Improving accuracy of neural networks compressed using fixed structures via doping

Urmish Thakker, Ganesh Dasika, Paul Whatmough, Matthew Mattina, Jesse Beu
Arm ML Research Lab

A matrix with pre-defined structure (e.g. circular, Kronecker) can be expressed using far fewer parameters than the equivalent sized unstructured matrix leading to almost 50x-100x compression [1-6,8,9-12], albeit at an accuracy cost. A pre-defined structure also assists in developing a fast inference run-time library for CPU/GPU, or a power/area efficient neural network (NN) accelerator [3,8,11]. Recent theoretical results indicate that NNs expressed in this way also have a function approximation property similar to a regular neural network [2,7]. These factors have led to a flurry of recent work that has shown the potential of structured matrices to compress RNNs and FC layers with minor loss in accuracy. Building on this prior work, we push the compression achieved by pre-defined structure-based matrices significantly further (>100x). To do this, we express a weight matrix using a *Kronecker* product of two smaller matrices. However, this comes at a cost of accuracy. In order to recover this accuracy, we propose a novel method to allow certain parameters of the structured matrix additional degrees of freedom. We call this method doping. By providing additional degrees of freedom during training, we are able recover the lost accuracy. Formally, doping is the process of adding an extremely sparse overlay matrix on top of the pre-defined structure. Thus, we replace the matrices in a NN with the sum of a Kronecker product-based matrix and an extremely sparse matrix. We train this model end-to-end to allow back-propagation to determine what parameters of this structured matrix require additional degrees of freedom. We start by adding a dense matrix to a structured matrix. As training progresses, we prune parameters from the dense matrix creating a sparse matrix. The compression factor determines the amount of pruning. We also discover the phenomenon of co-matrix adaption which can limit the impact of doping. To over this adaptation, we propose a new regularization scheme which allows for better accuracy. Our results have been very encouraging. We can compress a large language model with LSTM layers of size 25 MB by 25x with 1.4% loss in perplexity score. At 25x compression, an equivalent pruned network leads to 7.9% loss in perplexity score, while hybrid-matrix decomposition and low-rank matrix factorization lead to 15% and 27% loss in perplexity score respectively.

References

[1] Urmish Thakker, Igor Fedorov, Jesse Beu, Dibakar Gope, Chu Zhou, Ganesh Dasika, and Matthew Mattina. Pushing the limits of RNN Compression. NeurIPS-EMC2 2019.

[2] Anna T. Thomas, Albert Gu, Tri Dao, Atri Rudra, and Christopher Ré. Learning Compressed Transforms with Low Displacement Rank. NeurIPS 2018.

[3] Caiwen Ding, Siyu Liao, Yanzhi Wang, Zhe Li, Ning Liu, Youwei Zhuo, Chao Wang, Xuehai Qian, Yu Bai, Geng Yuan, Xiaolong Ma, Yipeng Zhang, Jian Tang, Qinru Qiu, Xue Lin, and Bo Yuan. CirCNN: Accelerating and Compressing Deep Neural Networks Using Block-Circulant Weight Matrices. MICRO 2017.

[4] Urmish Thakker, Jesse Beu, Dibakar Gope, Ganesh Dasika and Matthew Mattina. Run-Time Efficient RNN Compression for Inference on Edge Devices. ISCA-EMC2 2019.

[5] Vikas Sindhwani, Tara N. Sainath, and Sanjiv Kumar. Structured transforms for small-footprint deep learning. NeurIPS 2015.

[6] Siyu Liao, Zhe Li, Liang Zhao, Qinru Qiu, Yanzhi Wang, and Bo Yuan .CircConv: A Structured Convolution with Low Complexity. AAAI 2019.

[7] Liang Zhao, Siyu Liao, Yanzhi Wang, Zhe Li, Jian Tang, Victor Pan, Bo Yuan. Theoretical Properties for Neural Networks with Weight Matrices of Low Displacement Rank (ICML 2017)
[8] Morteza Hosseini, Mark Horton, Hiren Paneliya, Uttej Kallakuri, Houman Homayoun, and Tinoosh Mohsenin. On the Complexity Reduction of Dense Layers from O(N2) to O(NlogN) with Cyclic Sparsely Connected Layers. DAC 2019.
[9] Marcin Moczulski, Misha Denil, Jeremy Appleyard, and Nando de Freitas. ACDC: A Structured Efficient Linear Layer. ICLR 2016.
[10] An exploration of parameter redundancy in deep networks with circulant projections (ICCV 2015)
[11] PERMDNN: Efficient Compressed DNN Architecture with Permuted Diagonal Matrices (MICRO 2018)
[12] Factorization tricks for LSTM networks (ICLR 2017 Workshop)