The 22nd IEEE Mobile Ad-Hoc and Smart Systems (MASS 2025)

HOHAI UNIVERSITY

Efficient Zero-Cost Neural Architecture Search for Personalized AI Systems in Cloud-Edge Networks

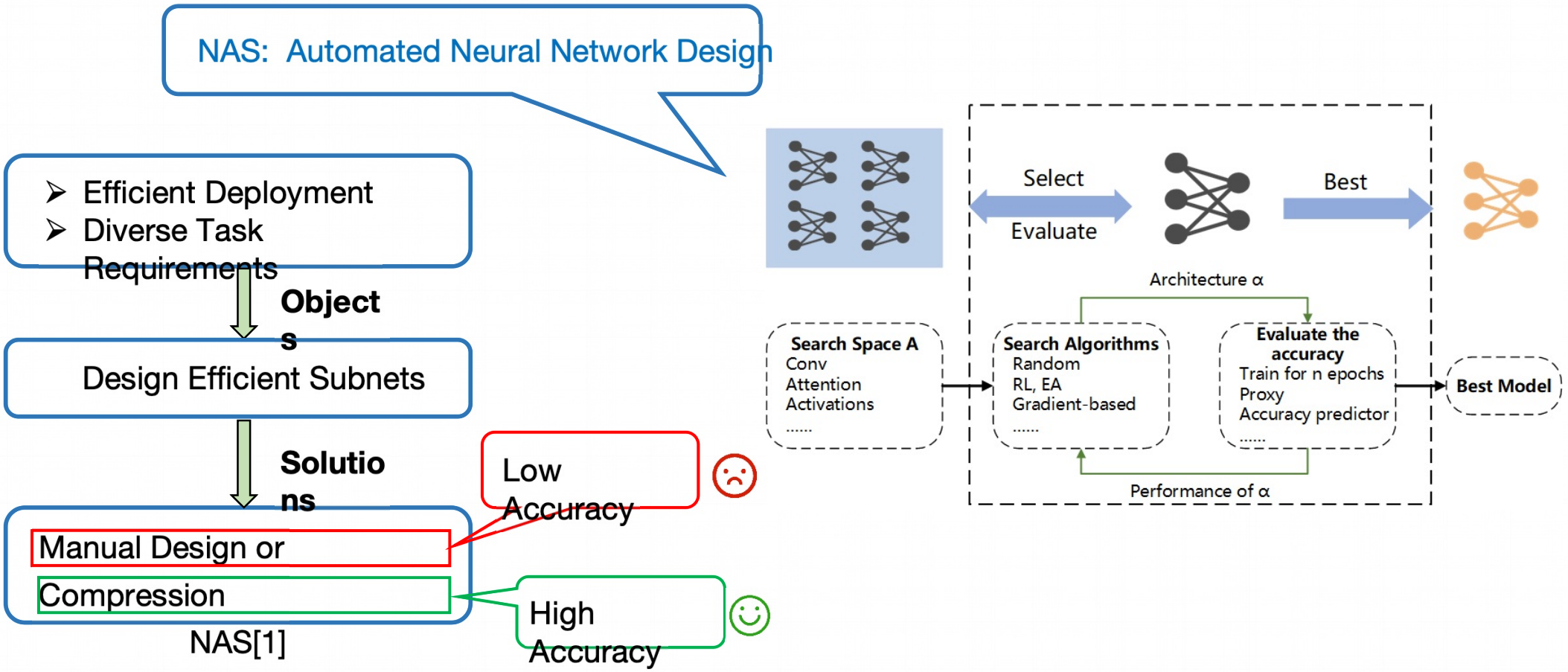
Kai Huang[a], Yingchi Mao[a], Benteng Zhang[a], Yihan Chen[a], Yuchu Chen[a], Jie Wu[b]

[a] College of Computer Science and Software Engineering, Hohai University, Nanjing, China

[b] Center for Networked Computing, Temple University, Philadelphia, USA

October 6, 2025

1.1 What is NAS (Neural Architecture Search)?



[1] B. Zoph and Q. V. Le, "Neural Architecture Search with Reinforcement Learning," in *Proc. Int. Conf. Learn. Represent (ICLR),* 2017, p. 2.

