TA-GATES: An Encoding Scheme for Neural Network Architectures

Xuefei Ning[*12], Zixuan Zhou[*1], Junbo Zhao[1], Tianchen Zhao[1], Yiping Deng[2],

Changcheng Tang[3], Shuang Liang[3], Huazhong Yang[1], Yu Wang[1]

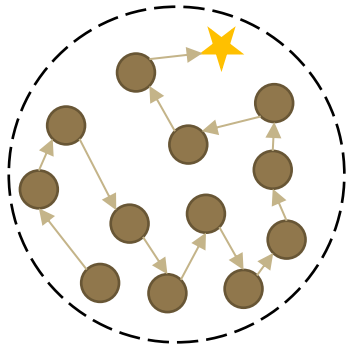
foxdoraame@gmail.com yu-wang@tsinghua.edu.cn

[1]NICS-EFC Lab, EE, Tsinghua University
[2]TCS Lab, Huawei
[3]Novauto Technology Co Ltd.

Background: NAS and The Importance of Architecture Encoding

Target: A better and dedicated encoding scheme for neural architectures

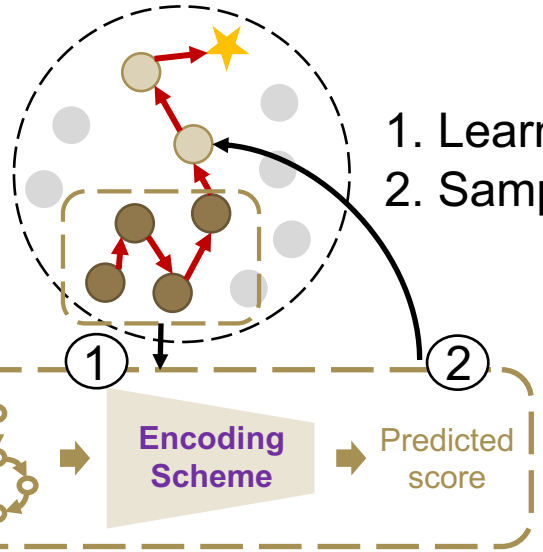
Encoding: Map an architecture DAG into a continuous embedding



Neural Architecture Search (NAS) has a large search space

⇒ Need to sample and evaluate thousands of architectures, slow

e.g., NASRL[1] 12k arch



Predictor-based NAS[2,3]

1. Learn a performance predictor
2. Sample worth-exploring architectures

A better encoding brings better sample efficiency

e.g., GATES[3] 800 arch

A good encoding of architectures can be used for…

Exploration Acceleration

Improve the sample efficiency [2,3]

Better Evaluation

Improve the parameter-sharing evaluation[4]

