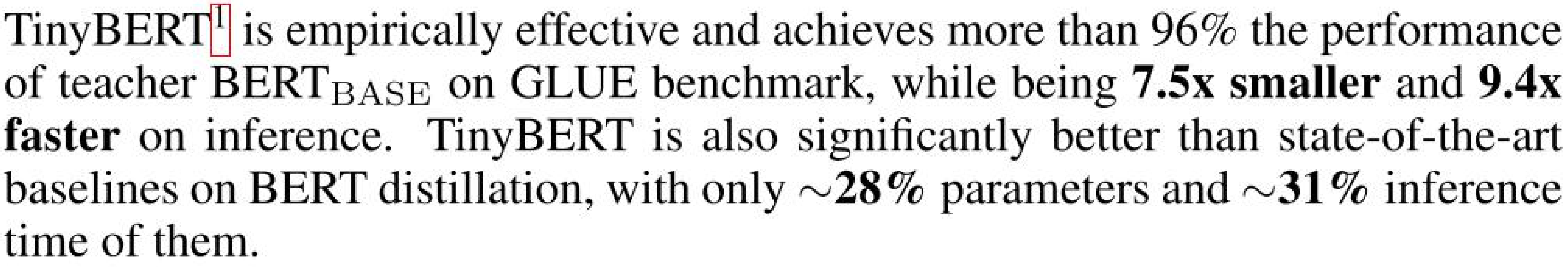
模型蒸馏

李威

TinyBERT

- Noval Transformer distillation method
- Two-Stage learning framework



TinyBERT[1] is empirically effective and achieves more than 96% the performance of teacher $\text{BERT}_{\text{BASE}}$ on GLUE benchmark, while being **7.5x smaller** and **9.4x faster** on inference. TinyBERT is also significantly better than state-of-the-art baselines on BERT distillation, with only $\sim$**28%** parameters and $\sim$**31%** inference time of them.

