Introduction:

Deep learning for Natural Language Processing (NLP) has undergone explosive growth in the past 12-24 months, with massive increases in the performance of state-of-the-art models, the size of the datasets required to train these models, and the number of parameters used to obtain top performance. The spread of Transformer-based architectures such as BERT [1], RoBERTa [2], ALBERT, MT-DNN, XLNet, and GPT-3 [3] has been called "NLP's ImageNet moment" in analogy to the performance leaps in computer vision earlier this decade.

Overview:

A major challenge in deploying transformer models is their prohibitive inference cost, which quadratically scales with the input sequence length. This makes it especially difficult to use transformers for processing long sequences. Both sparse weights and feature maps and low bit widths can accelerate inference performance. Quantization and pruning are two promising techniques that can be applied to make Transformers more efficient. Specifically, we investigate this problem from the angle of token-pruning, a structural pruning algorithm that reduces redundant tokens as the data passes through the different layers of the transformer. We focused on BERT for now and set RoBERTa and the GPT family of models as the future target models.

Literature Review:

Pruning: Pruning for Transformers involves reducing the sparsity of the weight matrix (e.g. magnitude pruning [6], head pruning [7]) and token-pruning (e.g. cascade pruning [8]). Magnitude pruning is an unstructured pruning method which will prune the weight value close to zero then retrain the subnetwork. Both head pruning and cascade pruning are structured pruning methods that will provide actual speedup against the original model. Cascade pruning also explored token sparsity, head sparsity, and quantization opportunities jointly for speeding up the attention computation in Transformers. However, the token pruning in cascade pruning relies mainly on the inefficient top-k algorithm, which hinders the inference speed of their methods.

Quantization + Pruning: Existing work that explores the combination of pruning and quantization is also available in Computer Vision like APQ [9] and OPQ [10]. In APQ, they optimize neural architecture, pruning policy, and quantization policy jointly by training a quantization-aware accuracy predictor to quickly get the accuracy of the quantized model and feed it to the search engine to select the best fit. OPQ analytically solves the compression allocation with pre-trained weight parameters only. During fine-tuning, the compression module is fixed and only weight parameters are updated so the compression is one-shot.

Methods and current results:

We will submit our work Learned Token Pruning for Transformers [11] to the incoming conference, where we propose the novel Learned Token Pruning (LTP) method that reduces redundant tokens as the data passes through the different layers of the transformer.

Table I: *Detailed performance and efficiency comparison of* LTP *applied to RoBERTa$_{base}$*

	Model	MNLI-m	MNLI-mm	QQP	QNLI	SST-2	STS-B	MRPC	RTE
Accuracy	RoBERTa$_{base}$	87.53	87.36	90.39	92.86	94.27	90.89	92.14	77.98
	LTP	86.53	86.37	89.69	91.98	93.46	90.03	91.59	77.98
GFLOPs	RoBERTa$_{base}$	6.83	7.15	5.31	8.94	4.45	5.53	9.33	11.38
	LTP	3.64	3.63	2.53	4.77	2.13	2.84	4.44	6.30
Speedup	LTP	1.88×	1.97×	2.10×	1.87×	2.09×	1.95×	2.10×	1.81×

