Dynamic-TinyBERT: Boost TinyBERT's Inference Efficiency by Dynamic Sequence Length

Shira Guskin (Intel Labs), Moshe Wasserblat (Intel Labs),
Ke Ding (Intel), Gyuwan Kim (UCSB)

Transformers: Powerful but Inefficient

Many techniques have been developed for improving transformers efficiency without compromising on performance. Among them are:

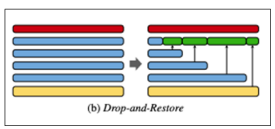
- Distillation (TinyBERT[1])
- Length Adaptive Transformer (LAT[2])

We generate **Dynamic-TinyBERT** by combining both methods and adapt to any given computational budget.

LAT: Best Performance per any Budget

Drop-and-Restore

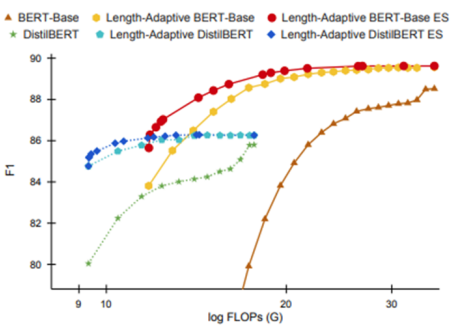
Tokens are being dropped at each layer according to a given sequence of lengths, then being restored in the last hidden layer, extending the applicability into token-level classification.



https://arxiv.org/pdf/2010.07003.pdf

Evolutionary Search

Generate a pareto-curve of accuracy-efficiency tradeoff to get the best length-configuration to be used per any computational budget.



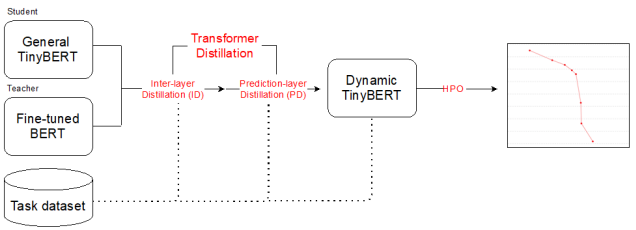
Results of training BERT and DistillBERT models with the LAT method

https://arxiv.org/pdf/2010.07003.pdf

TinyBERT: 6 Layers, 67M parameters

- Trained by Transformer Distillation- learn the knowledge resides in BERT's attention matrices and hidden states
- Runs x2 times faster with <1% accuracy loss

Dynamic-TinyBERT



We implement the Drop-and-Restore method into TinyBERT and train it similarly to the original TinyBERT for a larger number of epochs.

SigOpt - Hyperparameter Optimization (HPO)

- Generate a pareto-curve using Hyperparameter Optimization (HPO) over length-configurations
- Use SigOpt HPO for better search efficiency with a much smaller budget comparing to the evolutionary-search method

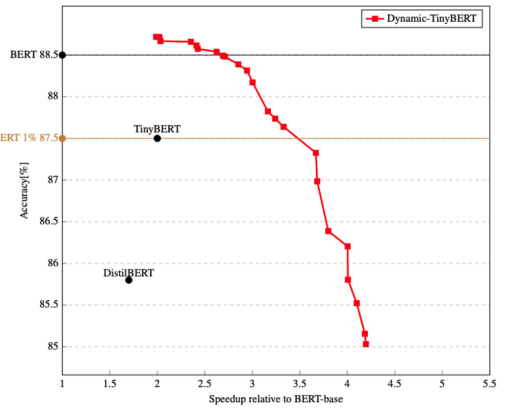
Results

Table 1: Models performance analysis

Model	Max F1 (full model)	Best Speedup within BERT-1%
BERT-base	88.5	1x
DistilBERT	85.8	-
TinyBERT	87.5	2x
Dynamic-TinyBERT	**88.71**	3.3x

Accuracy-efficiency tradeoff results of Dynamic-TinyBERT on SQuAD1.1

x3.3 speedup on CPU, <1% accuracy loss



- 2.7x faster than BERT-base with no accuracy loss
- With Drop-and-Restore: 3.3x faster with <1% accuracy loss
- Popular DistilBERT (67M): 1.7x speedup, 2.7% accuracy loss

Summary

- Dynamic-TinyBERT: effective approach to boost transformers speedup
- Other methods like Sparsity and Low-bit Quantization can be user to further improve the performance

References

[1] X. Jiao, Y. Yin, L. Shang, X. Jiang, X. Chen, L. Li, F. Wang, and Q. Liu (2020): Tinybert: Distilling bert for natural language understanding. In: *ArXiv*, abs/1909.10351, 2020.

[2] G. Kim and K. Cho. Length-adaptive transformer (2021): Train once with length drop, use anytime with search. In: roceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 6501–6511, Online, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.508. URL https://aclanthology.org/2021.acl-long.508.

Paper Code Model