Task-agnostic Distillation of Encoder-Decoder Language Models

Chen Zhang[♣], Yang Yang[♦], Qiuchi Li[♥], Jingang Wang[♦], Dawei Song[♣♠]

[♣]Beijing Institute of Technology [♦]Meituan NLP [♥]University of Copenhagen

chenzhang9702@outlook.com {yangyang113,wangjingang02}@meituan.com

qiuchi.li@di.ku.dk dwsong@bit.edu.cn

Abstract

Finetuning pretrained language models (LMs) have enabled appealing performance on a diverse array of tasks. The intriguing task-agnostic property has driven a shifted focus from task-specific to task-agnostic distillation of LMs. While task-agnostic, compute-efficient, performance-preserved LMs can be yielded by task-agnostic distillation, previous studies mainly sit in distillation of either encoder-only LMs (e.g., BERT) or decoder-only ones (e.g., GPT) yet largely neglect that distillation of encoder-decoder LMs (e.g., T5) can posit very distinguished behaviors. Frustratingly, we discover that existing task-agnostic distillation methods can fail to handle the distillation of encoder-decoder LMs. To the demand, we explore a few paths and uncover a path named as MiniEnD that successfully tackles the distillation of encoder-decoder LMs in a task-agnostic fashion. We examine MiniEnD on language understanding and abstractive summarization. The results showcase that MiniEnD is generally effective and is competitive compared to other alternatives. We further scale MiniEnD up to distillation of 3B encoder-decoder language models with interpolated distillation. The results imply the opportunities and challenges in distilling large language models (e.g., LLaMA).

Keywords: encoder-decoder language models, task-agnostic distillation, scaling.

1. Introduction

Pretrained language models (LMs) powered by finetuning have achieved remarkable performance on a wide range of downstream tasks (Devlin et al., 2019; Liu et al., 2019; Radford et al., 2019). Driven by the pursued task-agnostic property, distillation of LMs has witnessed a paradigm shift from task-specific to task-agnostic distillation (Sanh et al., 2019). Under a teacher-student regime, task-agnostic distillation distils pretrained LMs into ones of small compute on pretraining data so that these small LMs can be applied to tasks by finetuning (Jiao et al., 2020; Wang et al., 2020; Liang et al., 2023; Zhang et al., 2023c,a). In contrast, task-specific distillation distils finetuned LMs on finetuning data and consumed resource can be even huge when the number of tasks explode (Hinton et al., 2015; Sun et al., 2019; Xia et al., 2022; Yang et al., 2022). Additionally, it is acknowledged that task-agnostic distillation typically brings performance gain over task-specific distillation does (Zhang et al., 2022a, 2023b).

The above-mentioned merits has inspired prior studies to study the task-agnostic distillation of encoder-only LMs (e.g., BERT, Devlin et al., 2019) and decoder-only LMs (e.g., GPT, Radford et al., 2019). However, the study on the distillation of the encoder-decoder LMs (e.g., T5, Raffel et al., 2020) is solely limited to a task-specific aspect (Shleifer and Rush, 2020; Zhang et al., 2022b; Li et al., 2022; Tao et al., 2022), while task-specific distillation of encoder-decoder LMs remains unexplored. Core and unique to encode-decoder LMs, the interplay between encoder and decoder in the cross-attention layers should be explicitly rendered in the distillation of encoder-decoder LMs. Failing to account for the encoder-decoder interplay, existing distillation methods invariably fail to handle task-agnostic distillation of encoder-decoder LMs, as shown in Figure 1.

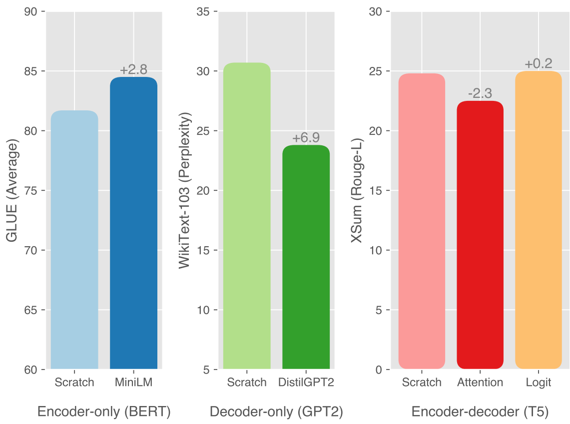

Figure 1: The failures of prior distillation methods. The setting is to distil a base-scale teacher to a 6-layer student. Either distilling last layer self-attention distributions (Wang et al., 2021) or logits (Sanh et al., 2019) for encoder-decoder LMs yields severe degradation or only marginal gain compared to pretraining from scratch, in contrast to significant improvements for either encoder-only or decoder-only LMs. Note that the lower the perplexity, the better.

[♠] Corresponding author.

[♦] The code is available at https://github.com/GeneZC/MiniEnD

LREC-COLING 2024, pages 15629–15639

To this end, we carry out the first investigation ever on task-agnostic distillation of encoder-decoder LMs, with a focus set on the encoder-decoder interplay. In a preliminary study, we show that distillation of encoder-decoder LMs can contribute to stable gradient norms of last hidden states, a key desiderata to the expressiveness of LMs. Motivated by it, we offer a path named as MiniEnD that successfully tackles the distillation of encoder-decoder LM by alternatively distilling the cross-attention to explicitly fall to both the encoder and the decoder.

We check MiniEnD on language understanding and abstractive summarization in sense that encoder-decoder LMs are more capable of sequence-to-sequence tasks. For evaluation on language understanding, we take GLUE (Wang et al., 2019) to benchmark the performance. For evaluation on abstractive summarization, we adopt CNN/DailyMail (See et al., 2017) and XSum (Narayan et al., 2018) as two testbeds. The results of both distilling T5 and BART indicate that MiniEnD is effective and competitive to other compression options such as quantization. We further scale our method up to the distillation of 3B $\text{T5}_{\text{xlarge}}$ with the aid of progressive distillation. The results suggest that distilling large language models (e.g., LLaMA, Touvron et al., 2023) should be promising but can be challenging.

