AUTODISC: AUTOMATIC DISTILLATION SCHEDULE FOR LARGE LANGUAGE MODEL COMPRESSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Driven by the teacher-student paradigm, knowledge distillation is one of the de facto ways for language model compression. Recent studies have uncovered that conventional distillation is less effective when facing a large capacity gap between the teacher and the student, and introduced teacher assistant-based distillation to bridge the gap. As a connection, the scale and the performance of the teacher assistant is crucial for transferring the knowledge from the teacher to the student. However, existing teacher assistant-based methods manually select the teacher assistant, requiring many trials before identifying the optimal teacher assistant. To this end, we propose an Automatic Distillation Schedule (AUTODISC) for large language model compression to discover the optimal teacher assistant in only one trial. In particular, motivated by the finding that the performance of the student is positively correlated to the scale-performance tradeoff of the teacher assistant, AUTODISC designs a *λ-tradeoff* to measure the optimality of the teacher assistant. AUTODISC then yields the *λ-tradeoff*s of all teacher assistant candidates in a once-for-all optimization with two approximations. The optimal teacher assistant can be automatically selected by uncovering the best *λ-tradeoff*. AUTODISC is evaluated with an extensive set of experiments on a language understanding benchmark GLUE. Experimental results demonstrate the improved efficiency with similar or even better effectiveness of our AUTODISC compared to several state-of-the-art baselines. We further apply AUTODISC to a language model with over one billion parameters and show the scalability of AUTODISC.

1 INTRODUCTION

Large language models (LLMs) (Devlin et al., 2019; Liu et al., 2019; Radford et al., 2019; Brown et al., 2020; Raffel et al., 2020) achieve promising results in various downstream tasks (Wang et al., 2019; Rajpurkar et al., 2018), but are inapplicable to those requiring limited compute or low latency (Liu et al., 2021b). To fulfill the computational requirement, LLMs can be compressed via a range of strategies such as model quantization (Zafrir et al., 2019; Bai et al., 2021), pruning (Michel et al., 2019; Hou et al., 2020), etc., among which knowledge distillation (Sun et al., 2019; Wang et al., 2020) is an appealing choice under the teacher-student paradigm. In knowledge distillation, LLMs serve as teachers and are distilled to small students.

Recent advances (Mirzadeh et al., 2020) have shown that *conventional distillation* suffers from severe performance decline when facing a large capacity gap between the teacher and the student. To alleviate the shortcoming, *teacher assistant-based distillation* (Son et al., 2021) has been proposed, where the teacher is first distilled into an teacher assistant of an intermediate scale. This teacher assistant then serves as an alternative teacher to better transfer the knowledge to the student. While teacher assistant-based distillation generally lifts the performance of the student (Wang et al., 2020; Wu et al., 2021), teacher assistants of different scales and performance may lead to different students and it is thus conjectured careful tuning is required to identify the teacher assistant that yields the best student. To verify this, we conduct a pilot study and illustrate in Figure 1 that the performance of the student is largely impacted by the choice of the teacher assistant. In fact, we observe in Figure 1 that the performance of the student is impacted by the teacher assistant from a perspective of scale-performance tradeoff. However, existing studies only manually schedule the teacher assistant (*manual distillation schedule*, in short MANDISC), resulting in an inefficient solution that requires many trials to meet the optimal teacher assistant.

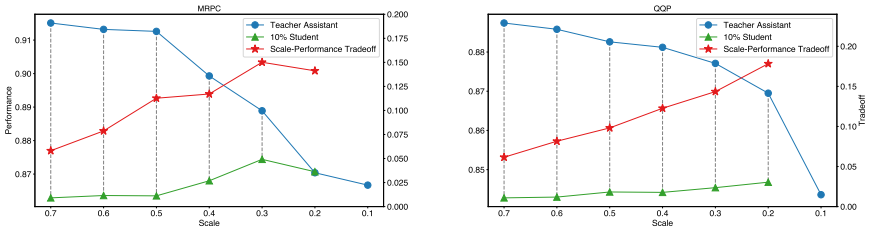


Figure 1: The impact of teacher assistants of different scales and performance on the performance of students. In the study, a $\text{BERT}_{\text{base}}$ model is used as the teacher and distilled to a pruned student (10% parameters remained) via different teacher assistants (Mirzadeh et al., 2020) on MRPC and QQP. There are several observations: (1) The blue curve shows that the performance of the teacher assistant degrades with the decreasing of its scale, which is obvious. (2) The green curve validates that the performance of the student varies with different teacher assistants. (3) The red curve represents our defined quantitative measure of the scale-performance tradeoff of the teacher assistant, which is positively correlated with the performance of the student.

To this demand, we propose an automatic distillation schedule (AUTODISC) that automatically schedules the optimal teacher assistant in only one trial. A quantitative measure, *λ-tradeoff*, is defined to empirically measure the scale-performance tradeoff for a given teacher assistant so that the optimality of the teacher assistant in terms of the student performance can be determined without trial distillation to the student. Specifically, AUTODISC is implemented in three stages. **Specification**, we leverage gridding and pruning techniques (Li et al., 2017; Frankle & Carbin, 2019) to generate a series of teacher assistant candidates of different scales and structures. **Optimization**, we demonstrate that the generated candidates satisfy the *incremental property* and the *sandwich rule*, and present two optimization approximations that yields the *λ-tradeoff* of the teacher assistant candidate of each scale in a one-run optimization. **Selection**, we select the optimal teacher assistant with the highest *λ-tradeoff*. It is noteworthy that our method can be flexibly extended to the case with multiple teacher assistants by recursively applying AUTODISC. However, this work only focuses on one teacher assistant between the teacher and student due to its sufficiency. To verify the effectiveness of AUTODISC, we conduct experiments on a language understanding benchmark GLUE (Wang et al., 2019) with both task-specific and task-agnostic distillation settings. Experimental results exhibit the competitive performance of AUTODISC compared with an array of state-of-the-art baselines, with significantly improved efficiency ($\sim$5-6$\times$) of AUTODISC compared with that of MANDISC. Further, we apply AUTODISC to a LLM EncT5_{xl} (Liu et al., 2021a) and show the scalability of AUTODISC.

Our main contributions are summarized as follows:

- We investigate the impact of teacher assistants with different scales on the performance of the student, and introduce a quantitative scale-performance tradeoff measure, *λ-tradeoff*, on the teacher assistant that is positively correlated with the student performance.
- We show two properties of the specified candidates. These properties lead to a novel optimization framework that jointly achieves yielding *λ-tradeoff*s of teacher assistants of all scales in one run. The optimization framework, together with the *λ-tradeoff*, enables a one-trial distillation schedule.
- We validate the effectiveness and efficiency of the AUTODISC. Our results of a language model with over one billion parameters show the scalability of our approach. To our best knowledge, our work is the first one exploring the distillation of true LLMs.

2 METHODOLOGY

2.1 PROBLEM FORMULATION

We formally define the problem of AUTODISC. Assuming we have a teacher $(\mathcal{T}, s_t, m_t)$ that should be distilled to a student $(\mathcal{S}, s_s, m_s)$, the goal is to find a teacher assistant $(\mathcal{A}, s_a, m_a)$ such that the student performance can be maximized when distilling the teacher to the student via the teacher

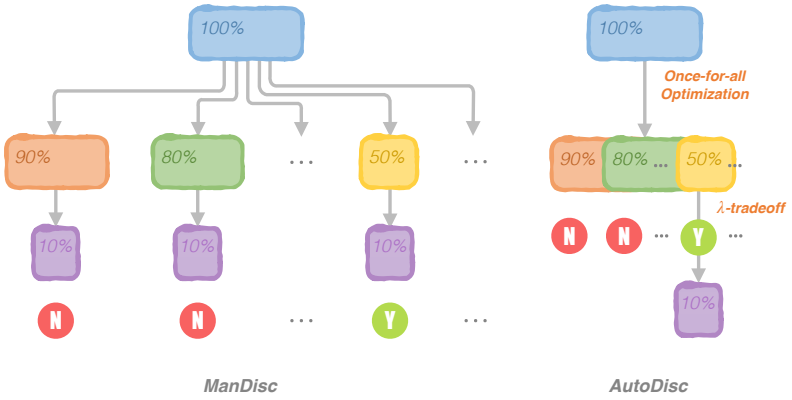


Figure 2: An overview of AUTODISC by contrasting it to MANDISC, where one arrow denotes a distillation step. AUTODISC uses only one trial while MANDISC uses many trials to discover the optimal teacher assistant. *Specification*: The scales and structures of candidates are specified by gridding the scale and pruning the structure of the teacher. *Optimization*: Candidates are sub-sampled and assembled into a sandwich-like model, and optimized in an *once-for-all optimization*. *Selection*: The candidate with the best λ-*tradeoff* measure is selected, thus the student is distilled.

