, and Kurt Keutzer. 2021. I-bert: Integer-only bert quantization. In *International conference on machine learning*, pages 5506–5518. PMLR.

Jongwoo Ko, Seungjoon Park, Minchan Jeong, Sukjin Hong, Euijai Ahn, Du-Seong Chang, and Se-Young Yun. 2023. Revisiting intermediate layer distillation for compressing language models: An overfitting perspective. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 158–175, Dubrovnik, Croatia. Association for Computational Linguistics.

François Lagunas, Ella Charlaix, Victor Sanh, and Alexander Rush. 2021. Block pruning for faster transformers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10619–10629, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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

Jianquan Li, Xiaokang Liu, Honghong Zhao, Ruifeng Xu, Min Yang, and Yaohong Jin. 2020. BERT-EMD: Many-to-many layer mapping for BERT compression with earth mover's distance. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3009–3018, Online. Association for Computational Linguistics.

Kevin J Liang, Weituo Hao, Dinghan Shen, Yufan Zhou, Weizhu Chen, Changyou Chen, and Lawrence Carin. 2021. Mix{kd}: Towards efficient distillation of large-scale language models. In *International Conference on Learning Representations*.

Kaiyuan Liao, Yi Zhang, Xuancheng Ren, Qi Su, Xu Sun, and Bin He. 2021. A global past-future early exit method for accelerating inference of pre-trained language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2013–2023, Online. Association for Computational Linguistics.

Chang Liu, Chongyang Tao, Jiazhan Feng, and Dongyan Zhao. 2022a. Multi-granularity structural knowledge distillation for language model compression. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1001–1011, Dublin, Ireland. Association for Computational Linguistics.

Chang Liu, Chongyang Tao, Jianxin Liang, Tao Shen, Jiazhan Feng, Quzhe Huang, and Dongyan Zhao. 2022b. Rethinking task-specific knowledge distillation: Contextualized corpus as better textbook. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10652–10658, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Weijie Liu, Peng Zhou, Zhiruo Wang, Zhe Zhao, Haotang Deng, and Qi Ju. 2020. FastBERT: a self-distilling BERT with adaptive inference time. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6035–6044, Online. Association for Computational Linguistics.

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

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Xinge Ma, Jin Wang, Liang-Chih Yu, and Xuejie Zhang. 2022. Knowledge distillation with reptile meta-learning for pretrained language model compression. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4907–4917, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Yihuan Mao, Yujing Wang, Chufan Wu, Chen Zhang, Yang Wang, Quanlu Zhang, Yaming Yang, Yunhai Tong, and Jing Bai. 2020. LadaBERT: Lightweight adaptation of BERT through hybrid model compression. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3225–3234, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Seyed Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. 2020. Improved knowledge distillation via teacher assistant. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 5191–5198.

Subhabrata Mukherjee, Ahmed Hassan Awadallah, and Jianfeng Gao. 2021. Xtremedistiltransformers: Task transfer for task-agnostic distillation. *arXiv preprint arXiv:2106.04563*.

Geondo Park, Gyeongman Kim, and Eunho Yang. 2021. Distilling linguistic context for language model compression. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 364–378, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Peyman Passban, Yimeng Wu, Mehdi Rezagholizadeh, and Qun Liu. 2021. Alp-kd: Attention-based layer projection for knowledge distillation. In *Proceedings of the AAAI Conference on artificial intelligence*, volume 35, pages 13657–13665.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

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*.

Sheng Shen, Zhen Dong, Jiayu Ye, Linjian Ma, Zhewei Yao, Amir Gholami, Michael W Mahoney, and Kurt Keutzer. 2020. Q-bert: Hessian based ultra low precision quantization of bert. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8815–8821.

Wonchul Son, Jaemin Na, Junyong Choi, and Wonjun Hwang. 2021. Densely guided knowledge distillation using multiple teacher assistants. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9395–9404.

Siqi Sun, Yu Cheng, Zhe Gan, and Jingjing Liu. 2019. Patient knowledge distillation for BERT model compression. 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 4323–4332, Hong Kong, China. Association for Computational Linguistics.

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.

Marzieh Tahaei, Ella Charlaix, Vahid Nia, Ali Ghodsi, and Mehdi Rezagholizadeh. 2022. KroneckerBERT: Significant compression of pre-trained language models through kronecker decomposition and knowledge distillation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2116–2127, Seattle, United States. Association for Computational Linguistics.