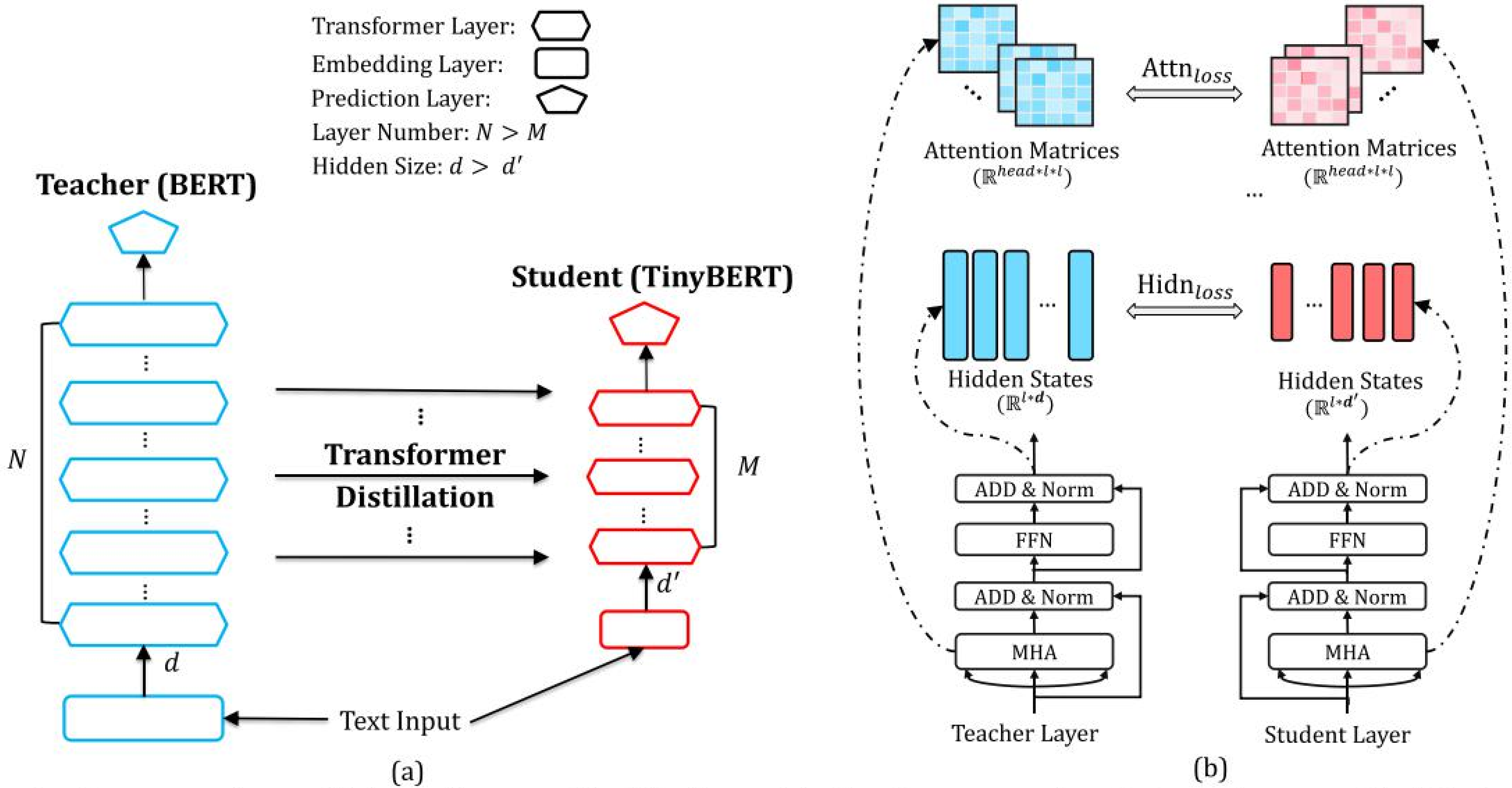


Figure 1: An overview of Transformer distillation: (a) the framework of Transformer distillation, (b) the details of Transformer-layer distillation consisting of Attn$_{loss}$(attention based distillation) and Hidn$_{loss}$(hidden states based distillation).

TinyBERT

Loss Function

- the output of the embedding layer

$$\mathcal{L}_{\text{embd}} = \texttt{MSE}(\boldsymbol{E}^S \boldsymbol{W}_e, \boldsymbol{E}^T),$$

- the hidden states and attention matrices derived from the Transformer layer

$$\mathcal{L}_{\text{hidn}} = \texttt{MSE}(\boldsymbol{H}^S \boldsymbol{W}_h, \boldsymbol{H}^T)$$

$$\mathcal{L}_{\text{attn}} = \frac{1}{h} \sum_{i=1}^{h} \texttt{MSE}(\boldsymbol{A}_i^S, \boldsymbol{A}_i^T),$$

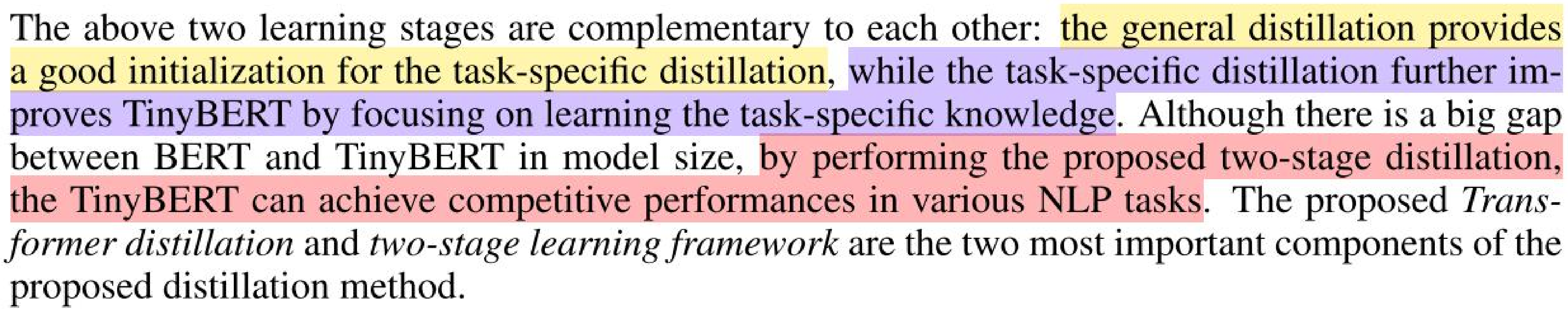
- the logits output by the prediction layer.