assistant (i.e., $\mathcal{T} \rightarrow \mathcal{A} \rightarrow \mathcal{S}$). Here the first, second and third elements in a tuple denote the model structure, the scale, and the performance respectively. It is straightforward that the scale and the performance of the teacher assistant are bounded by the teacher and the student, i.e., $s_s \leq s_a \leq s_t$ and $m_s \leq m_a \leq m_t$. Ideally, we are seeking a teacher assistant with small scale and high performance so that the best student can be realized. However, the scale-performance tradeoff is not easy to measure. To empirically quantify scale-performance balance, we first introduce a new tradeoff measure below:

Definition 1 (**λ-*tradeoff***) *The λ scale-performance tradeoff measure of a teacher assistant $(\mathcal{A}, s_a, m_a)$ is defined as $t_a = m_a/m_t + \lambda \cdot (1 - s_a/s_t)$, where $\lambda \in [0, 1]$.*

It is clear that the value of λ-*tradeoff* is bounded by $1 + \lambda \cdot (1 - s_s/s_t)$ when the teacher assistant can achieve the performance of the teacher ($m_a = m_t$) with the student scale ($s_a = s_s$). However, in practice, this is impossible as smaller models usually lead to lower performance as shown by the blue curves in Figure 1. We further observe that the λ-*tradeoff* (red curves) of the teacher assistant is positively correlated with the performance of the student (green curves). Theoretically, due to the linear property of the λ-*tradeoff* and the concave property of the teacher assistant scale-performance correlation, there should always be one and only one maximum value of λ-*tradeoff*. Therefore, the problem can be reformulated as finding an optimal teacher assistant with the maximum value of λ-*tradeoff*:

$$(\mathcal{A}^*, s_a^*, m_a^*) = \underset{\mathcal{A}, s_a, m_a}{\operatorname{argmax}}\, t_a = \underbrace{\underbrace{\underset{s_a}{\operatorname{argmax}}\, (\, \underset{\mathcal{A}}{\operatorname{argmax}}}_{\text{Specification}}\, \underbrace{(\, \underset{m_a}{\operatorname{argmax}}\, t_a\,)}_{\text{Optimization}}\,)}_{\text{Selection}} \tag{1}$$

Based on the above reformulation, our methodology can be decomposed into three main stages: *specification*, *optimization*, and *selection*. Essentially, during *specification*, a set of teacher assistant candidates are generated of different scales. Then the performance metric of teacher assistant of each scale is obtained through a one-run *optimization*. These two stages form a feasible region for the above reformulation. Finally, the optimal teacher assistant $\mathcal{A}^*$ is selected with a linear programming of the feasible region during *selection*. After the discovery of the optimal teacher assistant, the teacher assistant can subsequently be distilled to the expected student. An overview of the methodology is given in Figure 2.

2.2 SPECIFICATION

Gridding. Theoretically, one needs to generate candidates at every possible scale to find the optimal solution. However, it is impossible to enumerate all possibilities in a continuous space. Therefore, we discretize the candidate scales into n discrete values, $\{\mathcal{A} = (\mathcal{A}_k, s_{a_k}, m_{a_k}) \mid \Delta s_a = (s_t - s_s)/n\}$, with equal slicing between the teacher scale and student scale.

Pruning. For candidates at each scale, there are still an infinite number of possible structures, e.g., different combinations of width and depth. A number of approaches have been proposed to identify a good structure at a scale, including dynamic search (Hou et al., 2020), layer dropping (Fan et al., 2020) and pruning (Michel et al., 2019). In this work, we adopt pruning to assign structures $\mathcal{A}_k$ to the candidates due to its known advantages in knowledge distillation (Xia et al., 2022). Concretely, following previous work (Michel et al., 2019), the pruning starts with the least important parameters/features based on their importance scores, which are approximated by masking the parameterized structures. The technical details of our pruning are supplied in Appendix A.

Essentially, gridding positions the scales of candidates between the scales of the teacher and student with equal intervals and pruning assigns candidates with pruned structures.

2.3 OPTIMIZATION

A straightforward solution to unearth the optimality of each candidate is exhaustively measuring the student performance distilled from each, e.g., MANDISC. *λ-tradeoff* offers a chance to measure the optimality without actual distillation. However, the memory footprints and the computational costs apparently can also be extremely large considering the number of candidates. To reduce the memory overhead and the computation complexity, we introduce two effective approximations, *parameter-sharing* and *sandwich-optimization*, so that the *λ-tradeoff*s of all candidates at different scales can be yielded in a once-for-all optimization. The feasibility of the approximations are guarded by the following two Properties.

Property 1 (Incremental Property) *For two candidates $\mathcal{A}_i$ and $\mathcal{A}_j$ in the teacher assistant candidate set $\mathcal{A}$, if $s_i < s_j$, then we have $\mathcal{A}_i \subset \mathcal{A}_j$.*

This incremental property is an outcome of the pruning approach (Li et al., 2017; Frankle & Carbin, 2019), which essentially tells that among all candidates obtained from the specification, the structure of a candidate at a smaller scale is a subset of the structure for a candidate at a larger scale.

Remark 1 *The incremental property affirms that a larger candidate can result in a smaller one by continuously pruning less significant parameters, which enables these candidates to be assembled into one sandwich-like model in a parameter-sharing fashion. The memory scale of the sandwich-like model is exactly that of the largest candidate.*

Property 2 (Sandwich Rule) *For two candidates $\mathcal{A}_i$ and $\mathcal{A}_j$ from candidate set $\mathcal{A}$, if $s_i < s_j$, then we have $m_s \leq m_i \leq m_j \leq m_t$.*

The sandwich rule (Yu & Huang, 2019; Cai et al., 2020) states that the performance of a candidate is bounded by the best performance of a larger candidate and a smaller one, due to the subset structure. Therefore, a candidate can be optimized by alternatively distilling its larger and smaller candidates, without direct distillation.

Remark 2 *The sandwich rule allows us to sub-sample η out of all n ($\eta \leq n$) filling-like candidates and conduct sandwich-optimization over the sampled candidates, which substantially reduces the computational cost.*

With the two approximations, we reduce the memory footprints of all candidates to a distinguished one with the largest scale. The computational costs are also largely reduced with the sub-sampling. Finally, we formulate the distillation objectives for task-specific distillation (TSD) and task-agnostic

distillation (TAD) respectively as below:

$$\mathcal{L}_{\mathsf{TSD}} = \sum_{i=1}^{\eta} \left(\mathrm{CE}(\mathbf{y}_{\mathcal{T}}, \mathbf{y}_{\mathcal{A}_i}) + \sum_{j=1}^{l} \mathrm{MSE}(\mathbf{X}^j_{\mathcal{T}}, \mathbf{X}^j_{\mathcal{A}_i}) \right) \tag{2}$$

$$\mathcal{L}_{\mathsf{TAD}} = \sum_{i=1}^{\eta} \sum_{j=1}^{h} \left(\mathrm{KL}({}^{Q}\mathbf{R}^j_{\mathcal{T}}, {}^{Q}\mathbf{R}^j_{\mathcal{A}_i}) + \mathrm{KL}({}^{K}\mathbf{R}^j_{\mathcal{T}}, {}^{K}\mathbf{R}^j_{\mathcal{A}_i}) + \mathrm{KL}({}^{V}\mathbf{R}^j_{\mathcal{T}}, {}^{V}\mathbf{R}^j_{\mathcal{A}_i}) \right) \tag{3}$$

where MSE, CE and KL stand for mean squared error, cross entropy and kullback-leibler divergence respectively. $\mathbf{X}^j$ is the intermediate output of the j-th layer within totally l layers, $\mathbf{y}$ is the final prediction. As is taken from MiniLMv2 (Wang et al., 2021), ${}^{Q}\mathbf{R}^j$ is the query relation matrix of the j-th head within totally h attention heads from the last layer, likewise ${}^{K}\mathbf{R}^j$ and ${}^{V}\mathbf{R}^j$ are the key and value relation matrices. Since heads can be pruned for a teacher assistant candidate, an additional self-attention module is employed as the last layer for TAD. The teacher assistants with the best performance at different scales can be obtained after the above *sandwich-optimization*. The unsampled teacher assistants can be retrieved based on the larger teacher assistant from the sampled pool using the shared parameters.

2.4 SELECTION

The intuition for imposing the teacher assistant is to adequately sacrifice teacher scale for student performance, where the adequacy should be assured by retaining teacher assistant performance as much as possible. In other words, a good teacher assistant is the one at a small scale yet with nice performance. As observed in the pilot study, the λ-*tradeoff* measure is positively correlated with the final student performance and thus is directly used as the selection criterion. The optimal teacher assistant can be identified by selecting the candidate with the best tradeoff measure. The optimal teacher assistant is then distilled to the expected student again following above distillation objectives. Note that the tradeoff measure is also dependent on λ. However, we empirically find that the optimal solution of AUTODISC is relatively stable with a wide range of λ, and we fix λ to 0.2 in all our experiments. More discussion on the impact of λ is provided in the experiments.

