# Towards One Shot Search Space Poisoning in Neural Architecture Search (Student Abstract)

**Nayan Saxena**[1], **Robert Wu , Rohan Jain**
University of Toronto
ML Collective
[1]nayan.saxena@mail.utoronto.ca

### Abstract

We evaluate the robustness of a Neural Architecture Search (NAS) algorithm known as Efficient NAS (ENAS) against data agnostic poisoning attacks on the original search space with carefully designed ineffective operations. We empirically demonstrate how our one shot search space poisoning approach exploits design flaws in the ENAS controller to degrade predictive performance on classification tasks. With just two poisoning operations injected into the search space, we inflate prediction error rates for child networks upto 90% on the CIFAR-10 dataset.

## Introduction

The problem of finding optimal deep learning architectures has recently been automated by neural architecture search (NAS) algorithms. These algorithms continually sample operations from a predefined search space to construct neural networks to optimize a performance metric over time, eventually converging to better child architectures. This intuitive idea greatly reduces human intervention by restricting human bias in architecture engineering to just the selection of the predefined search space (Elsken et al. 2019). While NAS has been studied to further develop more adversarially robust networks through addition of dense connections (Guo et al. 2020), little work has been done in the past to assess the adversarial robustness of NAS itself. Search phase analysis has shown that computationally efficient algorithms such as ENAS are worse at truly ranking child networks due to their reliance on weight sharing (Yu et al. 2019). Finally, most traditional poisoning attacks involve injecting mislabeled examples in the training data and have been executed against classical machine learning approaches (Schwarzschild et al. 2021). We validate these concerns by evaluating the robustness of one such NAS algorithm known as Efficient NAS (ENAS) (Pham et al. 2018) against data-agnostic search space poisoning (SSP) attacks on the CIFAR-10 dataset. Throughout this paper, we focus on the pre-optimized ENAS search space $\hat{\mathcal{S}}$ = {Identity, 3x3 Separable Convolution, 5x5 Separable Convolution, Max Pooling (3x3), Average Pooling (3x3)} (Pham et al. 2018).



## Search Space Poisoning (SSP)

The idea behind SSP, as shown in Figure 1, is to inject precisely designed multiset $\mathcal{P}$ of ineffective operations into the ENAS search space, making the search space $\mathcal{S} := \hat{\mathcal{S}} \cup \mathcal{P}$. Our approach exploits the core functionality of the ENAS controller to sample child networks from a large computational graph of operations by introducing highly ineffective local operations into the search space. On the attacker's behalf, this requires no *a priori* knowledge of the problem domain or dataset being used, making this new approach more favourable than traditional data poisoning attacks.

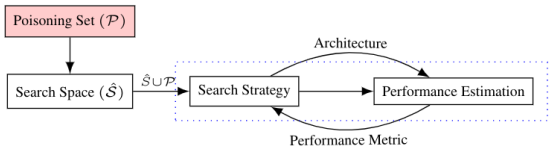


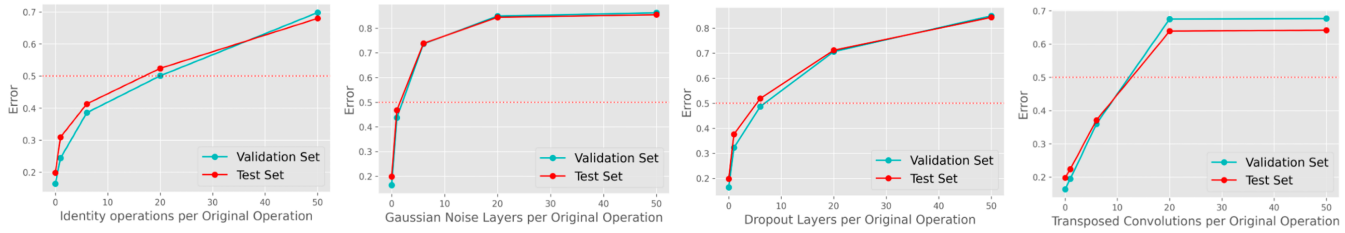

Figure 1: Overview of Search Space Poisoning (SSP)

### Multiple-Instance Poisoning

As a naïve strategy, we first propose multiple-instance poisoning which increases the likelihood of sampling bad operations by including duplicates of these bad operations in the search spaces. Through experimental results we discovered that biasing the search space this way resulted in final networks that are mostly comprised of these poor operations with error rates exceeding 80%. However, as shown in Figure 2, to perform well this approach requires overwhelming the original search space with up to 300 bad operations (50:1 ratio of bad operations per each good operation) which is unreasonable. The motivation then is to reduce the ratio of bad to good operations down to 1:1, or even lower, to make search space poisoning more feasible and effective.