$$\mathcal{L}_{\text{pred}} = -\texttt{softmax}(\boldsymbol{z}^T) \cdot \texttt{log\_softmax}(\boldsymbol{z}^S / t),$$

TinyBERT

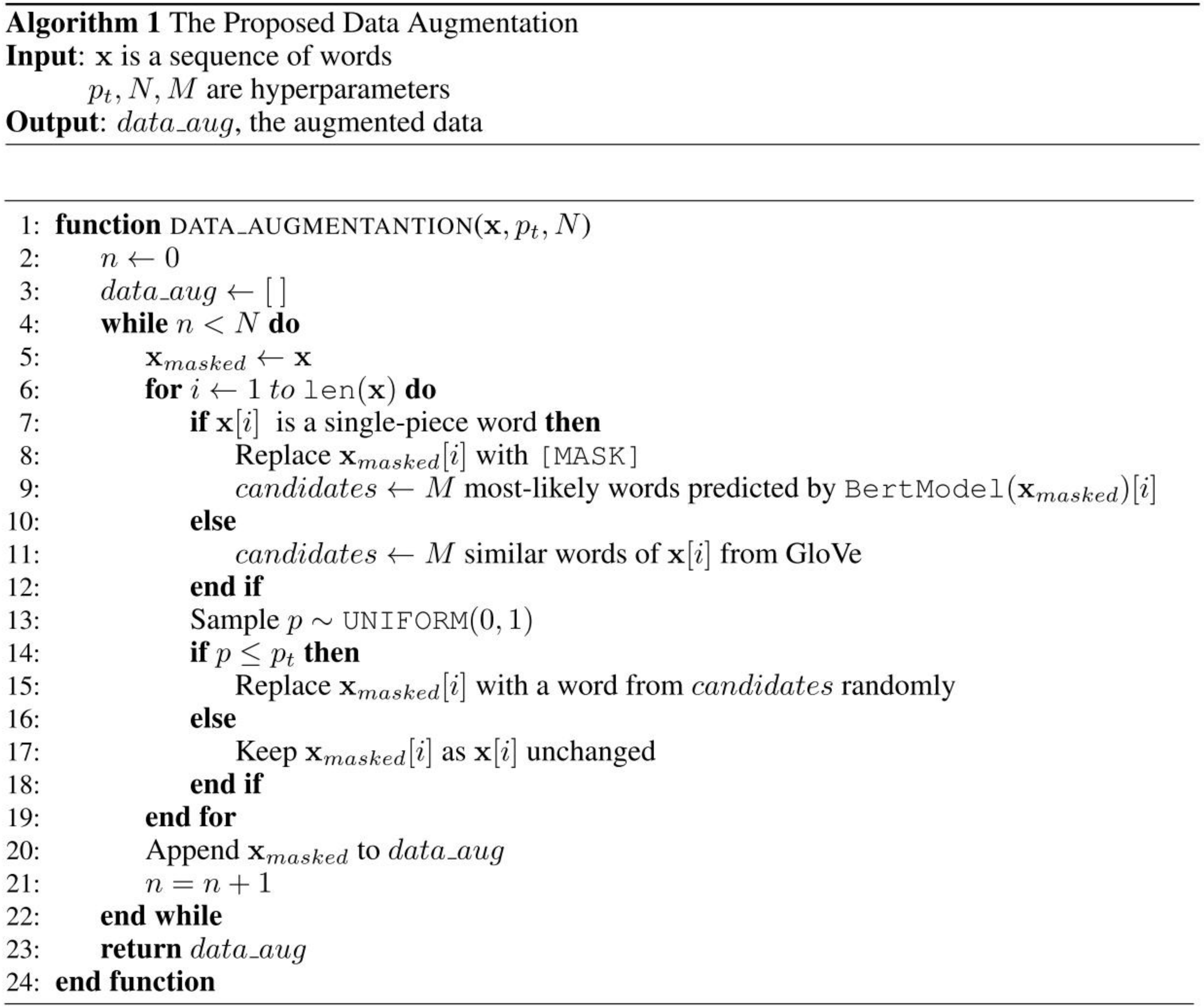
Two-Stage learning

- General distillation
    - BERT without fine-tuning as teacher model、large scale corpus from general domain as data
- Task-Specific distillation
    - Bert with fine-tuning as teacher model 、task-specific data with data agumentation



