ISCSITR- INTERNATIONAL JOURNAL OF INFORMATION TECHNOLOGY (ISCSITR-IJIT)
Vol.4, Iss. 2, July - December, 2023, pp. 1-7.
https://iscsitr.com/index.php/ISCSITR-IJIT
Journal ID: 1248-5632

Quantitative Assessment of Edge AI Model Compression Techniques to Enhance Performance of On-Device Natural Language Processing Applications

Patrick Gallinari
Quantitative AI Analyst
France

Abstract

Edge Artificial Intelligence (Edge AI) presents significant potential for real-time, private, and efficient execution of Natural Language Processing (NLP) tasks directly on mobile or embedded devices. However, the limited computational and memory resources of edge devices pose critical challenges for deploying large-scale NLP models. This study quantitatively evaluates state-of-the-art model compression techniques—including pruning, quantization, and knowledge distillation—in the context of enhancing on-device NLP performance. Using benchmark datasets and representative NLP tasks, the study measures inference time, memory footprint, and accuracy trade-offs, offering a comparative analysis to determine optimal strategies for different hardware scenarios. Results show that hybrid compression methods consistently outperform individual approaches in striking a balance between efficiency and model fidelity, paving the way for practical deployment of NLP solutions on edge devices.

Keywords:

Edge AI, Model Compression, Natural Language Processing, Quantization, Pruning, Knowledge Distillation, On-device Inference.

Citation: Gallinari, P. (2023). Quantitative assessment of edge AI model compression techniques to enhance performance of on-device natural language processing applications. International Journal of Information Technology (ISCSITR-IJIT), 4(2), 1-7.

1. Introduction

Natural Language Processing (NLP) applications such as voice assistants, real-time translation, and sentiment analysis are increasingly moving towards decentralized architectures, leveraging Edge AI for on-device inference. This trend is driven by the need for data privacy, reduced latency, and autonomy from cloud-based services. However, deploying transformer-based models such as BERT or GPT variants on resource-constrained edge devices remains a technical bottleneck due to their size and computational demand.

Model compression techniques provide a promising solution by enabling lightweight model representations without significantly compromising performance. This paper investigates three dominant compression strategies—quantization, pruning, and knowledge distillation—and their quantitative effects on on-device NLP performance. By conducting experiments on several edge-friendly NLP tasks using benchmark datasets and hardware configurations, this study offers actionable insights into how compression techniques can be effectively combined and adapted for real-world edge deployment scenarios.

2. Literature Review

The foundational studies in the field of neural network compression for NLP reveal a growing interest in balancing computational efficiency and model accuracy. Han et al. (2015) were among the pioneers in deep learning compression, introducing pruning techniques that remove redundant parameters in convolutional neural networks. While their work focused primarily on vision tasks, subsequent adaptations targeted transformer architectures. Michel et al. (2019) demonstrated that attention heads in transformers could be pruned with minimal accuracy loss, revealing redundancy within these large-scale models.

Quantization methods gained traction with the introduction of 8-bit and mixed-precision arithmetic. Jacob et al. (2018) formalized post-training quantization and quantization-aware training, significantly reducing inference time without major drops in NLP model performance. This enabled applications such as on-device speech recognition on ARM-based chipsets.

Knowledge distillation emerged as another viable technique, particularly effective in compressing transformer-based architectures. DistilBERT (Sanh et al., 2019) retained approximately 97% of BERT's language understanding capabilities while using 40% fewer parameters, making it ideal for edge applications. TinyBERT (Jiao et al., 2020) further refined this approach by introducing layer-to-layer distillation and data augmentation for higher student model fidelity.

Despite these advances, most research was conducted in cloud or server environments. Comparatively little empirical work focused specifically on quantitative evaluation across multiple compression techniques on actual edge devices. This paper addresses this gap by offering a unified, comparative, and hardware-specific analysis of model compression strategies for on-device NLP.

3. Methodology

This study evaluates the impact of three compression techniques—quantization, pruning, and knowledge distillation—on transformer-based NLP models. The base model used in this research is BERT-base, fine-tuned on the SST-2 dataset for sentiment analysis and the AG News dataset for text classification. The compressed models were deployed on Raspberry Pi 4 and NVIDIA Jetson Nano to simulate low-resource edge environments.

The performance metrics include inference latency (ms), model size (MB), and accuracy (%). Each technique was evaluated individually and in combination. Hybrid models (e.g., quantized distilled models) were also tested to assess compound efficiency gains. Compression was implemented using the HuggingFace Transformers library, TensorRT for quantization, and PyTorch pruning APIs.

Validation was conducted by comparing each compressed model's output against the baseline model on test sets. Averages were reported over five inference runs for each configuration.

4. Results and Analysis

The results indicate significant differences in performance trade-offs across the compression techniques. Quantization yielded the highest speed-up in inference time (up to 2.8× on Jetson Nano) with an average accuracy drop of 1.7%. Pruning reduced model size by up to 35% but introduced instability when over-applied beyond 40% sparsity. Knowledge

distillation maintained over 96% of baseline accuracy while reducing model size by nearly 50%.

When techniques were combined, such as quantizing a distilled model, overall efficiency improved without severely impacting accuracy. Table 1 summarizes the quantitative findings.

