Neural Architecture Search

Кудрявцева Софья

Цель NAS:

Создать алгоритм, который будет выдавать описание нейронной сети для определенной задачи.

- Поле поиска
- Стратегия алгоритма поиска
- Стратегия оценки эффективности поиска

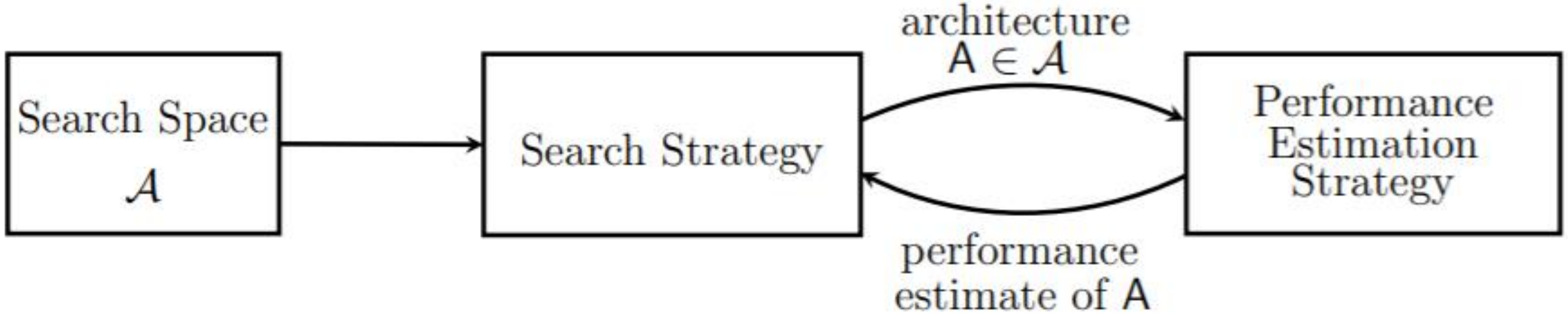
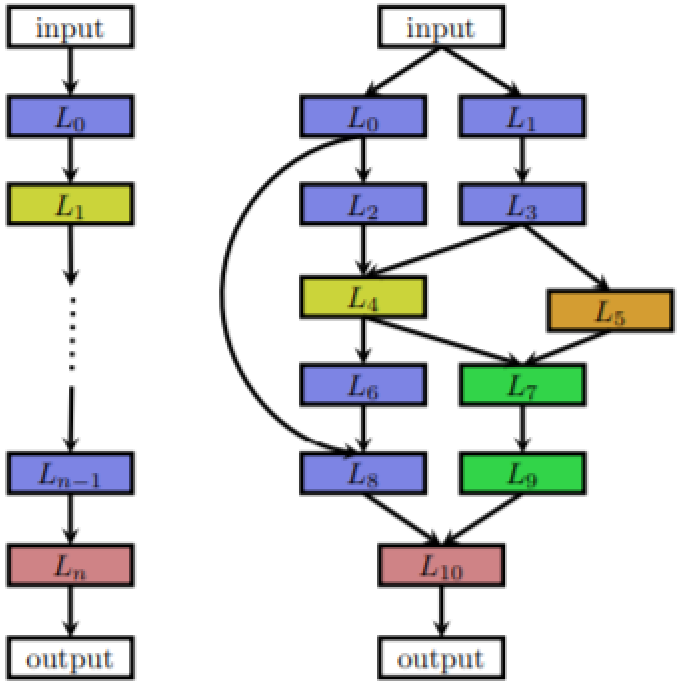
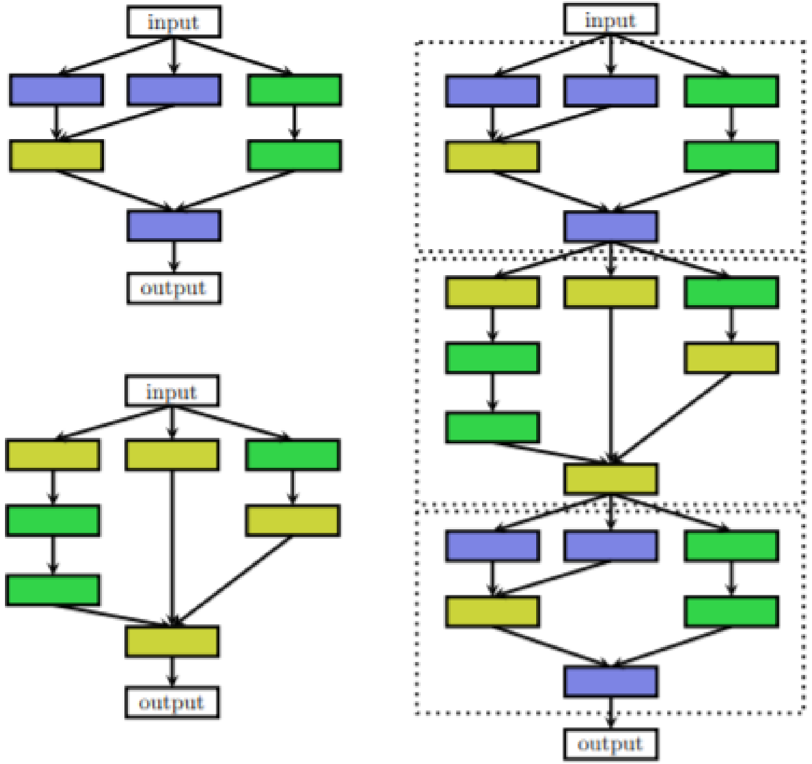