2. Background

2.1. Encoder-decoder LM

Typically, an encoder-decoder LM is composed of an encoder and a decoder, each of which is essentially a stack of transformer layers (Vaswani et al., 2017). Concretely, a transformer layer in the encoder contains a multihead self-attention (MSA) module and a feedforward network (FFN) module. Similarly, a transformer layer in the decoder comprises an MSA module, an FFN module, and additionally a multihead cross-attention (MCA) module that is inserted between the MSA and the FFN modules and accounts for absorption of encoded information from the encoder. Around each of these modules is attached necessarily a layer normalization and a residual connection.

MSA and FFN Mathematically, the procedure that a transformer encoder layer consumes an intermediate encoder input $\mathbf{X} \in \mathbb{R}^{n\times d}$ containing a n-length sequence of d-dimension vectors from last layer and gives an output to next layer can be depicted as a composition of MSA and FFN:

$$\begin{aligned}&\text{MSA}(\mathbf{X}; \mathbf{W}^{\mathsf{Q}}, \mathbf{W}^{\mathsf{K}}, \mathbf{W}^{\mathsf{V}}) \\ &= \sum_{i}^{A} \text{SelfAttn}(\mathbf{X}; \mathbf{W}_i^{\mathsf{Q}}, \mathbf{W}_i^{\mathsf{K}})\mathbf{X}\mathbf{W}_i^{\mathsf{V}}\mathbf{W}_i^{\mathsf{O}},\end{aligned}$$

$$\begin{aligned}&\text{SelfAttn}(\mathbf{X}; \mathbf{W}_i^{\mathsf{Q}}, \mathbf{W}_i^{\mathsf{K}}) \\ &= \text{Softmax}(\mathbf{X}\mathbf{W}_i^{\mathsf{Q}}\mathbf{W}_i^{\mathsf{K}\top}\mathbf{X}^{\top}/d^{\mathsf{A}}),\end{aligned}$$

$$\text{FFN}(\mathbf{X}; \mathbf{W}^{\mathsf{I}}, \mathbf{W}^{\mathsf{O}}) = \sum_{j}^{I} g(\mathbf{X}\mathbf{W}_j^{\mathsf{I}})\mathbf{W}_j^{\mathsf{O}},$$

where potential details (e.g., linear bias and layer normalization) are omitted. i is used to indicate i-th head parameterized by $\mathbf{W}_i^{\mathsf{Q}}$, $\mathbf{W}_i^{\mathsf{K}}$, $\mathbf{W}_i^{\mathsf{V}} \in \mathbb{R}^{d\times d^{\mathsf{A}}}$, $\mathbf{W}_i^{\mathsf{O}} \in \mathbb{R}^{d^{\mathsf{A}}\times d}$ among A heads, and j is used to indicate j-th intermediate neuron parameterized by $\mathbf{W}_j^{\mathsf{I}} \in \mathbb{R}^{d\times 1}$ and $\mathbf{W}_j^{\mathsf{O}} \in \mathbb{R}^{1\times d}$ among I neurons. g is an activation function (e.g., GELU).

MCA Likewise, the procedure that a transformer decoder layer processes an intermediate decoder input $\mathbf{Z} \in \mathbb{R}^{m\times d}$ based on the final encoder output $\mathbf{E} \in \mathbb{R}^{n\times d}$ can be incrementally described as an insertion of MCA:

$$\begin{aligned}&\text{MCA}(\mathbf{Z}, \mathbf{E}; \mathbf{W}^{\mathsf{Q}'}, \mathbf{W}^{\mathsf{K}'}, \mathbf{W}^{\mathsf{V}'}) \\ &= \sum_{i}^{A} \text{CrossAttn}(\mathbf{Z}, \mathbf{E}; \mathbf{W}_i^{\mathsf{Q}'}, \mathbf{W}_i^{\mathsf{K}'})\mathbf{E}\mathbf{W}_i^{\mathsf{V}'}\mathbf{W}_i^{\mathsf{O}'},\end{aligned}$$

$$\begin{aligned}&\text{CrossAttn}(\mathbf{Z}, \mathbf{E}; \mathbf{W}_i^{\mathsf{Q}'}, \mathbf{W}_i^{\mathsf{K}'}) \\ &= \text{Softmax}(\mathbf{Z}\mathbf{W}_i^{\mathsf{Q}'}\mathbf{W}_i^{\mathsf{K}'\top}\mathbf{E}^{\top}/d^{\mathsf{A}}),\end{aligned}$$

Here, each cross-attention head is parameterized by another set of parameters $\mathbf{W}_i^{\mathsf{Q}'}$, $\mathbf{W}_i^{\mathsf{K}'}$, $\mathbf{W}_i^{\mathsf{V}'} \in \mathbb{R}^{d\times d^{\mathsf{A}}}$, $\mathbf{W}_i^{\mathsf{O}'} \in \mathbb{R}^{d^{\mathsf{A}}\times d}$.

2.2. Task-agnostic Distillation

Give a teacher model $\mathcal{T}$, task-agnostic distillation aims at distilling the teacher into a smaller student model $\mathcal{S}$ on pretraining data so that the student can at least outperform its pretrain-from-scratch counterpart on as many tasks as possible, as opposing one task in task-specific distillation.

3. Distillation of Encoder-decoder Interplay

In an encoder-decoder LM, the decoder is architecturally connected to the encoder through MCA modules. In spite that state-of-the-art methods mainly manipulate the decoder during distillation (e.g., logits, Zhang et al., 2022b), the encoder could be learned anyway through the connections offered by MCA modules. However, it is still not clear to what

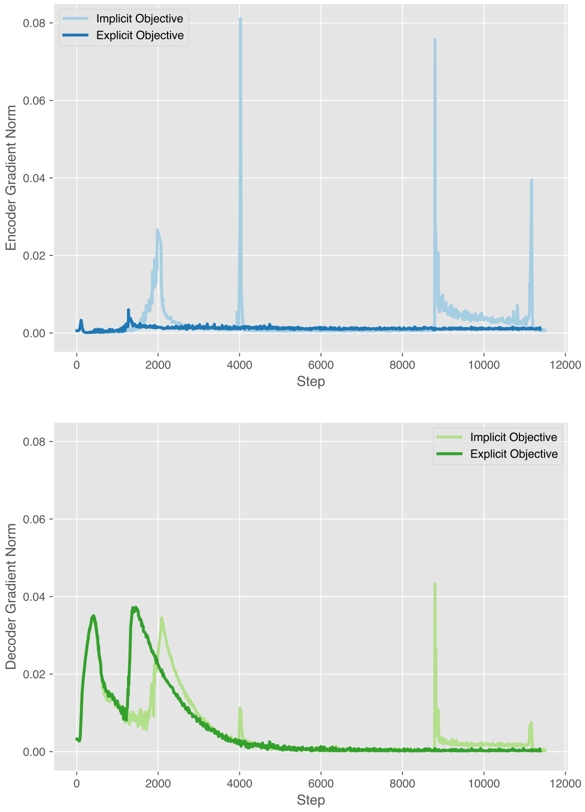

Figure 2: The preliminary results of gradient norms when using the implicit or explicit objective. The implicit objective imposes distinct gradient variations and unexpected gradient spikes during the distillation.

extent the encoder-decoder interplay is significant in the distillation and whether the implicit connections mentioned above are enough for alignment of the interplay.