1.2 The Challenges of NAS

RL(Reinforcement Learning)
EA(Evolutionary Algorithm)

- Huge Search Space
- SubNets Training

Cause

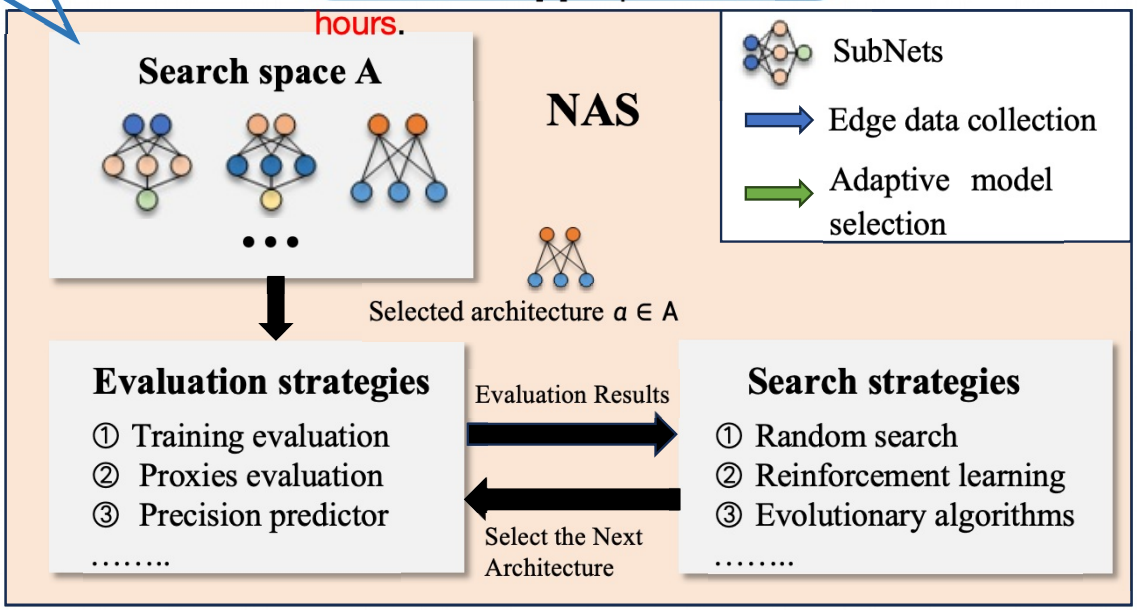
- High Computational Power
- Long Search Time

RL-based NAS[1] requires 2000 days.
EA-based NAS[2] requires 3000 hours.