Weiting Tan, Kevin Heffernan, Holger Schwenk, and Philipp Koehn. 2023. Multilingual representation distillation with contrastive learning. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1477–1490, Dubrovnik, Croatia. Association for Computational Linguistics.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Aashka Trivedi, Takuma Udagawa, Michele Merler, Rameswar Panda, Yousef El-Kurdi, and Bishwaranjan Bhattacharjee. 2023. Neural architecture search for effective teacher-student knowledge transfer in language models. *arXiv preprint arXiv:2303.09639*.

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, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations*.

Jue Wang, Ke Chen, Gang Chen, Lidan Shou, and Julian McAuley. 2022. SkipBERT: Efficient inference with shallow layer skipping. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7287–7301, Dublin, Ireland. Association for Computational Linguistics.

Wenhui Wang, Hangbo Bao, Shaohan Huang, Li Dong, and Furu Wei. 2021. MiniLMv2: Multi-head self-attention relation distillation for compressing pretrained transformers. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2140–2151, Online. Association for Computational Linguistics.

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. In *Advances in Neural Information Processing Systems*, volume 33, pages 5776–5788. Curran Associates, Inc.

Xinpeng Wang, Leonie Weissweiler, Hinrich Schütze, and Barbara Plank. 2023. How to distill your BERT: An empirical study on the impact of weight initialisation and distillation objectives. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1843–1852, Toronto, Canada. Association for Computational Linguistics.

Yimeng Wu, Peyman Passban, Mehdi Rezagholizadeh, and Qun Liu. 2020. Why skip if you can combine: A simple knowledge distillation technique for intermediate layers. In *Proceedings of the 2020 Conference*

on Empirical Methods in Natural Language Processing (EMNLP), pages 1016–1021, Online. Association for Computational Linguistics.

Yimeng Wu, Mehdi Rezagholizadeh, Abbas Ghaddar, Md Akmal Haidar, and Ali Ghodsi. 2021. Universal-KD: Attention-based output-grounded intermediate layer knowledge distillation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7649–7661, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Mengzhou Xia, Zexuan Zhong, and Danqi Chen. 2022. Structured pruning learns compact and accurate models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1513–1528, Dublin, Ireland. Association for Computational Linguistics.

Ji Xin, Raphael Tang, Yaoliang Yu, and Jimmy Lin. 2021. BERxiT: Early exiting for BERT with better fine-tuning and extension to regression. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 91–104, Online. Association for Computational Linguistics.

Canwen Xu and Julian McAuley. 2023. A survey on model compression and acceleration for pretrained language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 10566–10575.

Canwen Xu, Wangchunshu Zhou, Tao Ge, Furu Wei, and Ming Zhou. 2020. BERT-of-theseus: Compressing BERT by progressive module replacing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7859–7869, Online. Association for Computational Linguistics.

Dongkuan (DK) Xu, Subhabrata Mukherjee, Xiaodong Liu, Debadeepta Dey, Wenhui Wang, Xiang Zhang, Ahmed Awadallah, and Jianfeng Gao. 2022. Few-shot task-agnostic neural architecture search for distilling large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 28644–28656. Curran Associates, Inc.

Ofir Zafrir, Guy Boudoukh, Peter Izsak, and Moshe Wasserblat. 2019. Q8bert: Quantized 8bit bert. In *2019 Fifth Workshop on Energy Efficient Machine Learning and Cognitive Computing-NeurIPS Edition (EMC2-NIPS)*, pages 36–39. IEEE.

Minjia Zhang, Niranjan Uma Naresh, and Yuxiong He. 2022. Adversarial data augmentation for task-specific knowledge distillation of pre-trained transformers. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10):11685–11693.

Wangchunshu Zhou, Canwen Xu, and Julian McAuley. 2022. BERT learns to teach: Knowledge distillation with meta learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7037–7049, Dublin, Ireland. Association for Computational Linguistics.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 19–27.

A Related Work

MobileBERT (Sun et al., 2020) is an effective technique to compress BERT into a specially designed student with a bottleneck architecture. In BERT-of-Theseus (Xu et al., 2020), the modules of the teacher are progressively replaced with smaller ones to improve efficiency. However, these approaches constrain the architecture of the students. In contrast, we focus on the *architecture-agnostic* distillation methods for better flexibility.

