



EPFL

Overcoming Multi-Model Forgetting

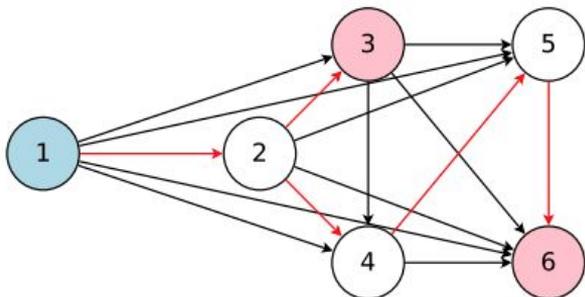
Y. Benyahia, K. Yu, K. Bennani-Smires, M. Jaggi, A. Davison, M. Salzmann, C. Musat



The Weight Sharing

In One of the first NAS papers using Reinforcement Learning, Zoph et Al. (Google) used more than **800 gpus** in parallel for **two weeks**.

Weight Sharing was introduced in NAS to **speed up** the process



Efficient Neural Architecture Search (Pham et al.)



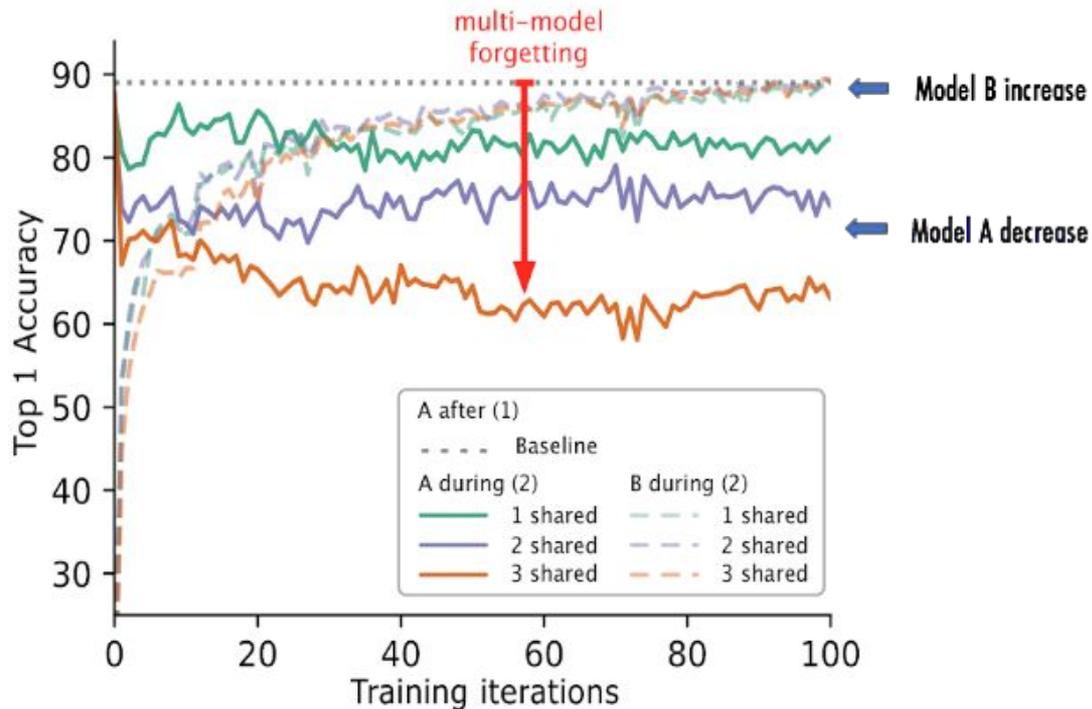
Assumptions

Our hypothesis:

1. Weight-sharing **can negatively affect architectures**.
2. If justified, this can lead to a **wrong evaluation** of candidates in **NAS**, making the evaluation phase **closer to random**



Multi-Model Forgetting





Study of Weight-Sharing

Simple scenario of two models **sharing** parameters:

$$f_1(\mathcal{D}; \theta_1, \theta_s) \text{ and } f_2(\mathcal{D}; \theta_2, \theta_s)$$

Assume that we have access to the optimal parameters $(\hat{\theta}_1, \hat{\theta}_s)$ of the first model $f_1(\mathcal{D}; \theta_1, \theta_s)$

Maximizing the posterior distribution $p(\theta \mid \mathcal{D})$, $\theta = (\theta_1, \theta_2, \theta_s)$

$$\mathcal{L}_{\text{WPL}}(\theta_2, \theta_s) = \mathcal{L}_2(\theta_2, \theta_s) + \frac{\lambda}{2} (\|\theta_s\|^2 + \|\theta_2\|^2) + \frac{\alpha}{2} \sum_{\theta_{s_i} \in \theta_s} F_{\theta_{s_i}} (\theta_{s_i} - \hat{\theta}_{s_i})^2$$

