Assignment 1: Selfish Sparse RNN Training

Jordi Nadeu Ferran

0.1 What are sparse neural networks, and why are they used in deep learning?

Sparse neural networks are deep learning models in which only a subset of the weights are active, the others are set to zero. They do this to reduce the computational and memory training costs and deploying deep neural networks.

0.2 Explain the difference between dense-to-sparse training and sparse-to-sparse training.

- Dense-to-sparse training involves first training a dense neural network and then pruning unimportant weights to create a sparse network. This approach ensures performance retention but don't reduce the training cost.
- Sparse-to-sparse training starts with a sparse network and maintains sparsity throughout training, avoiding the need for dense training and reducing computational costs.

0.3 What are the main advantages of sparse-to-sparse training over dense-to-sparse training?

- Reduces memory and computational cost during training.
- Avoid the cost of pretraining a dense model.
- Allows continuous weight redistribution, which can do a better model generalization.

0.4 Why have previous sparse-to-sparse methods struggled with Recurrent Neural Networks (RNNs)?

RNNs have long term dependencies, making them sensitive to changing weights. The optimization technique that works well for dense RNNs was incompatible with dense-to-sparse methods. Also the standard approach don't allow to effective weight redistribution across different RNN parts. [1]

0.5 What are the main contributions of the paper to improve sparse training for RNNs?

- Selfish RNN algorithm to train RNNs with fixed parameter from scratch while allowing dynamic weight updates.
- Demonstrate state of the art sparse training performance that is better than the dense-to-sparse methods with various RNN models.

0.6 Which datasets are used in the experiments, and why are they relevant for RNN evaluation?

- Penn TreeBank (PTB) is a large benchmark dataset for evaluating language models which tests the ability of an RNN to capture long range dependencies.
- WikiText 2 is also a larger but more complex language model dataset that allows better evaluation of RNN generalization and performance on real world.

0.7 What do the authors claim about the performance of their method compared to dense-to-sparse methods?

The authors claim that selfish RNN achieves better results than dense-to-sparse methods. Their method outperforms dense-to-sparse techniques such as Gradual Magnitude Pruning (GMP) and Intrinsic Sparse Structures (ISS) by maintaining a fixed sparse structure during the entire training.

0.8 What practical implications could this work have for deploying RNNs in real-world applications?

- Efficient deployment: sparse RNNs reduce memory and computation costs making them more available for everyone.
- Faster training: that helps to leading to reduce energy consumption.
- Improved scalability: allows larger models to be trained with limited resources.

References

[1] Wikipedia contributors: Recurrent Neural Networks. https://en.wikipedia.org/wiki/Recurrent_neural_network