Improvements on distillation objectives are also made, e.g. transferring the relational, structural or holistic representations of the language models may provide more useful signals for students (Park et al., 2021; Liu et al., 2022a; Tan et al., 2023). When the transfer set is limited, various methods of data augmentation (Liang et al., 2021; Zhang et al., 2022; Liu et al., 2022b) can be applied successfully. To alleviate the *capacity gap* between the teacher and student, previous works proposed scheduled annealing in OD transfer (Jafari et al., 2021), multi-stage distillation with intermediate-sized teacher assistants (Mirzadeh et al., 2020; Son et al., 2021), and meta-learning to optimize the teacher for student distillation (Zhou et al., 2022; Ma et al., 2022). We leave the exploration of such advanced techniques as future work.

Layer mapping strategies for HS transfer have also been studied extensively. Jiao et al. (2021) proposed an evolutionary search process to obtain the optimal layer mapping for specific downstream tasks. Li et al. (2020) applied Earth Mover's Distance to prioritize mappings with smaller cost (i.e. distillation loss). The attention mechanism can also be applied to map student layers to *similar* teacher layers, where the similarity is computed based on the cosine similarity (Passban et al., 2021) or the predictions of internal classifiers (Wu et al., 2021). Finally, random mapping has been shown to work surprisingly well, potentially working as a regularizer to prevent overfitting (Haidar et al., 2022).

In this study, we focus instead on the carefully designed and easily applicable heuristic strategies.

Finally, there are different approaches to reducing the inference costs of large language models, such as quantization (Zafrir et al., 2019; Shen et al., 2020; Kim et al., 2021; Bai et al., 2021), pruning (Fan et al., 2020; Lagunas et al., 2021; Xia et al., 2022), early exit mechanisms (Liu et al., 2020; Xin et al., 2021; Liao et al., 2021; Wang et al., 2022), and matrix decomposition (Ben Noach and Goldberg, 2020; Mao et al., 2020; Chen et al., 2021; Tahaei et al., 2022). Many of these approaches are complementary to our distillation methods and can be combined for further efficiency.

B Distillation Setup

We train our monolingual students on the entire Wikipedia and BookCorpus using the AdamW Optimizer (Loshchilov and Hutter, 2019) with $\beta_1 = 0.9, \beta_2 = 0.98$. For HS and MHA transfer, students are trained for 7 epoch with a peak learning rate (LR) of $5e-4$. For OD transfer, we train for 3 epochs with a peak LR of $3e-4$ after HS transfer. We use a linear LR warmup over the first 5% of the training steps and then a linear decay. We use a batch size of 32 with the maximum sequence length set to 256 and train on 30 V100 GPUs.

For multilingual distillation, we use a small subset of CC-100 containing 7M sentences, which we found to be sufficient for developing competitive students. We generally use the same setup as monolingual distillation, except we use the peak LR of $8e-4$ for MHA transfer. Multilingual students are trained on 2 A100-80GB GPUs.

Finally, the method-specific hyperparameters (§3) are as follows. For OD transfer, we set the output temperature $\mathcal{T}$ to the default value of 1. For MiniLMv2, we use $A_r > A_h$ to transfer more fine-grained knowledge in the Q/K/V mappings: specifically, we set $A_r = 48$, which is also used in Wang et al. (2021). For DirectMiniLM, we found using $A_r = A_h$ without the orthogonal constraints on $\mathbf{W}_{\alpha,a}$ led to the best performance and used this setting throughout our experiments.

C Finding Smaller Student Models

Our smallest students, a 4 layer and a 3 layer model, were obtained as recommendations from a Neural Architecture Search process to find good student architectures for task-agnostic distillation from an XLM-RoBERTa teacher, conditioned to minimize the latency of inference on a CPU. Specifically, we follow the KD-NAS method of Trivedi et al. (2023) and modify the reward to reduce the distillation loss $\mathcal{L}_{\text{HS}}$ defined in Eq. (6), along with the CPU latency of the student ($lat(S)$) normalized by the teacher's latency ($lat(T)$):

$$reward(S) = (1 - \mathcal{L}_{\text{HS}}) * \left(\frac{lat(S)}{0.6 * lat(T)} \right)^{-0.06} \quad (12)$$

Please refer to their original paper for more details.