A Gradient Perspective We take a closer look at the connections between the encoder and the decoder through the lens of gradients. The gradient norms of last layer hidden states of both the encoder and the decoder are examined under implicit and explicit objectives, detailed in the following paragraphs, when distilling from BART (Lewis et al., 2020). We hypothesize that a distillation objective explicitly involving the encoder-decoder interplay alignment could behave much differently in terms of gradients if the interplay is central to the distillation of encoder-decoder LMs. And naturally, if suboptimal cases are identified in the implicit objective, we can further highlight that the implicit objective suffers from the limited interplay alignment and the explicit objective can provide a more effective one.

Implicit versus Explicit Objective We instantiate the implicit objective as aligning logits and last decoder layer self-attention distributions, and the explicit objective as aligning logits, last decoder layer self-attention distributions, and last decoder layer cross-attention distributions. The core idea of last layer attention distribution alignment is borrowed from MiniLM (Wang et al., 2021). Any alignment can be abstracted as $\mathcal{L}(\mathcal{S};\mathcal{T},\mathcal{D}_*)$, where $\mathcal{D}_*$ denotes, with slight abuse of notation, the distribution of the input. As a crucial part, the alignment of self-attention is like the following:

$$
\begin{aligned}
&\mathcal{L}^{\mathsf{SelfAttn}}(\mathcal{S};\mathcal{T},\mathcal{D}_{\mathbf{Z}}) = \mathbb{E}_{\mathbf{Z}\sim\mathcal{D}_{\mathbf{Z}}}\sum_{k=1}^{R} \\
&\mathsf{KL}(\mathsf{Reln}(\mathbf{Z};{}^{\mathcal{T}}\mathbf{W}_k^{\mathsf{Q}}),\mathsf{Reln}(\mathbf{Z};{}^{\mathcal{S}}\mathbf{W}_k^{\mathsf{Q}})) \\
&+\mathsf{KL}(\mathsf{Reln}(\mathbf{Z};{}^{\mathcal{T}}\mathbf{W}_k^{\mathsf{K}}),\mathsf{Reln}(\mathbf{Z};{}^{\mathcal{S}}\mathbf{W}_k^{\mathsf{K}})) \\
&+\mathsf{KL}(\mathsf{Reln}(\mathbf{Z};{}^{\mathcal{T}}\mathbf{W}_k^{\mathsf{V}}),\mathsf{Reln}(\mathbf{Z};{}^{\mathcal{S}}\mathbf{W}_k^{\mathsf{V}})),
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{Reln}(\mathbf{Z};{}^{\mathcal{T}}\mathbf{W}_k^{\mathsf{Q}}) \\
&\qquad = \mathsf{Softmax}(\mathbf{Z}\,{}^{\mathcal{T}}\mathbf{W}_k^{\mathsf{Q}}\,{}^{\mathcal{T}}\mathbf{W}_k^{\mathsf{Q}\top}\mathbf{Z}^\top/d^{\mathsf{R}}),
\end{aligned}
$$

where KL stands for kullback-leibler divergence. Particularly, attention heads are first merged from the original A attention heads and then split to R heads for alignment of the number of attention heads. ${}^{\mathcal{T}/\mathcal{S}}\mathbf{W}_k^{\mathsf{Q}}$ is the redistributed query parameter of the k-th head within totally R heads from the last decoder layer, likewise ${}^{\mathcal{T}/\mathcal{S}}\mathbf{W}_k^{\mathsf{K}}$ and ${}^{\mathcal{T}/\mathcal{S}}\mathbf{W}_k^{\mathsf{V}}$ are the key and value parameters.

The alignment of cross-attention is similar but sort of different in that the keys and the values are aligned in fact from the encoder side, as the following:

$$
\begin{aligned}
&\mathcal{L}^{\mathsf{CrossAttn}}(\mathcal{S};\mathcal{T},\mathcal{D}_{\mathbf{Z}},\mathcal{D}_{\mathbf{E}}) = \mathbb{E}_{\mathbf{Z}\sim\mathcal{D}_{\mathbf{Z}},\mathbf{E}\sim\mathcal{D}_{\mathbf{E}}}\sum_{k=1}^{R} \\
&\mathsf{KL}(\mathsf{Reln}(\mathbf{Z};{}^{\mathcal{T}}\mathbf{W}_k^{\mathsf{Q}'}),\mathsf{Reln}(\mathbf{Z};{}^{\mathcal{S}}\mathbf{W}_k^{\mathsf{Q}'})) \\
&+\mathsf{KL}(\mathsf{Reln}(\mathbf{E};{}^{\mathcal{T}}\mathbf{W}_k^{\mathsf{K}'}),\mathsf{Reln}(\mathbf{E};{}^{\mathcal{S}}\mathbf{W}_k^{\mathsf{K}'})) \\
&+\mathsf{KL}(\mathsf{Reln}(\mathbf{E};{}^{\mathcal{T}}\mathbf{W}_k^{\mathsf{V}'}),\mathsf{Reln}(\mathbf{E};{}^{\mathcal{S}}\mathbf{W}_k^{\mathsf{V}'})),
\end{aligned}
$$

where the notations should be self-contained by referring to previously mentioned ones. The implicit and explicit objectives are therefore as follows:

$$
\mathcal{L}^{\mathsf{Imp}} = \mathcal{L}^{\mathsf{Logit}}(\mathcal{S};\mathcal{T},\mathcal{D}_{\mathbf{Z}}) + \mathcal{L}^{\mathsf{SelfAttn}}(\mathcal{S};\mathcal{T},\mathcal{D}_{\mathbf{Z}}),
$$

$$
\begin{aligned}
\mathcal{L}^{\mathsf{Exp}} = &\mathcal{L}^{\mathsf{Logit}}(\mathcal{S};\mathcal{T},\mathcal{D}_{\mathbf{Z}}) + \mathcal{L}^{\mathsf{SelfAttn}}(\mathcal{S};\mathcal{T},\mathcal{D}_{\mathbf{Z}}) \\
&+ \mathcal{L}^{\mathsf{CrossAttn}}(\mathcal{S};\mathcal{T},\mathcal{D}_{\mathbf{Z}},\mathcal{D}_{\mathbf{E}})
\end{aligned}
$$