Solution

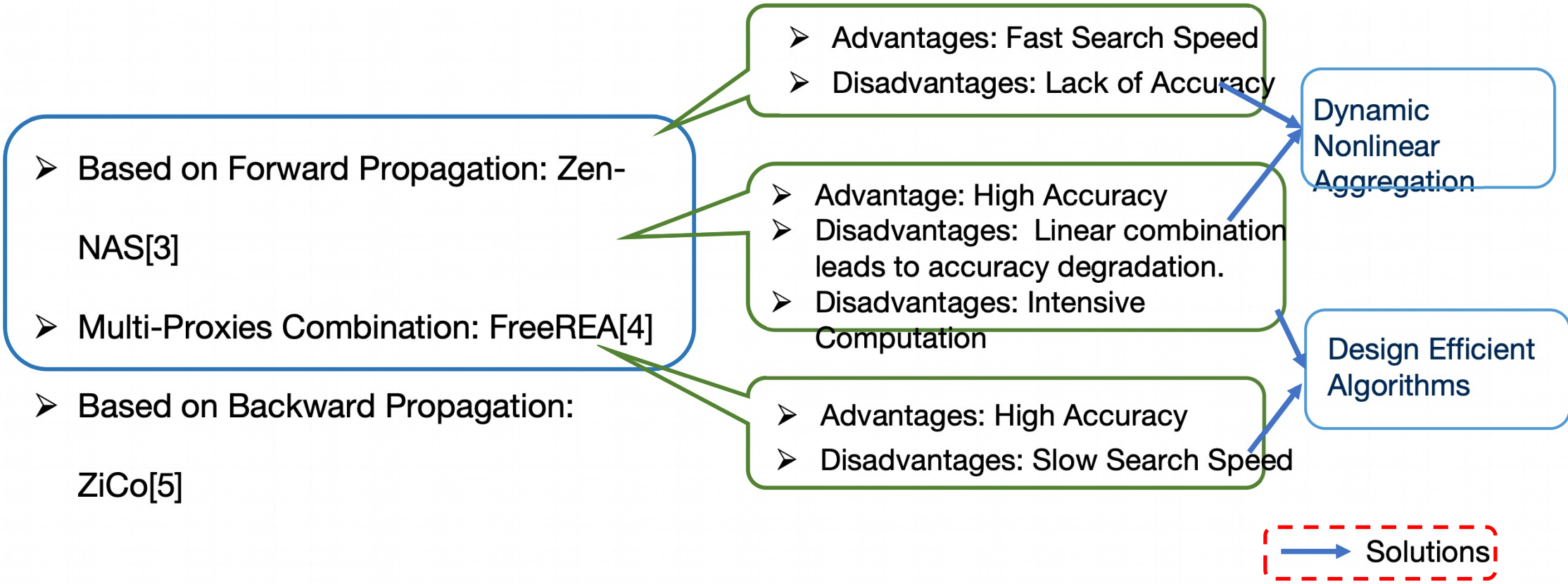
Zero-cost Proxy NAS

Design Train-Free Proxy Based on Features or Gradients



[1] B. Zoph and Q. V. Le, "Neural Architecture Search with Reinforcement Learning," in *Proc. Int. Conf. Learn. Represent (ICLR),* 2017, p. 2.
[2] E. Real et al., "Large-Scale Evolution of Image Classifiers," in *Proceedings of the 34th International Conference on Machine Learning, PMLR,* July 2017, pp. 2902–2911.

1.3 The Challenges of Current Zero-cost Proxy NAS



[3] Lin M, Wang P, Sun Z, et al. “Zen-Nas: A Zero-Shot Nas for High-Performance Image Recognition,”in *International Conference on Computer Vision(ICCV)*. 2021: 347-356.

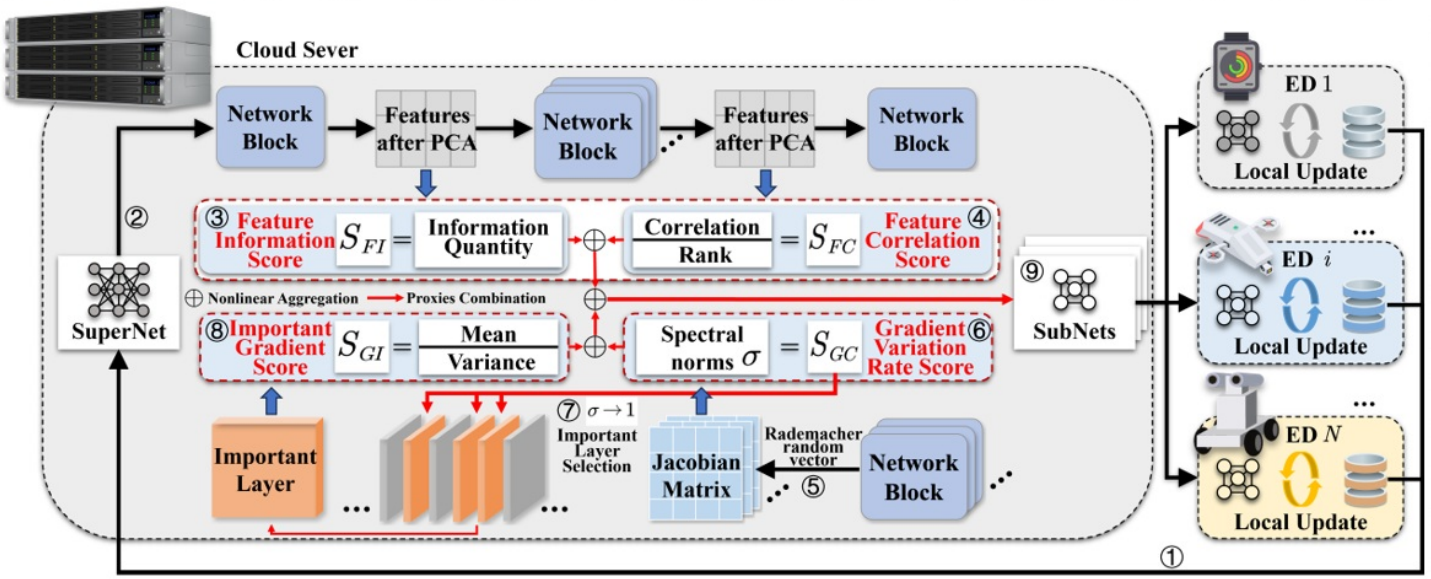
[4] N. Cavagnero, L. Robbiano, B. Caputo, and G. Averta, “FreeREA: Training-Free Evolution-based Architecture Search,” in *Winter Conference on Applications of Computer Vision (WACV)*, 2023 pp. 1493–1502.

[5] G. Li, Y. Yang, K. Bhardwaj, and R. Marculescu, "ZiCo: Zero-shot NAS via Inverse Coefficient of Variation on Gradients," in *Proc. Int. Conf. Learn. Representations (ICLR)*, 2023.

