ECCV'20
ONLINE
23-28 AUGUST 2020

16TH EUROPEAN CONFERENCE ON

COMPUTER VISION

WWW.ECCV2020.EU

A Generic Graph-based Neural Architecture Encoding Scheme for Predictor-based NAS

Xuefei Ning (宁雪妃)[1], Yin Zheng[2], Tianchen Zhao[13], Yu Wang[1], Huazhong Yang[1]

NICS-EFC Lab, Department of Electronic Engineering, Tsinghua University[1]

Weixin Group, Tencent[2]

Department of Electronic Engineering, Beihang University[3]

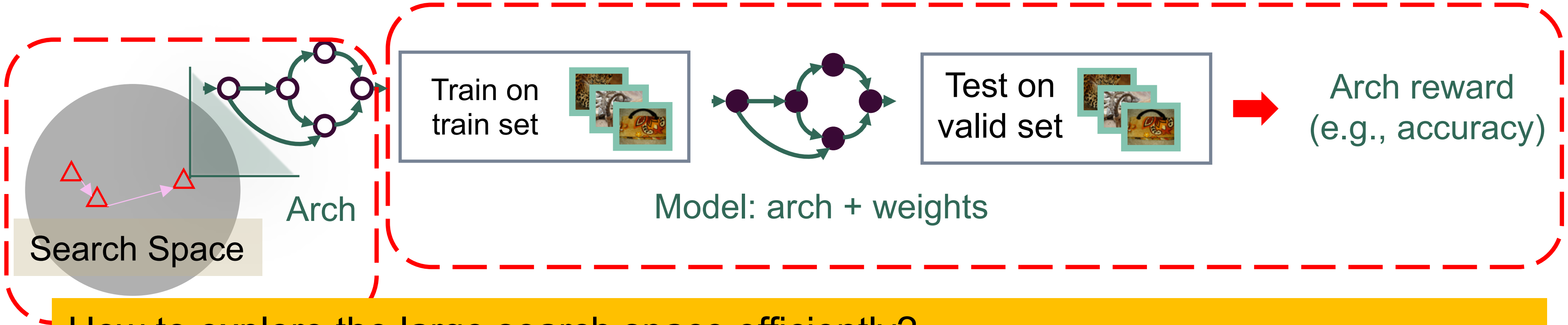
Xuefei Ning foxdoraame@gmail.com, Prof. Yu Wang yu-wang@tsinghua.edu.cn

Background

- Computational challenge of Neural Architecture Search (NAS)

Total time cost for NAS algorithm: N x T

- **N** architectures in the search space are actually evaluated
- **T** for evaluating each architecture on average



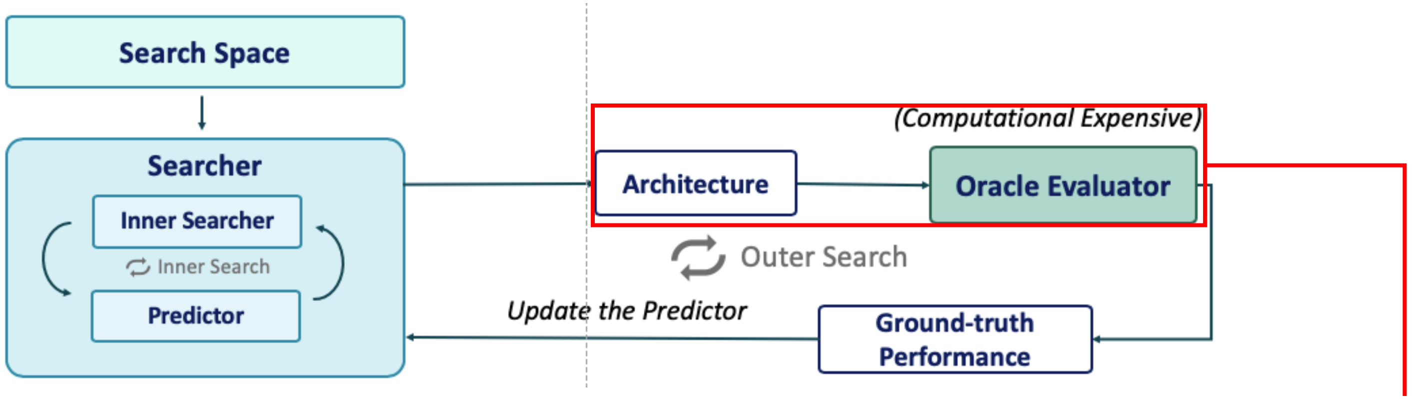
How to explore the large search space efficiently?
Decrease the number of architectures (N) need to be evaluated to discover a good architecture.