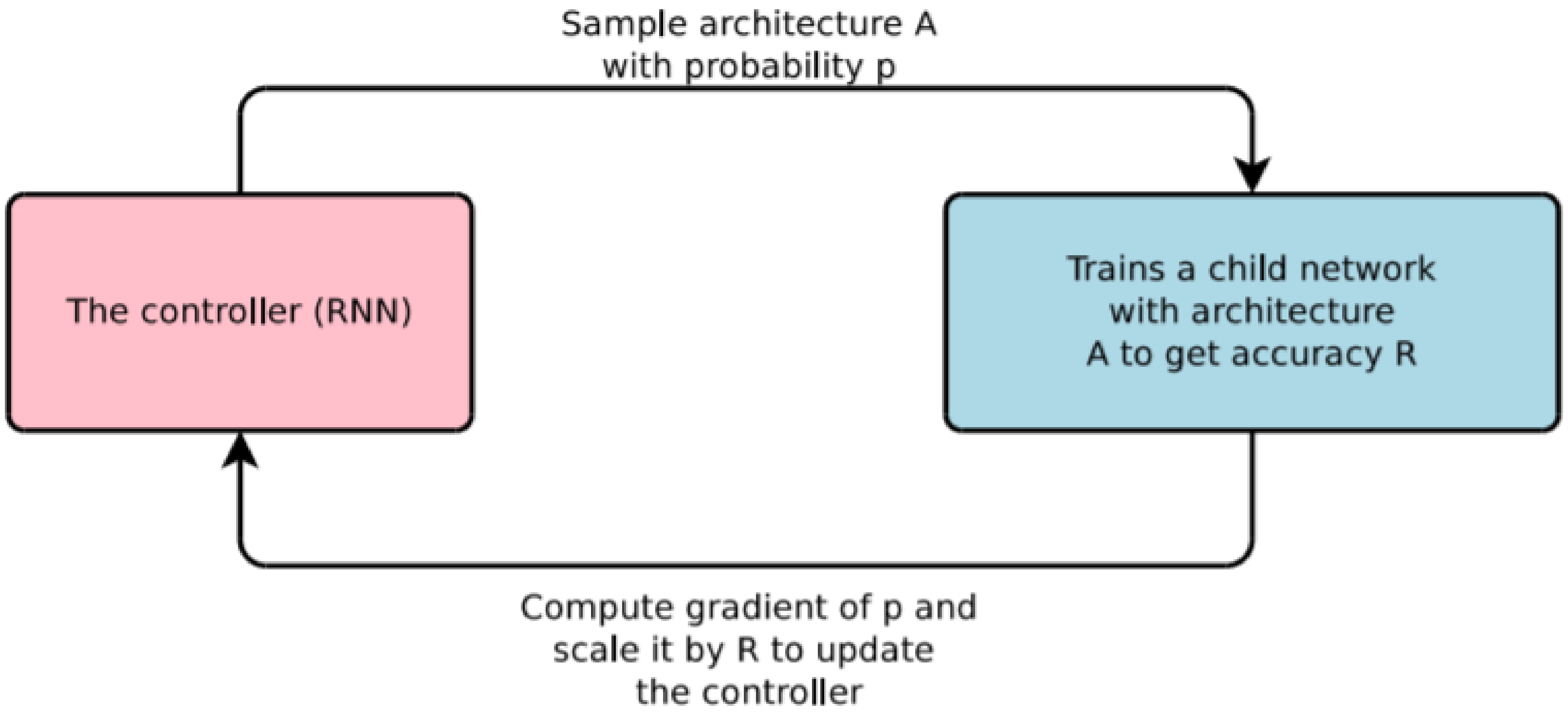
Figure 1: Abstract illustration of Neural Architecture Search methods. A search strategy selects an architecture A from a predefined search space $\mathcal{A}$. The architecture is passed to a performance estimation strategy, which returns the estimated performance of A to the search strategy.

