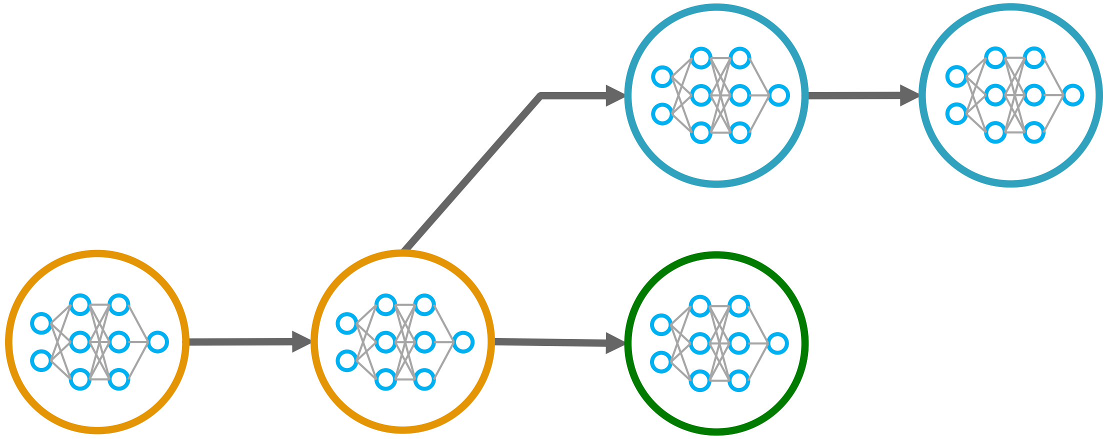
University of St.Gallen



Hyper-Representations as Generative Models: Sampling Unseen Neural Network Weights

Konstantin Schürholt[1], Boris Knyazev[2], Xavier Giró-i-Nieto[3], Damian Borth[1]

[1] AI:ML Lab, School of Computer Science, University of St. Gallen
[2] Samsung - SAIT AI Lab, Montreal
[3] Institut de Robòtica i Informàtica Industrial, Universitat Politècnica de Catalunya

Introduction

Learning from populations of Neural Network models is an emerging topic.

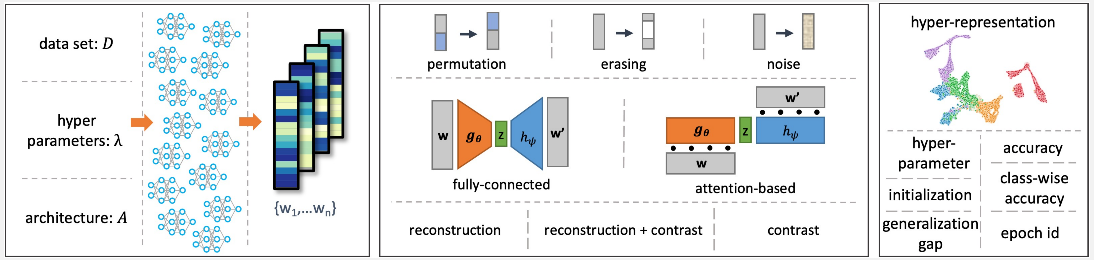
Discriminative: predict model properties

- Predict: accuracy, generalization gap, hyperparameters
- Features: weights *[Unterthiner et al., 2020; Martin et al., 2021]*, activations *[Jiang et al., 2019]*, graph-metrics *[Corneanu et al., 2020]*

