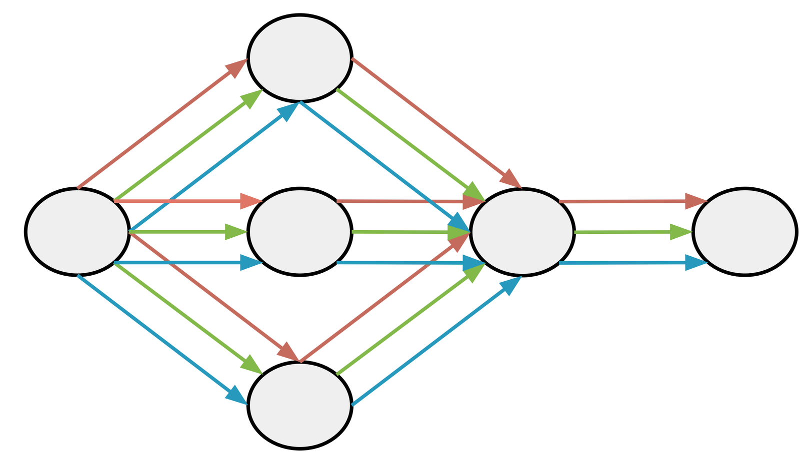
Gradient-based Search for Activation Functions

Sharat Patil, Jake Robertson, Lukas Strack

Supervisor: Mahmoud Safari

Motivation

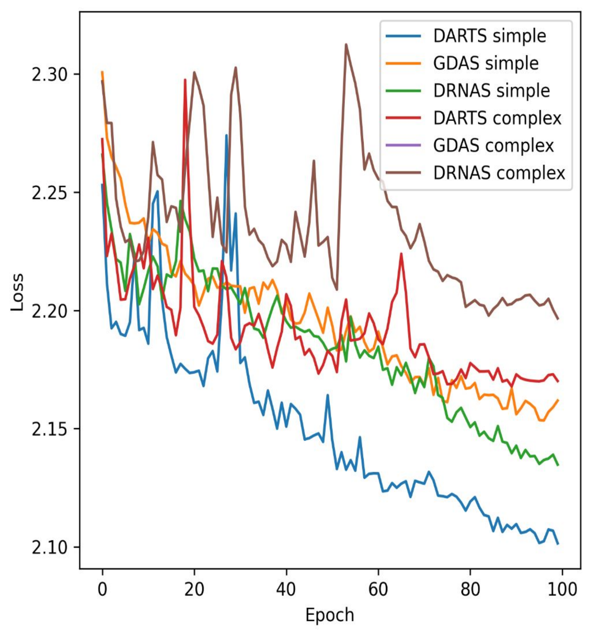
- Activation functions are a crucial aspect of neural architectures, but are relatively underexplored in NAS research
- Previous work uses black-box optimization and thus comes with a high computational cost [1]
- One-shot methods have offered a huge boost in efficiency in NAS research, which represent search spaces as a supergraph of architectures
- **Idea:** We apply Gradient based one-shot methods to search for novel activation functions and evaluate the results with respect to their performance and transferability