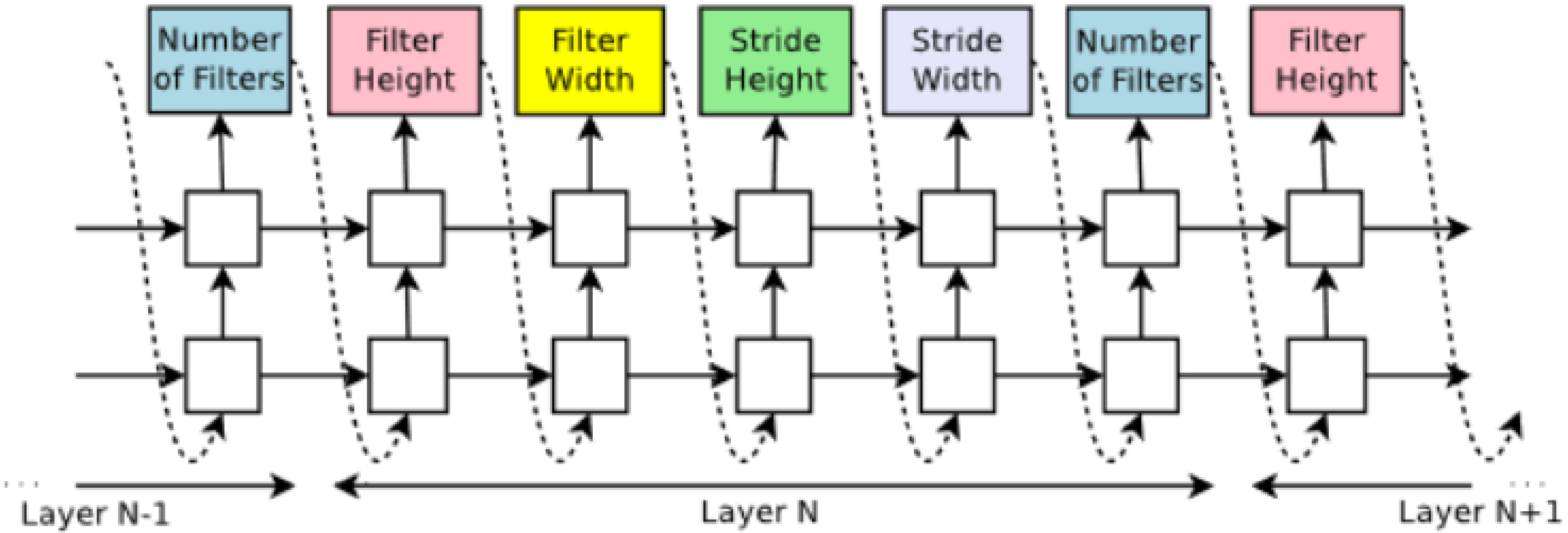
Поле поиска


input
L_0
L_1
L_{n-1}
L_n
output
input
L_0
L_1
L_2
L_3
L_4
L_5
L_6
L_7
L_8
L_9
L_{10}
output