3 EXPERIMENTS

3.1 SETUP

Datasets and Metrics We conduct experiments on a language understanding benchmark GLUE (Wang et al., 2019). The GLUE originally consists of two sequence classification tasks, SST-2 (Socher et al., 2013) and CoLA (Warstadt et al., 2019), with seven sequence-pair classification tasks, i.e., MRPC (Dolan & Brockett, 2005), STS-B (Cer et al., 2017), QQP, MNLI (Williams et al., 2018), QNLI (Rajpurkar et al., 2016), RTE (Bentivogli et al., 2009) and WNLI (Levesque et al., 2012). We exclude WNLI and CoLA due to the evaluation inconsistency (in other words, compressed LMs get dramatically worse results while original LMs get much better ones as found out in (Xia et al., 2022)) and use the other seven tasks for evaluation. Following the work in BERT (Devlin et al., 2019), we report F1 on MRPC and QQP, Spearman Correlation scores (Sp Corr) on STS-B, and Accuracy (Acc) on other tasks. Macro average scores (Average) over these seven tasks are computed for overall performance. Results on development sets are reported. We also adopt Wikipedia for pretraining in task-agnostic disitllation. The detailed statistics, maximum sequence lengths, and metrics of GLUE and Wikipeida are supplied in Appendix B.

Implementation Details Experiments are carried out on $\mathrm{BERT}_{\mathrm{base}}$ (Devlin et al., 2019) and $\mathrm{EncT5}_{\mathrm{xl}}$ (Liu et al., 2021a). EncT5 is a language model which achieves competitive performance as $\mathrm{T5}_{\mathrm{3b}}$ (Raffel et al., 2020) on GLUE with a nearly encoder-only T5 (incorporated with a decoder layer). Our task-specific experiments are carried out on either one Nvidia A100 for $\mathrm{EncT5}_{\mathrm{xl}}$ or one Nvidia V100 for $\mathrm{BERT}_{\mathrm{base}}$, and η is set to 9 according to our empirical investigation. On the other hand, the task-agnostic experiments are carried out on eight Nvidia A100s with $\mathrm{BERT}_{\mathrm{base}}$ only[1]. The pre-training is armed with a dynamic masking strategy (Liu et al., 2019). η is set to

[1]We are not able to make it work for $\mathrm{EncT5}_{\mathrm{xl}}$ and there is no existing guidance as of now.

Table 1: The results of task-specific distillation upon $\text{BERT}_{\text{base}}$. The best and second best results are boldfaced and underlined. Training time is included for our methods.

Model (Training time)	FLOPs	SST-2	MRPC	STS-B	QQP	MNLI-m/mm	QNLI	RTE	Average
$\text{BERT}_{\text{base}}$	10.9G	93.8	91.5	87.1	88.4	84.9/84.9	91.9	71.5	86.7
$\text{BERT}_{4\text{L}}$-KD (Hinton et al., 2015)	3.6G	89.6	86.9	86.4	86.1	77.7/77.7	85.1	65.3	81.9
$\text{BERT}_{4\text{L}}$-PKD (Sun et al., 2019)	3.6G	89.9	87.6	86.4	86.0	77.7/77.7	85.0	65.3	82.0
$\text{BERT}_{4\text{L}}$-CKD (Park et al., 2021)	3.6G	89.6	87.2	86.4	86.2	77.7/77.9	85.0	64.6	81.8
$\text{DynaBERT}_{30\%}$ (Hou et al., 2020)	3.3G	90.3	87.4	87.2	86.6	81.5/81.8	89.1	66.1	83.7
$\text{BERT}_{30\%}$-FT (Li et al., 2017)	3.3G	91.9	88.5	87.2	87.7	82.0/82.6	89.5	69.0	84.8
$\text{BERT}_{30\%}$-KD (Hinton et al., 2015)	3.3G	92.0	88.9	86.8	87.8	82.2/82.7	89.9	68.2	<u>84.8</u>
$\text{BERT}_{30\%}$-$\mathcal{L}_{\text{TSD}}$ (2.5h)	3.3G	91.9	89.5	86.4	88.0	82.5/82.8	89.9	68.6	**84.9**
$\text{BERT}_{2\text{L}}$-KD (Hinton et al., 2015)	1.8G	86.8	82.5	46.8	83.7	73.5/73.1	79.6	58.1	73.0
$\text{BERT}_{2\text{L}}$-PKD (Sun et al., 2019)	1.8G	86.7	82.4	46.8	83.7	73.4/73.0	79.7	57.4	72.9
$\text{BERT}_{2\text{L}}$-CKD (Park et al., 2021)	1.8G	86.4	82.3	48.6	83.6	73.3/73.0	79.1	56.7	72.9
$\text{DynaBERT}_{15\%}$ (Hou et al., 2020)	2.2G	89.1	85.1	84.7	84.3	78.3/79.0	86.6	61.4	81.1
$\text{BERT}_{15\%}$-FT (Li et al., 2017)	1.6G	89.9	87.1	85.6	86.1	79.9/80.1	85.7	63.9	82.3
$\text{BERT}_{15\%}$-KD (Hinton et al., 2015)	1.6G	89.9	88.6	85.1	86.2	79.8/80.2	85.6	63.9	82.4
$\text{BERT}_{15\%}$-$\mathcal{L}_{\text{TSD}}$ (2.3h)	1.6G	90.1	88.9	85.1	86.5	80.0/80.2	86.0	65.3	82.8
w/ MANDISC (40.0h)	1.6G	89.8	87.7	85.4	86.9	81.0/80.1	86.1	68.2	<u>83.2</u>
w/ AUTODISC (7.9h)	1.6G	89.8	88.2	85.8	86.6	80.3/79.9	87.3	68.2	**83.3**
$\text{BERT}_{10\%}$-FT (Li et al., 2017)	1.1G	88.2	84.8	84.7	84.4	77.6/77.3	84.3	65.3	80.8
$\text{BERT}_{10\%}$-KD (Hinton et al., 2015)	1.1G	88.2	87.6	84.0	84.4	77.6/77.4	84.3	67.2	81.3
$\text{BERT}_{10\%}$-$\mathcal{L}_{\text{TSD}}$ (1.9h)	1.1G	88.8	87.8	84.0	84.6	77.6/77.5	84.9	66.4	81.5
w/ MANDISC (37.7h)	1.1G	89.0	88.2	84.8	84.8	78.3/77.8	85.3	66.8	<u>81.9</u>
w/ AUTODISC (7.0h)	1.1G	89.1	88.4	85.4	84.9	78.2/78.6	86.3	68.2	**82.4**
$\text{BERT}_{5\%}$-FT (Li et al., 2017)	0.5G	85.4	82.8	84.1	82.6	72.5/73.3	81.7	63.9	78.3
$\text{BERT}_{5\%}$-KD (Hinton et al., 2015)	0.5G	85.6	84.0	83.8	82.5	72.6/73.2	81.6	63.2	78.3
$\text{BERT}_{5\%}$-$\mathcal{L}_{\text{TSD}}$ (1.3h)	0.5G	85.4	85.5	83.9	82.7	73.0/73.4	82.7	63.2	78.7
w/ MANDISC (33.8h)	0.5G	86.1	87.0	84.1	83.8	73.7/73.6	82.9	65.7	<u>79.6</u>
w/ AUTODISC (6.9h)	0.5G	86.9	87.6	84.8	83.5	72.7/74.5	84.0	66.8	**80.1**

1 to substantially reduce computational burden. The number of relation heads is set to 32 since we use deep relation distillation as the task-agnostic distillation objective. Other implementation details are supplied in Appendix C. Generally, the sampling is performed from candidates at scales $\{100\%, 95\%, 90\%, \ldots, 10\%, 5\%\}$.

Baselines We compare our model with several state-of-the-art baselines. $\text{BERT}_{*\text{L}}$, $\text{MiniLMv2}_{*\text{L};*\text{H}}$, and $\text{TinyBERT}_{*\text{L};*\text{H}}$ denote methods via dropping layers and hidden dimensions, while $\text{DynaBERT}_{*\%}$, $\text{BERT}_{*\%}$, and $\text{EncT5}_{*\%}$ represent structured pruning with either local ranking or our global ranking (see Appendix A).

- **Conventional Distillation:** FT (Li et al., 2017) indicates direct fine-tuning after pruning. KD (Hinton et al., 2015), PKD (Sun et al., 2019) and CKD (Park et al., 2021) are methods with different objectives, i.e., KD directly distills logits, PKD distills both logits and hidden states and CKD distills token and layer relations. DynaBERT (Hou et al., 2020) uses structured pruning with a local ranking in each layer. MiniLMv2 (Wang et al., 2021) is distilled with the deep relation alignment. TinyBERT (Jiao et al., 2020) is distilled with a combination of various feature distillations.
- **Teacher Assistant-based Distillation:** MANDISC manually selects the best teacher assistant among available trials. TA (Wang et al., 2020) is specifically incorporated with MiniLMv2 for task-agnostic distillation.

