GNAS: A Greedy Neural Architecture Search Method for Multi-Attribute Learning
Siyu Huang\(^1\), Xi Li\(^1\), Zhi-Qi Cheng\(^2\), Zhongfei Zhang\(^1\), Alexander Hauptmann\(^3\)
\(^1\)Zhejiang University, \(^2\)Southwest Jiaotong University, \(^3\)Carnegie Mellon University

**Problem Formulation**

**Goal:** To find the optimal tree-like neural network architecture

Difficult black-box optimization problem
- A huge number of candidate architectures
- Huge evaluation cost
  - Training every architecture until convergence

**Related Work**

AutoML: Towards the automation of machine learning pipelines
- To make ML available for non-ML experts
- To accelerate research on ML

Neural Architecture Search (NAS)
To automate the architecture design of neural networks

**GNAS Framework**

**GNAS Strategy 1:** Global ---\> Layers
- Architectures of the other layers are fixed

**GNAS Strategy 2:** Layer ---\> Connections
- To find the best-1 connection w.r.t each attribute
- Number of candidate architectures within one layer:
  \[ B_{l+1}^{a} = B_{l} \cdot B_{l+1} \]

**GNAS Strategy 3:** Evaluate connections in together
- Number of candidate architectures within one layer:
  \[ B_{l}^{a} \]

**GNAS Strategy 4:** Neural weight sharing \([\text{ENAS, ICML’18}]\)
- Training \(W[A]\) on training set
- Evaluating \(W[A]\)

**Performance of GNAS**

Facial Attribute
- Method: Attribute \(1\) vs. \(2\)
- Person Attribute: Attribute \(1\) vs. \(2\)
- Per-attribute Performance

Related attributes grouped hierarchically

**Efficiency of GNAS**

Training cost: 1 GPU * 1 day on LFWA (6k images), Market-1501 (17k images)
1 GPU * 2 days on CelebA (180k images)

GNAS vs. Random Search
- Results
  1) GNAS has better performance and faster convergence speed
  2) Better with larger validation batch

**References**


**Contact**

Siyu Huang
siyuhuang@zju.edu.cn
siyuhuang.github.io