The above two learning stages are complementary to each other: the general distillation provides a good initialization for the task-specific distillation, while the task-specific distillation further improves TinyBERT by focusing on learning the task-specific knowledge. Although there is a big gap between BERT and TinyBERT in model size, by performing the proposed two-stage distillation, the TinyBERT can achieve competitive performances in various NLP tasks. The proposed *Transformer distillation* and *two-stage learning framework* are the two most important components of the proposed distillation method.

Data argmentation



Algorithm 1 The Proposed Data Augmentation
Input: $\mathbf{x}$ is a sequence of words
p_t, N, M are hyperparameters
Output: $data\_aug$, the augmented data

```
function DATA_AUGMENTANTION($\mathbf{x}, p_t, N$)
    $n \leftarrow 0$
    $data\_aug \leftarrow [\,]$
    while $n < N$ do
        $\mathbf{x}_{masked} \leftarrow \mathbf{x}$
        for $i \leftarrow 1$ to len($\mathbf{x}$) do
            if $\mathbf{x}[i]$ is a single-piece word then
                Replace $\mathbf{x}_{masked}[i]$ with [MASK]
                $candidates \leftarrow M$ most-likely words predicted by BertModel($\mathbf{x}_{masked}$)[$i$]
            else
                $candidates \leftarrow M$ similar words of $\mathbf{x}[i]$ from GloVe
            end if
            Sample $p \sim$ UNIFORM(0, 1)
            if $p \leq p_t$ then
                Replace $\mathbf{x}_{masked}[i]$ with a word from $candidates$ randomly
            else
                Keep $\mathbf{x}_{masked}[i]$ as $\mathbf{x}[i]$ unchanged
            end if
        end for
        Append $\mathbf{x}_{masked}$ to $data\_aug$
        $n = n + 1$
    end while
    return $data\_aug$
end function
```

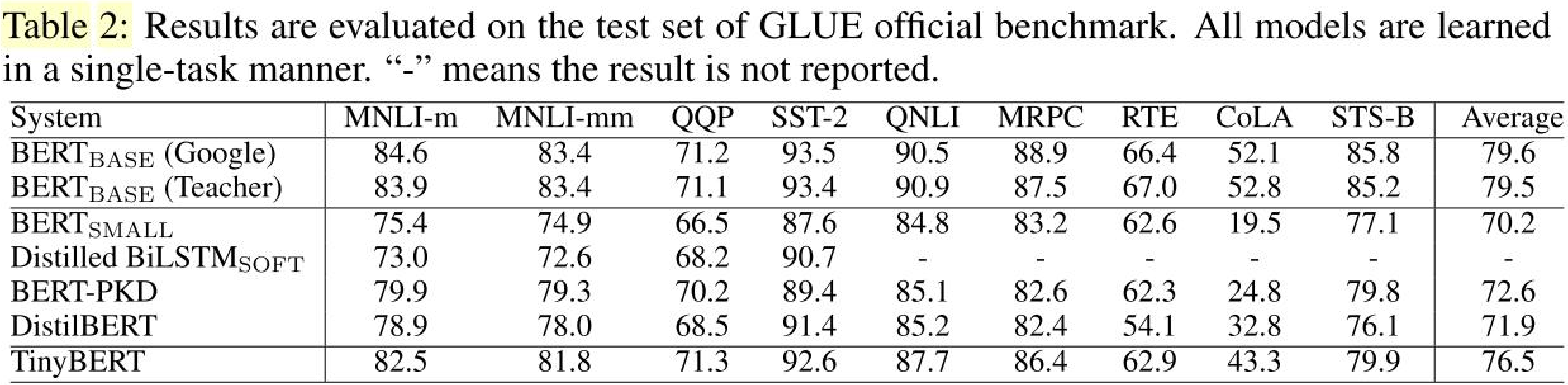


Table 2: Results are evaluated on the test set of GLUE official benchmark. All models are learned in a single-task manner. "-" means the result is not reported.