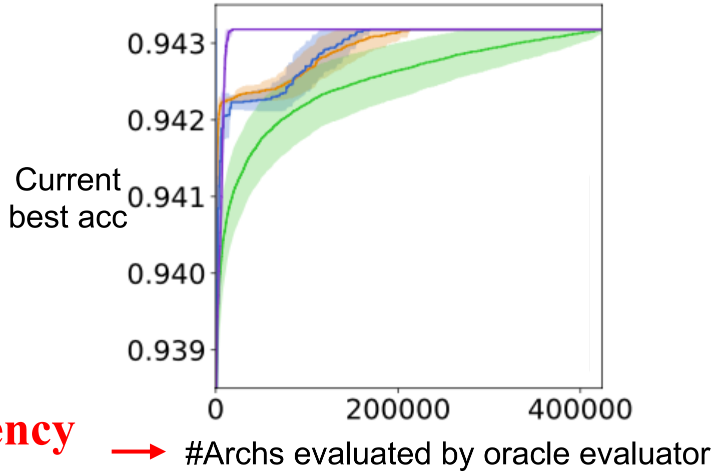
Predictor-based NAS

Use a predictor that predict the arch's performance (optionally with uncertainty) to guide the sampling/searching

Background

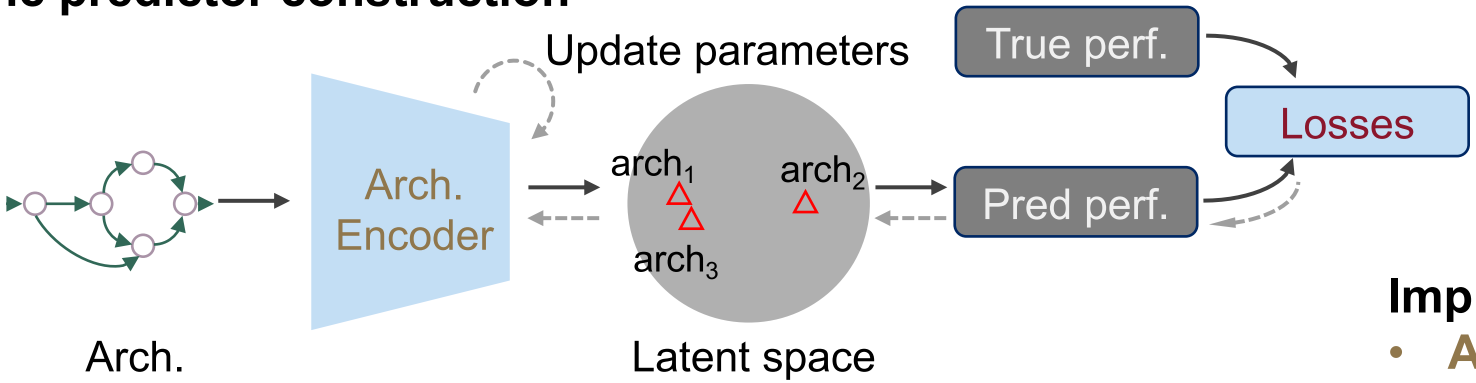
- Predictor-based NAS





The predictor's fitness is vital to the predictor-based searcher's sample efficiency

Typical parametric predictor construction

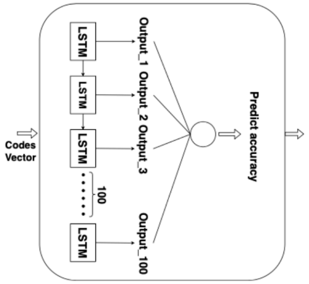


Can we use less true perf. data to learn better representation of archs (better latent space)?

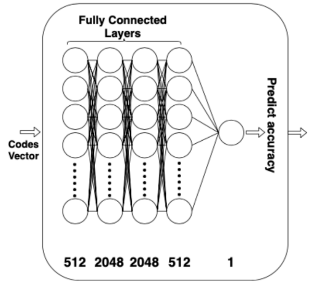
Improvements from 2 aspects
- **Arch. encoder**
- **Training loss**

Motivation

- **Arch. Encoder**



RNN

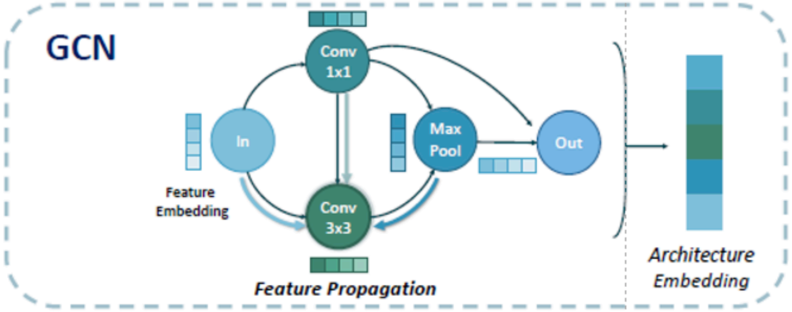


MLP

Sequence-based encoder [Luo et al. NIPS 2018]