Kendall's Tau 0.56 -> 0.67

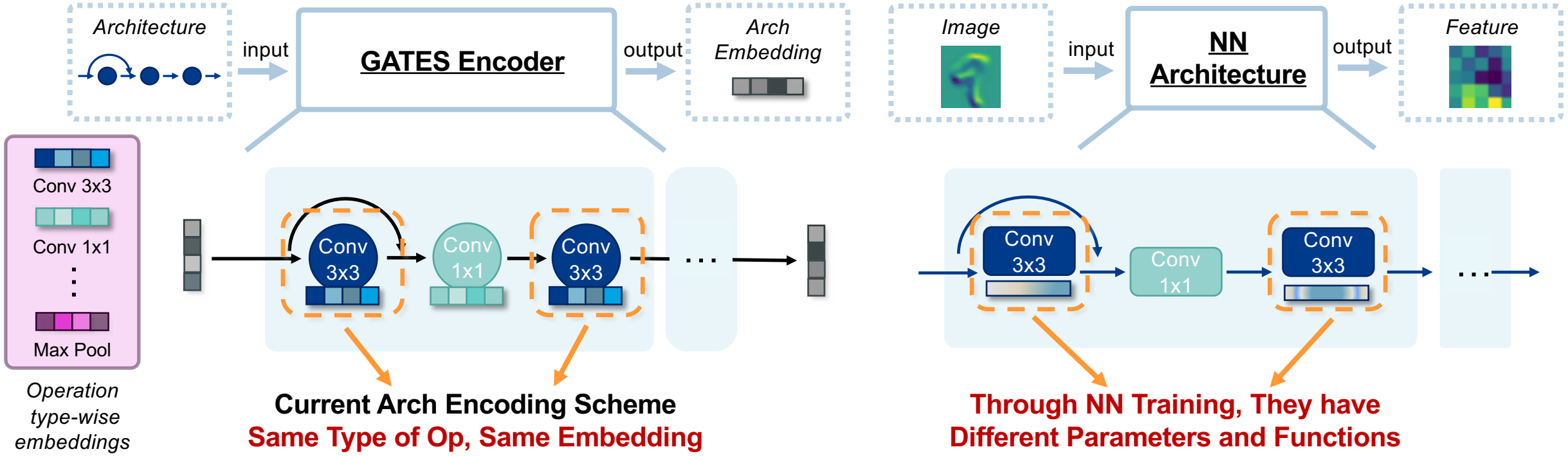
Interpretation

Interpret which architectural pattern is beneficial[5]

[1] Zoph et al., Neural Architecture Search with Reinforcement Learning, ICLR'17.
[2] Luo et al., Neural Architecture Optimization, NeurIPS'18.
[3] Ning et al., A Generic Graph-based Neural Architecture Encoding Scheme for Predictor-based NAS, ECCV'20.
[4] Zhou et al., CLOSE: Curriculum Learning On the Sharing Extent Towards Better One-shot NAS, ECCV'22.
[5] Ru et al., Interpretable Neural Architecture Search via Bayesian Optimisation with Weisfeiler-Lehman kernels, ICLR'21.

Motivation: A Drawback of SOTA Encoding

SOTA "information flow-based" encoders[1,2] view an architecture as a **DAG with operations**.
Drawback: Neglect the "operations are trainable" property of NN architecture.



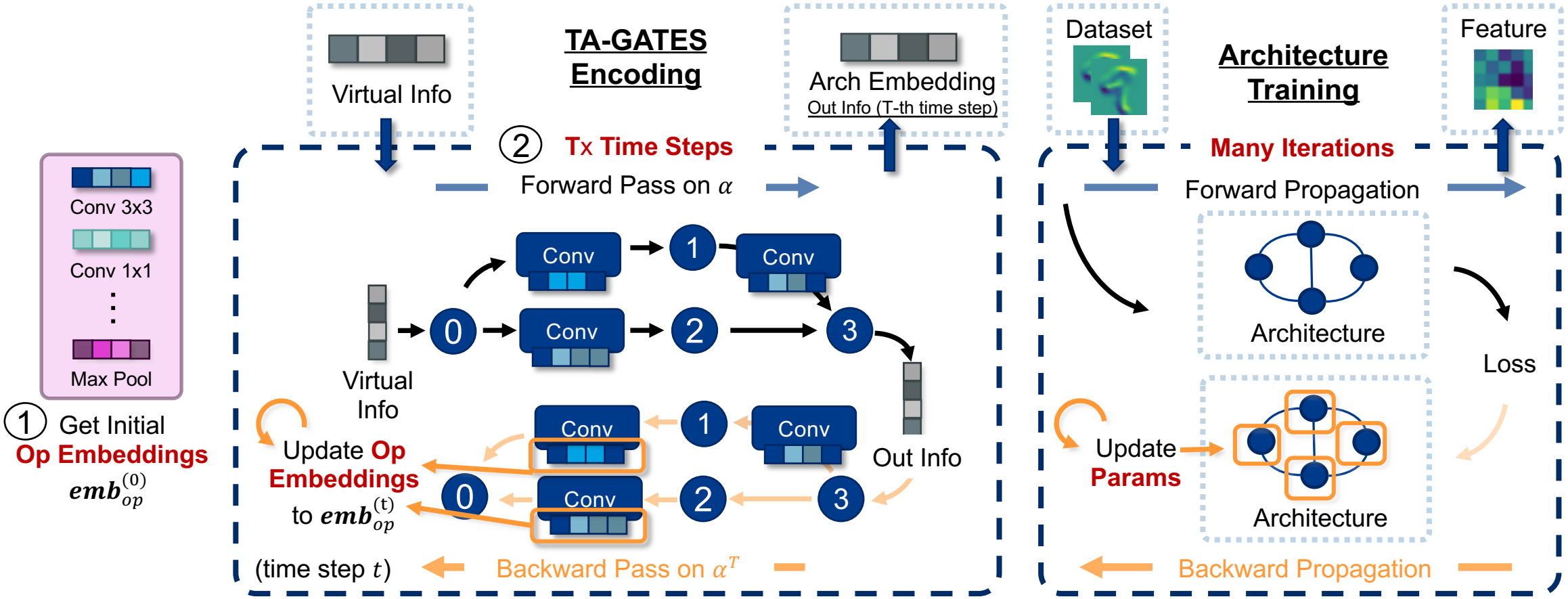
To improve the discriminative modeling of operation and architecture,
should **give contextualized embeddings for operations according to the architectural context**.