System	MNLI-m	MNLI-mm	QQP	SST-2	QNLI	MRPC	RTE	CoLA	STS-B	Average
BERT$_{\text{BASE}}$ (Google)	84.6	83.4	71.2	93.5	90.5	88.9	66.4	52.1	85.8	79.6
BERT$_{\text{BASE}}$ (Teacher)	83.9	83.4	71.1	93.4	90.9	87.5	67.0	52.8	85.2	79.5
BERT$_{\text{SMALL}}$	75.4	74.9	66.5	87.6	84.8	83.2	62.6	19.5	77.1	70.2
Distilled BiLSTM$_{\text{SOFT}}$	73.0	72.6	68.2	90.7	-	-	-	-	-	-
BERT-PKD	79.9	79.3	70.2	89.4	85.1	82.6	62.3	24.8	79.8	72.6
DistilBERT	78.9	78.0	68.5	91.4	85.2	82.4	54.1	32.8	76.1	71.9
TinyBERT	82.5	81.8	71.3	92.6	87.7	86.4	62.9	43.3	79.9	76.5

- There is a large performance gap between BERT SMALL and BERT BASE due to the big reduction in model size.

- TinyBERT is consistently better than BERT SMALL in all the GLUE tasks and achieves a large improvement of 6.3% on average. This indicates that the proposed KD learning framework can effectively improve the performances of small models regardless of downstream tasks.

- TinyBERT significantly outperforms the state-of- the-art KD baselines(i.e., BERT-PKD and DistillBERT)

