

2 Related Work

Task-specific Distillation Sun et al. (2019b) task-specifically compressed BERT by learning from the every k -th layer of the teacher. To avoid leaving out some of the teacher layers, many follow-up works (Wu et al., 2020, Passban et al., 2021, Wu et al., 2021) designed new layer mapping strategies to fuse the teacher layers. Jiao et al. (2020) used data augmentation to further improve the performance. Initialising the student model with pre-trained weights is crucial for performance since the student learns from the teacher only shortly in downstream tasks. Common choices for initialization are: (1) task-agnostically distilling models first, (2) using publicly available distilled models, or (3) initializing with teacher layers. As part of this study, we examine how to maximize the benefits of initializing from teacher layers.

Task-agnostic Distillation In the field of task-agnostic distillation, one line of work is to compress the teacher model into a student model with the same depth but narrower blocks (Sun et al., 2020b, Zhang et al., 2022). Another line of work is to distill the teacher into a student with fewer layers (Sanh et al., 2019, Jiao et al., 2020, Wang et al., 2020, Wang et al., 2021), which is our focus.

Comparative Studies Li et al. (2021) conducted out-of-domain and adversarial evaluation on three KD methods, which used hidden state transfer or data augmentation. Lu et al. (2022) is closely related to our work, where they also evaluated knowledge types and initialisation schemes. However, they did not consider layer choice when initialising from the teacher, and the evaluation was only for task-specific settings. Hence, our work complements theirs.

3 Distillation Objectives

Prediction Layer Transfer Prediction layer transfer minimizes the soft cross-entropy between the logits from the teacher and the student: $\mathcal{L}_{\text{pred}} = \text{CE}(z^T/t, z^S/t)$, with z^T and z^S the logits from the teacher/student and t is the temperature value.

Following the vanilla KD approach (Hinton et al., 2015), the final training loss is a combination of $\mathcal{L}_{\text{pred}}$ and supervision loss \mathcal{L}_{ce} (masked language modelling loss \mathcal{L}_{mlm} in the pertaining stage). We denote this objective as **vanilla KD**.

Hidden States Transfer Hidden states transfer penalizes the distance between the hidden states of specific layers from the teacher and the student. Common choices for the representation are the embedding of the [CLS] token (Sun et al., 2019b) and the whole sequence embedding (Jiao et al., 2020). We use Mean-Squared-Error (MSE) to measure the distance between the student and teacher embedding, which can be formulated as $\mathcal{L}_{\text{hid}} = \text{MSE}(\mathbf{h}^S \mathbf{W}_h, \mathbf{h}^T)$, where $\mathbf{h}^S \in \mathbb{R}^d$ and $\mathbf{h}^T \in \mathbb{R}^{d'}$ are the [CLS] token embedding of specific student and teacher layer, d and d' are the hidden dimensions. The matrix $\mathbf{W}_h \in \mathbb{R}^{d \times d'}$ is a learnable transformation. We denote this objective as **Hid-CLS**. In the case of transferring the sequence embedding, one can replace the token embeddings with sequence embeddings $\mathbf{H}^S \in \mathbb{R}^{l \times d}$ and $\mathbf{H}^T \in \mathbb{R}^{l \times d'}$, where l is the sequence length. The objective that transfers the sequence embedding with MSE loss is denoted as **Hid-Seq**.

We also evaluated a contrastive representation learning method which transfers the hidden state representation from the teacher to the student with a contrastive objective (Sun et al., 2020a). We inherited their code for implementation and refer our readers to the original paper for details. We denote this objective as **Hid-CLS-Contrast**.

Attention and Value Transfer The attention mechanism has been found to capture rich linguistic knowledge (Clark et al., 2019), and attention map transfer is widely used in transformer model distillation. To measure the similarity between the multi-head attention block of the teacher and the student, MSE and Kullback-Leibler divergence are the two standard loss functions. The objective using MSE is formulated as $\mathcal{L}_{\text{att}} = \frac{1}{h} \sum_{i=1}^h \text{MSE}(\mathbf{A}_i^S, \mathbf{A}_i^T)$, where h is the number of attention heads, matrices $\mathbf{A}_i \in \mathbb{R}^{l \times l}$ refers to the i -th attention head (before the softmax operation) in the multi-head attention block. We denote this objective as **Att-MSE**.

Since the attention after the softmax function is a distribution over the sequence, we can also use the KL-divergence to measure the distance: $\mathcal{L}_{\text{att}} = \frac{1}{TH} \sum_{t=1}^T \sum_{h=1}^H D_{KL}(a_{t,h}^T \| a_{t,h}^S)$, where T is the sequence length and H is the number of attention heads. We will denote this objective as **Att-KL**. In addition to attention transfer, value-relation transfer was proposed by Wang et al. (2020), to which we refer our readers for details. Value-relation transfer objective will be denoted as **Val-KL**.

