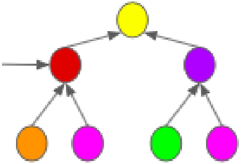
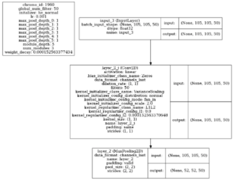
Evolving Multitask Neural Network Structure

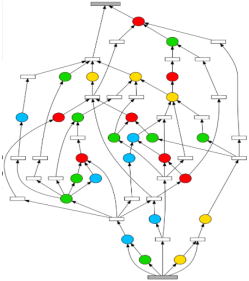
Risto Miikkulainen

The University of Texas at Austin
and Sentient Technologies, Inc.

What is Metalearning?



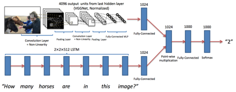




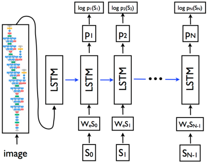
- In this talk: Discovering effective NN structure...
  - Nodes, modules, topology
- ...so that the networks learn better

Structure Matters!

Szegedy et al. 2014



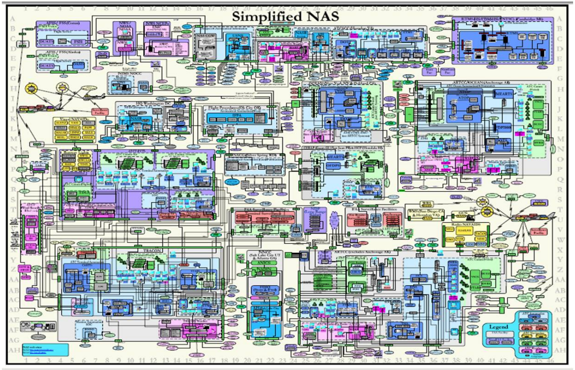
Agrawal et al. 2016



(Vinyals et al. 2016)

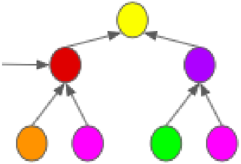
- Different architectures work better in different tasks
- Too complex to be discovered by hand
  - How to discover principles of organization?
  - How to cover enough of the space?

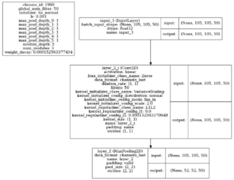
Configuring Complex Systems

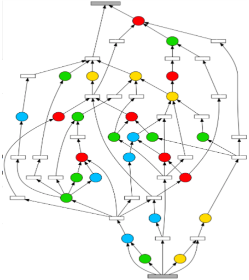


- A new general approach to engineering
  - Humans design just the framework
  - Machines optimize the details
- Design by optimization

How to discover structure?

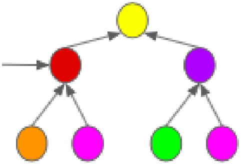


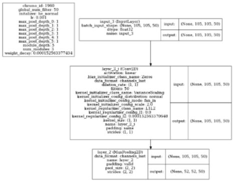


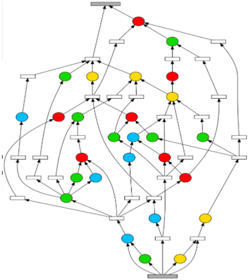


- Evolutionary optimization is a natural fit
  - Crossover between structures discovers principles
  - Population-based search covers space

Examples in Three Levels

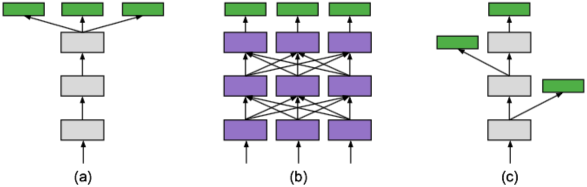






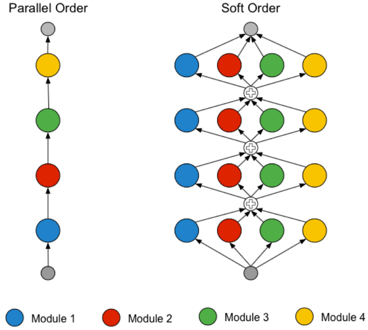
- Structured gated memory nodes (e.g. LSTM, GRU...)
  - Complex structure within nodes
- Modules that are used as building blocks
  - Tasks benefit from each other
- Network topologies
  - How to choose and connect modules

Multitask Learning Domain



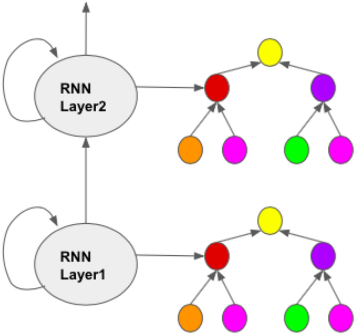
- Learning in multiple tasks at once
  - More generalizable embeddings
  - Each task can learn better
- Network structure can have a large effect
  - A good domain to test metalearning ideas