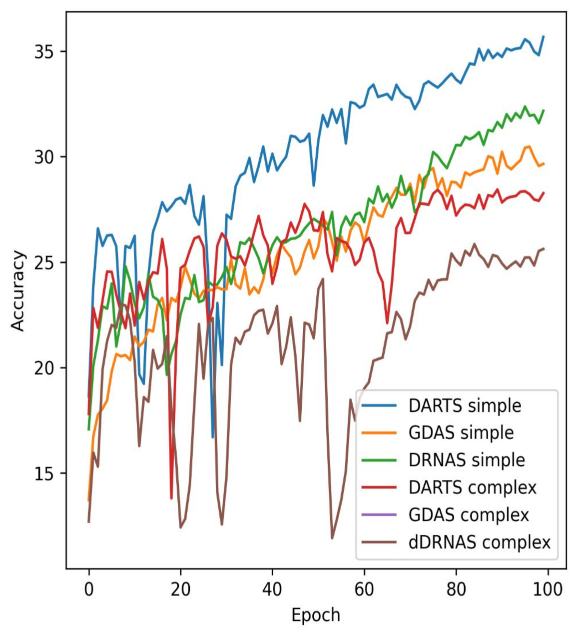


Technical Approach

Experimental Setup

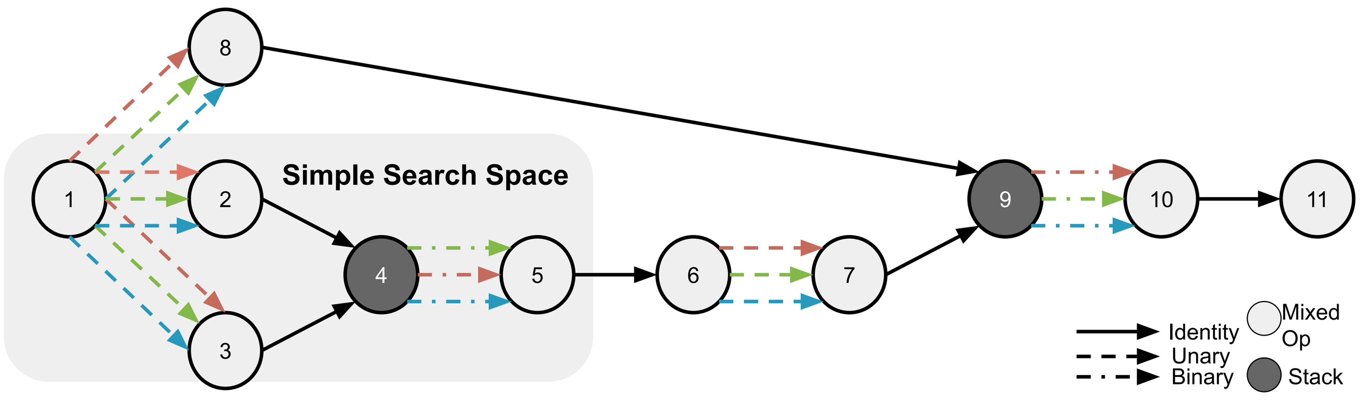
Dataset: CIFAR-10
Optimizers: DARTS[2], GDAS[3], DRNAS[4]
Benchmarks: ReLU, Swish
Macro-architectures: ResNet8, ResNet20
Unary Operators: x, −x, |x|, √x, βx, x + β, log(x), sin(x), cos(x), tanh(x), sinh−1(x), tan−1(x), sinc(x), max(x, 0), min(x, 0), σ(x), β
Binary Operators: x1 + x2, x1 · x2, x1 − x2,max(x1, x2), min(x1, x2), σ(x1) · x2, βx1 + (1 − β)x2
(β is a trainiable parameter)



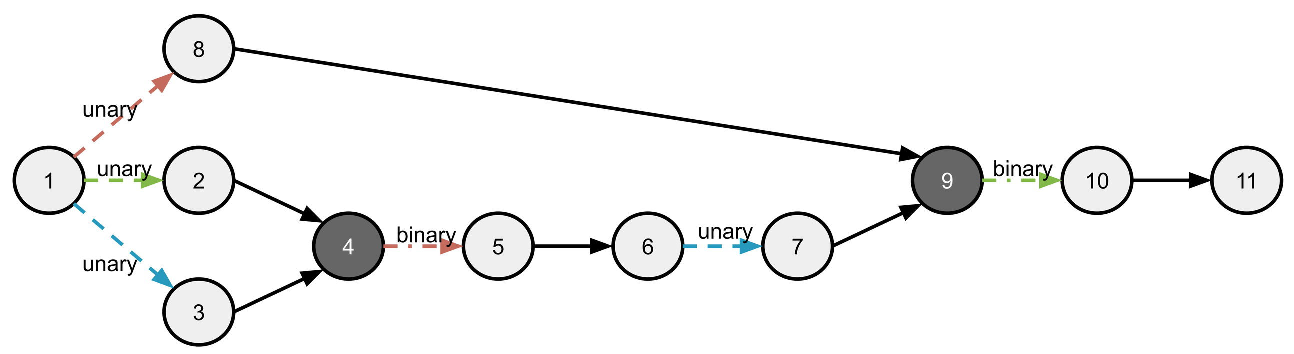


Search Space

- Using NASLib[5], we represent activation functions as cell-search-spaces of unary and binary operations as seen in [5]