Not suitable for handling DAGs

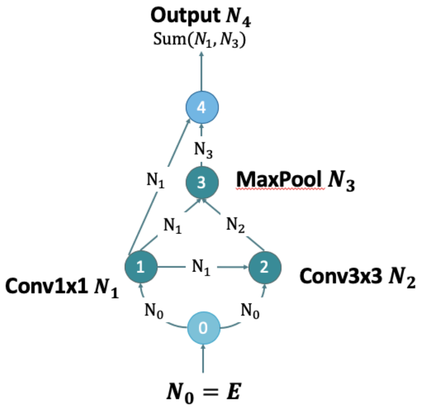
An architecture and its isomorphic counterparts
can have multiple different encodings



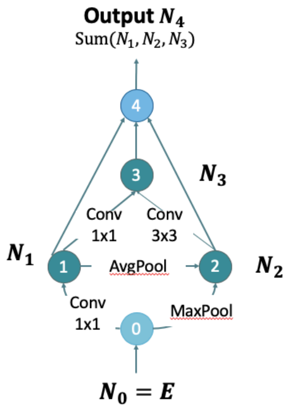
GCN-based encoder [Guo et al. NIPS 2019, Shi et al. 2019]

Not suitable for handling **data-processing DAG** (NN architecture)

1. Existing GCN encoder models the operation (Conv, Pooling)
as the information to propagate on the graph, which is not intuitive for data-processing DAG
2. Existing GCN encoder cannot encode architectures from **"operation-on-edge"** search spaces



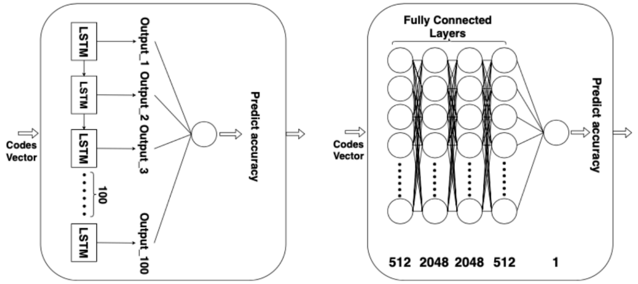
Operation on Node



Operation on Edge

Motivation

- **Arch. Encoder**

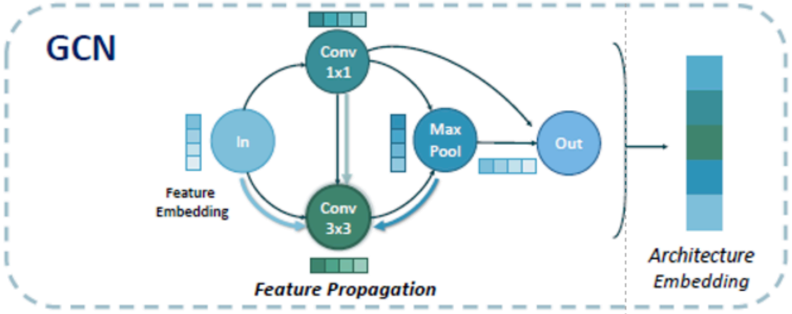


RNN MLP

Sequence-based encoder [Luo et al. NIPS 2018]

Not suitable for handling DAGs

An architecture and its isomorphic counterparts can have multiple different encodings



GCN-based encoder [Guo et al. NIPS 2019, Shi et al. 2019]

Not suitable for handling **data-processing DAG** (NN architecture)

1. Existing GCN encoder models the operation (Conv, Pooling) as the information to propagate on the graph, which is not intuitive for data-processing DAG
2. Existing GCN encoder cannot encode architectures from **"operation-on-edge"** search spaces

- **Training loss**

What is important in NAS is the relative ranking order of architectures, not the absolute score