D Evaluation Results for Best Models

We include detailed results of each distillation method for the best configuration (i.e. layer mapping strategy). Specifically, we show the results of each GLUE task for monolingual and multilingual distillation in Table 5 and 6. We show language-wise performance on XNLI in Table 7. All downstream tasks are evaluated on 3 random seeds.

For the sake of efficient evaluation, we did not conduct expensive grid search for finetuning hyperparameters. After some manual tuning, we used the same LR of $2e-5$ and batch size of 32 for finetuning all models on all tasks. We used 3 epochs of finetuning for GLUE tasks (except CoLA, where we used 6 and 10 epochs for monolingual and multilingual models) and 5 epochs for XNLI.

E Architecture Constrained Distillation: DistilBERT

DistilBERT (Sanh et al., 2019) is one of the earliest and most widely used baseline. This method comprises (1) layer initialization from the teacher layers, (2) HS transfer based on cosine similarity loss, and (3) OD transfer. The first two techniques restrict the architecture of each student layer to be identical to the teacher model, which limits our analysis to the 6L-DistilBERT student architecture.

	6L-DsitilBERT	Teacher
Avg. GLUE (Monolingual)	82.9 (0.5)	85.5 (0.6)
Avg. GLUE (Multilingual)	79.7 (0.5)	84.8 (0.3)
Avg. XNLI (Multilingual)	61.8 (0.5)	70.9 (0.8)

Table 4: DistilBERT Performance. Average GLUE scores reported for all tasks w/o CoLA. Average XNLI scores reported for all languages. Average taken over 3 random seeds with standard deviation in parenthesis.

As shown in the results of Table 4, the performance of DistilBERT is generally not competitive with our distillation methods from Table 3, especially in the multilingual setting.

Model	Distillation Method	Best Strategy	GLUE Performance								Avg.	Avg. (-CoLA)
			MNLI	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE		
6L-DistilBERT	HS Transfer	Uniform+Last	82.6	86.2	88.7	90.8	45.9	85.9	89.7	65.1	79.4 (0.5)	84.1 (0.4)
	OD Transfer	Uniform+Last	82.7	86.5	88.3	91.3	50.8	85.5	89.7	64.4	79.9 (0.3)	84.1 (0.2)
	MiniLMv2	$(L^T-2)^{\text{th}}$	83.0	86.6	90.1	91.6	53.1	86.7	89.0	64.2	**80.5 (0.4)**	**84.4 (0.3)**
	DirectMiniLM	$(L^T-1)^{\text{th}}$	82.9	86.6	90.0	91.4	52.7	86.4	89.0	64.9	**80.5 (0.5)**	**84.4 (0.4)**
6L	HS Transfer	Uniform-Cons.	78.3	85.0	85.9	90.9	31.2	83.2	84.4	56.3	74.4 (0.4)	80.6 (0.3)
	OD Transfer	Uniform+Last	79.1	84.6	86.3	89.7	38.6	82.3	83.7	57.9	75.3 (0.6)	80.5 (0.3)
	MiniLMv2	$(L^T-1)^{\text{th}}$	80.8	84.9	88.0	90.3	36.2	84.5	86.2	62.5	**76.7 (0.1)**	**82.5 (0.1)**
	DirectMiniLM	$(L^T-1)^{\text{th}}$	80.0	85.1	87.2	90.9	36.1	83.3	85.9	59.7	76.0 (0.2)	81.7 (0.2)
4L	HS Transfer	Uniform-Cons.	77.3	84.9	85.7	90.0	26.9	83.4	83.0	60.1	73.9 (0.4)	80.6 (0.3)
	OD Transfer	Uniform+Last	78.2	84.6	85.1	90.1	32.2	83.3	83.2	55.1	74.0 (0.2)	79.9 (0.4)
	MiniLMv2	$(L^T-2)^{\text{th}}$	78.8	83.8	86.0	90.8	30.9	83.0	84.3	58.2	**74.5 (0.2)**	**80.7 (0.3)**
	DirectMiniLM	$(L^T-2)^{\text{th}}$	79.0	84.2	85.7	90.0	29.7	82.5	84.9	56.6	74.1 (0.4)	80.4 (0.4)
3L	HS Transfer	$L^{T^{\text{th}}}$	74.3	82.8	84.0	89.4	20.0	80.8	83.4	57.5	**71.5 (0.1)**	**78.9 (0.3)**
	OD Transfer	Uniform+Last	73.8	81.9	83.4	86.6	15.1	78.8	82.7	52.8	69.4 (0.3)	77.1 (0.4)
	MiniLMv2	$(L^T-2)^{\text{th}}$	75.1	81.9	84.8	87.3	13.3	81.6	82.0	55.1	70.1 (0.4)	78.3 (0.2)
	DirectMiniLM	$(L^T-2)^{\text{th}}$	75.7	82.2	84.0	88.5	16.8	81.0	83.3	53.5	70.6 (0.2)	78.3 (0.3)
Teacher			84.4	88.0	91.5	92.9	57.4	88.0	89.0	64.8	82.0 (0.6)	85.5 (0.6)

