ProTran: Profiling the Energy of Transformers on Embedded Platforms

Shikhar Tuli, *Student Member, IEEE,* and Niraj K. Jha, *Fellow, IEEE*

***Abstract*—The abstract goes here.**

***Index Terms*—Embedded platforms, machine learning, transformers.**

I. Introduction

In recent years, self-attention-based transformer models [1, 2] have achieved state-of-the-art results on tasks that span the natural language processing (NLP), and recently, even the computer vision (CV) domain [3]. Increasing computational power and large-scale pre-training datasets has resulted in an explosion in the architecture sizes [4], much beyond the state-of-the-art CNNs. For instance, the Megatron Tuning-NLG [4] has 530B trainable parameters compared to only 928M trainable model parameters in BiT (which uses ResNet-152 with every hidden layer widened by a factor of four, i.e., ResNet-152x4) [5, 6]. However, such massive transformer architectures are not amenable to be run on mobile edge devices, due to a much lower compute budget and memory size.

Even if such a large model is run on these mobile devices, it would incur an extremely high latency. Smaller models may have reasonable latencies, however, they may still not meet the energy or power budget for running on edge devices. This could be due to a limited battery size or an intermittent power supply. Thus, there is a need for profiling and benchmarking the latency, energy and peak power consumption of a diverse set of mobile-friendly transformer architectures. This would aid frameworks to leverage hardware-aware neural architecture search (NAS) [7, 8] techniques to find the optimal architecture that maximizes model accuracy while meeting latency, energy and peal power budgets.

Several works have tried to prune transformer models to reduce the number of parameters [9, 10]. Some have also proposed novel attention mechanisms to reduce the number of trainable parameters [11, 12, 13]. Others have run NAS in a design space of transformer architectural hyperparameters to obtain efficient architectures [7, 14, 15]. However, most of these works have only shown gains in the number of model parameters or the number of floating-point operations per second (FLOPs). Such works do not consider latency, energy and power consumption in their optimization loop, while searching for the optimal transformer architecture (Wang et. al [7] only consider latency for running around 2000 transformer architectures on certain edge devices, and Li et. al [16] consider only a single FPGA). Thus, there is a need to profile not only the accuracy [8], but also the latency, energy and peak power consumption of a diverse set of architectures on various mobile devices for inclusive design in edge-AI.

Nevertheless, profiling all models in vast design spaces is a challenging endeavor. Hence, in this work, we make the following contributions:

- We implement the FlexiBERT benchmarking framework with its design space of diverse transformer architectures on multiple edge-AI devices, for both training and inference. We measure the latency, energy consumption and peak power draw for the transformer models in the design space. We call this profiling framework that can obtain all hardware measures for a design space of transformer architectures as ProTran.
- We leverage ProTran to train a surrogate model that exploits gradient-based optimization using backpropogation of inputs and heteroscedastic modelling [8, 17] to minimize the overall uncertainty in estimation of each measure in our active learning loop.
- We then leverage the surrogate models along with previously proposed performance predictors [8] to run hardware-aware NAS that gives equivalent performing models in terms of accuracy, but with much lower latency, energy consumption and peak power draw.

The rest of the article is organized as follows. Section II discusses the background and related work in hardware-aware NAS, pruning methods and profiling of transformer models. Secion III motivates the need for the ProTran framework using a toy example in our design space.

II. Background and Related Work

- A section on background and related work in both hardware-aware NAS and pruning methods. Every section highlights the drawbacks of that work and how our paper suggests improvements.

A. Transformer Architectures

- Mention popular transformer architectures proposed in the literature.
- Hint towards a design space of transformer architectures. Show previous works on design space generation.
- Show how transformers are getting larger-and-larger. Resulting problems in memory footprint and energy consumption on edge devices. Taking motivation from HAT,

This work was supported by NSF Grant No. —. S. Tuli and N. K. Jha are with the Department of Electrical and Computer Engineering, Princeton University, Princeton, NJ, 08544, USA (e-mail: {stuli, jha}@princeton.edu).
Manuscript received —; revised —.

show how activations can be reduced and network can be 8-bit quantized, etc.

B. Hardware-Aware Neural Architecture Search

- Show traditional NAS works in CNNs. How these works have shown model reduction and improvements in energy/latency measures. Cite TCAD.
- Show transformer-specific NAS works. Cite FlexiBERT, HAT. Previous works do not target simultaneous latency/energy/power consumption for optimization.
- Need for a benchmarking platform that gives model accuracy/latency/energy/power on diverse edge platforms. Show how FlexiBERT can be leveraged.

C. Energy Profiling of ML Models

- Show previous works for CNN design spaces. Cite ChamNet. Show co-design approaches like MCUNet that use a range of off-the-shelf micro-controllers.
- Show how FTRANS profiles energy for FPGA platforms, but HAT, FlexiBERT, etc. do not. Need for surrogate models (trained using active learning) on diverse embedded platforms for a design space of transformer architectures. Will aid co-design of transformer architectures in edge applications.