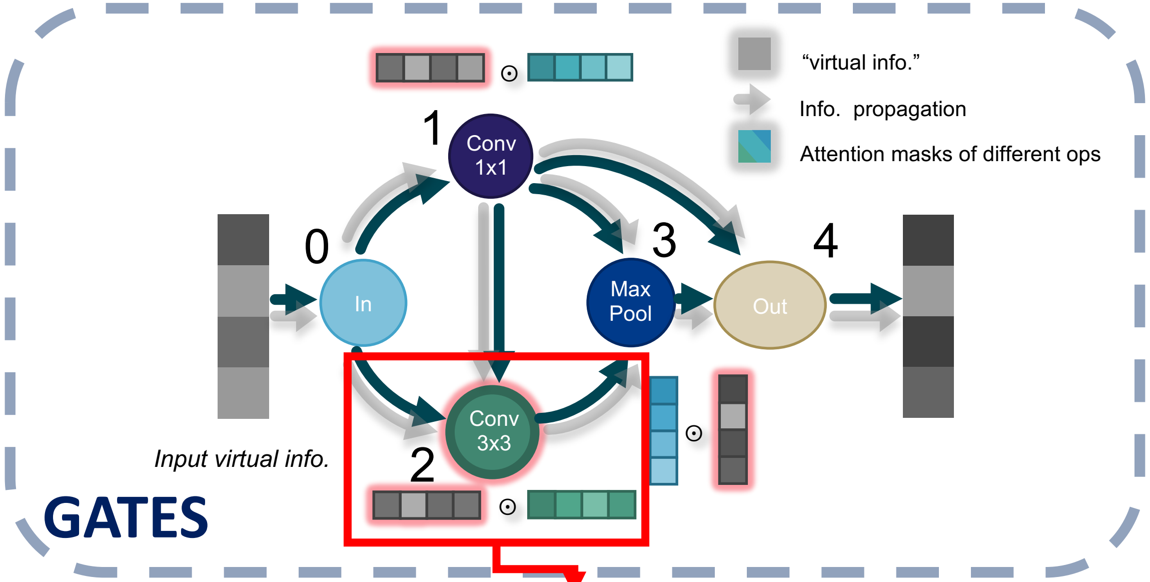
- Regression loss: make predicted score P(a_j) close to true performance y_j

$$L(\{a_j, y_j\}_{j=1,\cdots,N}) = \sum_{j=1}^{N} (\mathrm{P}(a_j) - y_j)^2$$

L is not a good surrogate of the ranking measures

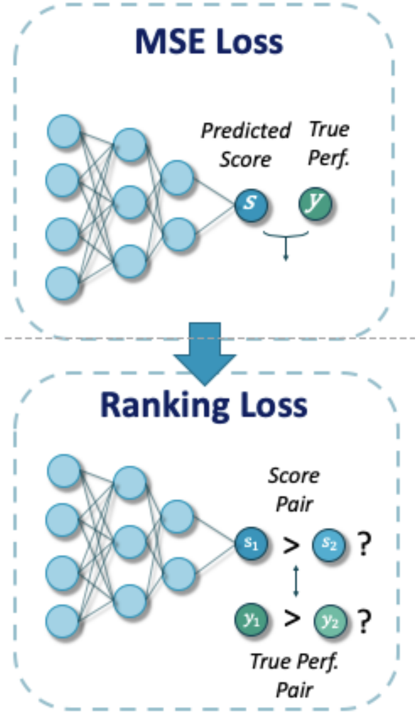
GATES

- Improve **Encoder** and **Training losses**
  - A more generic **G**raph-based neural **A**rchi**T**ecture **E**ncoding **S**cheme (**GATES**)
    - Mimic the information propagation in the architecture to encode it
  - Learning to Rank (LtR) losses (Relative order matters rather than absolute perf.)
    - Ranking Losses are better surrogate of ranking measures than regression losses



- feature map computation: F2=Conv3x3(F0 + F1)
- "Virtual info transformation" during architecture encoding: N2=$m_2 \odot$ (N0+N1)

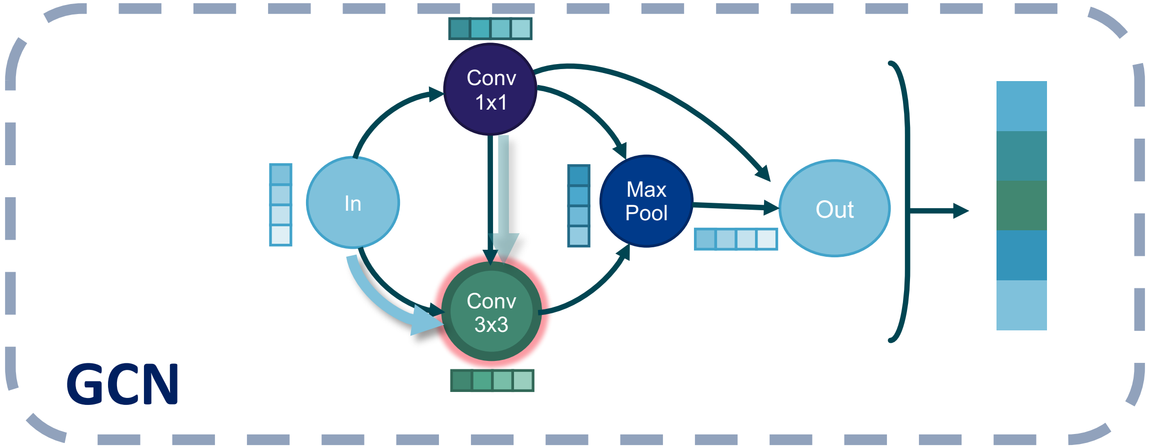
$m_2 = \sigma(\mathrm{EMB}_{Conv3x3}\mathrm{W_o})$is the **attention mask** of Conv3x3