input
output
input
output
input
output

Обучение с подкреплением




Number of Filters
Filter Height
Filter Width
Stride Height
Stride Width
Number of Filters
Filter Height
Layer N-1
Layer N
Layer N+1

Метод REINFORCE

- Контроллер максимизирует ожидаемое вознаграждение

$$J(\theta_c) = E_{P(a_{1:T};\theta_c)}[R]$$

$$\nabla_{\theta_c} J(\theta_c) = \sum_{t=1}^{T} E_{P(a_{1:T};\theta_c)}\big[\nabla_{\theta_c} \log P(a_t|a_{(t-1):1};\theta_c)R\big]$$

- По методу Монте – Карло эмпирически приближаем градиент

$$\frac{1}{m}\sum_{k=1}^{m}\sum_{t=1}^{T}\nabla_{\theta_c}\log P(a_t|a_{(t-1):1};\theta_c)(R_k-b)$$

- m – количество дочерних архитектур в батче, выданном контроллером
- T – количество гиперпараметров дочерней модели
- b – бейзлайн, который добавляется для уменьшения дисперсии оценки градиента. Равен скользящему среднему точности предыдущих дочерних моделей

Skip connections

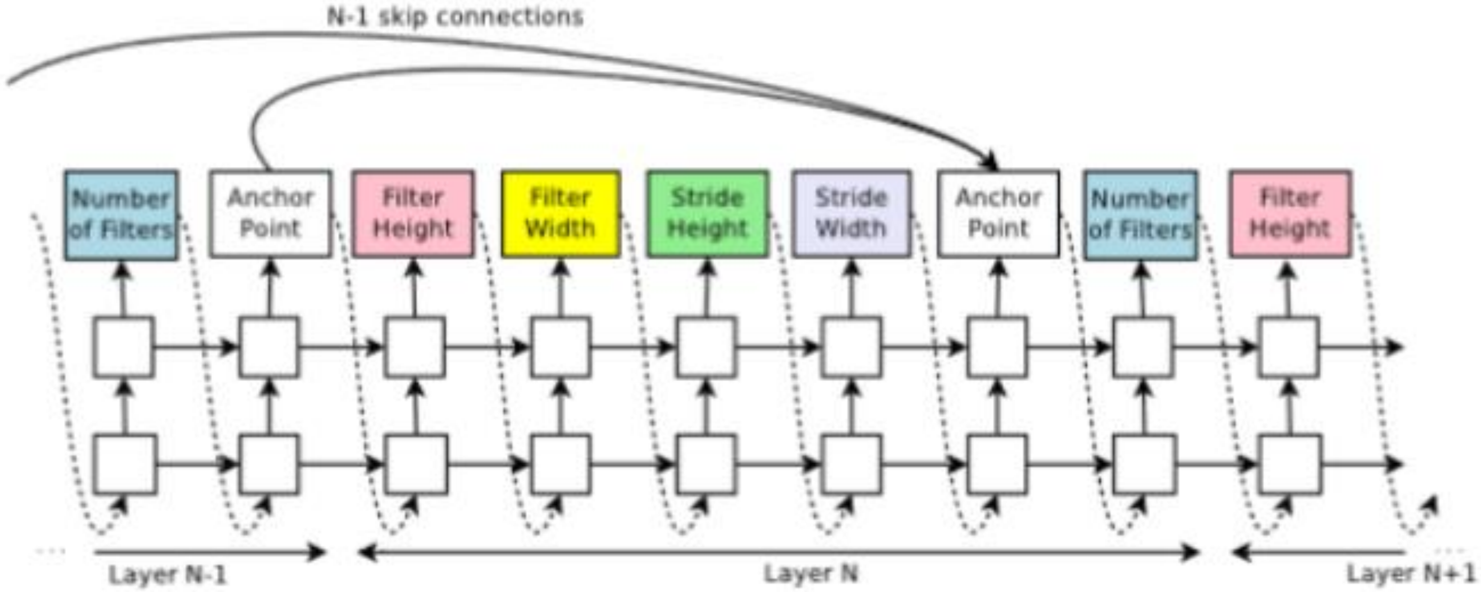


Fig.5: increasing the complexity with anchor points. [source]