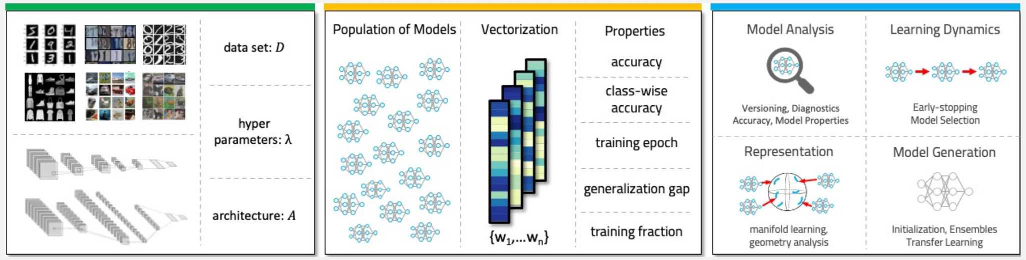
Generative: generate new models

- HyperNetworks *[Ha et al.. 2016; Deutsch, 2018; Zhang et al., 2020; Knyazev et al., 2021; Zhmoginov et al., 2022; Ratzlaff and Fuxin, 2019.]*
- Transfer Learning, Knowledge Distillation *[Shu et al., 2021; Liu et al., 2019.]*

Hyper-Representations: SSL representations of NN weights *[Schürholt et al., NeurIPS 2021]*



Model Zoos: Dataset of Diverse NN populations *[Schürholt et al., NeurIPS 2022]*



This work: Generative Hyper-Representations

Goal:
- Better initializations for fine-tuning and transfer-learning
- Knowledge distillation from populations
- Generate diverse ensembles

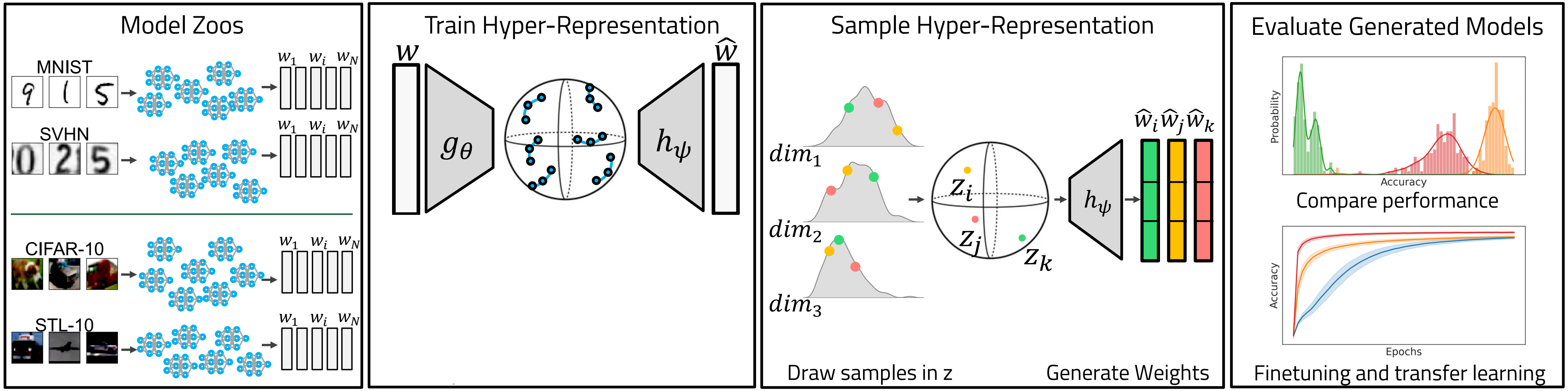
Approach

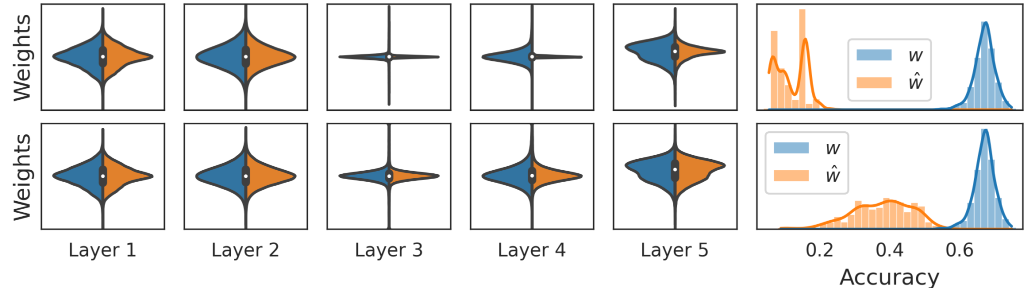
Zoo Generation Details
- Small CNNs: 3 conv, 2 FC layers
- ~2500 parameters
- 1000 models, trained for 25 epochs
- Initialized with different random seeds