Outline

1. Introduction
2. Approach
3. Experimental Evaluation
4. Conclusion

2.1 Approach: FG-NAS(Featrue-Gradient Aggregated Zero-Cost Neural Architecture Search)



① Upload data characteristics.

② Input SuperNet.

③ Compute Feature Information Score.

④ Compute Feature Correlation Score.

⑤ Generate Jacobian Matrix.

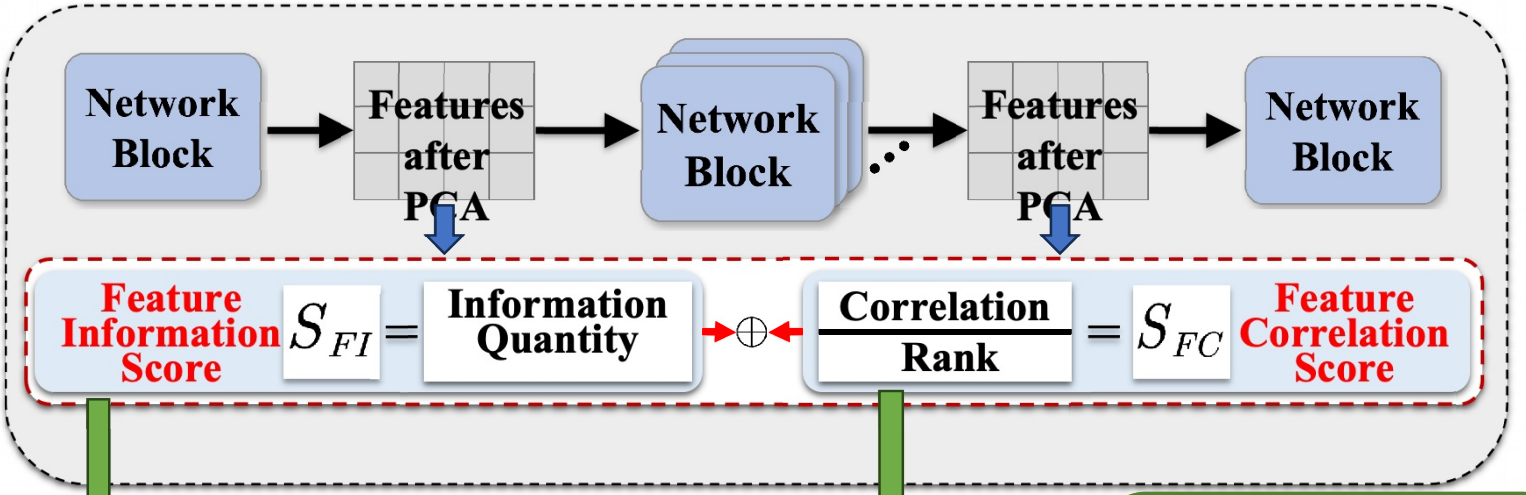
⑥ Compute Gradient Variation Rate Score.

⑦ Select Important Layer.

⑧ Compute Important Gradient Score.

⑨ Output the SubNet ranking and send the optimal SubNet to the ED.