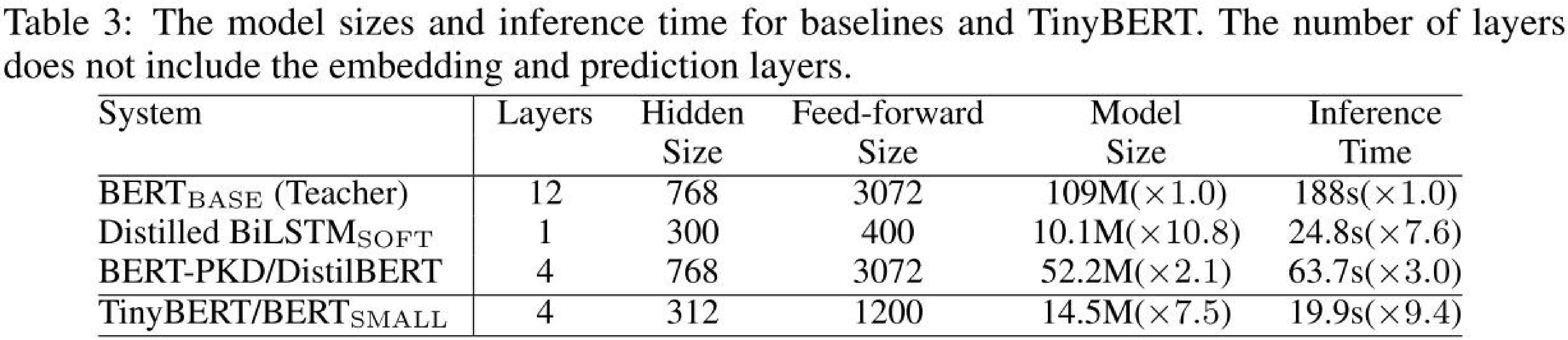


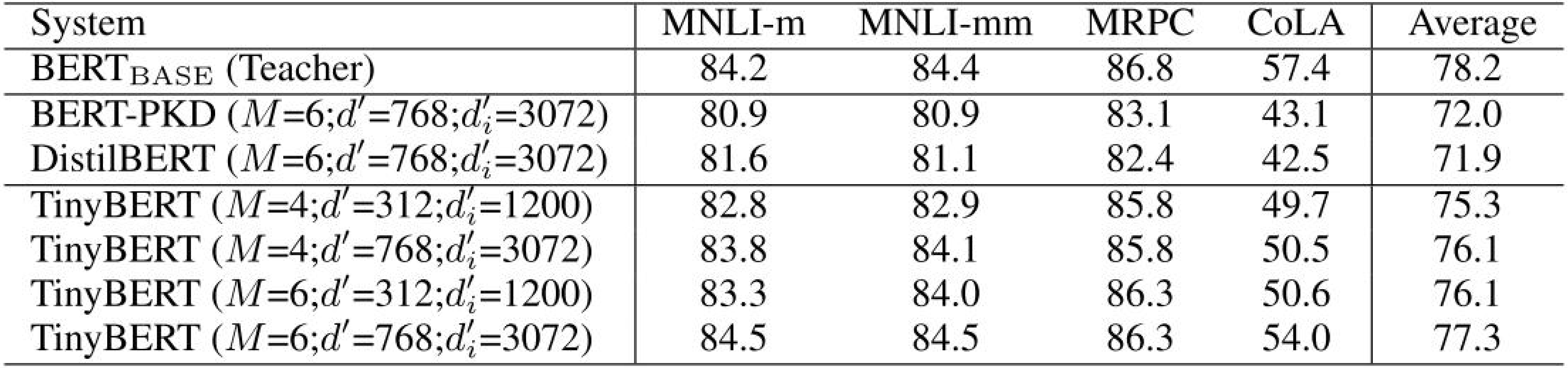
Table 3: The model sizes and inference time for baselines and TinyBERT. The number of layers does not include the embedding and prediction layers.

System	Layers	Hidden Size	Feed-forward Size	Model Size	Inference Time
BERT$_{\text{BASE}}$ (Teacher)	12	768	3072	109M(×1.0)	188s(×1.0)
Distilled BiLSTM$_{\text{SOFT}}$	1	300	400	10.1M(×10.8)	24.8s(×7.6)
BERT-PKD/DistilBERT	4	768	3072	52.2M(×2.1)	63.7s(×3.0)
TinyBERT/BERT$_{\text{SMALL}}$	4	312	1200	14.5M(×7.5)	19.9s(×9.4)

- Compared with the teacher BERTBASE, TinyBERT is 7.5x smaller and 9.4x faster in the model efficiency, while main-taining competitive performances.

- TinyBERT has a comparable model efficiency (slightly larger in size but faster in inference) with Distilled BiLSTMSOFT and obtains substantially better performances in all tasks reported by the BiLSTM baseline.

Table 4: Results (dev) of wider or deeper TinyBERT variants and baselines.



System	MNLI-m	MNLI-mm	MRPC	CoLA	Average
BERT$_{\text{BASE}}$ (Teacher)	84.2	84.4	86.8	57.4	78.2
BERT-PKD (M=6;d'=768;d'_i=3072)	80.9	80.9	83.1	43.1	72.0
DistilBERT (M=6;d'=768;d'_i=3072)	81.6	81.1	82.4	42.5	71.9
TinyBERT (M=4;d'=312;d'_i=1200)	82.8	82.9	85.8	49.7	75.3
TinyBERT (M=4;d'=768;d'_i=3072)	83.8	84.1	85.8	50.5	76.1
TinyBERT (M=6;d'=312;d'_i=1200)	83.3	84.0	86.3	50.6	76.1
TinyBERT (M=6;d'=768;d'_i=3072)	84.5	84.5	86.3	54.0	77.3

- All the three TinyBERT variants can consistently outperform the original smallest TinyBERT
- For the CoLA task, the improvement is slight when only increasing the number of layers (from 49.7 to 50.6) or hidden size (from 49.7 to 50.5). To achieve more dramatic improvements, the student model should become deeper and wider (from 49.7 to 54.0).
- Another interesting observation is that the smallest 4-layer TinyBERT can even outperform the 6-layers baselines, which further confirms the effectiveness of the proposed KD method.

