Optimization of the neural network architecture with hardware difficulty control

Firsov Sergey
Intelligent Systems Phystech
firsov.sa@phystech.edu

Oleg Bakhteev
Swiss Data Science Center
bakhteev@phystech.edu

Аннотация

The paper examines the issue of choosing deep learning models. It suggests a method for determining neural network architectures based on their hardware needs and model complexity. Model complexity is measured by the number of parameters in the model, and hardware requirements are based on the overall latency of network operations. This approach is inspired by Differentiable Architecture Search (DARTS), a technique for discovering optimal architectures. Instead of manually adjusting network structural parameters, they are treated as functions of a complexity parameter, with regularization used to control latency. This enables the generation of multiple architectures during one optimization process, allowing the selection of the best architecture.

Ключевые слова neural network, neural architecture search, DARTS, hypernetwork

1 Introduction

Selecting an appropriate architecture for deep learning models is a crucial task that directly impacts model efficiency and performance. With deep learning continuing to push computational limits, researchers face the challenge of finding a balance between model complexity, accuracy, and resource consumption. Recent advances in Neural Architecture Search (NAS) techniques, such as Differentiable Architecture Search (DARTS), seek to automate this process by exploring large search spaces of possible network structures. However, these methods often struggle with high computational requirements and the need for architecture adjustments when model complexity or target hardware changes[1].

One of the significant developments in NAS is the introduction of hardware-aware models. For example, the FBNet method [2] incorporates latency into the architecture search process, optimizing not only for model performance but also for efficiency on specific hardware devices. This approach addresses the mismatch between FLOPs and actual hardware performance, a limitation of many prior NAS methods.

Similarly, ProxylessNAS[3] addresses the issue of high memory consumption and computational cost during the search process by directly optimizing architectures on large-scale tasks and hardware platforms without relying on proxy tasks. This allows for direct specialization of architectures for target tasks, leading to models that are both more efficient and better suited to the given hardware constraints.

Building on these ideas, our work improves upon DARTS-CC[1], a NAS approach that uses hypernetworks to control model complexity during architecture search. Unlike other methods that search for individual architectures at different complexity levels, DARTS-CC generates multiple architectures in a single optimization process. This allows us to select the optimal architecture based on computation budget, further reducing the time and resources needed for NAS. Inspired by FBNet and ProxylessNAS, we use Gumbel-Softmax sampling to control architecture complexity and hardware efficiency, ensuring that the models produced are well-suited for deployment across different environments.

Список литературы

[1] Konstantin Yakovlev, Olga Grebenkova, Oleg Bakhteev, and Vadim Strijov. Neural architecture search with structure complexity control. In *Recent Trends in Analysis of Images, Social Networks and Texts*, pages 207–219, 2022.

[2] Bichen Wu, Xiaoliang Dai, Peizhao Zhang, Yanghan Wang, Fei Sun, Yiming Wu, Yuandong Tian, Peter Vajda, Yangqing Jia, and Kurt Keutzer. Fbnet: Hardware-aware efficient convnet design via differentiable neural architecture search. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[3] Han Cai, Ligeng Zhu, and Song Han. Proxylessnas: Direct neural architecture search on target task and hardware, 2019.