Preliminary Results Preliminary results are shown in Figure 2, from which we can see that 1) the implicit objective and the explicit objective lead to distinct gradient variations, and 2) in comparison to smooth gradient transitions under the explicit objective, implicit objective yields gradient

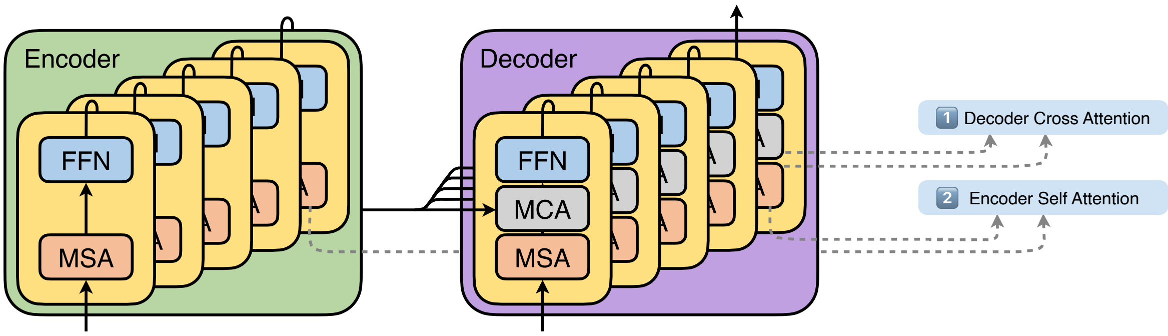

Figure 3: The overview of MiniEnD. Two directions are proposed to consider the encoder-decoder interplay alignment.

spikes, which may result in instability for a nice convergence (Zeng et al., 2022). Thereby, from the gradient perspective, we conclude that the encoder-decoder interplay is of crucial importance to the distillation of encoder-decoder LMs, and an explicit correspondence to the interplay is superior to an implicit one.

4. MiniEnD

The conclusion above drives us to the path dubbed as MiniEnD that places encoder-decoder interplay alignment in the core of encoder-decoder LM distillation. The path leads to two directions, as outlined in Figure 3.

Decoder Cross-Attention The first follows our practice in pilot study, where a fraction is always added towards the alignment of output logits. The overall distillation objective is therefore formulated as:

$$\mathcal{L}(\mathcal{S};\mathcal{T},\mathcal{D}_{\mathbf{Z}},\mathcal{D}_{\mathbf{E}}) = \mathcal{L}^{\text{Logit}}(\mathcal{S};\mathcal{T},\mathcal{D}_{\mathbf{Z}})+ \mathcal{L}^{\text{SelfAttn}}(\mathcal{S};\mathcal{T},\mathcal{D}_{\mathbf{Z}}) + \mathcal{L}^{\text{CrossAttn}}(\mathcal{S};\mathcal{T},\mathcal{D}_{\mathbf{Z}},\mathcal{D}_{\mathbf{E}}).$$

The alignment of logits can be further detailed as:

$$\mathcal{L}^{\text{Logit}}(\mathcal{S};\mathcal{T},\mathcal{D}_{\mathbf{Z}}) = \mathbb{E}_{\mathbf{Z}\sim\mathcal{D}_{\mathbf{Z}}} \text{CE}(\mathbf{Z}^{\mathcal{S}}\mathbf{W}^{\text{E}},\mathbf{Z}^{\mathcal{T}}\mathbf{W}^{\text{E}}),$$

where CE stands for soft cross entropy and ${}^{\mathcal{T}/\mathcal{S}}\mathbf{W}^{\text{E}}$ denotes output embedding.

Encoder Self-Attention In the second direction, the interplay part is instead accounted for by the last encoder self-attention distributions as:

$$\mathcal{L}(\mathcal{S};\mathcal{T},\mathcal{D}_{\mathbf{Z}},\mathcal{D}_{\mathbf{X}}) = \mathcal{L}^{\text{Logit}}(\mathcal{S};\mathcal{T},\mathcal{D}_{\mathbf{Z}})+ \mathcal{L}^{\text{SelfAttn}}(\mathcal{S};\mathcal{T},\mathcal{D}_{\mathbf{Z}}) + \mathcal{L}^{\text{EncSelfAttn}}(\mathcal{S};\mathcal{T},\mathcal{D}_{\mathbf{X}}),$$

The rationale of introducing the encoder self-attention alignment abides in that this term together with the decoder self-attention alignment can adequately replace the cross-attention term, aligning the encoder-decoder interplay by conducting alignment on each end in a separate manner.

5. Experiments

5.1. Data and Metrics

Following the pretraining of T5 and BART, we use C4 (Raffel et al., 2020) as the corpus for task-agnostic distillation of T5 and OpenWebText (Gokaslan et al., 2019) for that of BART. They are separately processed to follow the pretraining styles of T5 and BART. That is, C4 is converted to the masked language modeling style and OpenWebText is converted to the denoising style.