[1] Ning et al., A Generic Graph-based Neural Architecture Encoding Scheme for Predictor-based NAS, ECCV'20.
[2] Zhang et al., D-VAE: A Variational Auto-Encoder for Directed Acyclic Graphs, NeurIPS'19.

TA-GATES: "Training-Analogous" Encoding

An NN architecture not only describes what the forward computation semantics are (GATES[1] bases its design on this intrinsic property of NN architecture), but also **determines the NN training dynamics**.

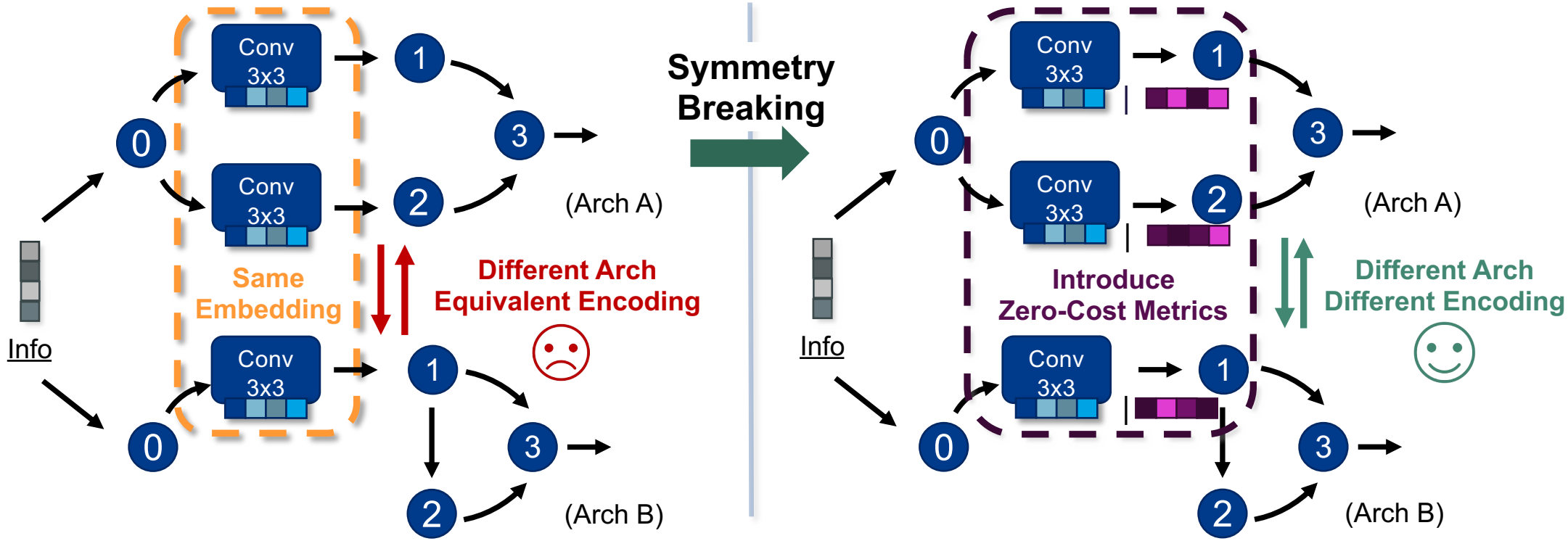
TA-GATES is designed considering this intrinsic property of NN architectures, and encodes architectures in an "**encoding by training-mimicking**" manner. **Naturally provide contextualized operation embeddings**.



