GENNAPE: Towards Generalized Neural Architecture Performance Estimators

Keith G. Mills[1], Fred X. Han[2], Jialin Zhang[3], Fabian Chudak[2], Ali Safari Mamaghani[1], Mohammad Salameh[2], Wei Lu[2], Shangling Jui[3] and Di Niu[1]

[1]University of Alberta [2]Huawei Technologies Canada [3]Huawei Kirin Solution, Shanghai, China

Link to data: https://github.com/Ascend-Research/GENNAPE

Neural Architecture Search (NAS) is about optimizing and automating network design.

A key resource bottleneck in the NAS process is Performance Evaluation, e.g., how to obtain the accuracy of an image classification network.

Neural Predictors enjoy high speed and low resource costs by learning to estimate performance.

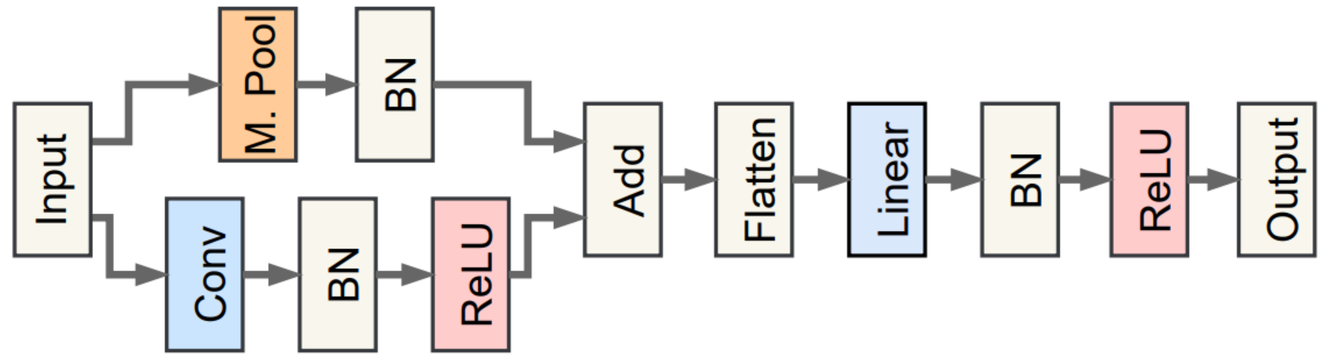
However, a key drawback of existing predictors is that they are confined to one search space, e.g., NAS-Benchmark networks, at a time.

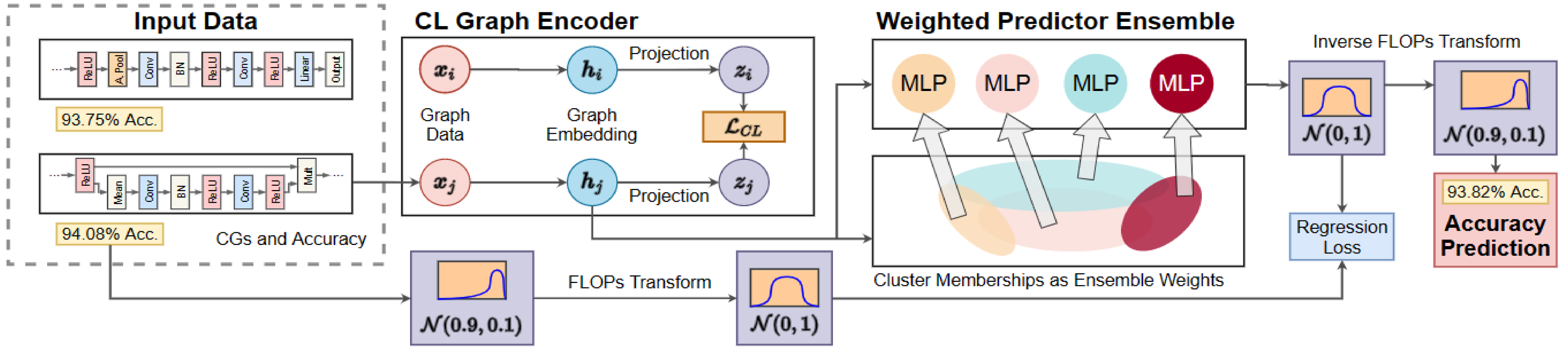
When new networks are introduced, there is a high resource cost incurred to obtain training samples.

In this paper, we propose GENNAPE: **GEN**eralized **N**eural **A**rchitecture **P**erformance **E**stimators in order to introduce search space transferability into the field of neural predictors.