III. Motivation

A. Energy Reduction on Mobile Platforms

- Show reduction in energy consumption of running inference of an off-the-shelf transformer architecture on a mobile device (*e.g.*, an iPhone), compared to CPU and GPU energy consumption.

B. Challenges and Proposed Solutions

- Show challenges in trying to fit large models. Where are the memory bottlenecks? Show latency increase due to gradient accumulation. Show advantages of quantization and reduced activation models.

References

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 5998–6008. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, vol. 1, 2019, pp. 4171–4186.

[3] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proceedings of the International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=YicbFdNTTy

[4] S. Smith, M. Patwary, B. Norick, P. LeGresley, S. Rajbhandari, J. Casper, Z. Liu, S. Prabhumoye, G. Zerveas, V. Korthikanti, E. Zheng, R. Child, R. Y. Aminabadi, J. Bernauer, X. Song, M. Shoeybi, Y. He, M. Houston, S. Tiwary, and B. Catanzaro, "Using deepspeed and megatron to train megatron-turing NLG 530b, A large-scale generative language model," *CoRR*, vol. abs/2201.11990, 2022. [Online]. Available: https://arxiv.org/abs/2201.11990

[5] A. Kolesnikov, L. Beyer, X. Zhai, J. Puigcerver, J. Yung, S. Gelly, and N. Houlsby, "Big Transfer (BiT): General visual representation learning," in *European Conference on Computer Vision*, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds., 2020, pp. 491–507.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[7] H. Wang, Z. Wu, Z. Liu, H. Cai, L. Zhu, C. Gan, and S. Han, "HAT: Hardware-aware transformers for efficient natural language processing," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Jul. 2020, pp. 7675–7688. [Online]. Available: https://aclanthology.org/2020.acl-main.686

[8] Anonymous, "FlexiBERT: Expanding the flexibility of transformer architectures and exploring a heterogeneous design space," *International Conference on Machine Learning*, 2022, in review.

[9] M. A. Gordon, K. Duh, and N. Andrews, "Compressing BERT: studying the effects of weight pruning on transfer learning," *CoRR*, vol. abs/2002.08307, 2020. [Online]. Available: https://arxiv.org/abs/2002.08307

[10] Z. Yan, H. Wang, D. Guo, and S. Han, "Micronet for efficient language modeling," in *Proceedings of the NeurIPS 2019 Competition and Demonstration Track*, ser. Proceedings of Machine Learning Research, H. J. Escalante and R. Hadsell, Eds., vol. 123. PMLR, 08–14 Dec 2020, pp. 215–231. [Online]. Available: https://proceedings.mlr.press/v123/yan20a.html

[11] J. Lee-Thorp, J. Ainslie, I. Eckstein, and S. Ontañón, "FNet: Mixing tokens with Fourier transforms," *CoRR*, vol. abs/2105.03824, 2021. [Online]. Available: https://arxiv.org/abs/2105.03824

[12] S. Wang, B. Z. Li, M. Khabsa, H. Fang, and H. Ma, "Linformer: Self-attention with linear complexity," *CoRR*, vol. abs/2006.04768, 2020. [Online]. Available: https://arxiv.org/abs/2006.04768

[13] F. Iandola, A. Shaw, R. Krishna, and K. Keutzer, "SqueezeBERT: What can computer vision teach NLP about efficient neural networks?" in *Proceedings of SustaiNLP: Workshop on Simple and Efficient Natural Language Processing*. Online: Association for Computational Linguistics, Nov. 2020, pp. 124–135. [Online]. Available: https://aclanthology.org/2020.sustainlp-1.17

[14] J. Xu, X. Tan, R. Luo, K. Song, J. Li, T. Qin, and T.-Y. Liu, "NAS-BERT: Task-agnostic and adaptive-size BERT compression with neural architecture search," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery Data Mining*, 2021, p. 1933–1943.

[15] Y. Yin, C. Chen, L. Shang, X. Jiang, X. Chen, and Q. Liu, "AutoTinyBERT: Automatic hyper-parameter optimization for efficient pre-trained language models," in *Proceedings 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)*. Online: Association for Computational Linguistics, Aug. 2021, pp. 5146–5157. [Online]. Available: https://aclanthology.org/2021.acl-long.400

[16] B. Li, S. Pandey, H. Fang, Y. Lyv, J. Li, J. Chen, M. Xie, L. Wan, H. Liu, and C. Ding, "FTRANS: Energy-efficient acceleration of transformers using fpga," in *Proceedings of the ACM/IEEE International Symposium on Low Power Electronics and Design*, ser. ISLPED '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 175–180. [Online]. Available: https://doi.org/10.1145/3370748.3406567

[17] S. Tuli, S. R. Poojara, S. N. Srirama, G. Casale, and N. R. Jennings, "COSCO: Container orchestration using co-simulation and gradient based optimization for fog computing environments," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 1, pp. 101–116, 2021.