[1] Ning et al., A Generic Graph-based Neural Architecture Encoding Scheme for Predictor-based NAS, ECCV'20.

TA-GATES: “Training-Analogous” Symmetry Breaking

Considering a special case that existing “information-based encoders” cannot discriminate…



Why these two operations are not equivalent in NN training? Random parameter initialization breaks the symmetry of these two convolutions when training begins.

Analogously, let’s break the symmetry of the initial operation embeddings $\boldsymbol{emb}_{op}^{(0)}$ by adding zero-cost saliency metrics!

TA-GATES: Empowering The Anytime Prediction Task

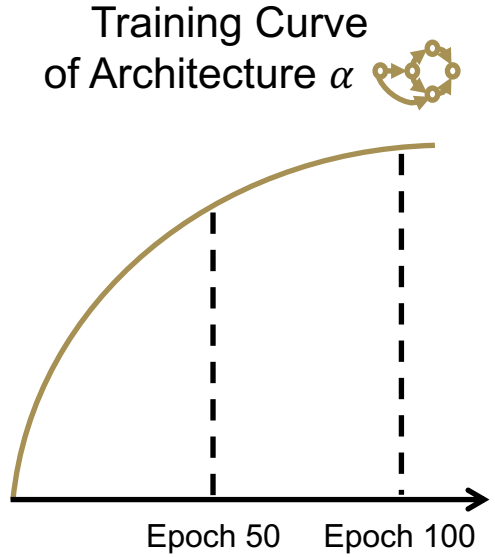
TA-GATES' correspondence with the actual training process enables it to conduct anytime predictions better!

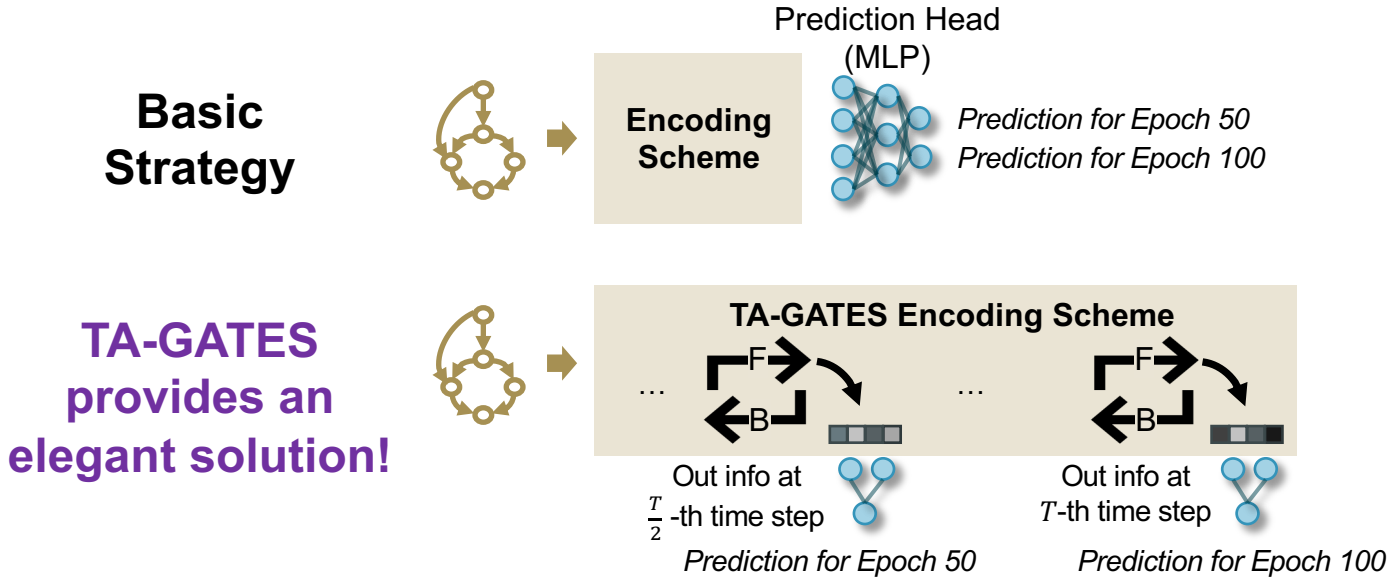
What is Anytime Training and Prediction?