Table 5: Monolingual Student GLUE Performance for all tasks. Each row shows performance based on the best layer mapping strategy. Each score reported as an average over 3 random seeds (standard deviation in parenthesis).

Model	Distillation Method	Best Strategy	GLUE Performance								Avg.	Avg. (-CoLA)
			MNLI	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE		
6L-DistilBERT	HS Transfer	Uniform+Last	80.8	86.8	87.9	90.2	32.3	84.7	88.5	62.6	76.7 (0.6)	83.1 (0.3)
	OD Transfer	Uniform+Last	80.1	86.4	86.2	89.8	33.1	84.1	87.5	60.5	76.0 (1.0)	82.1 (0.5)
	MiniLMv2	$(L^T-1)^{\text{th}}$	81.3	85.8	88.8	89.6	40.2	85.9	89.3	61.0	**77.7 (0.5)**	83.1 (0.3)
	DirectMiniLM	$(L^T-2)^{\text{th}}$	81.0	86.4	89.2	89.8	37.8	85.9	90.1	61.7	**77.7 (0.7)**	**83.4 (0.6)**
6L	HS Transfer	Uniform-Cons.	75.0	82.8	83.0	86.7	16.9	80.8	84.6	58.5	71.1 (0.6)	78.8 (0.4)
	OD Transfer	Uniform-Cons.	76.2	83.7	83.6	87.5	16.9	78.1	85.0	55.9	71.1 (0.6)	78.7 (0.5)
	MiniLMv2	$(L^T-1)^{\text{th}}$	78.3	83.7	86.9	89.1	29.2	83.6	85.1	60.3	**74.5 (0.5)**	81.0 (0.4)
	DirectMiniLM	$(L^T-1)^{\text{th}}$	78.3	84.3	86.1	89.4	25.5	84.5	86.9	58.0	74.1 (0.6)	**81.1 (0.5)**
4L	HS Transfer	Uniform+Last	75.6	83.7	83.8	87.8	18.3	81.2	83.3	59.0	71.6 (0.7)	79.2 (0.5)
	OD Transfer	Uniform	73.4	83.8	81.2	85.2	17.0	80.0	82.8	58.6	70.3 (0.7)	77.9 (0.7)
	MiniLMv2	$(L^T-1)^{\text{th}}$	76.8	83.4	85.2	87.6	17.1	83.9	86.0	58.1	**72.3 (0.7)**	80.2 (0.5)
	DirectMiniLM	$(L^T-1)^{\text{th}}$	77.0	83.6	85.2	88.5	19.2	83.5	85.2	59.1	72.7 (0.6)	**80.3 (0.4)**
3L	HS Transfer	Uniform-Cons.	71.0	80.7	82.1	84.6	11.0	75.8	82.2	54.9	67.8 (0.4)	**75.9 (0.4)**
	OD Transfer	Uniform+Last	68.1	79.4	79.7	81.9	2.6	61.5	81.2	54.6	63.6 (0.5)	72.3 (0.6)
	MiniLMv2	$(L^T-1)^{\text{th}}$	72.7	80.6	83.2	84.6	9.7	70.6	81.7	57.4	67.6 (0.6)	75.8 (0.5)
	DirectMiniLM	$(L^T-2)^{\text{th}}$	72.2	81.2	83.4	84.8	15.9	67.9	82.0	58.0	**68.2 (1.1)**	75.6 (1.1)
Teacher			84.1	87.9	90.2	91.9	51.7	86.6	91.4	61.4	80.6 (0.3)	84.8 (0.3)

Table 6: Multilingual Student GLUE Performance for all tasks. Each row shows performance based on the best layer mapping strategy. Each score reported as an average over 3 random seeds (standard deviation in parenthesis).