$$\mathrm{P}(\text{Layer j is an input to layer i}) = \text{sigmoid}(v^{\mathrm{T}}\tanh(W_{prev} * h_j + W_{curr} * h_i)),$$

- 12,800 архитектур
- 800 GPU
- 28 дней

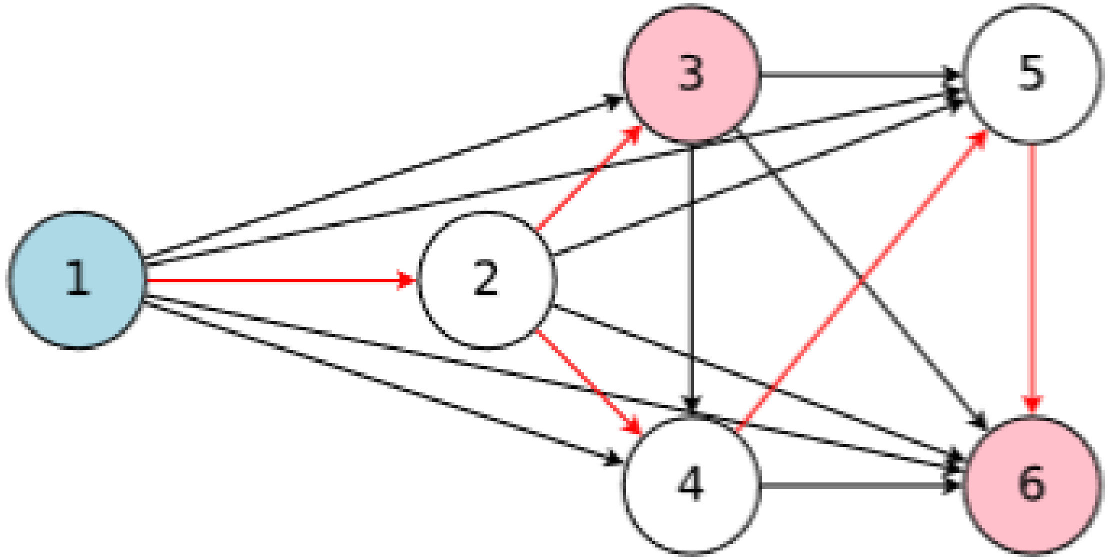
Model	Depth	Parameters	Error rate (%)
Network in Network (Lin et al., 2013)	-	-	8.81
All-CNN (Springenberg et al., 2014)	-	-	7.25
Deeply Supervised Net (Lee et al., 2015)	-	-	7.97
Highway Network (Srivastava et al., 2015)	-	-	7.72
Scalable Bayesian Optimization (Snoek et al., 2015)	-	-	6.37
FractalNet (Larsson et al., 2016)	21	38.6M	5.22
with Dropout/Drop-path	21	38.6M	4.60
ResNet (He et al., 2016a)	110	1.7M	6.61
ResNet (reported by Huang et al. (2016c))	110	1.7M	6.41
ResNet with Stochastic Depth (Huang et al., 2016c)	110	1.7M	5.23
	1202	10.2M	4.91
Wide ResNet (Zagoruyko & Komodakis, 2016)	16	11.0M	4.81
	28	36.5M	4.17
ResNet (pre-activation) (He et al., 2016b)	164	1.7M	5.46
	1001	10.2M	4.62
DenseNet ($L = 40, k = 12$) Huang et al. (2016a)	40	1.0M	5.24
DenseNet($L = 100, k = 12$) Huang et al. (2016a)	100	7.0M	4.10
DenseNet ($L = 100, k = 24$) Huang et al. (2016a)	100	27.2M	3.74
DenseNet-BC ($L = 100, k = 40$) Huang et al. (2016b)	190	25.6M	3.46
Neural Architecture Search v1 no stride or pooling	15	4.2M	5.50
Neural Architecture Search v2 predicting strides	20	2.5M	6.01
Neural Architecture Search v3 max pooling	39	7.1M	4.47
Neural Architecture Search v3 max pooling + more filters	39	37.4M	3.65