- Training: Using multiple-epoch performances as supervisory signals to train the predictor
- Prediction: Predict the performances at multiple epochs for unseen architectures

Why?

- Training: Potential to improve the prediction of final performances, since more information is used
- Prediction: Providing inspections into the learning dynamics / making surrogate benchmarks[1]



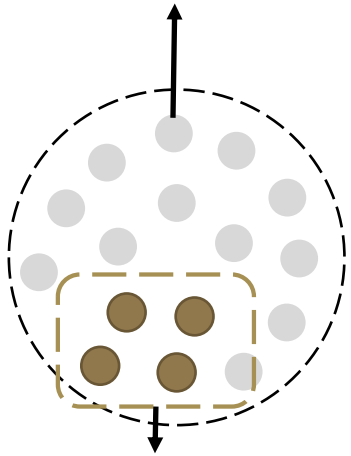


[1] Yan et al.,NAS-Bench-X11 and the Power of Learning Curves, NeurIPS'21.

Results: Comparison with Baseline Encoders

Spaces: NB101, NB201, NB301, NDS ENAS
Measures: Kendall's Tau, Precision@K, Mean Square Error (MSE), Pearson coefficient of LC

2. Predict the performances of unseen architectures, measure how close the predictions are to the GT performances (ranking and regression quality)



1. Train the encoder using the GT performances of some architectures

Kendall's Tau Comparison Example

	Encoder	Proportions of 7290 training samples				
		1%	5%	10%	50%	100%
NB101	MLP [42]	0.3937 (0.0302)	0.5318 (0.0185)	0.5703 (0.0167)	0.6225 (0.0078)	0.6307 (0.0069)
	LSTM [42]	0.5476 (0.0341)	0.5876 (0.0245)	0.6040 (0.0154)	0.6196 (0.0142)	0.6131 (0.0185)
	GCN (global node) [35]	0.3668 (0.0563)	0.5973 (0.0233)	0.6927 (0.0108)	0.7520 (0.0075)	0.7689 (0.0083)
	NASBOWL [33]	0.5850 (0.0232)	0.6416 (0.0241)	0.6536 (0.0193)	0.6833 (0.0022)	0.6872 (0.0000)
	SemiNAS [22]	0.5273 (0.0589)	0.6055 (0.0294)	0.5953 (0.0279)	0.6040 (0.0284)	0.6043 (0.0179)
	XGBoost [44]	0.4517 (0.0470)	0.5987 (0.0365)	0.5680 (0.0125)	0.5677 (0.0077)	0.6175 (0.0000)
	GATES [27]	0.6321 (0.0251)	0.7493 (0.0166)	0.7690 (0.0077)	0.7999 (0.0071)	0.8119 (0.0071)
	TA-GATES	**0.6686** (0.0338)	**0.7744** (0.0211)	**0.7839** (0.0063)	**0.8133** (0.0053)	**0.8217** (0.0057)
	Encoder	Proportions of 7813 training samples				
		0.1%	0.5%	1%	5%	10%
NB201	MLP [42]	0.0162 (0.0859)	0.0863 (0.0556)	0.1756 (0.0332)	0.3885 (0.0237)	0.5492 (0.0092)
	LSTM [42]	0.1935 (0.1806)	0.5079 (0.0715)	0.5691 (0.0110)	0.6690 (0.0189)	0.7395 0.0061)
	Line-Graph GCN [35]	0.2461 0.1549)	0.3113 0.0626)	0.4080 0.0369)	0.5461 0.0138)	0.6095 0.0164)
	NASBOWL [33]	0.4980 (0.0408)	0.6674 (0.0077)	0.5912 (0.0874)	0.7259 ((0.0098)	0.7625 (0.0083)
	XGBoost [44]	0.0706 (0.1238)	0.3719 (0.0560)	0.4178 (0.0288)	0.6412 (0.0053)	0.7084 (0.0123)
	GATES [27]	0.4309 0.1062)	0.6702 0.0254)	0.7571 0.0169)	0.8583 0.0019)	0.8823 0.0024)
	TA-GATES	**0.5382** (0.0478)	**0.6707** (0.0256)	**0.7731** (0.0249)	**0.8660** (0.0060)	**0.8890** (0.0049)