2.2 Feature Proxies



The Richness of Features Extracted by the Subnet

$$S_{FI} = \sum_{l=1}^{N} \frac{\ln(2\pi\sigma^2) + 1}{2}$$

Variance of the Feature Matrix ↑ σ → S_{FI} ↑

The Diversity of Features Extracted by the Subnet

$$S_{FC} = \sum_{l=1}^{N} \frac{\| f_l \|_{nuc}}{sum(P_{f_l})}$$

Nuclear Norm

Feature Extraction Capability

$$\| f_l \|_{nuc} = tr\left(\sqrt{f_l^T f_l}\right)$$

$\| f_l \|_{nuc}$ ↑ → S_{FC} ↑

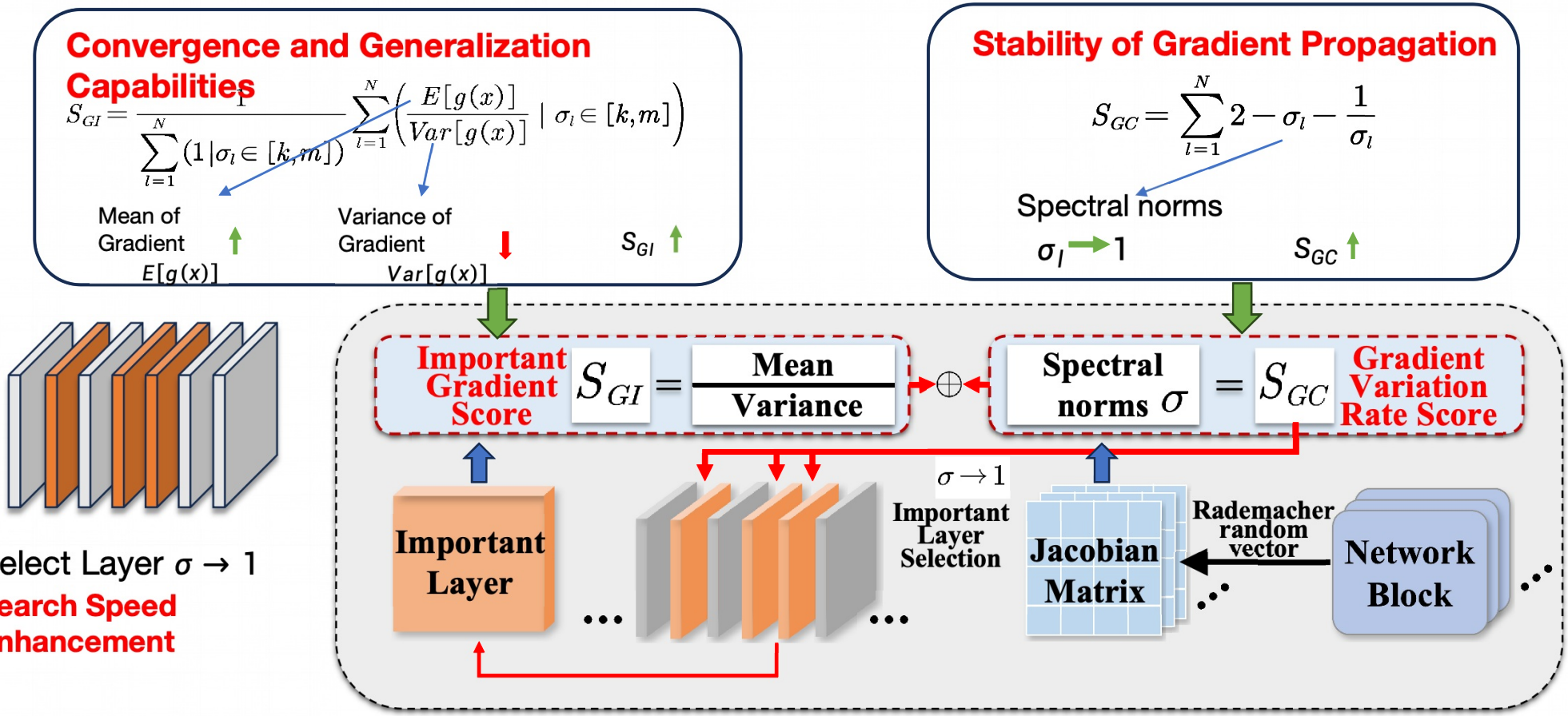
Pearson correlation coefficient