## Towards One Shot Poisoning

In an attempt to improve the attack, we further attempted to reduce the number of poisoning points to just 2 points by adding: (i) Dropout($p = 1$) (ii) Stretched Conv($k = 3$, padding, dilation $= 50$) to the original search space. Our rationale is that dropout operations with $p = 1$ would erase all information and produce catastrophic values such as 0 or




(a) (b) (c) (d)

Figure 2: Final validation and test classification errors as a function of multiple operation instances. (a) Identity layers were moderately effective (b) Gaussian noise reached high error rates even with fewer operations (c) Dropout proved most effective (d) Transposed convolutions plateaued after a saturation point.

not-a-number (NaN). The results were promising, with error rates shooting up to 90% very quickly during training as seen in Figure 3 and Table 1. An example final child network producing these high errors can be observed in Figure 4.

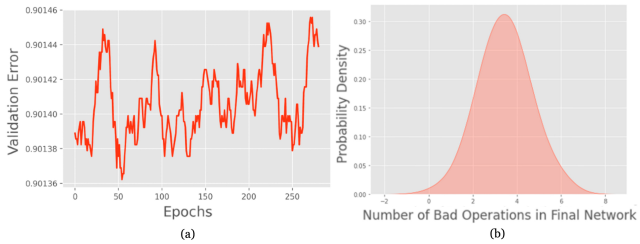



Figure 3: (a) Validation error for one shot poisoning over 300 epochs (b) Distribution of bad operations sampled by the ENAS controller after 300 epochs.

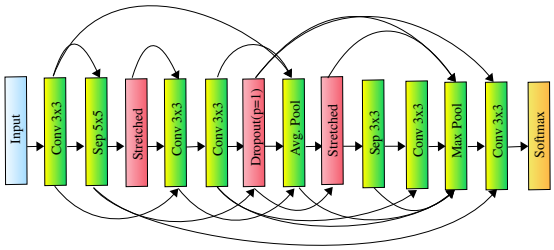



Figure 4: Network produced by ENAS after one shot poisoning with good operations highlighted in green and poisoning operations highlighted in red. Search space utilized is the same as shown in Table 1 with two poisoning points.

## Conclusion

In this paper, we focused on examining the robustness of ENAS under our newly proposed SSP paradigm. Our carefully designed poisoning sets demonstrated the potential to make it incredibly easy for an attacker with no prior knowledge or access to the training data to still drastically impact the quality of child networks. Finally, our one-shot poisoning results reveal an opportunity for future work in neural architecture design, as well as challenges to surmount in using NAS for more adversarially robust architecture search.

| Search Space | $\lvert\mathcal{P}\rvert$ | Val Error | Test Error |
|---|---|---|---|
| $\hat{\mathcal{S}}$ (Baseline) | 0 | 16.4% | 19.8% |
| $\hat{\mathcal{S}} + 300\{\text{Dropout}(1)\}$ | 300 | 84.8% | 84.3% |
| $\hat{\mathcal{S}} + \{\text{Conv}(3,50,50),\ \text{Dropout}(1)\}$ | **2** | **90.1%** | **90.0%** |

Table 1: Experimental results showing how one shot poisoning proves surprisingly effective with just 2 points as compared to its multiple instance counterpart with 300 points.

## Acknowledgements

The authors would like to thank Chuan-Yung Tsai & George-Alexandru Adam for their valuable comments. We thank the ML Collective community for the generous computational support and feedback on this research. We are also grateful to Kanav Singla and Benjamin Zhuo for their contributions to the codebase.

## References

Elsken, T.; Metzen, J. H.; Hutter, F.; et al. 2019. Neural Architecture Search: A Survey. *J. Mach. Learn. Res.*, 20(55): 1–21.

Guo, M.; Yang, Y.; Xu, R.; Liu, Z.; and Lin, D. 2020. When NAS Meets Robustness: In Search of Robust Architectures Against Adversarial Attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 631–640.

Pham, H.; Guan, M.; Zoph, B.; Le, Q.; and Dean, J. 2018. Efficient Neural Architecture Search via Parameters Sharing. In *International Conference on Machine Learning*, 4095–4104. PMLR.

Schwarzschild, A.; Goldblum, M.; Gupta, A.; Dickerson, J. P.; and Goldstein, T. 2021. Just how toxic is data poisoning? a unified benchmark for backdoor and data poisoning attacks. In *International Conference on Machine Learning*, 9389–9398. PMLR.

Yu, K.; Sciuto, C.; Jaggi, M.; Musat, C.; and Salzmann, M. 2019. Evaluating the Search Phase of Neural Architecture Search. *arXiv preprint arXiv:1902.08142*.