Trained with the same architecture-performance pairs, TA-GATES consistently outperforms other encoders

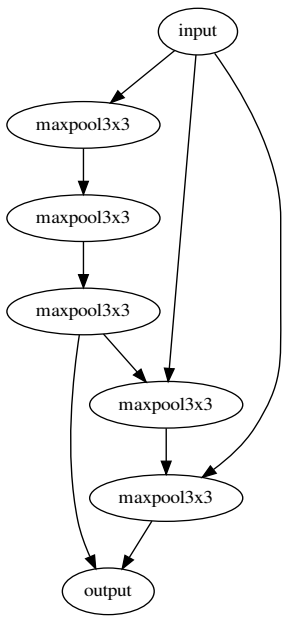
Results: Anytime Prediction

TA-GATES can offer better anytime predictions.

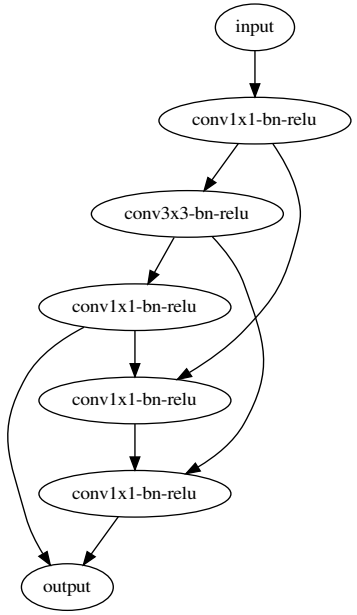
- TA-GATES' natural fit with the actual training process makes it easier to capture the learning speed of arch., thus giving correct relative order for different checkpoints (final and half).
- Baseline encoders using the "Multi-" strategy tend to give the same relative order for the half and final acc.

Single-: Output one score; Trained with one perf. only
Multi-: Output multiple scores; Trained with multiple supervisory signals
TA-GATES: Output multiple scores at different time steps; Trained with multiple supervisory signals

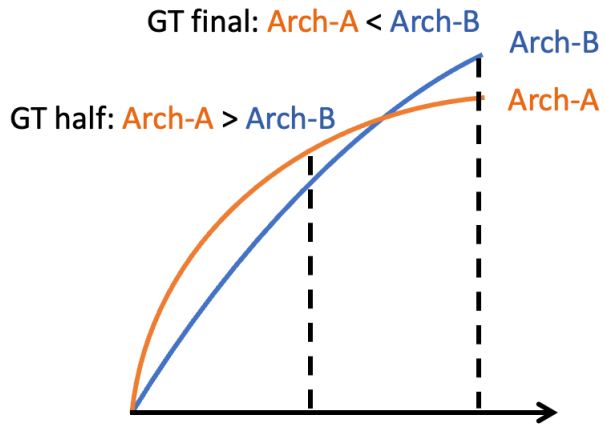
KD with the half accuracy		NB101 (7290 training)			
Encoder	Training	1%	5%	10%	50%
Single-GATES	half	0.3636	0.3473	0.3147	0.4796
Multi-LSTM	half+final	0.0123	0.0659	0.0723	0.0518
Multi-GATES	half+final	0.2862	0.2912	0.2883	0.1413
TA-GATES	half+final	**0.3921**	**0.4615**	**0.4805**	**0.5674**
KD with the final accuracy		NB101 (7290 training)			
Encoder	Training	1%	5%	10%	50%
Single-GATES	final	0.3856	0.3820	0.5034	0.5903
Multi-LSTM	half+final	-0.0372	0.1028	0.2191	0.1473
Multi-GATES	half+final	0.3455	0.3341	0.3370	0.1818
TA-GATES	half+final	**0.5463**	**0.5850**	**0.5950**	**0.6477**