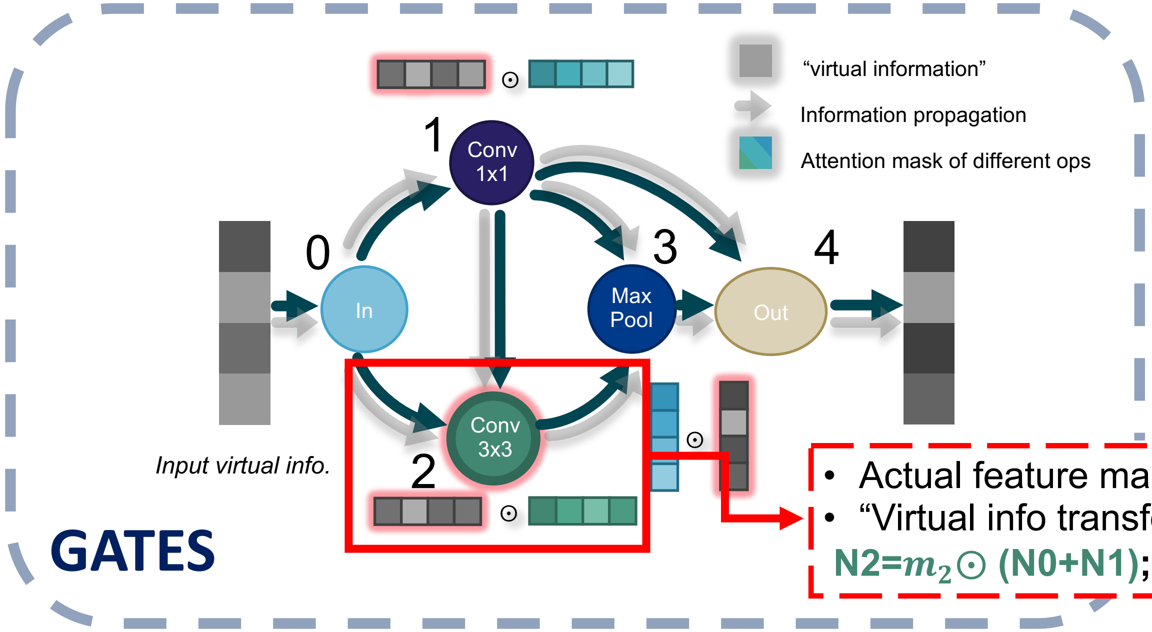
$$\sum_{j=1}^{N} (\mathrm{P}(a_j) - y_j)^2$$

$$\sum_{j=1}^{N} \sum_{i, y_i > y_j} \max[0, m - (P(a_i) - P(a_j))]$$

GATES



- **Operation** modeled as the information to be propagated on the graph
- **Architecture encoding**: After the information is propagated for several steps, the rep. of all nodes are read out (aggregated) as the architecture representation