Model	Distillation Method	Best Strategy	XNLI Performance															Avg.
			ar	bg	de	el	en	es	fr	hi	ru	sw	th	tr	ur	vi	zh	
6L-DistilBERT	HS Transfer	Uniform+Last	64.7	69.7	69.6	69.2	80.7	72.0	70.2	64.6	67.7	51.2	65.3	62.5	58.9	70.4	68.6	67.0 (0.4)
	OD Transfer	Uniform+Last	63.7	69.1	69.4	67.0	78.6	70.7	68.9	60.0	69.0	51.2	65.4	61.9	57.9	68.5	68.8	66.0 (0.6)
	MiniLMv2	$(L^T-1)^{\text{th}}$	65.5	71.6	72.1	71.5	81.4	75.0	73.5	65.3	70.6	58.1	65.1	67.1	60.9	69.7	69.3	**69.1 (0.5)**
	DirectMiniLM	$(L^T-1)^{\text{th}}$	63.8	69.4	69.3	68.5	79.2	73.2	71.2	64.1	67.2	55.1	63.9	65.6	59.7	66.6	67.0	66.9 (0.4)
6L	HS Transfer	Uniform+Last	59.7	67.2	63.4	65.6	75.9	68.7	66.8	58.3	62.4	48.9	62.7	59.1	53.4	63.2	65.1	62.7 (0.4)
	OD Transfer	Uniform+Last	55.7	62.6	63.7	59.2	76.5	66.9	63.7	54.1	62.0	45.7	57.9	56.3	51.0	62.8	62.2	61.0 (0.5)
	MiniLMv2	$(L^T-1)^{\text{th}}$	65.0	69.7	70.4	68.8	80.3	73.1	71.5	62.9	69.3	53.8	65.0	65.7	59.6	69.2	68.0	**67.5 (0.5)**
	DirectMiniLM	$L^{T^{\text{th}}}$	63.2	68.8	70.1	68.1	78.4	70.5	70.0	62.2	66.6	52.4	64.6	64.0	59.1	66.2	66.9	66.1 (0.5)
4L	HS Transfer	Uniform+Last	56.9	64.5	66.2	66.3	77.3	68.2	63.9	57.9	63.9	49.2	61.8	59.2	54.0	64.2	64.2	62.5 (0.5)
	OD Transfer	Uniform+Last	55.7	62.6	63.7	59.2	76.5	66.9	63.7	54.1	62.0	45.7	57.9	56.3	51.0	62.8	62.2	60.0 (0.5)
	MiniLMv2	$(L^T-1)^{\text{th}}$	62.9	67.5	67.8	68.2	77.8	70.7	68.2	62.4	67.0	51.0	63.6	64.7	57.7	67.2	67.4	**65.6 (0.8)**
	DirectMiniLM	$(L^T-2)^{\text{th}}$	63.2	68.3	67.9	67.6	78.3	69.7	69.6	63.1	64.9	49.0	64.2	62.4	58.6	67.2	66.3	65.4 (0.7)
3L	HS Transfer	Uniform	58.3	63.4	60.5	60.6	74.1	65.6	61.6	56.6	61.4	46.7	57.3	55.9	51.8	61.1	63.1	59.9 (0.5)
	OD Transfer	Uniform+Last	45.6	52.3	48.7	47.8	69.9	55.0	49.4	42.9	47.3	40.9	46.3	44.4	41.6	49.7	47.8	48.6 (0.5)
	MiniLMv2	$(L^T-1)^{\text{th}}$	60.0	64.9	63.6	64.3	74.1	66.7	64.2	58.2	61.8	49.4	59.7	60.7	55.3	64.2	62.4	**62.0 (0.8)**
	DirectMiniLM	$(L^T-1)^{\text{th}}$	57.4	63.0	64.1	63.3	74.3	66.1	65.1	57.2	62.1	46.7	56.7	58.1	55.2	63.6	61.8	61.0 (0.4)
Teacher			69.1	73.2	74.1	72.2	83.4	75.1	73.1	69	71.3	57.3	69.7	67.7	64.1	70.8	73.3	70.9 (0.8)

Table 7: Multilingual Student XNLI Performance for 15 languages. Each row shows performance based on the best layer mapping strategy. Each score reported as an average over 3 random seeds (standard deviation in parenthesis).