Arch-A



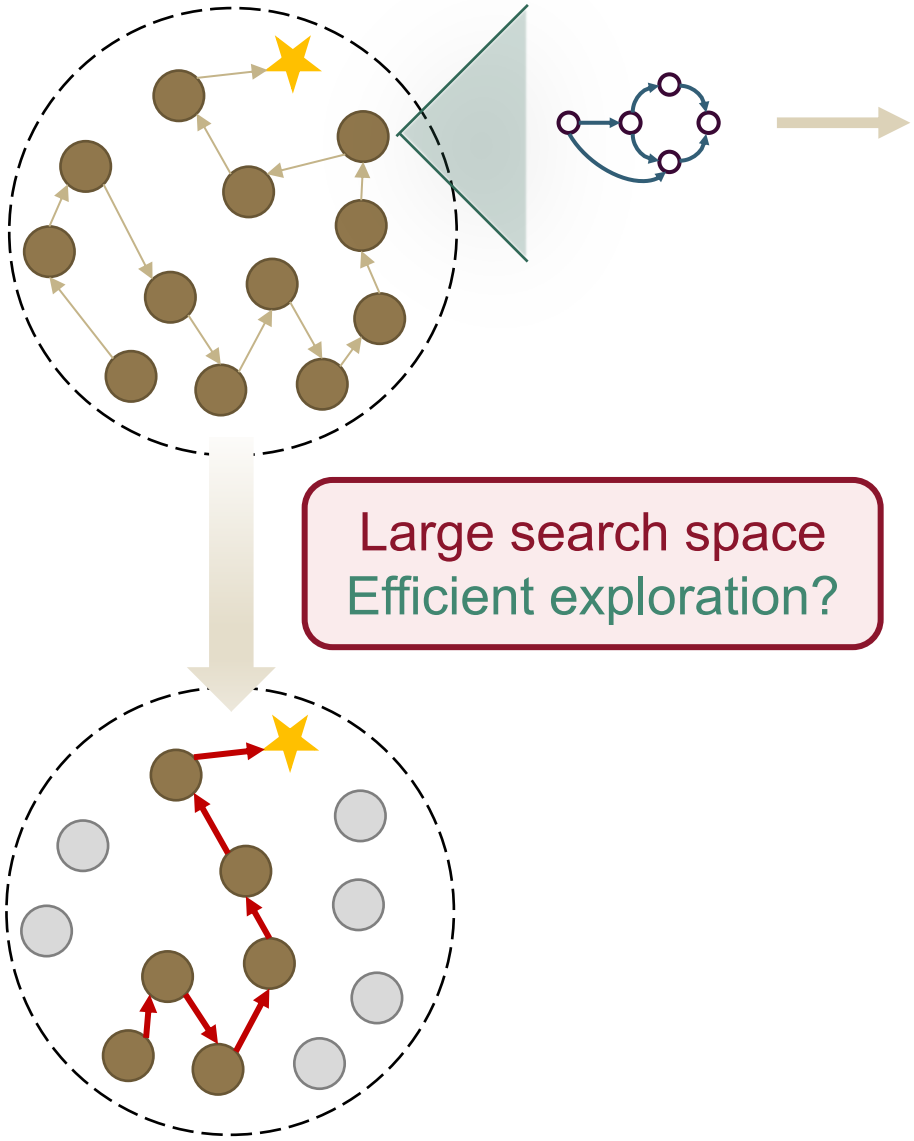
Arch-B



Conclusion and Future Work

- **A good encoding of neural architectures is useful**. Can be used for 1. NAS acceleration (sample efficiency improvement), 2. better parameter-sharing evaluation, 3. interpretation, and so on.
- To develop the most generalizable encoding, we should **identify the distinguishing property** of the neural architectures and **design the encoding scheme accordingly**.
- Neural architectures are **DAGs with trainable operations**. An NN architecture depicts **the forward computation semantics** and **the training dynamics**.
  - GATES mimics the forward process to encode an architecture.
  - TA-GATES further mimics the training process to encode an architecture.
- Besides having better predictive power, "Training-Analogous" brings other possibilities!
  - Better anytime performance training and prediction.
  - (future) TA-GATES as a learnable extrapolator of partial learning curves.
  - (future) TA-GATES as the joint encoder for other factors in AutoDL (e.g., training-time architecture, HPO, AutoAug).