For evaluation of MiniEnD, we mainly take GLUE (Wang et al., 2019) for language understanding. The GLUE benchmark consists of two sequence classification tasks, SST-2 (Socher et al., 2013), i.e., CoLA (Warstadt et al., 2019), and seven sequence-pair classification tasks, i.e., MRPC (Dolan and Brockett, 2005), STS-B (Cer et al., 2017), QQP, MNLI (Williams et al., 2018), QNLI (Rajpurkar et al., 2016), RTE (Bentivogli et al., 2011), WNLI (Levesque et al., 2012). We exclude WNLI and CoLA due to the evaluation inconsistency (in other words, MiniLMs get dramatically worse results while LMs get much better ones as found out in Xia et al., 2022) and use the left tasks. Following BERT (Devlin et al., 2019), we report Accuracy (Acc) on SST-2, MNLI, QNLI, RTE, Spearman Correlation scores (SpCorr) on STS-B, and F1 on MRPC, QQP, CoNLL. Average score over tasks from GLUE (GLUE Score) is additionally computed. Regarding that one of the most promising properties of encoder-decoder LMs is sequence-to-sequence modeling, we additionally adopt CNN/DailyMail (See et al., 2017) and XSum (Narayan et al., 2018) for abstractive summarization. We report Rouge-{1,2,L} (Rg-{1,2,L}) on both of them. Results are reported on development sets. GFLOPs are also attached as theoretical speedup references.

Table 1: The data statistics, maximum sequence lengths, and metrics. The maximum decoder sequence lengths of T5 and BART are indicated differently for language understanding tasks since they use different finetuning strategies.

Dataset	#Train exam	#Dev exam	Max enc len	Max dec len	Metric
C4	364.9M	–	512	114	–
OpenWebText	37.8M	–	512	512	–
SST-2	67.3K	0.9K	64	1 / 64	Accuracy
MRPC	3.7K	0.4K	128	1 / 128	F1
STS-B	7.0K	1.5K	128	1 / 128	Spearman Correlation
QQP	364.0K	40.0K	128	1 / 128	F1
MNLI-m/mm	393.0K	20.0K	128	1 / 128	Accuracy
QNLI	105.0K	5.5K	128	1 / 128	Accuracy
RTE	2.5K	0.3K	128	1 / 128	Accuracy
CNN/DailyMail	287.1K	13.4K	512	128	F1
XSum	204.0K	11.3K	512	128	F1

Table 2: The hyperparameters for both distillation and finetuning. In order to realize the global batch size, necessary gradient accumulations should be used. The beam search setting applies to BART only.

Hyperparameter	Distillation		Finetuning		
	C4	OpenWebText	GLUE	CNN/DailyMail	XSum
Batch size	1024	1024	{16,32}	{16,32}	{16,32}
Optimizer	AdamW	AdamW	AdamW	AdamW	AdamW
Learning rate	3e-4	3e-4	{1e-5,2e-5,3e-5}	{1e-4,2e-4,3e-4}	{1e-4,2e-4,3e-4}
Training epochs	1	5	10	10	10
Earlystop epochs	–	–	5	5	5
Warmup proportion	0.01	0.01	0.1	0.1	0.1
Weight decay	0.01	0.01	0.01	0.01	0.01
Number of beams	–	–	–	4	6
Length penalty	–	–	–	2.0	1.0

The detailed data statistics, maximum sequence lengths, and metrics for datasets we use are shown in Table 1, where the corpora used for distillation is also attached.

5.2. Implementation

The distillation is carried out on 16 Nvidia A100s. The number of relation heads is set to 32. After the distillation, the finetuning is carried out on one Nvidia A100. For language understanding tasks, T5 is finetuned with simplicity and performance guarantee following EncT5 (Liu et al., 2021) which uses the very first token (i.e., `[BOS]`) representation from the decoder, while BART is finetued following its original paper which uses the very last token (i.e., `[EOS]`) representation from the decoder. As for abstractive summarization tasks, both T5 and BART are finetuned in a sequence-to-sequence manner. For fast development, we use greedy search for T5 and beam search for BART only. The beam search setting strictly follows the original paper. In order to achieve higher training efficiency, we utilize fully-sharded data parallel (Zhao et al., 2023) to shard both the teacher and the student across GPUs during the distillation. For all cases, students are always randomly initialized before the distillation following MiniLM (Wang et al., 2020).

The details of hyperparameters for distillation and finetuning are shown in Table 2. We will be releasing our code and scripts in the final version for exact reproducibility.

5.3. Baselines

We name two variants of MiniEnD as MiniEnD-D and MiniEnD-E respectively, where MiniEnD-D uses decoder cross-attention for interplay alignment and MiniEnD-E uses encoder self-attention instead. As there are no existing work in task-agnostic distillation of encoder-decoder LMs, we mainly compare MiniEnD to task-agnostic baselines that are heavily adapted to encoder-decoder LMs and task-specific baselines that may be not super fair for comparison.

We compare MiniEnD-D and MiniEnD-E distilled from T5 to task-agnostic baselines on GLUE, CNN/DailyMail, and XSum: MlmKD (Hinton et al., 2015) that directly distils masked language modeling logits; MiniLM (Wang et al., 2021) that

Table 3: The results on GLUE. The best results are **boldfaced**.

Method	GFLOPs		SST-2 Acc	MRPC F1	STS-B SpCorr	QQP F1	MNLI-m/mm Acc	QNLI Acc	RTE Acc	GLUE Score
T5$_{\text{base}}$	25.4	1×	94.6	93.0	90.0	88.9	86.7/86.8	92.9	74.7	88.5
T5$_{\text{6L;384H}}$	3.18	3∼8×	92.2	90.2	86.0	87.3	81.2/81.7	88.2	**70.0**	84.6
MiniDisc$_{5\%}$[1]	7.80		**93.8**	89.8	85.3	86.7	**82.9**/82.7	**89.2**	64.6	84.4
MlmKD$_{\text{6L;384H}}$	3.18		92.3	88.7	86.2	87.5	81.6/82.1	88.2	67.9	84.3
MiniLM$_{\text{6L;384H}}$	3.18		92.1	89.6	85.2	87.0	81.2/81.5	88.0	68.6	84.1
MlmKD+MiniLM$_{\text{6L;384H}}$	3.18		92.4	89.2	86.0	87.3	81.7/82.1	89.1	67.9	84.5
MiniEnD-D$_{\text{6L;384H}}$	3.18	8×	92.1	**90.6**	85.8	**87.7**	81.8/82.3	89.0	68.6	84.7
w/o $\mathcal{L}^{\text{Logit}}$	3.18		92.2	90.1	**86.6**	87.6	82.2/82.8	89.1	68.6	84.9
MiniEnD-E$_{\text{6L;384H}}$	3.18		92.7	90.0	86.1	87.4	81.8/82.1	88.8	69.3	84.8
w/o $\mathcal{L}^{\text{Logit}}$	3.18		92.3	89.9	**86.6**	**87.7**	82.5/**83.1**	**89.2**	69.0	**85.0**

[1] MiniDisc is distilled from T5$_{\text{xlarge}}$, and owns larger GFLOPs.