Hyper-Representation Details
- Encoder-Decoder Transformer
- Trained with Reconstruction and Contrast

Sampling Details
- Properties like accuracy are embedded in latent
- Problem: space is relatively high-dimensional
- We propose 3 methods to sample good models

Evaluation Details
- Use sampled models as initialization:
  - finetuning in-distribution
  - transfer learning
  - generating diverse ensembles

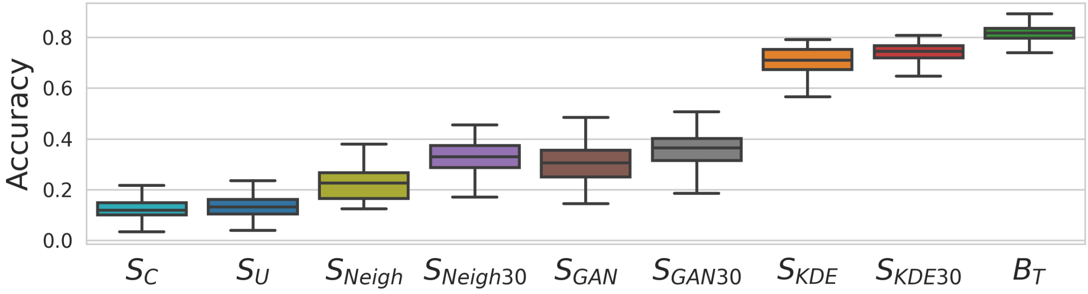




$$\mathcal{L}_{MSE} = \frac{1}{MN}\sum_{i=1}^{M}\sum_{l=1}^{L}\left\|\frac{\hat{\mathbf{w}}_i^{(l)} - \mu_l}{\sigma_l} - \frac{\mathbf{w}_i^{(l)} - \mu_l}{\sigma_l}\right\|_2^2 = \frac{1}{MN}\sum_{i=1}^{M}\sum_{l=1}^{L}\frac{\|\hat{\mathbf{w}}_i^{(l)} - \mathbf{w}_i^{(l)}\|_2^2}{\sigma_l^2}.$$

Sampling initializations

Sampling methods are targeted: distinguish high / low accuracy



Sampled populations are better than (or comparable to) baselines:

- As initialization
- In finetuning (often after 1 ep better than 25 ep trained from scratch)

Method	Ep.	MNIST	SVHN	CIFAR-10	STL-10
B_T	0	≈10% (random guessing)			
B_{KDE30}	0	63.2 ± 7.2	10.1 ± 3.2	15.5 ± 3.4	12.7 ± 3.4
S_{KDE30}	0	**68.6 ± 6.7**	**51.5 ± 5.9**	**26.9 ± 4.9**	**19.7 ± 2.1**
B_T	1	20.6 ± 1.6	19.4 ± 0.6	27.5 ± 2.1	15.4 ± 1.8
B_{KDE30}	1	83.2 ± 1.2	67.4 ± 2.0	39.7 ± 0.6	**26.4 ± 1.6**
S_{KDE30}	1	**83.7 ± 1.3**	**69.9 ± 1.6**	**44.0 ± 0.5**	25.9 ± 1.6
B_T	25	83.3 ± 2.6	66.7 ± 8.5	46.1 ± 1.3	35.0 ± 1.3
B_{KDE30}	25	**93.2 ± 0.6**	**75.4 ± 0.9**	48.1 ± 0.6	**38.4 ± 0.9**
S_{KDE30}	25	93.0 ± 0.7	74.2 ± 1.4	**48.6 ± 0.5**	38.1 ± 1.1
B_T	50	91.1 ± 2.6	70.7 ± 8.8	48.7 ± 1.4	39.0 ± 1.0