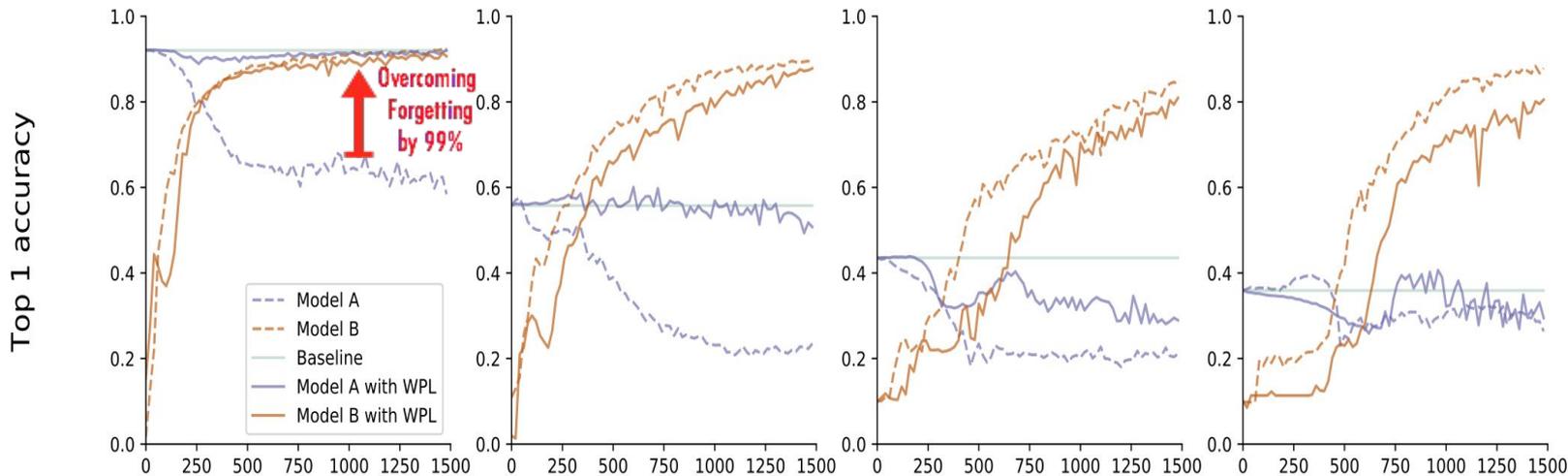
Cross-entropy loss

L2 regularization

Weight importance



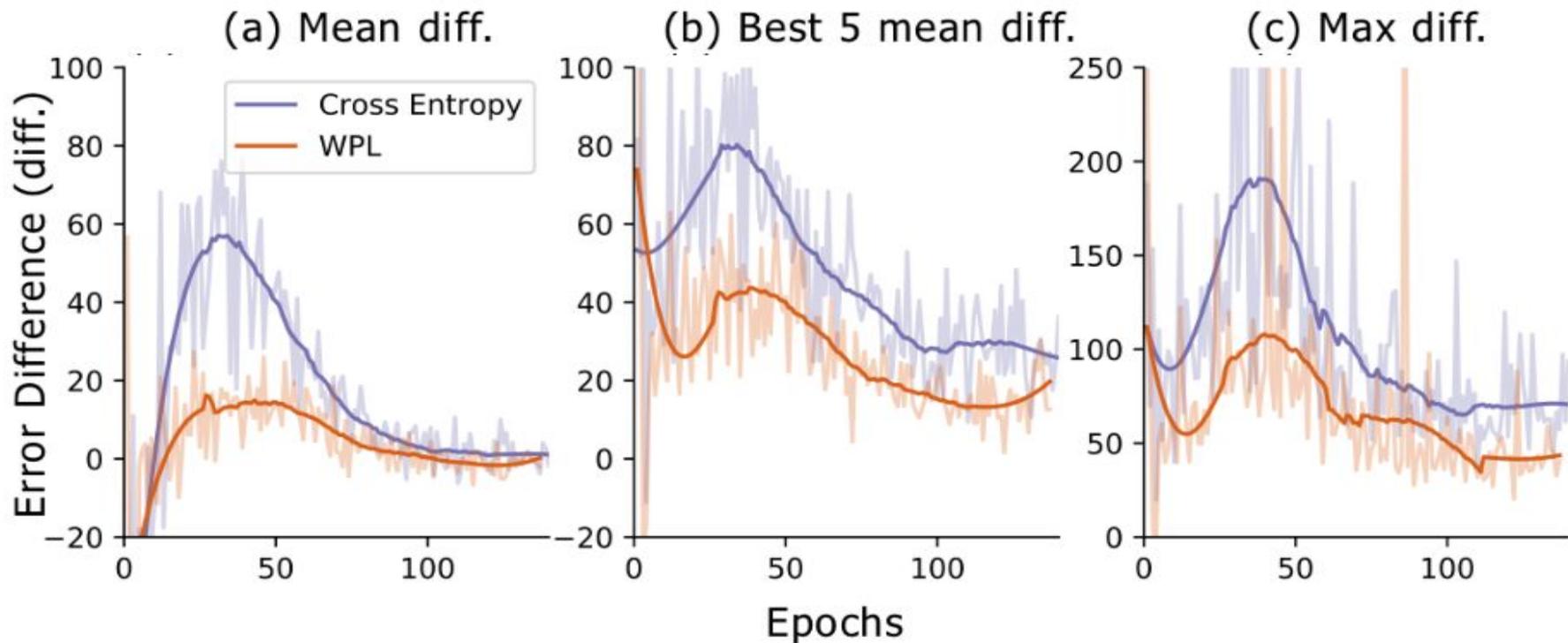
Experiments on Two Models



- **WPL** reduces multi-model forgetting
- **WPL** have a **minimal** effect on the learning of the second model



ENAS on PTB





Summing up

To recap, our main contributions are:

1. **Weight Sharing** negatively impacts NAS
2. **Weight Sharing** can cause the **search phase** in NAS to become closer to random
3. **WPL** reduces Multi-Model Forgetting

Overcoming Multi-Model Forgetting

Yassine Benyahia^{1,2,*}, Kakcheng Yu^{2,3}, Kamil Bernani-Smires¹, Martin Jaggi⁴, Anthony Davison¹, Mathieu Salzmann¹, Claudiu Musat¹
¹ Institute of Mathematics, EPFL - ICM, EPFL - Swisscom Digital Lab, ² Machine Learning and Optimization Lab, EPFL, ³ Digital Switzerland
*Correspondence to: Yassine.Benyahia@epfl.ch, Kakcheng.Yu@epfl.ch

1. Introduction

We identify the problem of Multi-Model Forgetting:

- Performance degrades when several models share their parameters

Weight sharing

- is commonly used in NAS
- but, as we show, has a negative impact on the NAS search phase

To overcome this, we introduce a **Weight Plasticity Loss (WPL)**

- it helps reduce multi-model forgetting

3. Comparison with EWC

Linear elastic system, based on F_1

$$F_1 = \frac{1}{2} \theta^T \Sigma_1 \theta$$

Linear elastic system, based on F_2

$$F_2 = \frac{1}{2} \theta^T \Sigma_2 \theta$$

WPL: Multi-jointly regularized $F(\theta) = \frac{1}{2} \theta^T \Sigma \theta$

$\Sigma = \Sigma_1 + \lambda \Sigma_2$

Multi-green: Σ_1

5. WPL for NAS

WPL reduces the negative impact on sampled architectures up to 95.2%

Dataset: PTB

- Efficient Neural Architecture Search (ENAS)