Summary of Our Solution for Efficient NAS



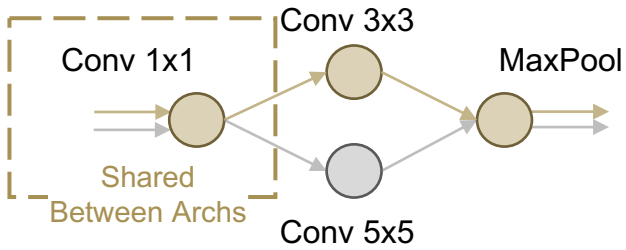
Large search space
Efficient exploration?

Evaluation: How to get the parameters

Standalone Training

Training on Train Set → Standalone-Trained Parameters → Low Efficiency

Parameter-Sharing Training

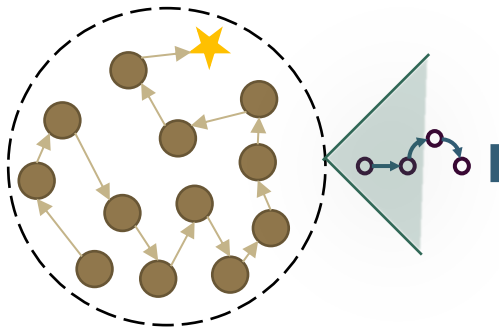


Low Quality

Cannot guarantee efficiency and quality in the meantime
How to improve the quality of parameter-sharing evaluation?

Summary of Our Solution for Efficient NAS

Utilize the learnable encoding of architecture to accelerate exploration and improve the quality of the evaluation, thus enabling efficient NAS



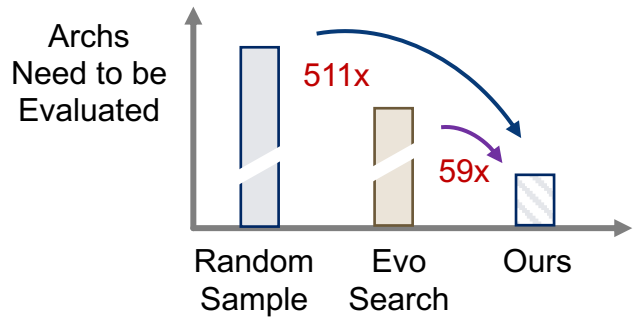
Architecture Encoder[1,2]

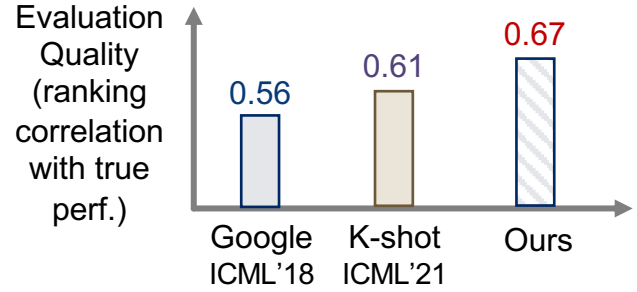
Architecture-level Encoding

Accelerate Exploration[1,4]
Which architectures are promising and need explore

Operation-level Encoding

Improve Evaluation[3]
Which operations to share parameters





[1] Ning et al., A Generic Graph-based Neural Architecture Encoding Scheme for Predictor-based NAS, ECCV'20.
[2] Ning, Zhou et al., TA-GATES: An Encoding Scheme for Neural Network Architectures, NeurIPS'22.
[3] Zhou, Ning et al., CLOSE: Curriculum Learning On the Sharing Extent Towards Better One-shot NAS, ECCV'22.
[4] Zhao, Ning et al., Dynamic Ensemble of Low-fidelity Experts: Mitigating NAS "Cold-Start", AAAI'23.

Thanks for Listening!

- Check our website introducing NAS and summarizing our work at https://sites.google.com/view/nas-nicsefc
- Check the code at https://github.com/walkerning/aw_nas (soon available)
- Contact us at
  - **foxdoraame@gmail.com** (Xuefei Ning)
  - **yu-wang@tsinghua.edu.cn** (Yu Wang)