Table 4: The results on CNN/DailyMail and XSum. The best results are **boldfaced**.

Method	GFLOPs		CNN/DailyMail Rg-1	CNN/DailyMail Rg-2	CNN/DailyMail Rg-L	XSum Rg-1	XSum Rg-2	XSum Rg-L
T5$_{\text{base}}$	25.4	1×	40.1	19.4	31.5	34.7	12.4	29.7
T5$_{\text{6L;384H}}$	3.18	8×	35.7	16.8	28.4	28.6	8.9	24.8
MlmKD$_{\text{6L;384H}}$	3.18		36.0	17.0	28.7	28.9	**9.2**	25.0
MiniLM$_{\text{6L;384H}}$	3.18		35.0	16.5	28.0	25.9	7.5	22.5
MlmKD+MiniLM$_{\text{6L;384H}}$	3.18		35.8	17.0	28.7	29.0	9.1	25.1
MiniEnD-D$_{\text{6L;384H}}$	3.18	8×	**36.2**	17.2	**28.9**	**29.5**	**9.2**	**25.4**
w/o $\mathcal{L}^{\text{Logit}}$	3.18		35.7	17.0	28.6	27.3	8.2	23.7
MiniEnD-E$_{\text{6L;384H}}$	3.18		36.1	**17.3**	**28.9**	28.9	9.1	24.9
w/o $\mathcal{L}^{\text{Logit}}$	3.18		35.8	17.1	28.7	27.2	8.0	23.6
BART$_{\text{base}}$	12.7	1×	39.4	18.5	30.6	36.9	14.7	31.9
LogitKD$_{\text{3/1L;768H}}$[1]	4.23	1∼3×	38.0	16.0	25.2	32.9	12.4	26.9
DQ-BART$_{\text{8bit}}$[2]	12.7		**42.4**	**19.3**	28.8	**38.2**	**15.7**	**30.7**
MiniEnD-D$_{\text{6L;384H}}$	3.18	4×	38.5	18.5	**29.7**	33.6	12.9	29.2

[1] LogitKD is distilled with an asymmetric layer setting, i.e., more encoder layers than decoder layers, for saved performance decline.
[2] DQ-BART only quantizes parameter precision to lower one, i.e., 8 bit, but does not reduce parameter amount. Quantization would not give any speedup in GFLOPs though nice reduction in model size.

distils last decoder layer attention distributions; MlmKD+MiniLM that is essentially a combination of preceding two. We also compare MiniEnD-D and MiniEnD-E distilled from T5 to a task-specific baseline that is as far as we know the most comparable one on GLUE: MiniDisc (Zhang et al., 2022a) that exploits a teacher assistant for large compression.

On the other hand, we compare MiniEnD-D distilled from BART to two recent task-specific baselines on CNN/DailyMail and XSum: LogitKD and DQ-BART (Li et al., 2022) that jointly quantizes and distils from the teacher.

For MiniEnD above baselines, student structures are denoted either with $_{\text{*L;*H}}$ for number of layers and dimension of hidden states in random initialization, or with $_{*\%}$ for preserved portion of parameters in pruning initialization.

5.4. Main Results

Baselines fail, yet MiniEnD triumphs. From results in Table 3 and Table 4, we can tell that baselines fail to handle the distillation of encoder-decoder LMs since they either underperform the baseline pretrained from scratch or outperform it by only a small margin. For example, MlmKD+MiniLM achieves 84.5 versus 84.6 from T5 in GLUE Score, and 35.8 versus 35.7 from T5 in CNN/DailyMail Rg-1.

Contrarily, MiniEnD can safely escape from performance degradation and bring further performance increment. For example, MiniEnD-D reaches 0.1 absolute improvement in GLUE Score, and 0.9 absolute improvement in XSum Rg-1. The improvement in GLUE Score seems to be not very

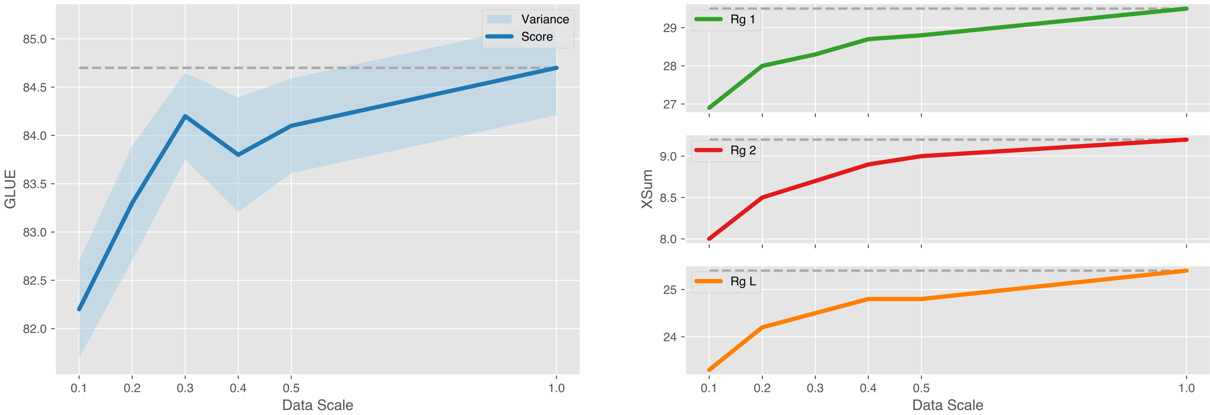

Figure 4: The results of data scaling using MiniEnD-D.

significant, but can be boosted according to the ablation. That is, MiniEnD-E w/o $\mathcal{L}^{\text{Logit}}$ goes up to 85.0, which is notably better than 84.6 from T5 in the average sense.

All count, and interplay forms the key. On another note, removing $\mathcal{L}^{\text{Logit}}$ will consistently produce performance deterioration on CNN/DailyMail and XSum. We conjecture there is a tradeoff of using between using $\mathcal{L}^{\text{Logit}}$ or not. Namely, the use of $\mathcal{L}^{\text{Logit}}$ will offer better generative ability but worse discriminative ability, and the removal of it will work reversely.