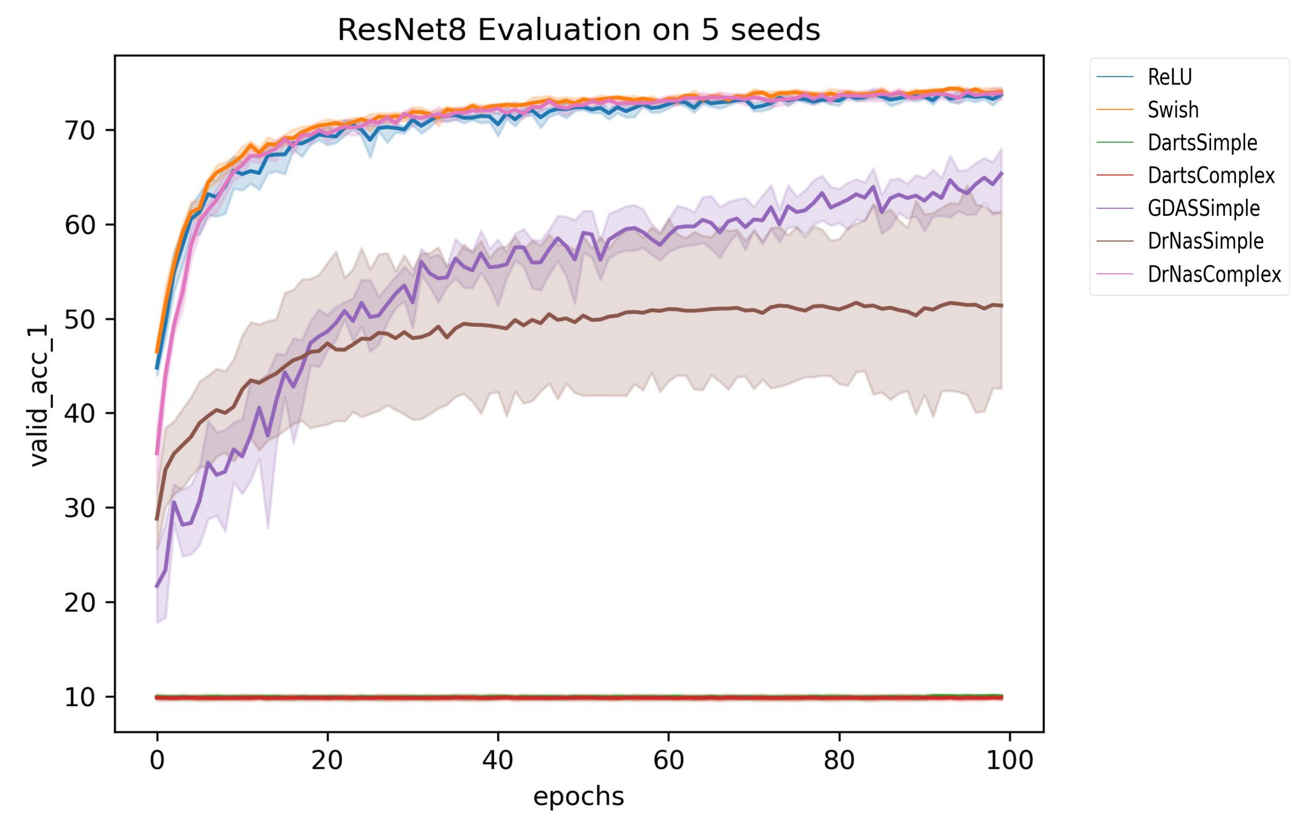


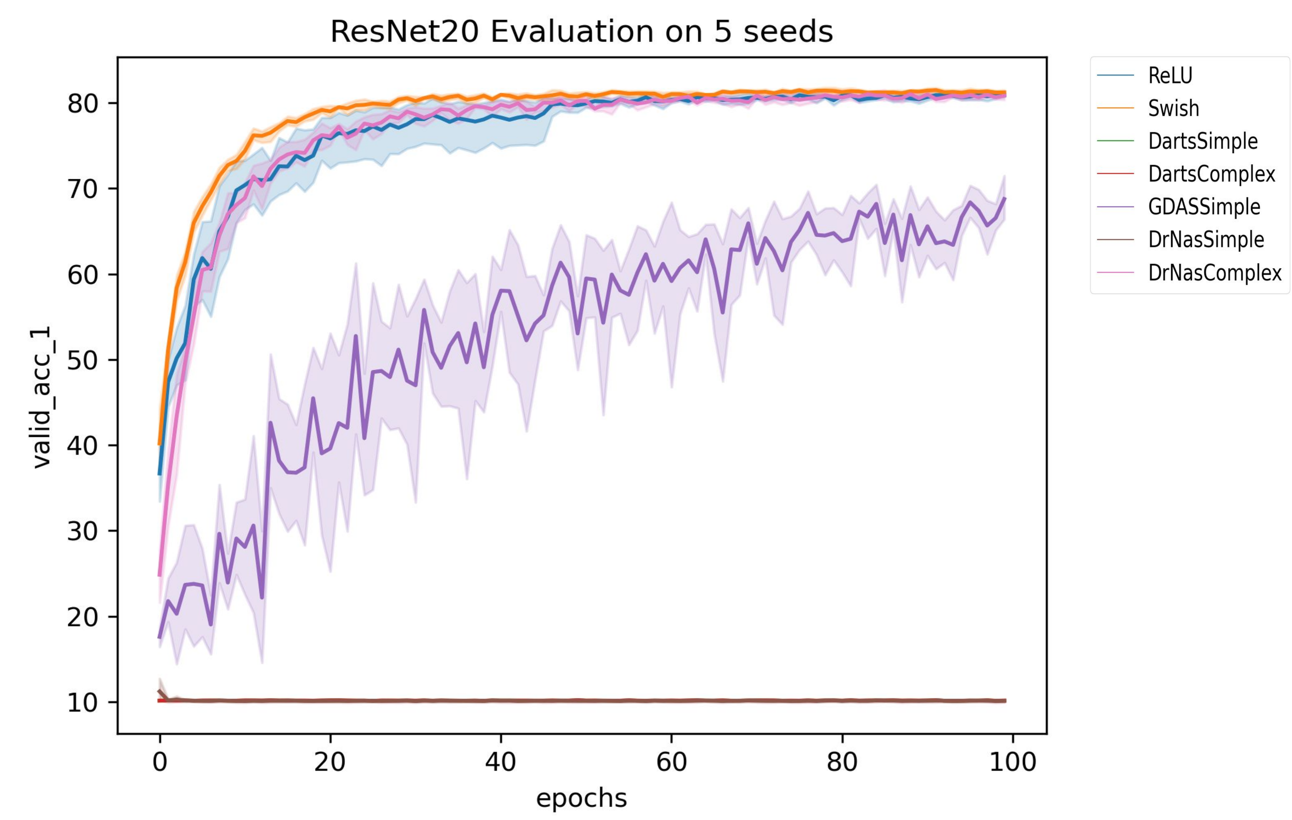
- Then we insert these cells into ResNet architectures and optimize both the network and activation cell weights (alphas)
- Discretized network:

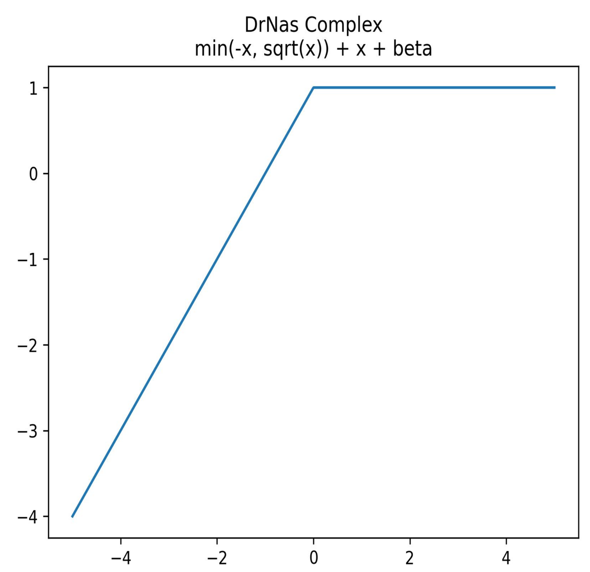


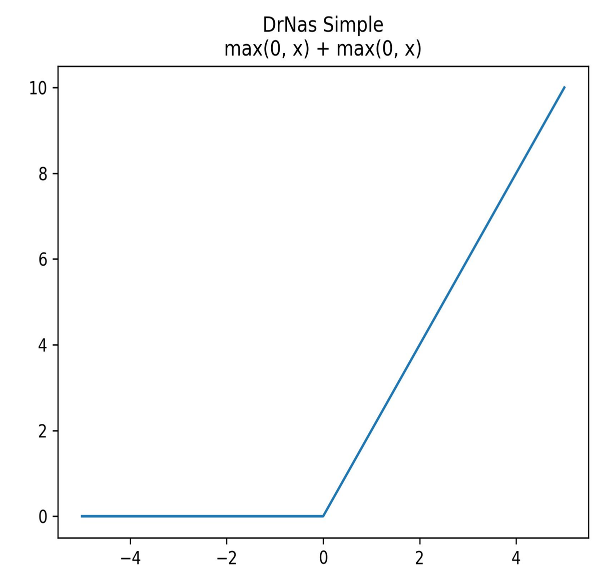
Experimental Results

- DrNas was able to find a function that has comparable performance to ReLU and Swish
- The functions found by DrNas are similar to ReLU (*ReLU(x)*2* and *Min(x, 0) + beta*).
- The training time for the DrNas activation is higher than ReLU and Swish without giving any additional performance increase
- Search loss and accuracy from Search search does not translate to evaluation.
- DARTS searched activations fail to learn in evaluation.
- Transferability across models seems to hold generally but the DrNasSimple activation fails to learn on the ResNet20
- Using operations such as exp(x) or Pow(x) which have large values and gradients along with small operators adversely affects the Oneshot methods.









Activation Function		ResNet8 Test Accuracy		ResNet20 Test Accuracy	
		mean	std	mean	std
DartsComplex	$(\lvert x\rvert^2 + \beta) + \beta x$	50.01	0.05	49.97	0.00
DartsSimple	$(\beta + x)^2$	50.01	0.05	49.97	0.00
DrNasComplex	$min(-x, \sqrt{x}) + \beta + x$	97.76	0.11	98.42	0.15
DrNasSimple	$max(0, x) + max(0, x)$	79.74	8.07	49.98	0.02
GDASSimple	$\sigma(\sigma(x)) \cdot \sigma(x)$	88.58	5.54	94.96	1.13
ReLU	$max(0, x)$	97.88	0.20	98.40	0.14
Swish	$x \cdot \sigma(x)$	97.76	0.14	98.48	0.11

Summary/Conclusion

- The best performing activation is very similar to ReLU.
- There are limitations of DARTS based methods that affect how expressive the search space can be.
- Ensemble behavior of DARTS based methods affects the searched function. After discretization some functions fail to learn.
- Gradient based search will require additional modifications to the algorithm to account for unique behavior of activations and a carefully designed search space to produce novel activation function.

[1] Ramachandran, Prajit, Barret Zoph, and Quoc V. Le. "Searching for activation functions." (2017). **[2]** Liu, Hanxiao, Karen Simonyan, and Yiming Yang. "Darts: Differentiable architecture search." (2018).
[3] Dong, X., & Yang, Y. (2019). Searching for A Robust Neural Architecture in Four GPU Hours (Version 2). **[4]** Chen, X., Wang, R., Cheng, M., Tang, X., & Hsieh, C.-J. (2020). DrNAS: Dirichlet Neural Architecture Search (Version 4).
[5] Ruchte, Michael, et al. "NASLib: a modular and flexible neural architecture search library." (2020).

Code: https://github.com/jr2021/DARTS_activations/