3.2 MAIN RESULTS

Results of Task-specific Distillation Table 1 presents the comparison results of different methods on task-specific distillation at four student scales. The highlighted rows are the results from this work, where $\mathcal{L}_{\text{TSD}}$ denotes the student directly obtained from distillation and AUTODISC denotes the student with additional distillation using the optimal teacher assistant. There are several observations: **First**, $\mathcal{L}_{\text{TSD}}$ achieves the best performance among all conventional distillation methods, while AUTODISC further improves the model and obtains similar or even better results compared to MANDISC. This validates the improved performance of AUTODISC for automatically identifying a

Table 2: The results of task-agnostic distillation upon $\text{BERT}_{\text{base}}$. The best and second best results are boldfaced and underlined. TA stands for teacher assistant. The results of TinyBERT are reproduced based on their released checkpoints without data augmentation for a fair comparison. Training time is included for our methods.

Model (Train time)	FLOPs	SST-2	MRPC	STS-B	QQP	MNLI-m/mm	QNLI	RTE	Average
$\text{BERT}_{\text{base}}$	10.9G	93.8	91.5	87.1	88.4	84.9/84.9	91.9	71.5	86.7
$\text{MiniLMv2}_{\text{4L;384H}}$ (Wang et al., 2021)	0.9G	90.0	88.6	87.2	86.1	80.0/80.3	87.9	67.2	83.4
w/ TA (Wang et al., 2020)	0.9G	90.0	88.5	87.3	86.3	80.1/80.7	88.0	66.4	83.4
$\text{BERT}_{\text{10\%}}$-FT (Li et al., 2017)	1.1G	84.6	83.1	83.8	84.5	75.3/75.4	83.2	56.7	78.3
$\text{BERT}_{\text{10\%}}$-$\mathcal{L}_{\text{TSD}}$	1.1G	90.7	89.0	87.0	85.9	78.4/78.2	86.0	66.4	82.7
$\text{BERT}_{\text{10\%}}$-$\mathcal{L}_{\text{TAD}}$ (∼1d)	1.1G	92.0	90.1	87.9	86.6	80.0/80.3	88.0	67.2	**84.0**
w/ MANDISC (∼20d)	1.1G	91.5	90.3	87.8	86.6	80.0/80.1	88.6	67.2	**84.0**
w/ AUTODISC (∼4d)	1.1G	91.4	90.0	87.5	86.6	79.8/80.0	88.0	67.2	<u>83.8</u>
$\text{TinyBERT}_{\text{4L;312H}}$ (Jiao et al., 2020)	0.6G	88.3	88.5	84.3	84.0	77.0/77.4	82.5	63.5	80.7
$\text{MiniLMv2}_{\text{3L;384H}}$ (Wang et al., 2021)	0.7G	89.1	89.1	86.6	85.4	77.8/78.4	87.2	66.1	82.5
w/ TA (Wang et al., 2020)	0.7G	89.8	85.9	86.0	85.5	77.6/78.5	86.8	66.1	82.0
$\text{BERT}_{\text{5\%}}$-FT (Li et al., 2017)	0.5G	84.1	82.4	81.8	83.7	74.4/74.9	82.5	57.0	77.6
$\text{BERT}_{\text{5\%}}$-$\mathcal{L}_{\text{TAD}}$ (∼1d)	0.5G	90.9	89.4	87.7	85.8	79.2/79.8	87.3	65.7	<u>83.2</u>
w/ MANDISC (∼20d)	0.5G	90.1	89.7	87.4	85.6	79.3/79.7	87.1	67.9	**83.4**
w/ AUTODISC (∼4d)	0.5G	89.3	89.7	87.4	85.9	79.2/79.4	86.9	69.7	**83.4**

good teacher assistant. Notably, for further smaller $\text{BERT}_{\text{3\%}}$, the improvement still holds, as supplied in Appendix D. **Second**, conventional distillations generate reasonable results at large student scale 30% but fail to maintain the student performance at small scale 15% (with extremely worse results at scales 10% and 5% which are not included). Nonetheless, AUTODISC consistently outperforms the baselines at all scales. Additional comparisons of practical inference measurement are supplied in Appendix E. **Third**, pruning based models perform much better compared to the layer dropping methods, e.g., $\text{BERT}_{\text{15\%}}$-KD achieves much higher score than FLOPs-matched BERT_{2L}-KD, which verifies the effectiveness of pruning approach in knowledge distillation. Moreover, we discover the global ranking strategy surpasses the local ranking one by comparing $\text{BERT}_{\text{15\%}}$-$\mathcal{L}_{\text{TSD}}$ to FLOPs-matched $\text{DynaBERT}_{\text{15\%}}$. We speculate the narrow structures induced by the local ranking strategy are not effective for rather small students. The distributions of example pruned structures are supplied in Appendix F.

Results of Task-agnostic Distillation We also apply AUTODISC to task-agnostic distillation and report the results in Table 2. The first glimpse is that $\mathcal{L}_{\text{TAD}}$ surpasses $\mathcal{L}_{\text{TSD}}$, indicating the deep relation alignment is more suitable for task-agnostic distillation. Surprisingly, we discover that the pruned structures can boost the performance of MiniLMv2, i.e., BERT-$\mathcal{L}_{\text{TAD}}$, and establish a new state-of-the-art for conventional task-agnostic distillation. Another interesting observation is that teacher assistant-based distillations do not improve the performance over conventional distillations until the scale is reduced to 5%, indicating that conventional distillations are already promising choices on task-agnostic distillations. Nonetheless, we still argue the applicability of AUTODISC to task-agnostic distillation for a performance guarantee. Note that TinyBERT is less effective without data augmentation, and its results with data augmentation are supplied in Appendix G.

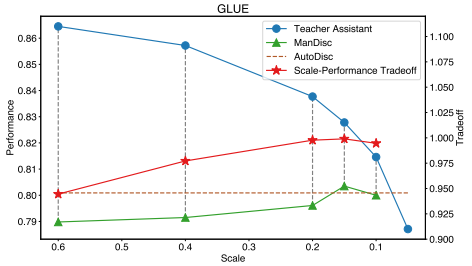
Results of Large-scale Distillation As is shown in Table 3, we conduct a similar comparison on a true LLM, EncT5_{xl}, with over one billion parameters. The very first results of a true LLM also exhibit an akin trend as the one in $\text{BERT}_{\text{base}}$. And the results on a moderate $\text{BERT}_{\text{large}}$ are supplied in Appendix H. We therefore conclude that the scalability of AUTODISC is also compelling. Reversely, the results of AUTODISC on small LMs are supplied in Appendix I.

3.3 ANALYSES

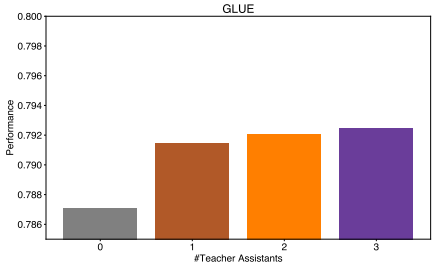
Scale-performance Tradeoff To validate the existence of scale-performance tradeoff, we use teacher assistants at different scales for MANDISC and plot performance variations of these schedules upon $\text{BERT}_{\text{base}}$ in Figure 3(a). It can be seen that reducing the scale can lead to performance improvement until a certain scale, after which performance degradation is witnessed. Almost all manual schedules underperform the automatic one. We attribute the inferiority of manual schedules to improper scale-performance tradeoffs, as concentrating only on either scale or performance will

Table 3: The results of task-specific distillation upon EncT5$_{xl}$. The best and second best results are boldfaced and underlined. Training time is included for our methods.

Model (Train time)	FLOPs	SST-2	MRPC	STS-B	QQP	MNLI-m/mm	QNLI	RTE	Average
EncT5$_{xl}$	155.9G	96.9	95.1	92.3	90.0	90.7/90.9	95.0	88.5	92.4
EncT5$_{10\%}$-FT (Li et al., 2017)	15.6G	91.6	87.1	86.7	87.9	81.9/87.0	66.1	91.6	83.8
EncT5$_{10\%}$-KD (Hinton et al., 2015)	15.6G	92.2	86.8	86.6	87.9	83.6/83.8	88.1	63.5	84.1
EncT5$_{10\%}$-$\mathcal{L}_{\text{TSD}}$ (2.3h)	15.6G	94.5	90.2	87.4	87.9	84.7/84.1	90.8	67.5	85.9
w/ ManDisc (42.1h)	15.6G	94.6	90.5	88.0	88.1	86.2/85.1	91.5	70.4	<u>86.8</u>
w/ AutoDisc (8.4h)	15.6G	94.6	91.5	87.8	87.3	85.9/85.0	91.1	72.2	**86.9**
EncT5$_{5\%}$-FT (Li et al., 2017)	7.8G	90.1	84.8	84.7	86.5	78.0/78.2	83.9	62.8	81.1
EncT5$_{5\%}$-KD (Hinton et al., 2015)	7.8G	89.9	85.1	85.4	86.6	79.4/79.6	84.2	55.6	80.7
EncT5$_{5\%}$-$\mathcal{L}_{\text{TSD}}$ (2.0h)	7.8G	92.9	88.0	83.4	85.4	79.6/80.0	87.0	58.8	81.9
w/ ManDisc (36.3h)	7.8G	93.0	88.0	83.9	86.5	81.2/81.6	88.1	67.5	<u>83.7</u>
w/ AutoDisc (7.8h)	7.8G	93.8	89.8	85.3	86.7	82.9/82.7	89.2	64.6	**84.4**



(a) The existence of scale-performance tradeoff.



