Deep Model Compression

As deep learning models continue to advance in complexity, they increasingly demand extensive computational resources and storage capacity. Conversely, the rise in resource-constrained devices such as mobile and edge devices has amplified the need for deploying deep learning models on these devices due to the advantages they offer, such as saving bandwidth, preserving privacy, reducing latency, and being cost-effective. However, deploying deep learning models on edge devices presents challenges due to limitations in power, computational resources, and memory capacity.

To tackle these challenges, Model Compression [1] techniques have emerged with the goal of reducing the size of deep learning models while maintaining their performance. By compressing models, we can enable more efficient deployment on resource-constrained devices like mobile phones and edge devices. In this project, our primary objective is to investigate various model compression techniques, including: • Pruning [2]: Removing unnecessary connections or parameters from the model to reduce its overall size. • Quantization [3]: Decreasing the precision of weights and activations to lower memory and computation requirements. • Weight Sharing [4]: Leveraging shared weights among network components to further reduce the model's size. • Knowledge Distillation [5]: Transferring knowledge from a larger model to a smaller model. • Low-Rank Approximation: Approximating weight matrices with lower-rank matrices to reduce computational complexity.

By employing these techniques, our aim is to strike a balance between model size and accuracy, thereby optimizing the performance of deep learning models for deployment in resource-limited environments. Accomplishing this will unlock the potential of deep learning models for a wider range of real-world applications on edge devices, where computational resources are limited.

Desired (but not mandatory) requirements/skills: • Programming skills in Python, • Familiarity with either the TensorFlow or PyTorch framework. • Knowledge of deep learning concepts. • Familiarity with edge devices like Raspberry Pi and Jetson Nano.

References [1] T. Choudhary, V. Mishra, A. Goswami en J. Sarangapani, „A comprehensive survey on model compression and acceleration," Artificial Intelligence Review, vol. 53, p. 5113–5155, 2020. [2] H. Li, A. Kadav, I. Durdanovic, H. Samet en H. P. Graf, „Pruning filters for efficient convnets," arXiv preprint arXiv:1608.08710, 2016. [3] W. Balzer, M. Takahashi, J. Ohta en K. Kyuma, „Weight quantization in Boltzmann machines," Neural Networks, vol. 4, p. 405–409, 1991. [4] S. Han, H. Mao en W. J. Dally, „Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," arXiv preprint arXiv:1510.00149, 2015. [5] G. Hinton, O. Vinyals en J. Dean, „Distilling the knowledge in a neural network," arXiv preprint arXiv:1503.02531, 2015.

Group	Interconnected Resource-aware Intelligent Systems
Contact	Nirvana Meratnia
Keywords	Edge AI, Model Compression, Pruning, Quantization, Knowledge Distillation
Recommended courses	• 2IMC82 ES - Embedded Software