Table 1: Performance of Neural Architecture Search and other state-of-the-art models on CIFAR-10.

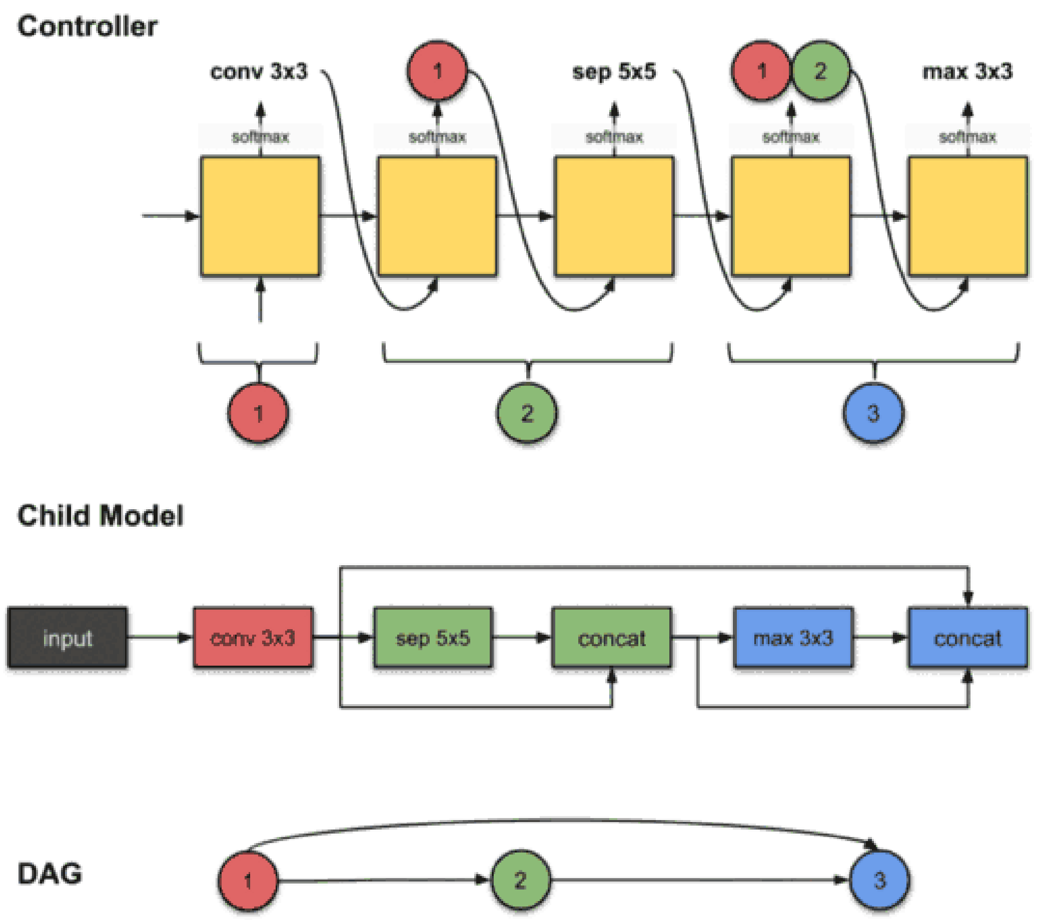
Слишком медленно

Speed-up method	How are speed-ups achieved?	References
Lower fidelity estimates	Training time reduced by training for fewer epochs, on subset of data, downscaled models, downscaled data, ...	Li et al. (2017), Zoph et al. (2018), Zela et al. (2018), Falkner et al. (2018), Real et al. (2019), Runge et al. (2019)
Learning Curve Extrapolation	Training time reduced as performance can be extrapolated after just a few epochs of training.	Swersky et al. (2014), Domhan et al. (2015), Klein et al. (2017a), Baker et al. (2017b)
Weight Inheritance/ Network Morphisms	Instead of training models from scratch, they are warm-started by inheriting weights of, e.g., a parent model.	Real et al. (2017), Elsken et al. (2017), Elsken et al. (2019), Cai et al. (2018a,b)
One-Shot Models/ Weight Sharing	Only the one-shot model needs to be trained; its weights are then shared across different architectures that are just subgraphs of the one-shot model.	Saxena and Verbeek (2016), Pham et al. (2018), Bender et al. (2018), Liu et al. (2019b), Cai et al. (2019), Xie et al. (2019)