Feature Extraction Capability

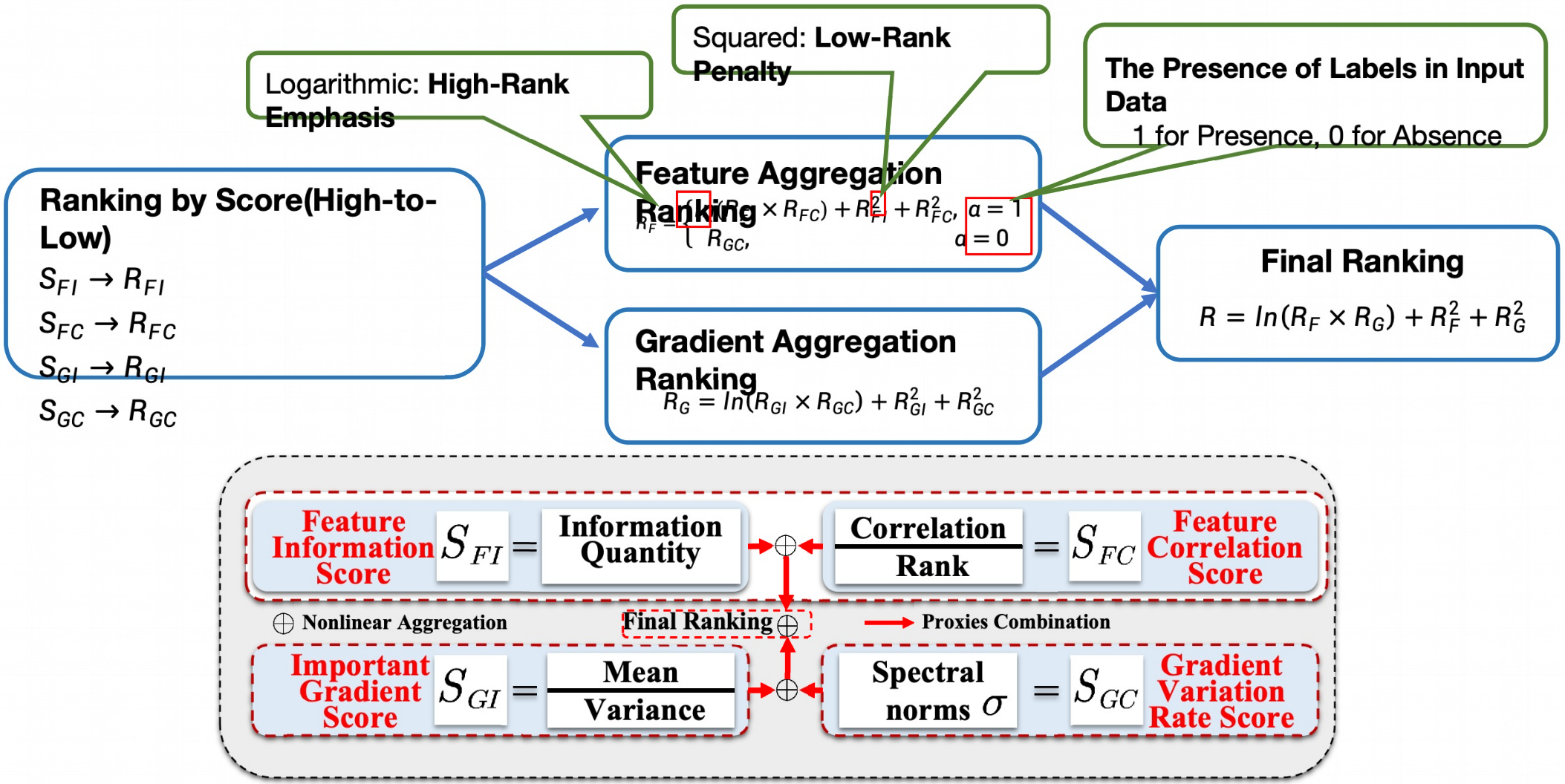
$$\rho(f_l^i, f_l^j) = \frac{\sum_{m=1}^{k}\left(f_l^{i,m} - \bar{f}_l^i\right)\left(f_l^{j,m} - \bar{f}_l^j\right)}{\sqrt{\sum_{m=1}^{k}\left(f_l^{i,m} - \bar{f}_l^i\right)^2 \sum_{m=1}^{k}\left(f_l^{j,m} - \bar{f}_l^j\right)^2}}$$

P_{f_l} ↓ → S_{FC} ↑

2.3 Gradient Proxies



2.4 Dynamic Nonlinear Aggregation



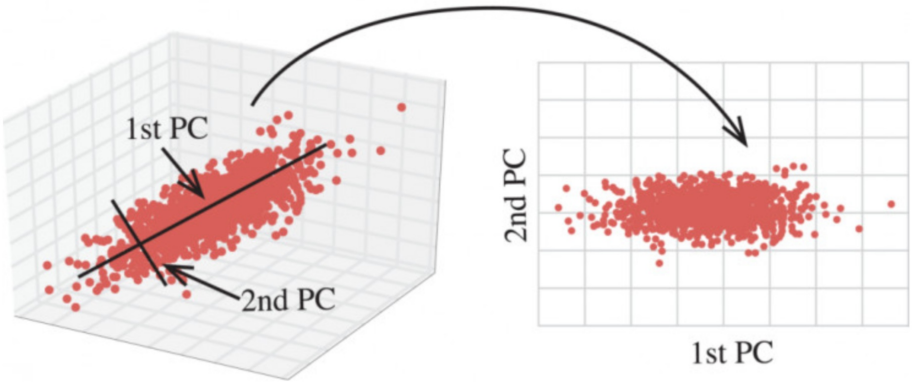
2.5 Dimensionality Reduction

Our Solution:

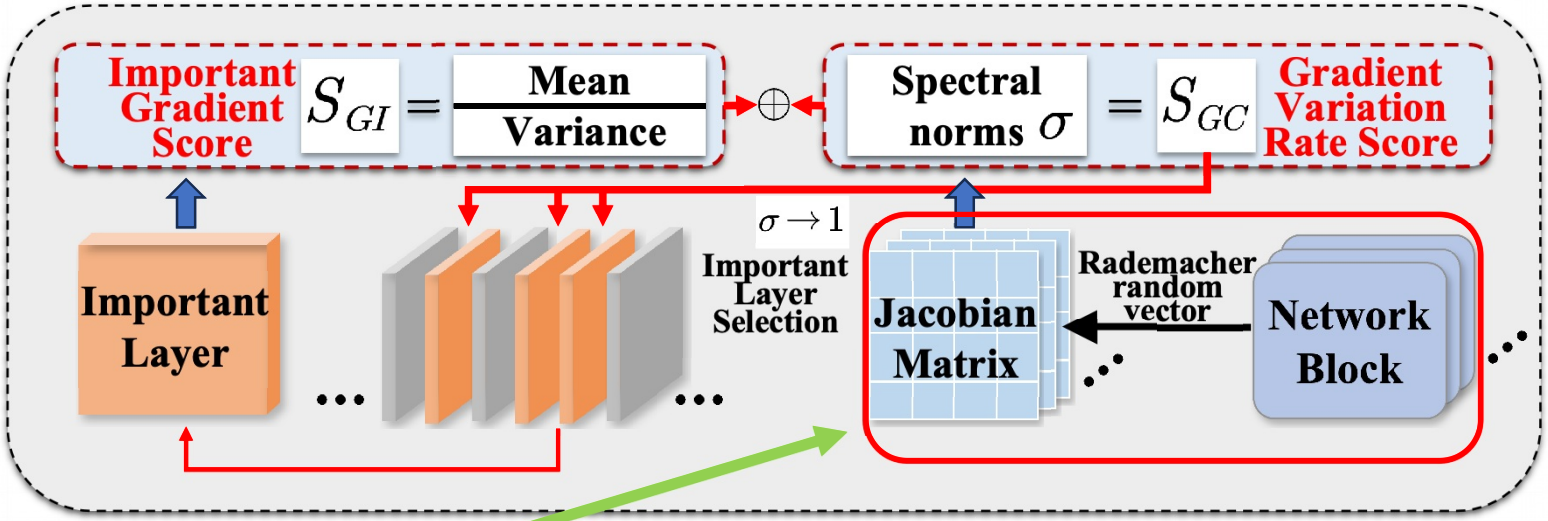
Principal Component Analysis (PCA) is a linear method for dimensionality reduction and feature extraction.

The purpose of dimensionality reduction

- Acceleration of the Subnet Evaluation Process
- Concentration on Important Features



2.6 Jacobian Matrix



Directly calculating Jacobain Martix needs high compute complexity.

$$J_l = \frac{\partial f_l}{\partial f_{l-1}}$$

Rademacher random vector

Reduce Compute Complexity with Rademacher Random Vector.

$$v_k' = Block(v_k^T), \quad J_l = \frac{1}{n}\sum_{k=1}^{n} v_k' v_k^T$$

Outline

1. Introduction
2. Approach
3. Experimental Evaluation
4. Conclusion

3.1 Experimental Environment and Search Space

Experimental Environment	
Cloud Servers	NVIDIA A100 GPUs with 80GB memory and 256GB RAM
End Devices(EDs)	Jetson Xavier NX equipped with 8GB memory and 6-core ARM CPUs
Datasets	CIFAR-10 CIFAR-100 ImageNet16-120

Jetson Xavier NX

Search Space

- **NAS-Bench-201**: Search Space with 15,625 Subnets
- **MobileNetV2**: Search Space with Inverted Residual Structures
- **AutoFormer**: Search Space for Evaluating Transformer Architectures

3.2 Baseline and Metrics

Baseline		
Zero-Cost Proxy Methods	/	#Params/FLOPS
	Based on Forward Propagation	NASWOT
	Based on Backward Propagation	SNIP ZiCo
	Multi-Proxies + Linear Aggregation	TE-NAS
Training-Based Methods	OFA	
	AutoFormer	

Metrics

- **Spearman ρ**: The Correlation Between Predicted Subnet Ranks and True Performance

$$\rho = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{n(n^2-1)}$$

- **Top-1 Accuracy**: Measure Accuracy of the Subnet

$$Top1\ acc. = \frac{\text{Number of Correctly Predicted Samples}}{\text{Total Number of Samples}}$$

- **Search Time**: Subnet Search Time

3.5 Spearman ρ

TABLE II
CORRELATION BETWEEN DIFFERENT ZERO-COST PROXIES AND MODEL ACCURACY IN THE NAS-BENCH-201 SUPERNET

Method	*Spearman* ρ			Time(ms)
	CIFAR-10	CIFAR-100	ImageNet16-120	
#Params/FLOPS	0.753	0.727	0.691	-
SNIP	0.615	0.619	0.539	304.8
NASWOT	0.743	0.769	0.760	**33.1**
TE-NAS	0.731	0.728	0.680	1209.1
ZiCo	0.809	0.785	0.778	342.6
FG-NAS	**0.853**	**0.849**	**0.843**	316.3

It demonstrates efficiency and accuracy of the FG-NAS.