(b) The sufficiency of one teacher assistant.

Figure 3: Scale-performance tradeoff studies by distilling the teacher to a student at 5% scale. The blue curve represents the performance of teacher assistants at different scales. The green curve represents the performance of ManDisc using these teacher assistants. The red curve represents the value of the scale-performance tradeoff measure. The brown dashed line represents the performance of AutoDisc. On the other hand, the brown, orange, and purple bars represent the performance of AutoDisc using one, two, and three teacher assistants.

give rise to a trivial solution with pareto optimality (Sener & Koltun, 2018; Lin et al., 2019). The overall phenomenon implies the existence of scale-performance tradeoff. Similar phenomenon is also observed in EncT5, which is supplied in Appendix J. One may note that AutoDisc does not achieve the best performance, which is expected as ManDisc exhaustively searches for the teacher assistant at all scales. However, AutoDisc is able to automatically identify a reasonably good teacher assistant in a much more efficient manner compared to ManDisc.

Sufficiency of One Teacher Assistant To examine whether one teacher assistant is sufficient, we insert more than one teacher assistant to AutoDisc and present the results in Figure 3(b). It is clear that there is no obvious performance gain when applying more than one teacher assistant (two and three) in schedules. Therefore, we alternatively choose to use only one teacher assistant in AutoDisc for training efficiency based on the sufficiency. The conclusion still holds for EncT5, which is supplied in Appendix J.

Impact of Candidate Sampling We then study the impact of the sandwich-optimization in AutoDisc by varying the number of sampled candidates η, and measuring the training cost and the student performance. From Table 4, we show the assembled sandwich together with sub-sampled fillings brings acceptable performance detriment and efficiency gain. In comparison with ManDisc which we conduct 10 trials with different teacher assistants, AutoDisc with just one candidate is able to achieve similar performance with much less training time.

Impact of λ To show λ-*tradeoff* is robust on the value of λ, we vary λ within $\{0.1,0.2,0.3\}$. It can be seen from Table 5 that the performance of AutoDisc is relatively stable with different values of

λ. Moreover, we offer a λ-independent solution using a negative derivative of performance to scale as the tradeoff measure, which yields slightly worse results, as supplied in Appendix K.

Table 4: The efficacy of approximations upon distilling $\text{BERT}_{\text{base}}$ to $\text{BERT}_{10\%}$.

Model	Train time	Average
$\text{BERT}_{10\%}$-$\mathcal{L}_{\text{TSD}}$	1.9h	81.5
w/ MANDISC	37.7h	81.9
w/ AUTODISC (η=1)	3.5h	82.1
w/ AUTODISC (η=3)	4.3h	81.9
w/ AUTODISC (η=6)	6.5h	81.9
w/ AUTODISC (η=9)	7.0h	82.4

Table 5: The impact of λ-*tradeoff* upon distilling $\text{BERT}_{\text{base}}$ to $\text{BERT}_{10\%}$.

Model	Average
$\text{BERT}_{10\%}$-$\mathcal{L}_{\text{TSD}}$	81.5
w/ MANDISC	81.9
w/ AUTODISC (λ=0.1)	82.1
w/ AUTODISC (λ=0.2)	82.4
w/ AUTODISC (λ=0.3)	81.8

4 RELATED WORK

Language Model Language models (LMs) (Devlin et al., 2019; Raffel et al., 2020) are widely adopted in various natural language tasks (Bao et al., 2020; Zhang et al., 2020). Typical LMs consist of a stack of transformer (Vaswani et al., 2017) encoder/decoder layers. Each encoder layer has two modules. The first is a self-attention module, and the second is a feed-forward module. A residual connection is employed around each of these modules, with a layer normalization placed either in (pre-norm) or out of (post-norm) the connection (Xiong et al., 2020). Each decoder layer additionally has a cross-attention module between the self-attention and feed-forward modules. While LMs exhibit excellent performance in various downstream tasks, their scales impede the deployment in real-world applications. Therefore, it is an important research problem of learning compact language models from the large ones. In our work, we aim to make LMs deployable via model compression.

Model Pruning Inspired by the idea that not all parameters contribute equally to the overall performance of a model, model pruning (Han et al., 2015) is widely adopted to waive the parameters with little impact. Model pruning spans from unstructured pruning (Frankle & Carbin, 2019; Louizos et al., 2018; Sanh et al., 2020; Chen et al., 2020) to structured pruning (Michel et al., 2019; Hou et al., 2020; Li et al., 2017; Xia et al., 2022; Lagunas et al., 2021). Unstructured pruning prunes parameters at neuron level referring to parameter magnitude (Han et al., 2015; Louizos et al., 2018) or learning dynamics (Sanh et al., 2020), while structured pruning (Michel et al., 2019; Xia et al., 2022) prunes parameters at module level relying on parameter sensitivity to performance. Although unstructured pruning enjoys a finer-grained pruning, it can only fit specialized devices. In contrast, structured pruning generally fits modern acceleration devices. In our work, we adopt structured pruning for deriving the structures of candidates for its benefits for distillation. Pruning also offers an opportunity to optimize the efficiency of our method due to its merits (Li et al., 2017; Frankle & Carbin, 2019; Yu & Huang, 2019; Cai et al., 2020).

5 CONCLUSIONS

In this paper, we propose AUTODISC to automatically identify an effective teacher assistant for teacher assistant-based distillation, bridging the large capacity gap between the teacher and student in only on trial in contrast to MANDISC. Based on the largely-neglected observation that scale-performance tradeoff of the teacher assistant is of great importance to the performance of the student, we introduce a λ-*tradeoff* measure that quantifies the scale-performance tradeoff of the teacher assistant, and show that it is positively correlated with the student performance. To compute the measures for possible teacher assistant candidates, we leverage gridding and pruning to specify these candidates and achieve a once-for-all optimization for these candidates based on two properties. The best teacher assistant is selected according to the λ-*tradeoff* value. Comprehensive experiments demonstrate the improved performance and applicability of AUTODISC. Experimental results of a language model over one billion parameters show the scalability of AUTODISC.

REFERENCES

Haoli Bai, Wei Zhang, Lu Hou, Lifeng Shang, Jin Jin, Xin Jiang, Qun Liu, Michael R. Lyu, and Irwin King. Binarybert: Pushing the limit of BERT quantization. In *ACL-IJCNLP*, pp. 4334–4348, 2021. URL https://doi.org/10.18653/v1/2021.acl-long.334.

Siqi Bao, Huang He, Fan Wang, Hua Wu, and Haifeng Wang. PLATO: pre-trained dialogue generation model with discrete latent variable. In *ACL*, pp. 85–96, 2020. URL https://doi.org/10.18653/v1/2020.acl-main.9.

Luisa Bentivogli, Bernardo Magnini, Ido Dagan, Hoa Trang Dang, and Danilo Giampiccolo. The fifth PASCAL recognizing textual entailment challenge. In *TAC*, 2009. URL https://tac.nist.gov/publications/2009/additional.papers/RTE5_overview.proceedings.pdf.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *NeurIPS*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html.

Han Cai, Chuang Gan, Tianzhe Wang, Zhekai Zhang, and Song Han. Once-for-all: Train one network and specialize it for efficient deployment. In *ICLR*, 2020. URL https://openreview.net/forum?id=HylxE1HKwS.

Daniel M. Cer, Mona T. Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. Semeval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *SemEval@ACL*, pp. 1–14, 2017. URL https://doi.org/10.18653/v1/S17-2001.

Tianlong Chen, Jonathan Frankle, Shiyu Chang, Sijia Liu, Yang Zhang, Zhangyang Wang, and Michael Carbin. The lottery ticket hypothesis for pre-trained BERT networks. In *NeurIPS*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/b6af2c9703f203a2794be03d443af2e3-Abstract.html.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, pp. 4171–4186, 2019. URL https://doi.org/10.18653/v1/n19-1423.

William B. Dolan and Chris Brockett. Automatically constructing a corpus of sentential paraphrases. In *IWP@IJCNLP*, 2005. URL https://aclanthology.org/I05-5002/.

Angela Fan, Edouard Grave, and Armand Joulin. Reducing transformer depth on demand with structured dropout. In *ICLR*, 2020. URL https://openreview.net/forum?id=SylO2yStDr.

Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *ICLR*, 2019. URL https://openreview.net/forum?id=rJl-b3RcF7.

Song Han, Jeff Pool, John Tran, and William Dally. Learning both weights and connections for efficient neural network. In *NeurIPS*, pp. 1135–1143, 2015. URL https://proceedings.neurips.cc/paper/2015/file/ae0eb3eed39d2bcef4622b2499a05fe6-Paper.pdf.