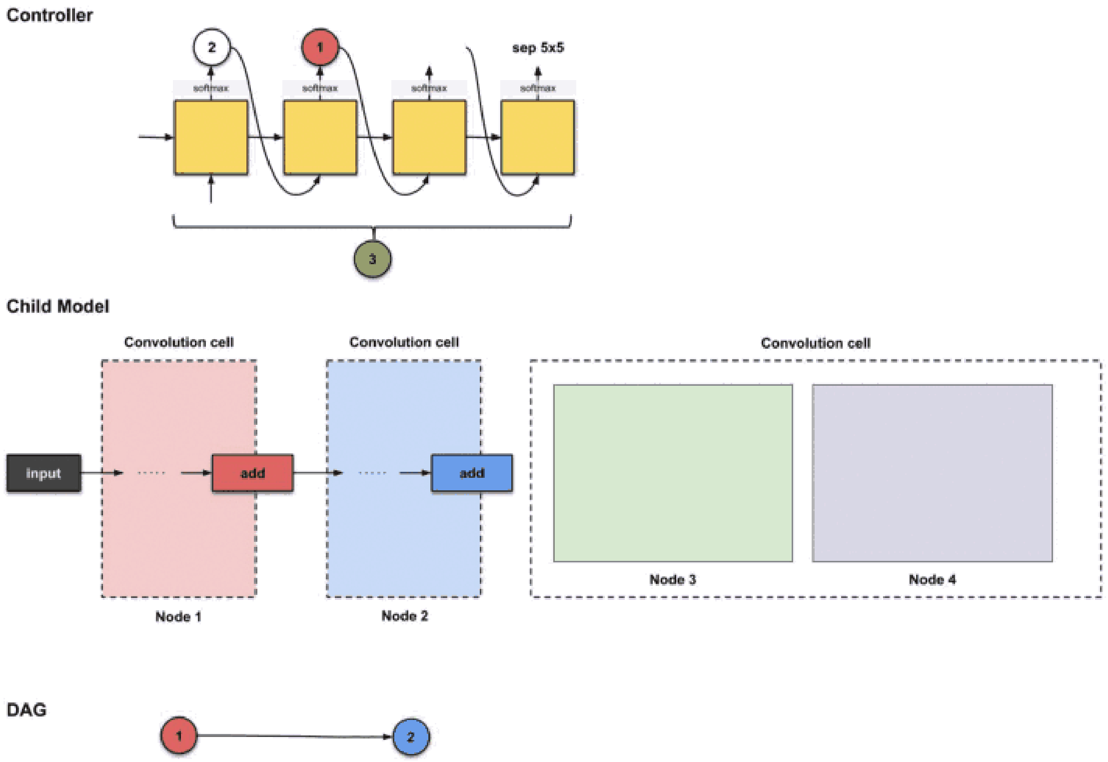
Efficient Neural Architecture Search



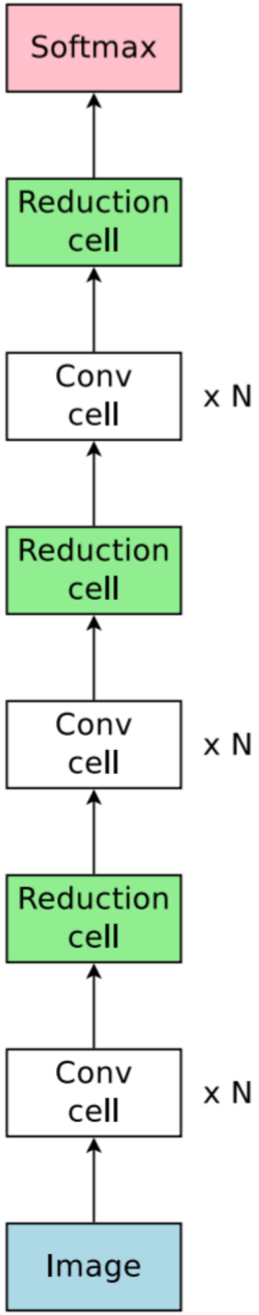
Макро поиск


Controller
conv 3x3
1
sep 5x5
1
2
max 3x3
softmax
softmax
softmax
softmax
softmax
1
2
3
Child Model
input
conv 3x3
sep 5x5
concat
max 3x3
concat
DAG
1
2
3

Микро поиск



Controller
2
1
sep 5x5
softmax
softmax
softmax
softmax
3
Child Model
Convolution cell
Convolution cell
Convolution cell
input
add
add
Node 1
Node 2
Node 3
Node 4
DAG
1
2



Softmax
Reduction cell
Conv cell
x N
Reduction cell
Conv cell
x N
Reduction cell
Conv cell
x N
Image

Результаты ENAS

Method	GPUs	Times (days)	Params (million)	Error (%)
DenseNet-BC (Huang et al., 2016)	–	–	25.6	3.46
DenseNet + Shake-Shake (Gastaldi, 2016)	–	–	26.2	2.86
DenseNet + CutOut (DeVries & Taylor, 2017)	–	–	26.2	**2.56**
Budgeted Super Nets (Veniat & Denoyer, 2017)	–	–	–	9.21
ConvFabrics (Saxena & Verbeek, 2016)	–	–	21.2	7.43
Macro NAS + Q-Learning (Baker et al., 2017a)	10	8-10	11.2	6.92
Net Transformation (Cai et al., 2018)	5	2	19.7	5.70
FractalNet (Larsson et al., 2017)	–	–	38.6	4.60
SMASH (Brock et al., 2018)	1	1.5	16.0	4.03