Mimic the information propagation of NN computation

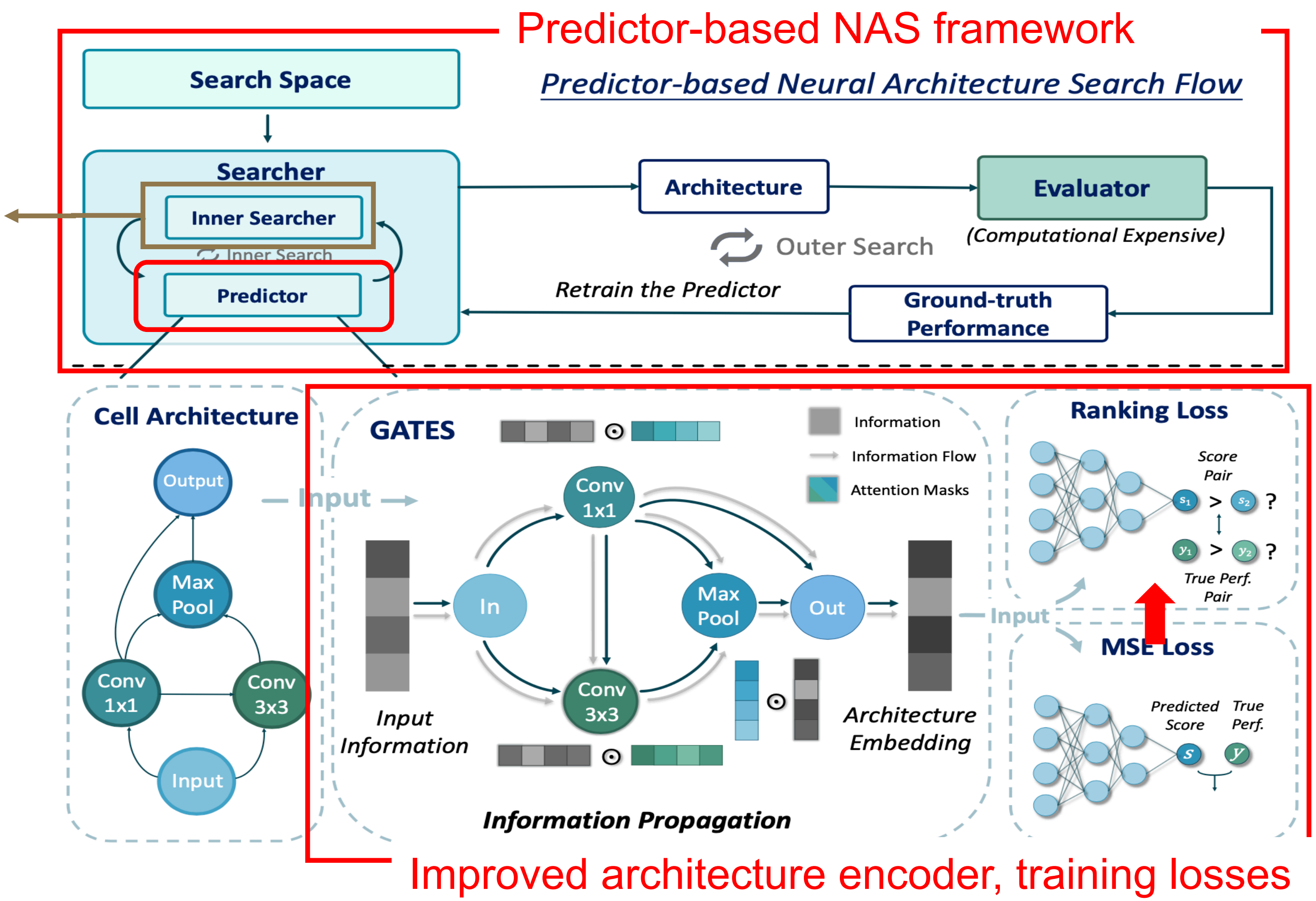
- **Operation** modeled as the transformation/processing of the propagating information (attention mask)
- **Architecture encoding** : output "information" is used as the architecture representation

- Actual feature map computation: **F2=Conv3x3(F0 + F1)**
- "Virtual info transformation" in the arch. encoding process: **N2=m_2⊙ (N0+N1)**; $m_2 = \sigma(\mathrm{EMB}_{Conv3x3}\mathrm{W}_o)$ is the **attention mask** of Conv3x3

Overall framework

- The overall framework of predictor-based NAS with GATES and LtR

- Evolutionary Algorithm (EA)
- Random Search (RS)



Results on NAS-Bench-101

- Ranking correlation (Kendall's Tau) of the predictors
  - Encoder comparison

Encoder	Proportions of 381262 training samples							
	0.05%	0.1%	0.5%	1%	5%	10%	50%	100%
MLP [21]	0.3971	0.5272	0.6463	0.7312	0.8592	0.8718	0.8893	0.8955
LSTM [21]	0.5509	0.5993	0.7112	0.7747	0.8440	0.8576	0.8859	0.8931
GCN (w.o. global node)	0.3992	0.4628	0.6963	0.8243	0.8626	0.8721	0.8910	0.8952
GCN (global node) [20]	0.5343	0.5790	0.7915	0.8277	0.8641	0.8747	0.8918	0.8950
GATES	**0.7634**	**0.7789**	**0.8434**	**0.8594**	**0.8841**	**0.8922**	**0.9001**	**0.9030**

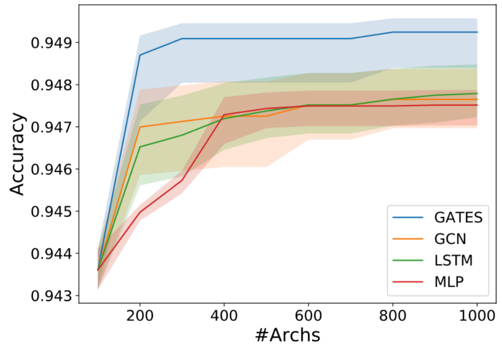
GATES outperform other encoders consistently, especially when there are few training samples