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. *arXiv*, abs/1503.02531, 2015. URL http://arxiv.org/abs/1503.02531.

Lu Hou, Zhiqi Huang, Lifeng Shang, Xin Jiang, Xiao Chen, and Qun Liu. Dynabert: Dynamic BERT with adaptive width and depth. In *NeurIPS*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/6f5216f8d89b086c18298e043bfe48ed-Abstract.html.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. Tinybert: Distilling BERT for natural language understanding. In *EMNLP*, volume EMNLP 2020 of *Findings of ACL*, pp. 4163–4174, 2020. URL https://doi.org/10.18653/v1/2020.findings-emnlp.372.

François Lagunas, Ella Charlaix, Victor Sanh, and Alexander M. Rush. Block pruning for faster transformers. In *EMNLP*, pp. 10619–10629, 2021. URL https://doi.org/10.18653/v1/2021.emnlp-main.829.

Hector J. Levesque, Ernest Davis, and Leora Morgenstern. The winograd schema challenge. In *KR*, 2012. URL http://www.aaai.org/ocs/index.php/KR/KR12/paper/view/4492.

Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. In *ICLR*, 2017. URL https://openreview.net/forum?id=rJqFGTslg.

Xi Lin, Hui-Ling Zhen, Zhenhua Li, Qingfu Zhang, and Sam Kwong. Pareto multi-task learning. In *NeurIPS*, pp. 12037–12047, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/685bfde03eb646c27ed565881917c71c-Abstract.html.

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

Xiangyang Liu, Tianxiang Sun, Junliang He, Lingling Wu, Xinyu Zhang, Hao Jiang, Zhao Cao, Xuanjing Huang, and Xipeng Qiu. Towards efficient NLP: A standard evaluation and A strong baseline. *arXiv*, abs/2110.07038, 2021b. URL https://arxiv.org/abs/2110.07038.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *arXiv*, abs/1907.11692, 2019. URL http://arxiv.org/abs/1907.11692.

Christos Louizos, Max Welling, and Diederik P. Kingma. Learning sparse neural networks through l_0 regularization. In *ICLR*, 2018. URL https://openreview.net/forum?id=H1Y8hhg0b.

Paul Michel, Omer Levy, and Graham Neubig. Are sixteen heads really better than one? In *NeurIPS*, pp. 14014–14024, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/2c601ad9d2ff9bc8b282670cdd54f69f-Abstract.html.

Seyed-Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. Improved knowledge distillation via teacher assistant. In *AAAI*, pp. 5191–5198, 2020. URL https://ojs.aaai.org/index.php/AAAI/article/view/5963.

Pavlo Molchanov, Stephen Tyree, Tero Karras, Timo Aila, and Jan Kautz. Pruning convolutional neural networks for resource efficient inference. In *ICLR*, 2017. URL https://openreview.net/forum?id=SJGCiw5gl.

Geondo Park, Gyeongman Kim, and Eunho Yang. Distilling linguistic context for language model compression. In *EMNLP*, pp. 364–378, 2021. URL https://doi.org/10.18653/v1/2021.emnlp-main.30.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *Preprint*, 2019. URL https://d4mucfpksywv.cloudfront.net/better-language-models/language-models.pdf.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*, 21:140:1–140:67, 2020. URL http://jmlr.org/papers/v21/20-074.html.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100, 000+ questions for machine comprehension of text. In *EMNLP*, pp. 2383–2392, 2016. URL https://doi.org/10.18653/v1/d16-1264.

Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for squad. In Iryna Gurevych and Yusuke Miyao (eds.), *ACL*, pp. 784–789, 2018. URL https://aclanthology.org/P18-2124/.

Victor Sanh, Thomas Wolf, and Alexander M. Rush. Movement pruning: Adaptive sparsity by fine-tuning. In *NeurIPS*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/eae15aabaa768ae4a5993a8a4f4fa6e4-Abstract.html.

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. In *NeurIPS*, pp. 525–536, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/432aca3a1e345e339f35a30c8f65edce-Abstract.html.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *EMNLP*, pp. 1631–1642, 2013. URL https://aclanthology.org/D13-1170/.

Wonchul Son, Jaemin Na, Junyong Choi, and Wonjun Hwang. Densely guided knowledge distillation using multiple teacher assistants. In *ICCV*, pp. 9375–9384, 2021. URL https://doi.org/10.1109/ICCV48922.2021.00926.

Siqi Sun, Yu Cheng, Zhe Gan, and Jingjing Liu. Patient knowledge distillation for BERT model compression. In *EMNLP-IJCNLP*, pp. 4322–4331, 2019. URL https://doi.org/10.18653/v1/D19-1441.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, pp. 5998–6008, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *ICLR*, 2019. URL https://openreview.net/forum?id=rJ4km2R5t7.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *NeurIPS*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Wenhui Wang, Hangbo Bao, Shaohan Huang, Li Dong, and Furu Wei. Minilmv2: Multi-head self-attention relation distillation for compressing pretrained transformers. In *ACL-IJCNLP*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pp. 2140–2151, 2021. URL https://doi.org/10.18653/v1/2021.findings-acl.188.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. Neural network acceptability judgments. *TACL*, 7:625–641, 2019. URL https://transacl.org/ojs/index.php/tacl/article/view/1710.

Adina Williams, Nikita Nangia, and Samuel R. Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *NAACL-HLT*, pp. 1112–1122, 2018. URL https://doi.org/10.18653/v1/n18-1101.

Yimeng Wu, Mehdi Rezagholizadeh, Abbas Ghaddar, Md. Akmal Haidar, and Ali Ghodsi. Universal-kd: Attention-based output-grounded intermediate layer knowledge distillation. In *EMNLP*, pp. 7649–7661, 2021. URL https://doi.org/10.18653/v1/2021.emnlp-main.603.

Mengzhou Xia, Zexuan Zhong, and Danqi Chen. Structured pruning learns compact and accurate models. *arXiv*, abs/2204.00408, 2022. URL https://doi.org/10.48550/arXiv.2204.00408.

Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tie-Yan Liu. On layer normalization in the transformer architecture. In *ICML*, volume 119 of *PMLR*, pp. 10524–10533, 2020. URL http://proceedings.mlr.press/v119/xiong20b.html.

Jiahui Yu and Thomas S. Huang. Universally slimmable networks and improved training techniques. In *ICCV*, pp. 1803–1811, 2019. URL https://doi.org/10.1109/ICCV.2019.00189.

Ofir Zafrir, Guy Boudoukh, Peter Izsak, and Moshe Wasserblat. Q8BERT: quantized 8bit BERT. In *EMC2@NeurIPS*, pp. 36–39, 2019. URL https://doi.org/10.1109/EMC2-NIPS53020.2019.00016.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. PEGASUS: pre-training with extracted gap-sentences for abstractive summarization. In *ICML*, volume 119 of *PMLR*, pp. 11328–11339, 2020. URL http://proceedings.mlr.press/v119/zhang20ae.html.

A TECHNICAL DETAILS OF PRUNING

Concretely, following previous work (Michel et al., 2019), the pruning always starts with the least important parameters/features, which are identified according to *importance scores*. The importance scores are approximated by first masking the parameterized structures. $\mu^{(i)}$, $\nu^{(i)}$, and $\xi^{(j)}$ denote the mask variables respectively for a self-attention head, optionally a cross-attention head, and a feed-forward neuron, such that for an intermediate input $\mathbf{X}$ and potentially an encoder-produced input $\mathbf{E}$:

$$\mathbf{Z}^{\circ} = \text{SelfAttention}^{\circ}(\mathbf{X}) = \sum_{i}^{h} \mu^{(i)} \text{softmax}(\mathbf{X}\mathbf{W}_{Q}^{(i)}\mathbf{W}_{K}^{(i)\top}\mathbf{X}^{\top})\mathbf{X}\mathbf{W}_{V}^{(i)}\mathbf{W}_{O}^{(i)}, \tag{4}$$

$$\mathbf{Z}^{\circ} = \text{CrossAttention}^{\circ}(\mathbf{Z}^{\circ}, \mathbf{E}) = \sum_{i}^{h} \nu^{(i)} \text{softmax}(\mathbf{Z}^{\circ}\mathbf{W}_{Q'}^{(i)}\mathbf{W}_{K'}^{(i)\top}\mathbf{E}^{\top})\mathbf{E}\mathbf{W}_{V'}^{(i)}\mathbf{W}_{O'}^{(i)}, \tag{5}$$

$$\widetilde{\mathbf{X}}^{\circ} = \text{FeedForward}^{\circ}(\mathbf{Z}^{\circ}) = \sum_{j}^{d} \xi^{(j)} g(\mathbf{Z}^{\circ}\mathbf{W}_{1}^{(j)})\mathbf{W}_{2}^{(j)}, \tag{6}$$

where potential bias terms (e.g., linear bias and position bias) are omitted, i means i-th head among h heads, j means j-th intermediate neuron among d neurons, and g is an activation function. We initialize all mask variables to ones to preserve the original structure at the very beginning.