- Neural Architecture Optimization (NAO)

2. Methodology

We consider two models $f_1(\mathcal{D}; \theta_1, \theta_2)$ and $f_2(\mathcal{D}; \theta_2, \theta_1)$

- Sharing parameters θ_2
- Both models are trained sequentially
- Assume we have access to the optimal parameters (θ_1^*, θ_2^*) after the training of $f_1(\mathcal{D}; \theta_1, \theta_2)$

We maximize the posterior probability $p(\theta | \mathcal{D})$, $\theta = (\theta_1, \theta_2, \theta_1)$

- with the diagonal Fisher information approximation to the Hessian which led to WPL.

$$\mathcal{L}_{\text{WPL}}(\theta_2, \theta_1) = \mathcal{L}_2(\theta_2, \theta_1) + \frac{\lambda}{2} (\|\theta_1\|_2^2 + \|\theta_2\|_2^2) + \frac{\lambda}{2} \sum_{i=1}^n \sum_{c=1}^C F_{\text{WPL}}(\theta_{1c} - \theta_{2c})^2$$

F_{WPL} is the Fisher diagonal element corresponding to θ_{1c} for the first trained model.

- Fisher information encodes the importance of each shared weight for the first model's performance
- WPL preserves shared parameters that were important for the first model

4. WPL for two Models

WPL leads to a 99% reduction in the degradation of the initial model

Dataset: MNIST

From strict to loose convergence

- We relax the assumption of convergence for the first model to different levels \rightarrow green line
- WPL \rightarrow 99.9% multi-model forgetting reduction for (a) (b)
- WPL \rightarrow 2% for (c)
- WPL is highly effective when $f_1(\mathcal{D}; \theta_1, \theta_2)$ is trained to at least 40% optimality

6. Conclusion

- Weight-sharing negatively affects architectures \rightarrow the search phase of NAS is degraded
- WPL reduces the negative impact of weight-sharing

Our code is available at:

Pacific Ballroom #19
(6:30pm - 9pm)