  - Loss function comparison

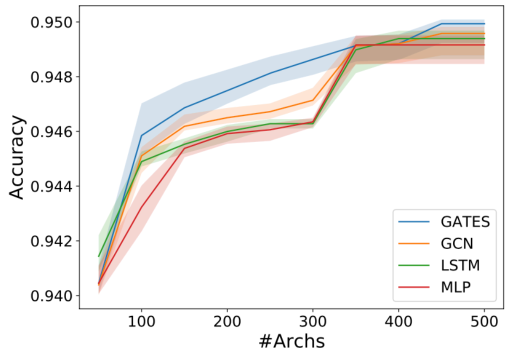
Loss	Proportions of 381262 training samples							
	0.05%	0.1%	0.5%	1%	5%	10%	50%	100%
Regression (MSE) + GCN†	0.4536	0.5058	0.5587	0.5699	0.5846	0.5871	0.5901	0.5941
Regression (MSE) + GATES†	0.4935	0.5425	0.5739	0.6323	0.7439	0.7849	0.8247	0.8352
Pairwise (BCE)	0.7460	0.7696	0.8352	0.8550	0.8828	0.8913	0.9006	0.9042
Pairwise (Comparator)	0.7250	0.7622	0.8367	0.8540	0.8793	0.8891	0.8987	0.9011
Pairwise (Hinge)	0.7634	0.7789	0.8434	0.8594	0.8841	0.8922	0.9001	0.9030
Listwise (ListMLE)	0.7359	0.7604	0.8312	0.8558	0.8852	0.8897	0.9003	0.9009

Ranking losses are better surrogate to ranking measures than regression losses

- Sample efficiency
  - Encoder comparison

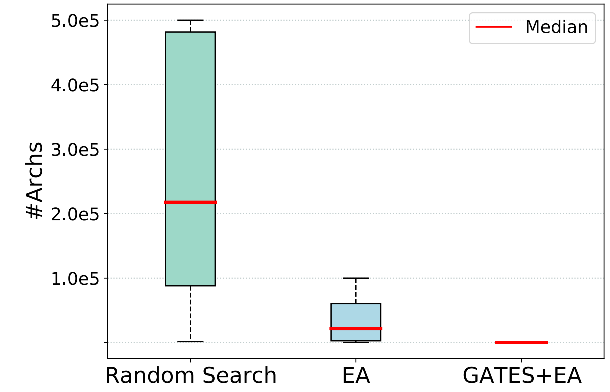


(a) RS inner search method ($r = 500$)



(b) EA inner search method ($r = 100$)

  - Comparison with baseline search strategies



Median: 220k 24k 0.4k

551.0× and 59.25× more efficient than RS/EA

Results on NAS-Bench-101/201

- Two ranking measures for NAS application
  - The Kendall’ s Tau treats all the discordant pairs equally
  - The ranking order among the poorly performed architectures is not important for NAS application

N@K

the best true ranking of the top K predicted architectures

NAS-Bench-101

Encoder	Ranking Loss N@5	Ranking Loss N@10	Regression Loss N@5	Regression Loss N@10
MLP [21]	57 (0.13%)	58 (0.13%)	1397 (3.30%)	552 (1.30%)
LSTM [21]	1715 (4.05%)	1715 (4.05%)	1080 (2.54%)	312 (0.73%)
GCN [19]	2025 (4.77%)	1362 (3.21%)	405 (0.95%)	405 (0.95%)
GATES	**22 (0.05%)**	**22 (0.05%)**	**27 (0.05%)**	**27 (0.05%)**

NAS-Bench-201

Encoder	Ranking Loss N@5	Ranking Loss N@10	Regression Loss N@5	Regression Loss N@10
MLP [21]	7 (0.09%)	7 (0.09%)	1538 (19.7%)	224 (3.87%)
LSTM [21]	8 (1.02%)	2 (0.01%)	250 (6.65%)	234 (2.99%)
GATES	**1 (0.00%)**	**1 (0.00%)**	**1 (0.00%)**	**1 (0.00%)**

Precision@K

the proportion of true top-K architectures among the top-K predicted architectures

