Adaptation of the architecture of the deep learning model with performance control

Savely Borodin

Moscow Institute of Physics and Technology

Course: My first scientific paper

Expert: O. Y. Bakhteev

Consultant: K. D. Yakovlev

2023

Goal of research

Goal

Prune deep learning model with respect to target hardware properties.

Task

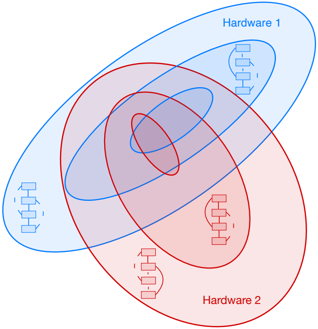
Train hypernetwork to generate pruning parameterization for arbitrary trade-off between accuracy and latency.

Method

Use Gumbel-Softmax trick to relax search space and use latency regularization to train hypernetwork end-to-end with backpropagation through the main model.

Basic idea

The task is to get pruned model, that gives comparable quality with significant speed-up on a particular device.



A loss function for our problem is a trade-off between loss function of original problem and execution time regularization:

$$\mathbb{E}_\lambda \mathbb{E}_\gamma \mathcal{L}_{\mathsf{val}}\big(\gamma(\lambda)\big) + \lambda T^\top \gamma(\lambda)$$

A hypernetwork is a model that generates the parameters of the target deep learning model.

Literature

- Ha, D., Dai, A. and Le, Q.V., 2016. Hypernetworks. arXiv preprint arXiv:1609.09106.
- He, K., Zhang, X., Ren, S. and Sun, J., 2016. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).
- Jang, E., Gu, S. and Poole, B., 2016. Categorical reparameterization with gumbel-softmax. arXiv preprint arXiv:1611.01144.
- Tan, M., Chen, B., Pang, R., Vasudevan, V., Sandler, M., Howard, A. and Le, Q.V., 2019. Mnasnet: Platform-aware neural architecture search for mobile. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 2820-2828).

Problem statement

- Given a model with structure $\Gamma = (V, E)$, where E is a set of atomic operations such as convolution, pooling, activation etc. The task is to find a subset of edges for which the pruned model gives comparable quality with significant speed-up on a particular device.
- Pruned model is parameterised with $\widetilde{\gamma} \in \{0, 1\}^{|E|}$ such that we compute the output of edge e iff $\widetilde{\gamma}_e = 1$.
- Since discrete search space of γ is finite, it is possible that obtained hypernetwork would result in piecewise-constant function. So we use piecewise-constant function as hypernerwok.

Method

We use hypernetwork to generate γ for regularization parameter $\lambda \in \mathbb{R}$, so the generated parameter is denoted as $\gamma(\lambda)$. We want to optimize for arbitrary λ, thus we assume $\lambda \sim P$. Since optimization on discrete space $\{0, 1\}^{|E|}$ is not differentiable we use Gumbel-softmax trick to relax space. Then the search space is replaced with relaxed space $[0; 1]^{|E|}$. Optimization problem:

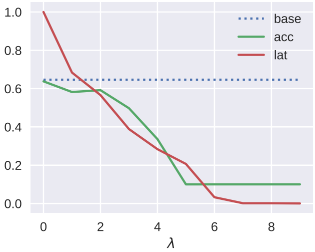
$$\mathbb{E}_\lambda \mathbb{E}_\gamma \mathcal{L}_{\text{valid}}\big(\gamma(\lambda)\big) + \lambda \sum_{e \in E} T(e)\gamma(\lambda)_e \to \min_{\gamma(\lambda)},$$

where $\mathcal{L}_{\text{valid}}(\gamma)$ is a loss function of the original problem for a pruned network on validation dataset, $T(e)$ is the time to execute corresponding atomic operation.

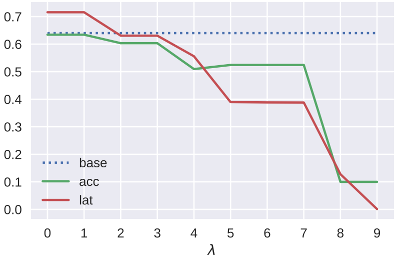
Computational experiment

Before experiment we wrap backbone-model into interpreter that allows us to use Gumbel-Softamx trick for pruning reparameterization. Specifically we use RelaxedBernoulli since there are only two categories: 0 and 1.

In the basic experiment we optimize γ for different λ to approximate Λ.



Then in the main experiment we train hypernetwork using approximated Λ.



AUC comparasion

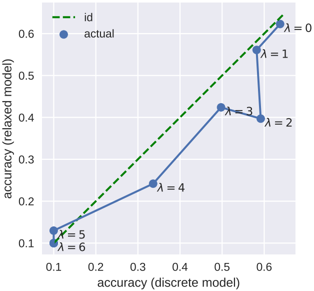
To present value of our method we obtain pruned models by our method and greedy algorithm. Then we measure latency and accuracy of each model and plot accuracy versus latency. To conduct numerical comparison we calculate area under the curve (AUC) for both methods. The results are written in table.

Table: Comparison of AUC for different methods

method	AUC
hypernetwork (ours)	0.342
random	0.065

Comparison of discrete and relaxed model

Since optimization of original task on discrete space is not differentiable, we relax optimization. This approach changes optimization problem. In this experiment we ensure that solution for the relaxed problem is an approximation of a solution for the original problem.



Initially we obtain pruned relaxed and discrete models for different λ. We measure accuracy for each model and plot the results. Complete equivalence between tasks should result in identity curve.

Conclusion

In this paper we have investigated the problem of structural pruning with respect to target hardware properties. To make search differentiable we relaxed the problem with Gumbel-Softmax trick. To solve relaxed problem we have trained piecewise-constant hypernetwork end-to-end for arbitrary trade-off parameter with backpropagation through the backbone-model. We compared accuracy of relax-pruned and discrete-pruned models, to ensure equivalence of relaxed and original problems. Thus achieving better accuracy for the same latency and better latency for the same accuracy than greedy algorithm.