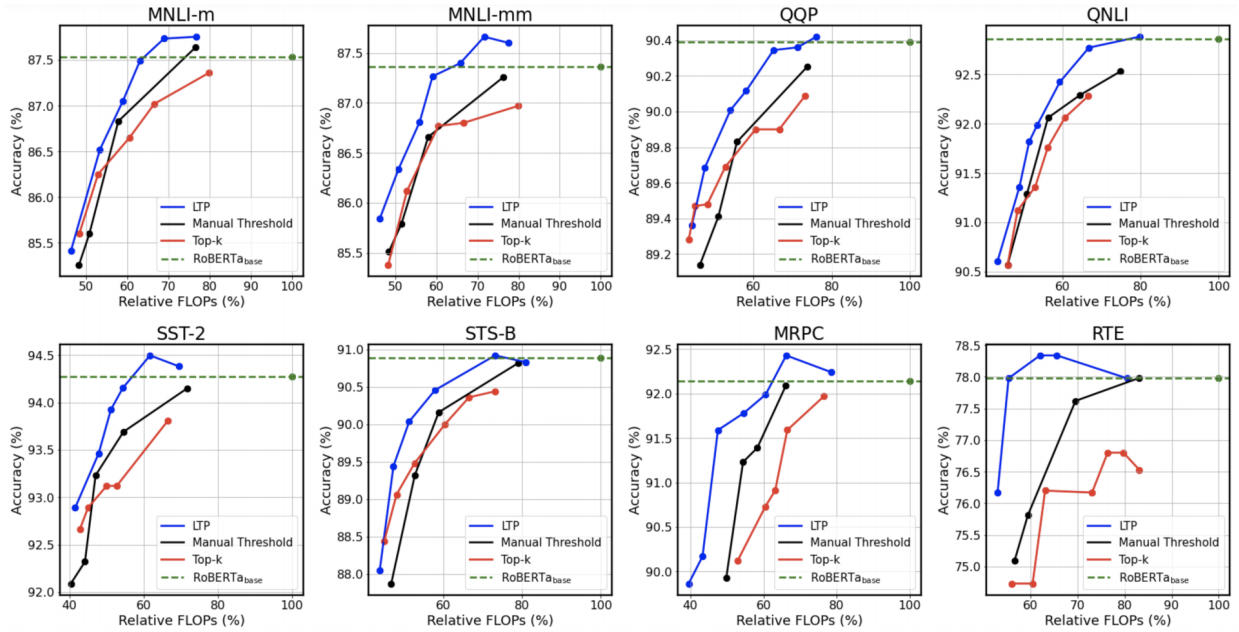

Fig. 5: *Performance of different pruning methods on GLUE tasks for different token pruning methods across different relative FLOPs, i.e., normalized FLOPs with respect to the the baseline model. Manual threshold assigns linearly raising threshold values for each layer instead of learning them. The performance of the baseline model without token pruning is dotted in a horizontal line for comparison.*

In particular, LTP prunes tokens with an attention score below a threshold value, which is learned during training. Importantly, our threshold based method avoids algorithmically expensive operations such as top-k token selection which are used in prior token pruning methods, and also leads to structured pruning. We extensively test the performance of our approach on multiple GLUE tasks and show that our learned threshold based method consistently outperforms the prior state-of-the-art top-k token based method by up to ~2% higher accuracy with the same amount of FLOPs. Furthermore, our preliminary results show up to 1.4x and 1.9x throughput improvement on Tesla T4 GPU and Intel Haswell CPU, respectively, with less than 1% of accuracy drop (and up to 2.1x FLOPs reduction). The main results are also attached here, where LTP can reduce the GFLOPs by around 2x within 1% performance degradation.

Future works:

We will continue to explore the combination effect of pruning (LTP [11]) and quantization (Q-BERT [4] and I-BERT [5]) for Transformer models using more techniques gathered from previous literature and our own work in more general NLU tasks.

Reference:

[1] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019, June). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. ACL 2019.

[2] Liu, Yinhan, 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 preprint arXiv:1907.11692 (2019).

[3] Brown, Tom B., Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan et al. "Language models are few-shot learners." arXiv preprint arXiv:2005.14165 (2020).

[4] Sheng Shen, Zhen Dong, Jiayu Ye, Linjian Ma, Zhewei Yao, Amir Gholami, Michael W Mahoney, and Kurt Keutzer. QBERT: Hessian based ultra low precision quantization of bert. AAAI 2020.

[5] Kim, Sehoon, Amir Gholami, Zhewei Yao, Michael W. Mahoney, and Kurt Keutzer. "I-BERT: Integer-only BERT Quantization." arXiv preprint arXiv:2101.01321 (2021).

[6] Gordon, Mitchell and Duh, Kevin and Andrews, Nicholas. "Compressing BERT: Studying the Effects of Weight Pruning on Transfer Learning." ACL. 2020.

[7] Michel, Paul, Omer Levy, and Graham Neubig. "Are sixteen heads really better than one?." NeurIPS 2019.

[8] Wang, Hanrui, Zhekai Zhang, and Song Han. "SpAtten: Efficient Sparse Attention Architecture with Cascade Token and Head Pruning." HPCA 2021.

[9] Wang, Tianzhe, Kuan Wang, Han Cai, Ji Lin, Zhijian Liu, Hanrui Wang, Yujun Lin, and Song Han. "Apq: Joint search for network architecture, pruning and quantization policy." CVPR. 2020.

[10] Hu, P., Xi Peng, Hongyuan Zhu, M. Aly and J. Lin. “OPQ: Compressing Deep Neural Networks with One-shot Pruning-Quantization.” AAAI 2021.

[11] Kim, Sehoon, Sheng Shen, David Thorsley, Amir Gholami, Joseph Hassoun, and Kurt Keutzer. "Learned Token Pruning for Transformers." arXiv preprint arXiv:2107.00910 (2021).