Contributions of GENNAPE

1. Use a robust Computational Graph (CG; example shown below) format that represents network architectures from different search spaces by casting primitive operations (e.g., Conv2D) as nodes.
2. Introduce a semi-supervised Contrastive Learning (CL) method for pre-training a graph encoder using a spectral distance based on the structural properties of Laplacian Eigenvalues.
3. Use Fuzzy C-Means to perform soft clustering on graph embeddings in order to train a weighted predictor ensemble to cover different regions of the latent space.
4. Introduce three new benchmark families and open-source our data in order to further transferable predictor research:
   HiAML: Used in Facial Landmark Detection.
   Inception: Used in Facial Recognition.
   Two-Path: Used in Super Resolution and 4k LivePhoto.





Graph Encoder Pre-Training and Ensemble Clustering

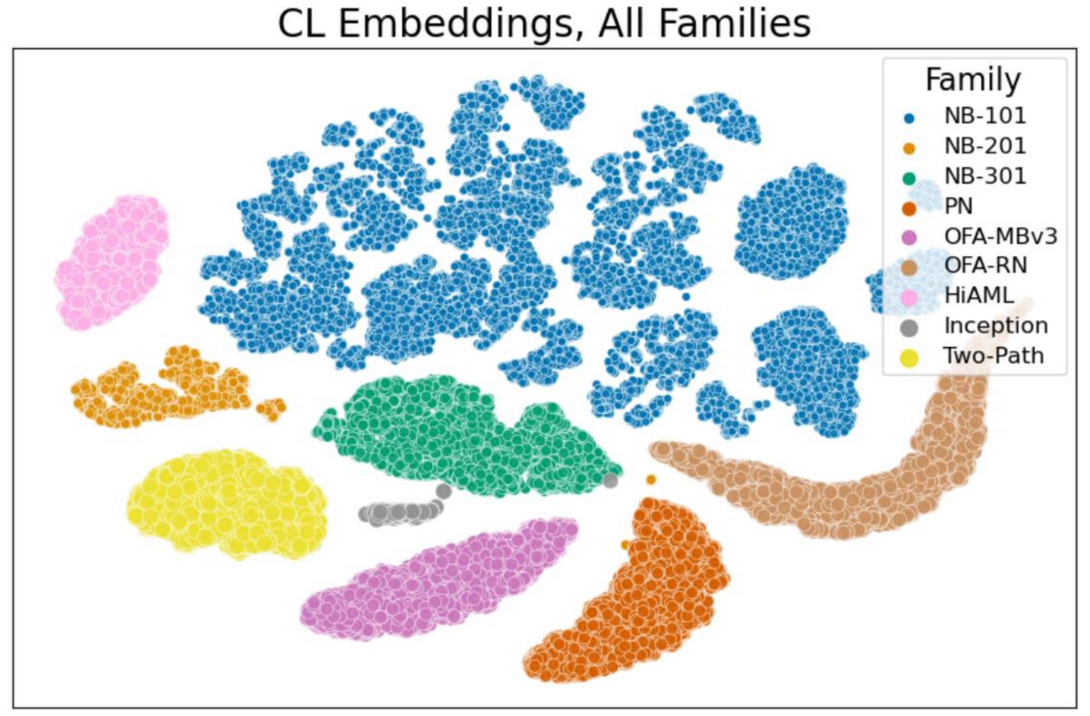
We pre-train a graph encoder using a semi-supervised Contrastive Learning (CL) loss like SimCLR or SupCon.

The goal of CL is to ensure similarity between the embeddings (latent representations) of similar data samples.

We determine similarity of two CGs i and l using a spectral distance based on Laplacian Eigenvalues: $\alpha_l^{(i)}$.

$$\mathcal{L}_{CL} = -\sum_{i \in I} \sum_{\ell \neq i} \alpha_\ell^{(i)} \log \frac{\exp(sim(z_i, z_\ell))}{\sum_{r \neq i} \exp(sim(z_i, z_r))},$$

We pre-train a graph encoder on NAS-Bench-101. The embeddings form small clusters. Inferred embeddings for other search spaces form distinct clusters:



We cluster the NB-101 embeddings using Fuzzy C-Means (FCM) to produce continuous cluster memberships. Clusters overlap and represent different regions of the latent space.

A single data point is represented by many clusters. We train a weighted ensemble using cluster membership.

Single Search Space Evaluation

First, test performance on NAS-Bench-101, in terms of Mean Absolute Error and Spearman's Correlation.

Compare our method to other single-search predictors like TNASP and BANANAS, and simple GNNs that can use CGs.