Proxy methods based on the forward propagation are significantly faster.

3.6 Top-1 Accuracy

TABLE III
EXPERIMENTAL RESULTS IN THE MOBILENETV2 SUPERNET

Constraint (Params, FLOPs)	Method	Type	*Top*-1	Search Cost (GPU Days)
(5M, 600M)	OFA	TB	78.7	50
	ZiCo	TF	79.1	0.4
	FG-NAS	TF	**79.5**	**0.35**
(3M, 450M)	OFA	TB	77.7	50
	ZiCo	TF	78.1	0.4
	FG-NAS	TF	**78.4**	**0.33**

TABLE IV
EXPERIMENTAL RESULTS IN THE AUTOFORMER SUPERNET

Constraint (Params, FLOPs)	Method	Type	*Top*-1	Search Cost (GPU Days)
(25M, 5G)	AutoFormer	TB	74.7	24
	TF-TAS	TF	75.3	0.5
	FG-NAS	TF	**75.9**	**0.25**
(5M, 1.5G)	AutoFormer	TB	81.7	24
	TF-TAS	TF	81.9	0.6
	FG-NAS	TF	**82.0**	**0.31**

TB(**T**raining-**b**ased methods)

TF(**T**raining-**f**ree methods)

Substantial Increase in Search Time Due to Training

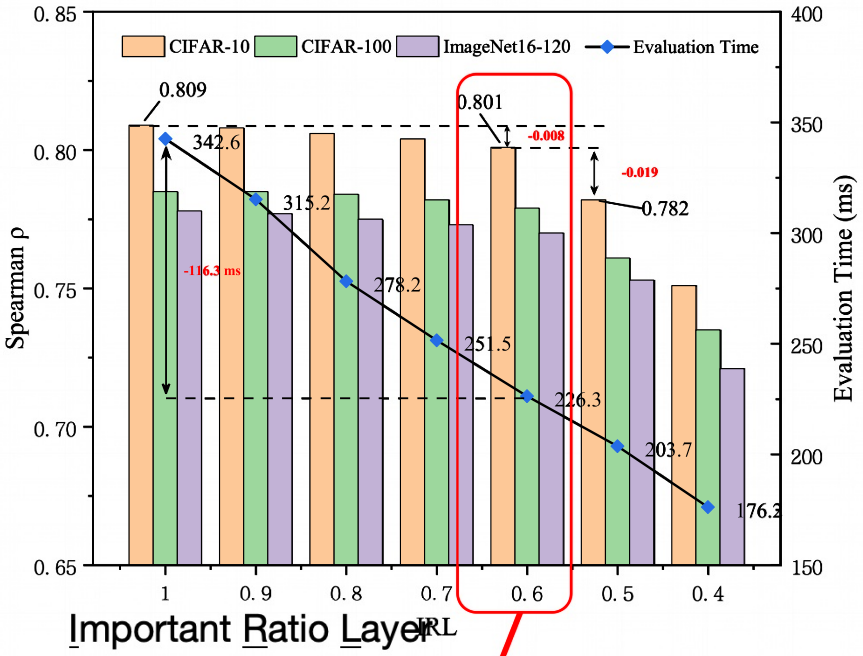
3.7 Search Time

TABLE V
EXPERIMENTAL RESULTS OF FG-NAS ABLATION

Method	*Spearman* ρ			Time(ms)
	CIFAR-10	CIFAR-100	ImageNet16-120	
-FP	0.803	0.782	0.779	282.6
-GP	0.787	0.771	0.762	**36.7**
-NL	0.839	0.815	0.798	311.4
FG-NAS	**0.853**	**0.849**	**0.843**	316.3

-FP(Remove Feature Proxy)
-GP(Remove Gradient Proxy)
-NL(Remove Dynamic Nonlinear Aggregation)

Proxy methods based on the forward propagation are significantly faster.



An important ratio layer of 0.6 achieves balance of the accuracy and search time.

Outline

1. Introduction
2. Approach
3. Experimental Evaluation
4. Conclusion

Conclusion

- The proposed FG-NAS framework introduces feature and gradient zero-cost proxies to achieve accurate and efficient subnet search.
- Dynamic nonlinear aggregation contributes to higher search accuracy.

Thank You!