Then expected absolute gradients over either fine-tuning or pre-training data gives the important scores:

$$\mathbb{I}_{\mu}^{(i)} = \mathbb{E}_{(x,y)\sim\mathcal{D}} \left| \frac{\partial \mathcal{L}(x,y)}{\partial \mu^{(i)}} \right|, \mathbb{I}_{\nu}^{(i)} = \mathbb{E}_{(x,y)\sim\mathcal{D}} \left| \frac{\partial \mathcal{L}(x,y)}{\partial \nu^{(i)}} \right|, \mathbb{I}_{\xi}^{(j)} = \mathbb{E}_{(x,y)\sim\mathcal{D}} \left| \frac{\partial \mathcal{L}(x,y)}{\partial \xi^{(j)}} \right|, \tag{7}$$

where (x, y) is a data point and $\mathcal{L}$ is the task-specific loss for task-specific models or the language modeling loss for pre-trained models. $\mathbb{E}$ represents expectation. The absolute value of gradient for a mask indicates how large the impact of pruning the corresponding structure is, thus implying how important the structure is.

Intuitively, we take a global ranking, in contrast to a local one as in other literature (Hou et al., 2020), for the structures of the same type (i.e., attention head or feed-forward element) from all stacking layers for pruning preference, before which we also normalize the importance scores for same-type structures in a layer with ℓ_2 norm, as suggested by Molchanov et al. (2017), for a balanced pruning. Therefore, for each candidate, we separately prune attention heads and feed-forward elements to the scale so that we reach a qualified structure. For the sake of a corner case that all structures in a module are pruned, we skip the module by feeding the input as the output. While we can alternate to an quite recent pruning method (Xia et al., 2022) exploiting both coarse-grained and fine-grained strategies for state-of-the-art performance, we argue that our framework is agnostic to pruning methods and keep the pruning method simple.

B DATASET STATISTICS

We conduct experiments on seven datasets. The detailed statistics, maximum sequence lengths, and metrics for datasets we use are shown in Table 6, where the Wikipedia corpus used for pretraining is also attached.

C ADDITIONAL IMPLEMENTATION DETAILS

The summary of hyperparameters for both task-specific and task-agnostic distillation is shown in Table 7. We will be releasing our code and scripts in the final version for exact reproducibility.

D ADDITIONAL RESULTS UPON $\text{BERT}_{\text{BASE}}$

We further conduct experiments on extremely small scale student model, i.e., $\text{BERT}_{3\%}$. The results are shown in Table 8.

Table 6: The statistics, maximum sequence lengths, and metrics.

Dataset	#Train exam.	#Dev exam.	Max. length	Metric
SST-2	67K	0.9K	64	Accuracy
MRPC	3.7K	0.4K	128	F1
STS-B	7K	1.5K	128	Spearman Correlation
QQP	364K	40K	128	F1
MNLI-m/mm	393K	20K	128	Accuracy
QNLI	105K	5.5K	128	Accuracy
RTE	2.5K	0.3K	128	Accuracy
Wikipedia	35M	-	128	-

Table 7: The hyperparameters for both task-specific and task-agnostic distillation. The learning rate is searched within different grids for $\text{BERT}_{\text{base}}$ and EncT5_{xl}.

Hyperparameter	Task-specific Distillation	Task-agnostic Distillation
Batch Size	{16,32}	8×128=1024
Optimizer	AdamW	AdamW
Learning Rate	{1e-5, 2e-5, 3e-5}/{1e-4, 2e-4, 3e-4}	3e-4
Training Epochs	10	5
Early-stop Epochs	5	-
Warmup Proportion	0.1	0.01
Weight Decay	0.01	0.01
Sampling Number η	9	1

E Inference Measurement

Since FLOPs only offers theoretical inference compute, we additionally provide throughput for empirical inference compute of each model with throughput (i.e., processed tokens per micro second) in Table 9. The test environment is established by feeding 32×128 tokens to models. The amount of decomposed parameters is also attached for a reference.

F Pruned Structure Distributions

We give the distributions of example pruned structures in Figure 4, which exactly show what pruned LMs consist of. While pruned $\text{BERT}_{\text{base}}$ tends to preserve bottom and middle layers, pruned EncT5_{xl} tends to preserve bottom layers. Meanwhile, neurons in feed-forward layers are more likely to be pruned than heads in attention layers, owing to the centrality of the attention module within an transformer layer.

G Data Augmentation for TinyBERT

We compare TinyBERT with and without data augmentation as in Table 10. The results with data augmentation are retrieved from the original paper, since the augmented data is not publicly available. The results demonstrate that TinyBERT is largely supported with data augmentation for good performance.

H Results upon $\text{BERT}_{\text{LARGE}}$

We show extended results of AutoDisc on $\text{BERT}_{\text{large}}$ for readers' interest in Table 11. Consistent patterns have been observed as in $\text{BERT}_{\text{base}}$.

Table 8: Additional results of task-specific distillation upon $\text{BERT}_{\text{base}}$.

Model	FLOPs	SST-2	MRPC	STS-B	QQP	MNLI-m/mm	QNLI	RTE	Average
$\text{BERT}_{3\%}$-$\mathcal{L}_{\text{TSD}}$	0.3G	85.2	83.6	81.9	82.1	71.9/72.7	81.9	57.4	77.1
w/ MANDISC	0.3G	85.6	85.0	82.7	82.7	72.7/72.8	82.0	59.6	77.9
w/ AUTODISC	0.3G	85.9	85.7	83.6	83.1	72.9/73.6	81.9	58.1	78.1

Table 9: Inference compute measurement.

Model	FLOPs	Throughput	Trm params	Emb params
$\text{BERT}_{\text{base}}$	10.9G	55.7tokens/ms	85.7M	23.8M
$\text{BERT}_{10\%}$	1.1G	278.2tokens/ms	9.1M	23.8M
$\text{BERT}_{5\%}$	0.5G	412.9tokens/ms	4.9M	23.8M
$\text{BERT}_{\text{large}}$	38.7G	17.9tokens/ms	303.3M	31.8M
$\text{BERT}_{10\%}$	3.9G	104.1tokens/ms	31.3M	31.8M
$\text{BERT}_{5\%}$	1.9G	154.2tokens/ms	16.3M	31.8M
EncT5_{xl}	155.8G	4.8tokens/ms	1275.1M	32.9M
$\text{EncT5}_{10\%}$	15.6G	38.8tokens/ms	127.4M	32.9M
$\text{EncT5}_{5\%}$	7.8G	64.0tokens/ms	64.0M	32.9M

I RESULTS OF SMALL-SCALE DISTILLATION

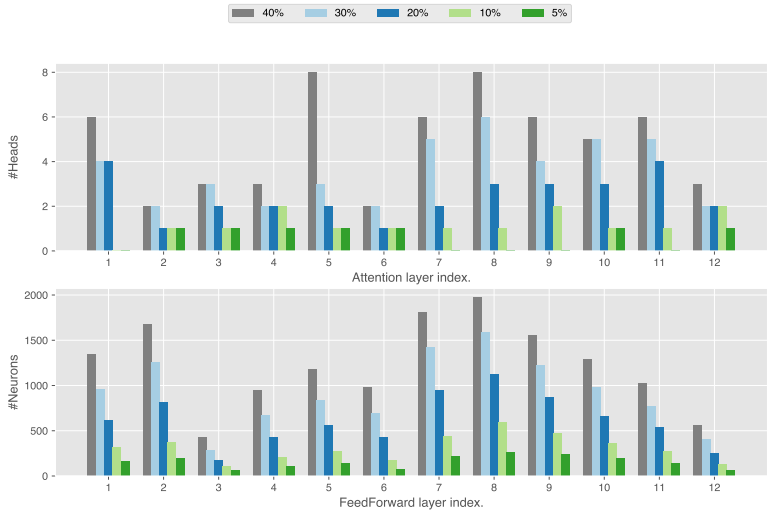
When AUTODISC is applied to small $\text{MiniLM}_{12;384\text{H}}$ and $\text{BERT}_{\text{mini}}$ as shown in Table 12, AUTODISC can reversely affect the performance of conventional distillation. Contrarily, MANDISC can still improve or at least retain the performance. However, it is less necessary to compress small LMs.