Method	MAE	SRCC
NPN†	$1.09 \pm 0.01\%$	**0.934 ± 0.003**
BANANAS†	$1.40 \pm 0.06\%$	0.834 ± 0.002
TNASP†	$1.23 \pm 0.02\%$	0.918 ± 0.002
GCN	$1.78 \pm 0.06\%$	0.732 ± 0.034
GIN	$1.72 \pm 0.04\%$	0.735 ± 0.035
k-GNN	$1.61 \pm 0.08\%$	0.814 ± 0.020
CL+MLP	$1.51 \pm 0.17\%$	0.874 ± 0.009
CL+FCM	$1.19 \pm 0.12\%$	0.896 ± 0.003
CL+MLP+T	*$0.65 \pm 0.08\%$*	0.921 ± 0.003
CL+FCM+T	**$0.59 \pm 0.01\%$**	*0.930 ± 0.002*

Result: The contributions of our method gradually reduce MAE until it is only 0.59% and improve SRCC until it is above 0.9 and it exceeds or is on-par with several single-space predictors.

Application to NAS

Model	Dataset	FLOPs	Top-1 Acc.(%)
NB-101-Best	CIFAR-10	11.72G	94.97
NB-101-Search	CIFAR-10	**9.49G**	**95.05**
NB-201-Best	CIFAR-10	313M	93.27
NB-201-Search	CIFAR-10	**283M**	**93.62**
OFA-ResNet-Input	ImageNet120	12.13G	80.62
OFA-ResNet-Search	ImageNet120	**9.46G**	**81.08**

We pair a predictor with a CG-based search algorithm.

- Algorithm operates on node-based mutations.
- Like changing the operation type, filter, or channels.
- Mutation results in networks outside of original family.
- E.g., for cell-based NAS benchmark families, we can mutate an operation node in a specific cell, rather than all of them.

We can eclipse the performance of the best NB-101/201 architectures by reducing FLOPs while increasing the accuracy.

Transferability Test: Spearman's Rank Correlation Coefficient

Train on NB-101, then infer on other families like NB-201.
Two scenarios:

1. Zero-shot transfer.
2. Fine-tuning on 50 labeled CGs.

For zero-shot transfer, GENNAPE achieves SRCC above 0.8 for PN, OFA-MBv3 and NB-201.

With fine-tuning, achieve above 0.85 SRCC for all public benchmarks.

SRCC (Tab. 5 in paper)

Family	k-GNN	GENNAPE
NB-201	0.4930	**0.8146**
w/ FT	0.8606 ± 0.0245	**0.9103 ± 0.0114**
NB-301	0.0642	**0.3214**
w/ FT	0.8584 ± 0.0290	**0.8825 ± 0.0134**
PN	0.0703	**0.8213**
w/ FT	0.7559 ± 0.0621	**0.9506 ± 0.0039**
OFA-MBv3	0.4345	**0.8660**
w/ FT	0.6862 ± 0.0253	**0.9449 ± 0.0015**
OFA-RN	**0.5721**	0.5115
w/ FT	0.9102 ± 0.0146	**0.9114 ± 0.0063**
HiAML	-0.1211	**0.4331**
w/ FT	**0.4300 ± 0.0507**	0.4169 ± 0.0479
Inception	-0.2045	**0.4249**
w/ FT	0.3340 ± 0.0793	**0.5524 ± 0.0166**
Two-Path	0.1970	**0.3413**
w/ FT	0.3694 ± 0.0406	**0.4875 ± 0.0311**

Transferability Test: Normalized Discounted Cumulative Gain

NDCG, originally from Information Retrieval (IR), prioritizes correctly ranking architectures with high accuracy.

Important for when a search algorithm needs to find good architectures.

In zero-shot setting, our method achieves over 0.65 in all cases.

NDCG@10 (Tab. 6 in paper)

Family	k-GNN	GENNAPE
NB-201	0.9270	**0.9793**
w/ FT	0.9751 ± 0.0082	**0.9855 ± 0.0030**
NB-301	0.5341	**0.7885**
w/ FT	0.9723 ± 0.0134	**0.9765 ± 0.0081**
PN	0.4426	**0.8736**
w/ FT	0.9287 ± 0.0271	**0.9800 ± 0.0057**
OFA-MBv3	0.8464	**0.9234**
w/ FT	0.8859 ± 0.0536	**0.9838 ± 0.0030**
OFA-RN	**0.9470**	0.6606
w/ FT	**0.9717 ± 0.0090**	0.9463 ± 0.0236
HiAML	0.5088	**0.6892**
w/ FT	0.7356 ± 0.0371	**0.7804 ± 0.0211**
Inception	0.6064	**0.8150**
w/ FT	0.7310 ± 0.0423	**0.8073 ± 0.0072**
Two-Path	0.6339	**0.8275**
w/ FT	0.7860 ± 0.0268	**0.8392 ± 0.0220**

With fine-tuning, this increases to over 0.94 on all public benchmarks and over 0.75 for all introduced families.