Anyway, either $\mathcal{L}^{\text{CrossAttn}}$ in MiniEnD-D or $\mathcal{L}^{\text{SelfEncAttn}}$ in MiniEnD-E shall be a crucial ingredient as the interplay alignment term is the only difference between MiniEnD and MlmKD+MiniLM but results in a considerable performance gap.

And it may be suspected that whether $\mathcal{L}^{\text{SelfAttn}}$ is still important given that MiniLM is not an ideal choice for the distillation of encoder-decoder LMs. We suggest the use of it in two aspects: 1) MlmKD+MiniLM is better than MlmKD alone; 2) the interplay alignment will witeness a subtle performance drop after the removal of $\mathcal{L}^{\text{SelfAttn}}$, say MiniEnD-D will decrease from 84.7 to 83.0 in GLUE Score.

Quantization has two sides. MiniEnD surpasses most of them except DQ-BART. However, we should emphasize that quantized LMs usually perform better but run much slower than distilled LMs do when compression is the same. In our case, DQ-BART uses 8 bit precision and gives rise to a 4$\times$ model size reduction which is the same as that of MiniEnD. In addition to that, MiniEnD is orthogonal to quantization and thus can be enhanced with other quantization schemes.

5.5. Analyses

Data Scaling Some would wonder whether the huge amounts of GPU hours due to the large pretraining corpus is necessary. So we inspect the performance variation of MiniEnD-D by varying data scale, which is shown in Figure 4.

The results generally hint that using a portion of data could hardly approximate the full data performance, though half data can achieve acceptable performance. Therefore, we suggest the use of full data in the distillation.

Model Scaling Inspired by pioneering work finding a curse that larger teachers induces worse students, we double check the existence of the curse and offer a trial solution to the curse so that we can scale the teacher up to 3B $\text{T5}_{\text{xlarge}}$.

From the results in Table 5, we observe that the curse of capacity gap still exists in our case. With the increase of teacher scale, the student performance decreases. We attempt to apply common solutions the circumvent the curse. The first is to make the student learn from a teacher assistant distilled from the teacher (Mirzadeh et al., 2020). The second is to make the student to learn from a smaller teacher and then from the teacher (Lin et al., 2023). Both two solutions inherit the idea of inserting an additional distillation step thus progressive distillation. We reveal that teacher assistant-based distillation is somewhat useful but not as excepted since $\text{T5}_{\text{xlarge}}\Rightarrow\text{T5}_{\text{12L;384H}}\Rightarrow\text{T5}_{\text{6L;384H}}$ still does not imrpove over $\text{T5}_{\text{xlarge}}\Rightarrow\text{T5}_{\text{6L;384H}}$ in some cases. Nonetheless, we unearth that progressive distillation is more promising in terms of consistent performance gains when comparing $\text{T5}_{\text{xlarge}}\Rightarrow\{\text{T5}_{\text{large}}\Rightarrow\text{T5}_{\text{12L;384H}}\}$ to $\text{T5}_{\text{xlarge}}\Rightarrow\text{T5}_{\text{12L;384H}}$. We claim that distilling large language models like LLaMA can therefore be appealing but challenging.

6. Conclusions

In this paper, we aim to provide a path that successfully tackles the distillation of encoder-decoder LMs, which fails most previous methods in the

Table 5: The results of model scaling using MiniEnD-D. $\Rightarrow$ denotes a distillation step, which should be operated sequentially otherwise $\{\}$ is prioritized. $\cdots \Rightarrow \cdots \Rightarrow \cdots$ indicates teacher assistant-based distillation and $\cdots \Rightarrow \{\cdots \Rightarrow \cdots\}$ indicates progressive distillation.

Method	**GLUE Score**	**CNN/DailyMail Rg-1**	**Rg-2**	**Rg-L**	**XSum Rg-1**	**Rg-2**	**Rg-L**
$\text{T5}_{\text{6L;384H}}$	84.6	35.7	16.8	28.4	28.6	8.9	24.8
$\text{T5}_{\text{12L;384H}}$	85.0	37.2	17.9	29.6	31.2	10.5	27.0
T5_{base}	88.5	40.1	19.4	31.5	34.7	12.4	29.7
T5_{large}	90.7	40.6	19.4	31.7	38.2	15.1	32.9
$\text{T5}_{\text{xlarge}}$	92.0	40.8	19.7	32.1	41.1	17.6	35.5
$\text{T5}_{\text{base}} \Rightarrow \text{T5}_{\text{6L;384H}}$	84.7	36.2	17.2	28.9	29.5	9.2	25.4
$\text{T5}_{\text{large}} \Rightarrow \text{T5}_{\text{6L;384H}}$	84.5	36.4	17.4	29.0	29.4	9.3	25.3
$\text{T5}_{\text{xlarge}} \Rightarrow \text{T5}_{\text{6L;384H}}$	84.2	36.1	17.2	28.8	29.1	9.1	25.1
$\text{T5}_{\text{xlarge}} \Rightarrow \text{T5}_{\text{12L;384H}} \Rightarrow \text{T5}_{\text{6L;384H}}$	84.6	36.6	17.5	29.2	29.2	9.1	25.1
$\text{T5}_{\text{large}} \Rightarrow \text{T5}_{\text{12L;384H}}$	85.5	38.3	18.4	30.4	32.4	11.2	27.9
$\text{T5}_{\text{xlarge}} \Rightarrow \text{T5}_{\text{12L;384H}}$	85.2	38.0	18.4	30.3	32.2	11.1	27.7
$\text{T5}_{\text{xlarge}} \Rightarrow \{\text{T5}_{\text{large}} \Rightarrow \text{T5}_{\text{12L;384H}}\}$	85.8	38.4	18.5	30.6	32.9	11.5	28.3

area. We find through a pilot study that the encoder-decoder interplay is a key component that should be aligned in the distillation so that the distilled encoder-decoder LMs are promising. Based on the idea, we propose two directions that the encoder-decoder interplay alignment can be incorporated and verify their effectiveness on a language understanding benchmark and two abstractive summarization datasets. We further scale the distillation of encoder-decoder LMs to a 3B teacher that requires additional distillation steps. In this sense, we recommend future research to devote more efforts to exploring how large language models can be distilled.