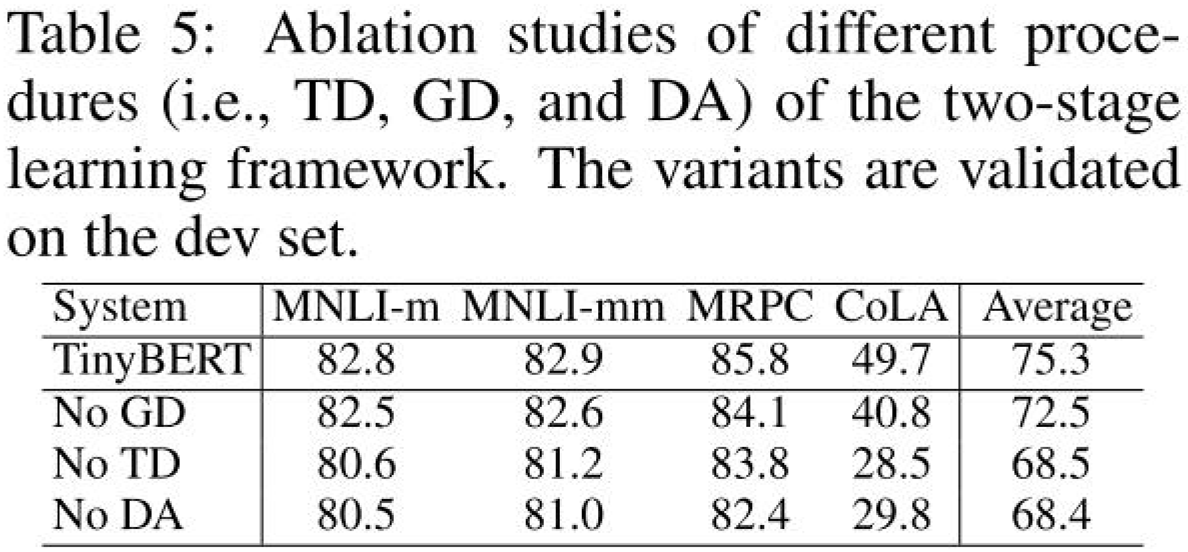


Table 5: Ablation studies of different procedures (i.e., TD, GD, and DA) of the two-stage learning framework. The variants are validated on the dev set.

System	MNLI-m	MNLI-mm	MRPC	CoLA	Average
TinyBERT	82.8	82.9	85.8	49.7	75.3
No GD	82.5	82.6	84.1	40.8	72.5
No TD	80.6	81.2	83.8	28.5	68.5
No DA	80.5	81.0	82.4	29.8	68.4

TD (Task-specific Distillation)

GD (General Distillation)

DA (Data Augmentation).

We can also find the task-specific procedures (TD and DA) are more helpful than the pre-training procedure (GD) in all the four tasks.
GD has more effect on CoLA than on MNLI and MRPC. We conjecture that the ability of linguistic generalization (Warstadt et al., 2018) learned by GD plays a more important role in the downstream CoLA task