Objectives	QNLI Acc	SST-2 Acc	MNLI Acc	MRPC F1	QQP Acc	RTE Acc	CoLA Mcc	Avg
Vanilla KD	66.5 \pm 1.49	84.7 \pm 0.16	75.1 \pm 0.05	71.2 \pm 0.80	81.9 \pm 0.10	54.0 \pm 1.24	69.1 \pm 0.00	71.8
Hid-CLS-Contrast	69.3 \pm 0.60	85.3 \pm 0.56	76.2 \pm 0.45	71.1 \pm 0.85	83.1 \pm 0.69	53.6 \pm 0.23	69.0 \pm 0.12	72.5
Hid-CLS	75.7 \pm 0.57	85.8 \pm 0.34	77.0 \pm 0.10	71.3 \pm 0.41	83.8 \pm 1.63	54.0 \pm 2.17	68.4 \pm 0.35	73.2
Hid-Seq	83.3 \pm 0.13	87.4 \pm 0.13	78.3 \pm 0.13	72.9 \pm 0.50	87.6 \pm 0.00	51.8 \pm 1.10	69.2 \pm 0.55	75.8
Att-MSE	84.3 \pm 0.18	89.2 \pm 0.40	78.6 \pm 0.25	71.1 \pm 0.41	88.7 \pm 0.05	54.4 \pm 1.03	69.3 \pm 0.17	76.5
+Hid-Seq	84.6 \pm 0.29	89.2 \pm 0.21	78.9 \pm 0.10	71.8 \pm 0.51	88.8 \pm 0.00	54.0 \pm 0.93	69.5 \pm 0.48	77.0
Att-KL	85.3 \pm 0.14	89.0 \pm 0.26	79.4 \pm 0.08	71.4 \pm 0.29	89.0 \pm 0.05	55.5 \pm 2.05	69.3 \pm 0.13	77.0
+Hid-Seq	84.6 \pm 0.21	89.1 \pm 0.46	79.5 \pm 0.17	72.4 \pm 0.39	89.0 \pm 0.06	57.2 \pm 0.86	69.3 \pm 0.21	77.3
+Val-KL	85.5 \pm 0.24	89.6 \pm 0.31	79.6 \pm 0.10	72.2 \pm 0.39	89.1 \pm 0.05	57.5 \pm 0.70	69.2 \pm 0.15	77.5

Table 1: Task-specific distillation results on GLUE dev sets. Student models are initialised with every 4th layer of the teacher model. We report the average and standard deviation over 4 runs. Attention based objectives consistently outperform hidden states transfer and vanilla KD.

Objectives	QNLI Acc	SST-2 Acc	MNLI Acc	MRPC F1	QQP Acc	RTE Acc	CoLA Mcc	Avg
DistilBERT*	89.2	91.3	82.2	87.5	88.5	59.9	51.3	78.5
TinyBERT [†]	90.5	91.6	83.5	88.4	90.6	72.2	42.8	79.9
MiniLM [§]	91.0	92.0	84.0	88.4	91.0	71.5	49.2	81.0
Vanilla KD*	88.6	91.4	82.4	86.5	90.6	61.0	54.4	79.3
Hid-CLS	86.5	90.6	79.3	73.0	89.7	61.0	33.9	73.4
Hid-Seq	89.2	91.5	82.3	89.2	90.3	67.2	48.2	79.7
Att-MSE	89.8	91.6	83.2	90.6	90.7	69.7	53.5	81.3
+Hid-Seq [†]	89.7	92.4	82.8	90.4	90.8	68.6	52.8	81.1
Att-KL	88.0	89.7	81.1	90.1	90.3	66.1	43.6	78.4
+Hid-Seq	88.9	91.6	82.4	90.0	90.5	66.8	47.9	79.7
+Val-KL [§]	89.8	91.6	82.4	91.0	90.6	66.7	47.7	80.0

Table 2: Task-agnostic distillation: Performance on GLUE dev sets of three existing distilled 6-layer Transformer models and our 6-layer students distilled. All the students are randomly initialised and distilled from BERT_{BASE}. We report the best fine-tuning result with grid search over learning rate and batch size. Att-MSE performs the best among all the objectives.

4 Experimental Setup

We evaluate our model on the General Language Understanding Evaluation (GLUE) benchmark (Wang et al., 2018) tasks, including linguistic acceptability (CoLA), sentiment analysis (SST-2), semantic equivalence (MRPC, QQP), and natural language inference (MNLI, QNLI, RTE).

For task-specific distillation, we distill a fine-tuned RoBERTa_{BASE} (Liu et al., 2019) into a 3-layer transformer model on each GLUE task, using the Fairseq (Ott et al., 2019) implementation and the recommended hyperparameters presented in Liu et al. (2019). We follow the training procedure from TinyBERT to perform *intermediate layer* and *prediction layer* distillation sequentially for 10 epochs each, freeing us from tuning the loss weights. For intermediate layer distillation, the

student learns from the same teacher’s layers that were used for initialising the student. In addition, we always initialise the embedding layer with the teacher’s embedding layer.

For task-agnostic distillation, we distill the uncased version of BERT_{base} into a 6-layer student model, based on the implementation by Izsak et al. (2021). Here we perform last-layer knowledge transfer since we see no improvement when transferring multiple layers in our experiments. We train the student model for 100k steps with batch size 1024, a peaking learning rate of 5e-4 and a maximum sequence length of 128. The distilled student model is then fine-tuned on the GLUE datasets with grid search over batch size {16, 32} and learning rate {1e-5, 3e-5, 5e-5, 8e-5}. We follow the original training corpus of BERT: English Wikipedia and BookCorpus (Zhu et al., 2015).