Limitations

This paper lacks a validation study on more recently advanced encoder-decoder LMs such as FLAN (Chung et al., 2022) and UL2 (Tay et al., 2022) as well as their instruction-tuned versions.

Acknowledgements

This work is funded in part by the Natural Science Foundation of China (grant no: 62376027) and Beijing Municipal Natural Science Foundation (grant no: 4222036 and IS23061).

Bibliographical References

Luisa Bentivogli, Peter Clark, Ido Dagan, and Danilo Giampiccolo. 2011. The seventh PASCAL recognizing textual entailment challenge. In *Proceedings of the Fourth Text Analysis Conference, TAC 2011, Gaithersburg, Maryland, USA, November 14-15, 2011*.

Daniel M. Cer, Mona T. Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. Semeval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation, SemEval@ACL 2017, Vancouver, Canada, August 3-4, 2017*, pages 1–14.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models. *CoRR*, abs/2210.11416.

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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing, IWP@IJCNLP 2005, Jeju Island, Korea, October 2005, 2005*.

Aaron Gokaslan, Vanya Cohen, Ellie Pavlick, and Stefanie Tellex. 2019. Openwebtext corpus.

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. *CoRR*, abs/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. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 4163–4174. Association for Computational Linguistics.

Hector J. Levesque, Ernest Davis, and Leora Morgenstern. 2012. The winograd schema challenge. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Thirteenth International Conference, KR 2012, Rome, Italy, June 10-14, 2012*.

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, ACL 2020, Online, July 5-10, 2020*, pages 7871–7880. Association for Computational Linguistics.

Zheng Li, Zijian Wang, Ming Tan, Ramesh Nallapati, Parminder Bhatia, Andrew O. Arnold, Bing Xiang, and Dan Roth. 2022. DQ-BART: efficient sequence-to-sequence model via joint distillation and quantization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 203–211. Association for Computational Linguistics.

Chen Liang, Haoming Jiang, Zheng Li, Xianfeng Tang, Bin Yin, and Tuo Zhao. 2023. Homodistil: Homotopic task-agnostic distillation of pre-trained transformers. *CoRR*, abs/2302.09632.

Zhenghao Lin, Yeyun Gong, Xiao Liu, Hang Zhang, Chen Lin, Anlei Dong, Jian Jiao, Jingwen Lu, Daxin Jiang, Rangan Majumder, and Nan Duan. 2023. PROD: progressive distillation for dense retrieval. In *Proceedings of the ACM Web Conference 2023, WWW 2023, Austin, TX, USA, 30 April 2023 - 4 May 2023*, pages 3299–3308. ACM.

Frederick Liu, Siamak Shakeri, Hongkun Yu, and Jing Li. 2021. Enct5: Fine-tuning T5 encoder for non-autoregressive tasks. *CoRR*, abs/2110.08426.

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. *CoRR*, abs/1907.11692.

Seyed-Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. 2020. Improved knowledge distillation via teacher assistant. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 5191–5198. AAAI Press.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 1797–1807. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100, 000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 2383–2392.

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.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1073–1083. Association for Computational Linguistics.

Sam Shleifer and Alexander M. Rush. 2020. Pre-trained summarization distillation. *CoRR*, abs/2010.13002.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1631–1642.

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 2019, Hong Kong, China, November 3-7, 2019*, pages 4322–4331. Association for Computational Linguistics.

Chaofan Tao, Lu Hou, Wei Zhang, Lifeng Shang, Xin Jiang, Qun Liu, Ping Luo, and Ngai Wong. 2022. Compression of generative pre-trained language models via quantization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 4821–4836. Association for Computational Linguistics.

Yi Tay, Mostafa Dehghani, Vinh Q. Tran, Xavier Garcia, Dara Bahri, Tal Schuster, Huaixiu Steven Zheng, Neil Houlsby, and Donald Metzler. 2022. Unifying language learning paradigms. *CoRR*, abs/2205.05131.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *CoRR*, abs/2302.13971.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

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 *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*.

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, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 2140–2151. 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 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. Neural network acceptability judgments. *Transactions on Association for Computational Linguistics*, 7:625–641.

Adina Williams, Nikita Nangia, and Samuel R. Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 1112–1122.

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), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 1513–1528. Association for Computational Linguistics.

Yi Yang, Chen Zhang, and Dawei Song. 2022. Sparse teachers can be dense with knowledge. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates,* *December 7-11, 2022*, pages 3904–3915. Association for Computational Linguistics.

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, Weng Lam Tam, Zixuan Ma, Yufei Xue, Jidong Zhai, Wenguang Chen, Peng Zhang, Yuxiao Dong, and Jie Tang. 2022. GLM-130B: an open bilingual pre-trained model. *CoRR*, abs/2210.02414.

Chen Zhang, Dawei Song, Zheyu Ye, and Yan Gao. 2023a. Towards the law of capacity gap in distilling language models. *CoRR*, abs/2311.07052.

Chen Zhang, Benyou Wang, and Dawei Song. 2023b. On elastic language models. *CoRR*, abs/2311.07204.

Chen Zhang, Yang Yang, Jiahao Liu, Jingang Wang, Yunsen Xian, Benyou Wang, and Dawei Song. 2023c. Lifting the curse of capacity gap in distilling language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 4535–4553. Association for Computational Linguistics.

Chen Zhang, Yang Yang, Qifan Wang, Jiahao Liu, Jingang Wang, Yunsen Xian, Wei Wu, and Dawei Song. 2022a. Minidisc: Minimal distillation schedule for language model compression. *CoRR*, abs/2205.14570.

Shengqiang Zhang, Xingxing Zhang, Hangbo Bao, and Furu Wei. 2022b. Attention temperature matters in abstractive summarization distillation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 127–141. Association for Computational Linguistics.

Yanli Zhao, Andrew Gu, Rohan Varma, Liang Luo, Chien-Chin Huang, Min Xu, Less Wright, Hamid Shojanazeri, Myle Ott, Sam Shleifer, Alban Desmaison, Can Balioglu, Bernard Nguyen, Geeta Chauhan, Yuchen Hao, and Shen Li. 2023. Pytorch FSDP: experiences on scaling fully sharded data parallel. *CoRR*, abs/2304.11277.