Sampling for New Tasks and Architectures

Sampled populations outperform or match baselines for transfer-learning

Method	**SVHN to MNIST**		
	Ep. 0	**Ep. 1**	**Ep. 50**
B_T	10.0 ± 0.6	20.6 ± 1.6	91.1 ± 1.0
B_F	**33.4 ± 5.4**	84.4 ± 7.4	95.0 ± 0.8
S_{KDE30}	31.8 ± 5.6	**86.9 ± 1.4**	**95.5 ± 0.4**

Sampled weights generalize to changed architectures and outperform random initialization

Initialization	**Epoch 1**	**Epoch 5**	**Epoch 50**
3-conv (r. i.) + res-skip (r. i.)	18.9 ± 1.6	31.4 ± 17	50.6 ± 28
3-conv (gen.) + res-skip (r. i.)	**34.5 ± 14**	**60.5 ± 21**	**68.0 ± 21**
4-conv (r. i.)	19.2 ± 1.0	19.2 ± 0.9	55.2 ± 11
4-conv (gen.)	**44.0 ± 4.5**	**57.8 ± 3.5**	**67.6 ± 1.9**
4-conv + id.-skip (r. i.)	18.9 ± 1.0	19.6 ± 1.7	56.4 ± 7.9
4-conv + id.-skip (gen.)	**48.0 ± 4.0**	**59.9 ± 2.5**	**66.4 ± 1.7**

Acknowledgements

Find our work at **hsg.ai/neurips22**

Funding:

- Google Research Scholar Award (Damian Borth)
- HSG Basic Research Fund
- MCIN/ AEI /10.13039/501100011033

References

[1] Thomas Unterthiner, Daniel Keysers, Sylvain Gelly, Olivier Bousquet, and Ilya Tolstikhin. Predicting Neural Network Accuracy from Weights. arXiv:2002.11448, February 2020.

[2] Charles H Martin, Tongsu Serena Peng, and Michael W Mahoney. Predicting trends in the quality of state-of-the-art neural networks without access to training or testing data. Nature Communications, 12(1):1–13, 2021.

[3] Yiding Jiang, Dilip Krishnan, Hossein Mobahi, Samy Bengio. Predicting the Generalization Gap in Deep Networks with Margin Distributions. ICLR 2019.

[4] Ciprian Corneanu, Meysam Madadi, Sergio Escalera, Aleix Martinez. Computing the Testing Error Without a Testing Set. CVPR 2020.

[5] David Ha, Andrew Dai, and Quoc V. Le. HyperNetworks. In arXiv:1609.09106 [Cs], 2016.

[6] Lior Deutsch. Generating Neural Networks with Neural Networks. arXiv:1801.01952 [cs, stat], April 2018.

[7] Chris Zhang, Mengye Ren, and Raquel Urtasun. Graph HyperNetworks for Neural Architecture Search. arXiv:1810.05749 [cs, stat], December 2020.

[8] Boris Knyazev, Michal Drozdzal, Graham W. Taylor, and Adriana Romero-Soriano. Parameter Prediction for Unseen Deep Architectures. In Conference on Neural Information Processing Systems (NeurIPS), 2021.

[9] Andrey Zhmoginov, Mark Sandler, and Max Vladymyrov. HyperTransformer: Model Generation for Supervised and Semi-Supervised Few-Shot Learning. arXiv:2201.04182 [cs], January 2022.

[10] Neale Ratzlaff and Li Fuxin. HyperGAN: A Generative Model for Diverse, Performant Neural Networks. In Proceedings of the 36th International Conference on Machine Learning, pages 5361–5369. PMLR, May 2019.

[11] Yang Shu, Zhi Kou, Zhangjie Cao, Jianmin Wang, and Mingsheng Long. Zoo-tuning: Adaptive transfer from a zoo of models. In International Conference on Machine Learning, pages 457 9626–9637. PMLR, 2021.

[12] Iou-Jen Liu, Jian Peng, and Alexander G. Schwing. Knowledge Flow: Improve Upon Your Teachers. In International Conference on Learning Representations (ICLR), April 2019.

[13] Konstantin Schürholt, Dimche Kostadinov, and Damian Borth. Self-Supervised Representation Learning on Neural Network Weights for Model Characteristic Prediction. In Conference on Neural Information Processing Systems (NeurIPS), volume 35, 2021.