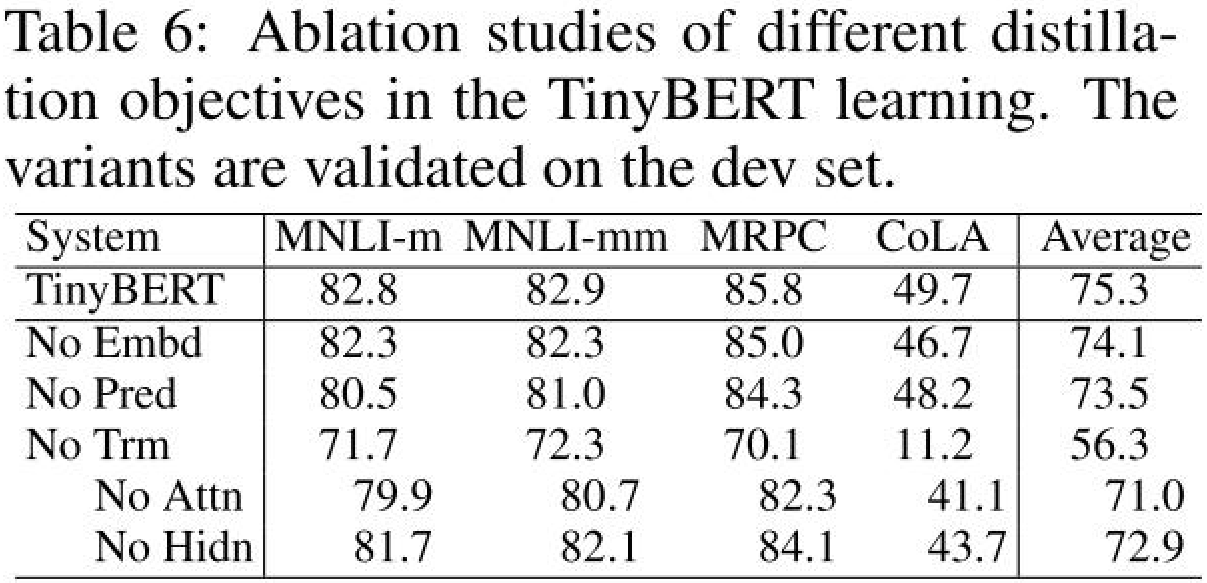


Table 6: Ablation studies of different distillation objectives in the TinyBERT learning. The variants are validated on the dev set.

System	MNLI-m	MNLI-mm	MRPC	CoLA	Average
TinyBERT	82.8	82.9	85.8	49.7	75.3
No Embd	82.3	82.3	85.0	46.7	74.1
No Pred	80.5	81.0	84.3	48.2	73.5
No Trm	71.7	72.3	70.1	11.2	56.3
No Attn	79.9	80.7	82.3	41.1	71.0
No Hidn	81.7	82.1	84.1	43.7	72.9

No Embd: embedding-layer distillation

No Pred : prediction- layer distillation

No Trm : Transformer-layer distillation

No Attn : contributions of attention

No Hidn : contributions of hidden states

- all the proposed distillation objectives are useful for the TinyBERT learning. The performance drops significantly from 75.3 to 56.3 under the setting (No Trm), which indicates Transformer-layer distillation is the key for TinyBERT learning.
- We can find the attention based distillation has a bigger effect than hidden states based distillation on TinyBERT learning.

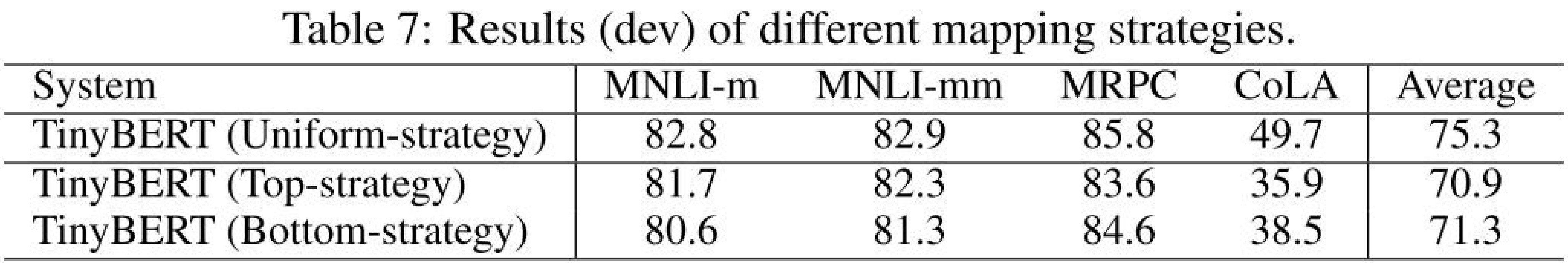


Table 7: Results (dev) of different mapping strategies.

System	MNLI-m	MNLI-mm	MRPC	CoLA	Average
TinyBERT (Uniform-strategy)	82.8	82.9	85.8	49.7	75.3
TinyBERT (Top-strategy)	81.7	82.3	83.6	35.9	70.9
TinyBERT (Bottom-strategy)	80.6	81.3	84.6	38.5	71.3

Uniform-strategy : $g(m) = 3 \times m$
Top-strategy : $g(m) = m+N-M; 0 < m \leq M$
Bottom-strategy : $g(m) = m ; 0 < m < M$

We find that the top-strategy performs better than the bottom-strategy in MNLI, while being worse in MRPC and CoLA tasks, which confirms the observations that different tasks depend on the different kinds of knowledge from BERT layers.