NAS (Zoph & Le, 2017)	800	21-28	7.1	4.47
NAS + more filters (Zoph & Le, 2017)	800	21-28	37.4	**3.65**
ENAS + macro search space	1	0.32	21.3	4.23
ENAS + macro search space + more channels	1	0.32	38.0	**3.87**
Hierarchical NAS (Liu et al., 2018)	200	1.5	61.3	3.63
Micro NAS + Q-Learning (Zhong et al., 2018)	32	3	–	3.60
Progressive NAS (Liu et al., 2017)	100	1.5	3.2	3.63
NASNet-A (Zoph et al., 2018)	450	3-4	3.3	3.41
NASNet-A + CutOut (Zoph et al., 2018)	450	3-4	3.3	**2.65**
ENAS + micro search space	1	0.45	4.6	3.54
ENAS + micro search space + CutOut	1	0.45	4.6	**2.89**

Вопросы

- Опишите алгоритм работы NAS с помощью reinforce.
- Как реализуются skip connections в NAS? Какие проблемы могут возникнуть и как они решаются?
- За счет чего ENAS эффективнее простой реализации поиска нейронной архитектуры?

Статьи про NAS

Обзорные:

- https://medium.com/@SmartLabAI/introduction-to-neural-architecture-search-reinforcement-learning-approach-55604772f173
- http://jmlr.org/papers/volume20/18-598/18-598.pdf

Про NAS

- https://arxiv.org/abs/1611.01578 - NAS
- https://arxiv.org/abs/1802.03268 - ENAS