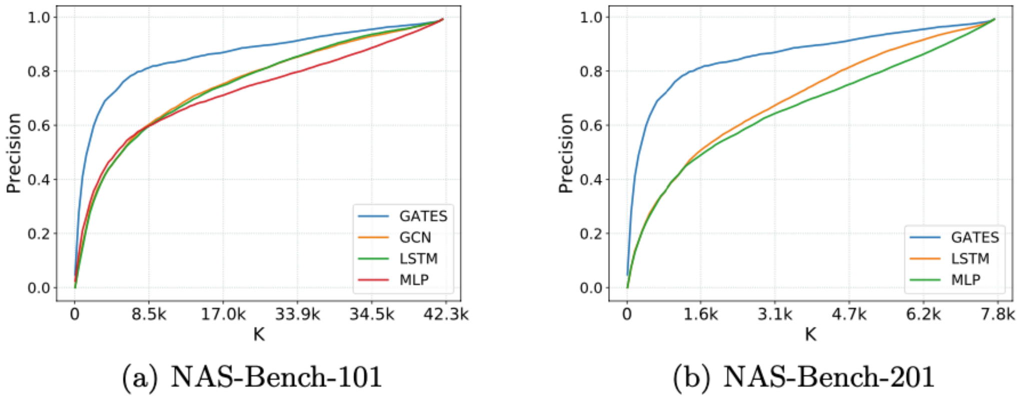
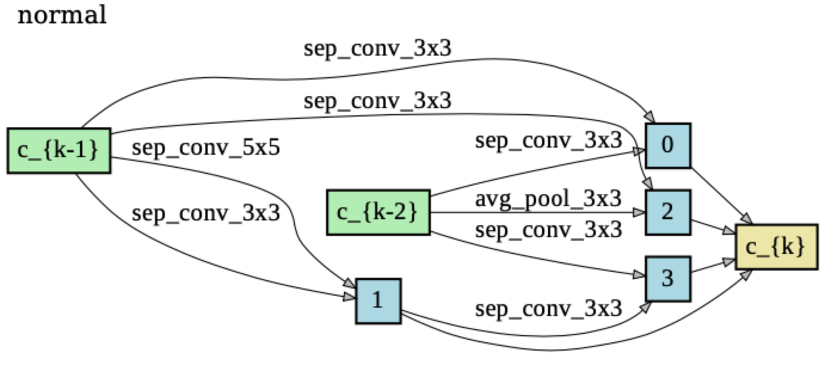


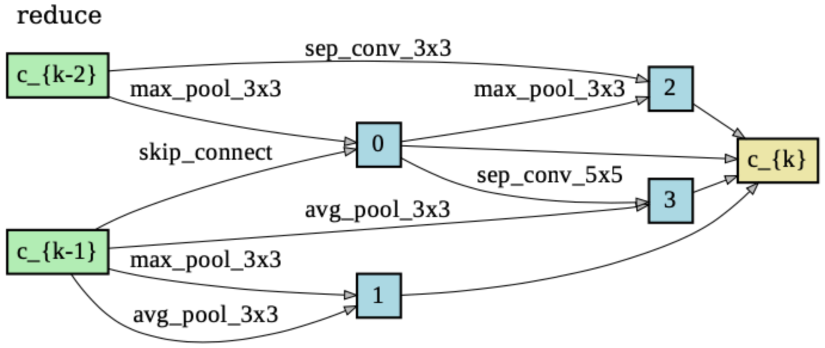
Fig. 3. Precision@K

Results on ENAS search space

- Search on large open search space (ENAS)



(a) Normal cell



(b) Reduction cell

CIFAR-10 results

Method	Test Error (%)	#Params (M)	#Archs Evaluated
NASNet-A + cutout [25]	2.65	3.3	20000
AmoebaNet-B + cutout [16]	2.55	2.8	27000
NAONet [13]	2.98	28.6	1000
PNAS [8]	3.41	3.2	1160
NAONet-WS† [13]	3.53	2.5	-
DARTS+cutout† [10]	2.76	3.3	-
ENAS + cutout† [15]	2.89	4.6	-
Ours + cutout	2.58	4.1	800

Transferring to ImageNet

Method	Top-1 Test Error (%)	#Params (M)
NASNet-A [16]	26.0	5.3
AmoebaNet-B [9]	27.2	5.3
PNAS [6]	25.8	5.1
DARTS [7]	26.9	4.9
GHN [15]	27.0	6.1
Ours	24.1	5.6

Conclusion & Future work

- Knowledge: Ranking measures N@K, Precision@k other than the Kendall's Tau ranking correlation are meaningful for NAS application
- Use GATES to encode topological architecture
  - An intuitive encoding method that is more suitable for data-processing DAGs
  - Correct handling of architecture isomorphism (map isomorphic architectures to the same rep.)
  - Encode both operation-on-edge and operation-on-node architectures
- Use learning-to-rank losses to train the architecture predictor
  - Correspond better with the ranking measures

- Future work
  - Employing GATES to larger or hierarchical search spaces with more complex topologies

Thanks for listening!

Contact us at: Xuefei Ning foxdoraame@gmail.com, Prof. Yu Wang yu-wang@tsinghua.edu.cn

Paper

https://arxiv.org/abs/2004.02164

Code

https://github.com/walkerning/aw_nas

Contributions, suggestions and discussions are all welcome!