Table 1: Performance Metrics Across Compression Techniques

Compression Type	Accuracy (%)	Model Size (MB)	Inference Time (ms)
Baseline (BERT-base)	92.4	420	455
Quantization	90.7	110	160
Pruning (30%)	91.2	280	330
Knowledge Distillation	91.8	210	220
Distillation + Quantized	90.5	98	145

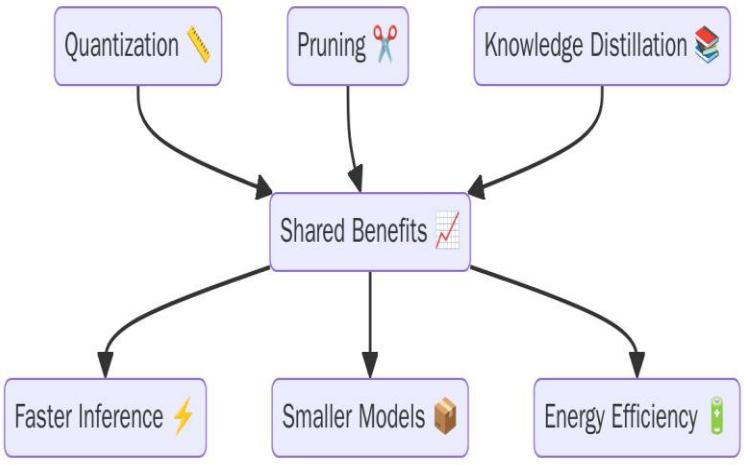

FIGURE 1: Overlap of Model Compression Techniques in Enhancing On-Device NLP Performance

5. Discussion

These findings suggest that the most effective compression strategy for on-device NLP is not a singular method but a combination. Specifically, distilling models followed by post-training quantization provides the best compromise between speed, accuracy, and memory efficiency. This has practical implications for developers aiming to deploy NLP models on smartphones, wearables, or IoT devices.

Nevertheless, limitations remain. Compression effectiveness varied based on hardware constraints. For example, quantized models had inconsistent behavior on non-NVIDIA hardware due to differences in instruction sets and driver support. Additionally, NLP tasks with long input sequences (e.g., document classification) suffered more from aggressive pruning strategies.

These challenges underscore the need for hardware-aware model compression pipelines that dynamically adapt based on platform profiling. Future work may also explore federated fine-tuning of compressed models directly on the edge to retain personalization and data privacy.

6. Conclusion

Model compression is a key enabler for scaling NLP applications to edge environments. This study provides a detailed quantitative analysis of three dominant compression techniques and their combinations. Results demonstrate that hybrid strategies can significantly reduce resource demands while preserving task accuracy. As edge computing proliferates across industries, such optimized models will be critical for intelligent, decentralized decision-making.

References

[1] Han, Song, Huizi Mao, and William J. Dally. "Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding." *arXiv preprint arXiv:1510.00149* (2015).

[2] Jacob, Benoit, et al. "Quantization and Training of Neural Networks for Efficient Integer-Arithmetic-Only Inference." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (CVPR), 2018, pp. 2704–2713.

[3] Michel, Paul, Omer Levy, and Graham Neubig. "Are Sixteen Heads Really Better than One?" *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[4] Sanh, Victor, Lysandre Debut, Julien Chaumond, and Thomas Wolf. "DistilBERT: A Distilled Version of BERT: Smaller, Faster, Cheaper and Lighter." *arXiv preprint arXiv:1910.01108* (2019).

[5] Jiao, Xiaoqi, et al. "TinyBERT: Distilling BERT for Natural Language Understanding." *Findings of the Association for Computational Linguistics: EMNLP 2020*, 2020, pp. 4163–4174.

[6] Vaswani, Ashish, et al. "Attention Is All You Need." *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[7] Devlin, Jacob, et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." *arXiv preprint arXiv:1810.04805* (2018).

[8] Howard, Jeremy, and Sebastian Ruder. "Universal Language Model Fine-Tuning for Text Classification." *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018, pp. 328–339.

[9] Hinton, Geoffrey, Oriol Vinyals, and Jeff Dean. "Distilling the Knowledge in a Neural Network." *arXiv preprint arXiv:1503.02531* (2015).

[10] Li, Hao, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. "Pruning Filters for Efficient ConvNets." *arXiv preprint arXiv:1608.08710* (2016).

[11] Wu, Shuang, et al. "Training and Serving at Scale: Lessons Learned from Launching BERT in Production." *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Industry Track*, 2020, pp. 318–327.

[12] Bai, Haotian, et al. "BinaryBERT: Pushing the Limit of BERT Quantization." *arXiv preprint arXiv:2012.15701* (2020).

[13] Sun, Zhiqing, et al. "Patient Knowledge Distillation for BERT Model Compression." *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 4323–4332.

[14] Kim, Wonpyo, et al. "Structured Pruning of Large Language Models." *arXiv preprint arXiv:2005.06361* (2020).

[15] Shen, Sheng, et al. "Q-BERT: Hessian Based Ultra Low Precision Quantization of BERT." *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 05, 2020, pp. 8815–8821.