J VARYING SCHEDULES FOR ENCT5

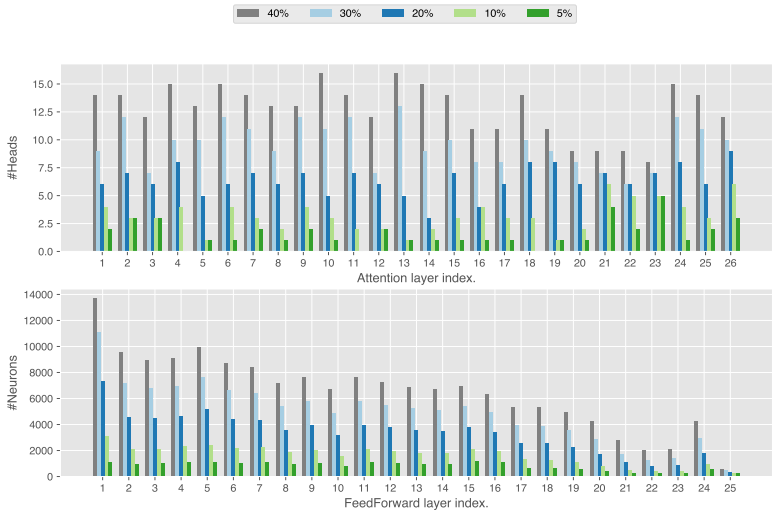
Performance variations among possible schedules for EncT5 are displayed in Figure 5, where the existence of scale-performance tradeoff and sufficiency of one teacher assistant can be verified.

K NEGATIVE DERIVATIVE-TRADEOFF

As mentioned in the main paper, although λ-*tradeoff* is able to provide stable tradeoff measurement, it is dependent on the value of λ. To eliminate this dependency, we design a new measure, negative derivative-tradeoff, which computes the negative derivative of performance to scale at each candidate scale as: $t_a = \lim_{\delta \to 0} \frac{-(m_{a+\delta} - m_a)}{s_{a+\delta} - s_a}$. In the discrete case, $t_{a_i} = \frac{-(m_{a_{i+1}} - m_{a_i})}{\Delta s_a}$. The idea of the measure is basically derived from saving the performance from a potentially significant drop. However, first-order estimation can lead to a high estimation variance and can be further tuned with second-order or so for better performance. The comparison results using λ-*tradeoff* and *ND-tradeoff* are shown in Table 13. It can be seen from the table that AUTODISC-ND also achieves comparable results.



(a) 12-layer $\text{BERT}_{\text{base}}$.



(b) 24-layer $\text{EncT5}_{1.5}$. Layer indices lager than 24 denote modules from the one-layer decoder (i.e., two more attention modules and one more feed-forward modules).

Figure 4: The distributions of example pruned structures. The structures are derived with MRPC dataset.

Table 10: The results of TinyBERT with and without DA.

Model	FLOPs	SST-2	MRPC	STS-B	QQP	MNLI-m/mm	QNLI	RTE	Average
$\text{TinyBERT}_{4\text{L};312\text{H}}$ (Jiao et al., 2020)	0.6G	88.3	88.5	84.3	84.0	77.0/77.4	82.5	63.5	80.7
w/ DA (Jiao et al., 2020)	0.6G	92.7	90.2	86.3	87.1	82.8/82.8	88.0	65.7	84.5
$\text{MiniLMv2}_{3\text{L};384\text{H}}$ (Wang et al., 2021)	0.7G	89.1	89.1	86.6	85.4	77.8/78.4	87.2	66.1	82.5

Table 11: The results of task-specific distillation upon $\text{BERT}_{\text{large}}$.

Model	FLOPs	SST-2	MRPC	STS-B	RTE	Average
$\text{BERT}_{\text{base}}$	10.9G	93.8	91.5	87.1	71.5	86.0
$\text{BERT}_{10\%}$-$\mathcal{L}_{\text{TSD}}$	1.1G	88.8	87.8	84.0	66.4	81.8
w/ MANDISC	1.1G	89.0	88.2	84.8	66.8	82.2
w/ AUTODISC	1.1G	89.1	88.4	85.4	68.2	82.7
$\text{BERT}_{5\%}$-$\mathcal{L}_{\text{TSD}}$	0.5G	85.4	85.5	83.9	63.2	79.5
w/ MANDISC	0.5G	86.1	87.0	84.1	65.7	80.7
w/ AUTODISC	0.5G	86.9	87.6	84.8	66.8	81.5
$\text{BERT}_{\text{large}}$	38.7G	94.2	92.5	90.1	75.5	88.1
$\text{BERT}_{10\%}$-$\mathcal{L}_{\text{TSD}}$	3.9G	90.4	88.1	87.0	66.1	82.9
w/ MANDISC	3.9G	90.6	88.9	87.1	67.2	83.4
w/ AUTODISC	3.9G	90.5	88.8	87.8	66.1	83.3
$\text{BERT}_{5\%}$-$\mathcal{L}_{\text{TSD}}$	1.9G	89.2	85.7	85.8	61.4	80.5
w/ MANDISC	1.9G	90.4	86.0	85.7	62.8	81.2
w/ AUTODISC	1.9G	89.6	87.4	87.3	61.4	81.4
EncT5_{xl}	155.9G	96.9	95.1	92.3	88.5	93.2
$\text{EncT5}_{10\%}$-$\mathcal{L}_{\text{TSD}}$	15.6G	94.5	90.2	87.4	67.5	84.9
w/ MANDISC	15.6G	94.6	90.5	88.0	70.4	85.9
w/ AUTODISC	15.6G	94.6	91.5	87.8	72.2	86.5
$\text{EncT5}_{5\%}$-$\mathcal{L}_{\text{TSD}}$	7.8G	92.9	88.0	83.4	58.8	80.8
w/ MANDISC	7.8G	93.0	88.0	83.9	67.5	83.1
w/ AUTODISC	7.8G	93.8	89.8	85.3	64.6	83.4

Table 12: The results of task-specific distillation upon small LMs.

Model	FLOPs	SST-2	MRPC	STS-B	QQP	MNLI-m/mm	QNLI	RTE	Average
$\text{MiniLM}_{\text{12L;384H}}$	2.72G	92.1	90.9	88.6	87.2	83.0/83.3	90.7	72.9	86.1
$\text{MiniLM}_{10\%}$-$\mathcal{L}_{\text{TSD}}$	0.26G	87.8	87.1	85.6	84.3	77.2/78.4	84.8	66.4	81.5
w/ MANDISC	0.26G	88.2	88.2	86.3	84.7	77.8/79.2	85.2	65.7	81.9
w/ AUTODISC	0.26G	87.6	86.0	86.5	84.4	77.8/78.6	84.4	64.6	81.3
$\text{BERT}_{\text{mini}}$	0.60G	87.5	86.4	85.3	85.0	76.1/77.2	84.5	66.8	81.1
$\text{BERT}_{10\%}$-$\mathcal{L}_{\text{TSD}}$	0.04G	83.3	83.8	81.6	81.6	66.3/71.4	82.7	58.8	76.2
w/ MANDISC	0.04G	83.8	84.1	80.7	82.0	66.4/71.6	82.9	58.1	76.2
w/ AUTODISC	0.04G	83.3	82.9	80.6	81.1	67.4/71.3	82.8	58.5	76.0

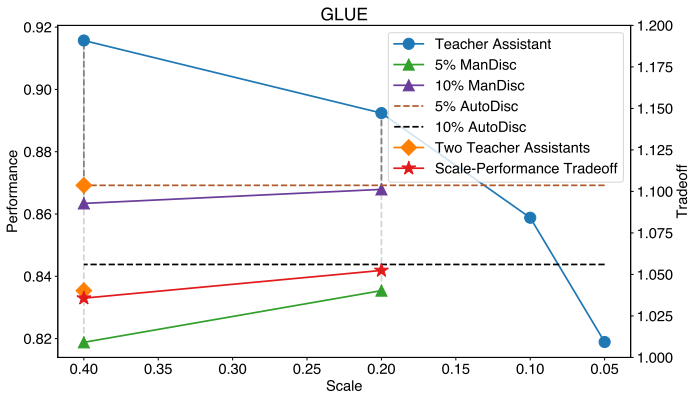


Figure 5: Performance comparisons among various schedules for EncT5. The dots represent performance variations using either one or two teacher assistants for MANDISC. The triangles represent performance resulting from AUTODISC using one teacher assistant. The rectangles represent performance resulting from AUTODISC using two teacher assistants.

Table 13: The results of negative derivative-tradeoff upon $\text{BERT}_{\text{base}}$.

Model	FLOPs	SST-2	MRPC	STS-B	RTE	Average
$\text{BERT}_{\text{base}}$	10.9G	93.8	91.5	87.1	71.5	86.0
$\text{BERT}_{10\%}$-$\mathcal{L}_{\text{TSD}}$	1.1G	88.8	87.8	84.0	66.4	81.8
w/ MANDISC	1.1G	89.0	88.2	84.8	66.8	82.2
w/ AUTODISC-λ	1.1G	89.1	88.4	85.4	68.2	82.7
w/ AUTODISC-ND	1.1G	89.8	87.9	85.4	66.4	82.4
$\text{BERT}_{5\%}$-$\mathcal{L}_{\text{TSD}}$	0.5G	85.4	85.5	83.9	63.2	79.5
w/ MANDISC	0.5G	86.1	87.0	84.1	65.7	80.7
w/ AUTODISC-λ	0.5G	86.9	87.6	84.8	66.8	81.5
w/ AUTODISC-ND	0.5G	86.8	86.0	84.9	66.8	81.1