Soft Ordering Framework



- Order and contribution of modules varies
  - Can be learned by gradient descent
- State of the art in Multitask learning (Meyerson et al. 2017)
  - Improves 10% over standard fixed ordering
- Evolve nodes, modules, and topology in this framework

Node-level Evolution



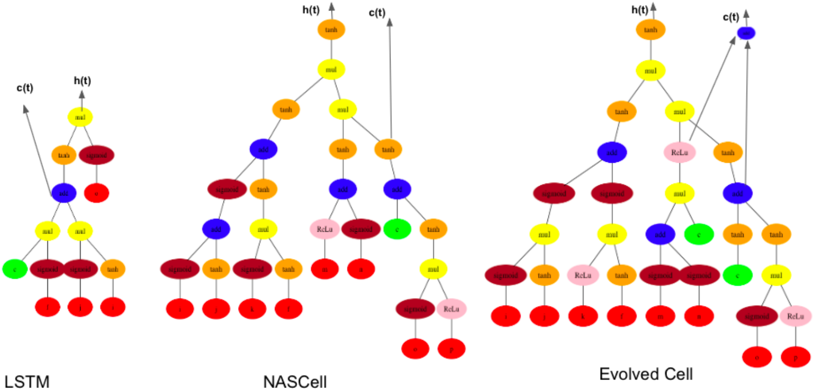
- Gated memory units for a fixed architecture
  - Tree representation for the nodes
  - Optimized through genetic programming
  - Placed into a fixed multilayer architecture
  - Evaluated in the language modeling benchmark

Node Evolution Results

Models with 20M parameters	Word Perplexity on Penn Tree Bank
LSTM	79.2
NASCELL (Zoph and Le, 2016)	77.2
Evolved Cell	76.0

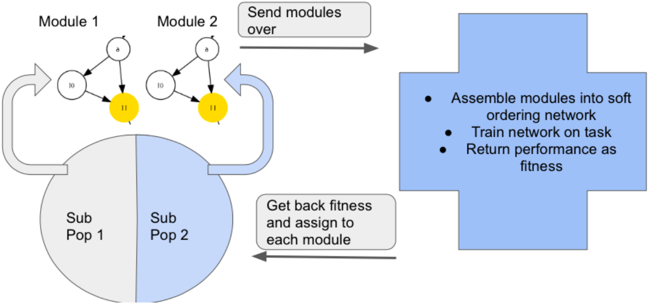
- Single task experiment so far
- Improves upon the state of the art
  - Results before hyperparameter tuning, ensembling, scaling

Evolved Solution



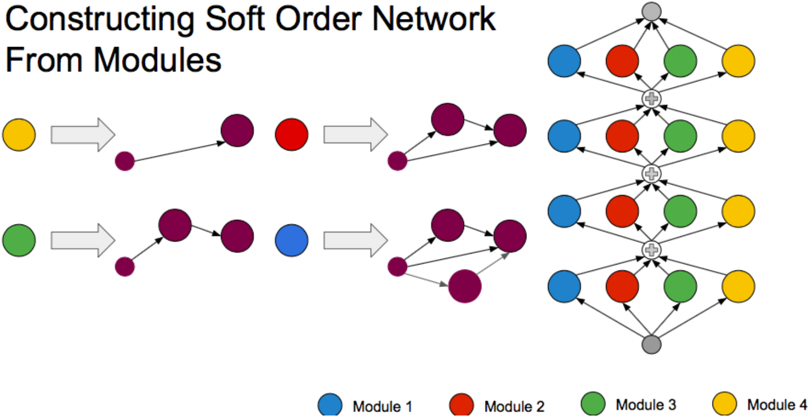
- NAS and Evolved use nonlinear paths from hidden
- Evolved adds a second memory cell path
  - Results from broader search in evolution

Module-level Evolution



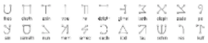
- Co-evolve modules in separate subpopulations
- NEAT method: structured crossover, mutation

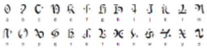
Constructing Soft Order Network From Modules



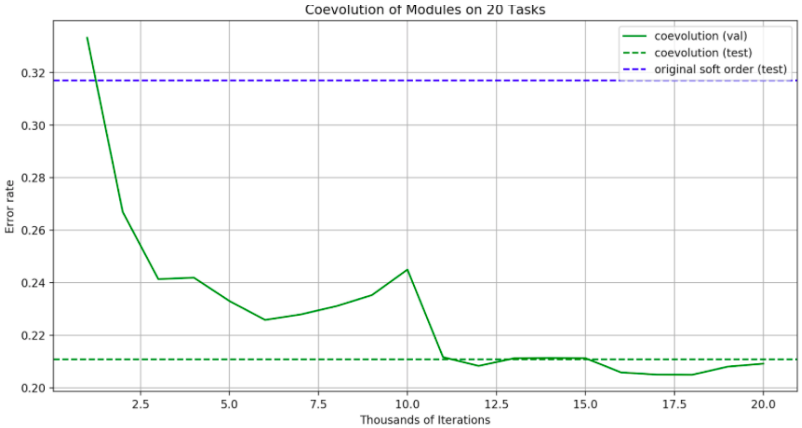
Omniglot Set of Tasks

- Experiment Details
  - Dataset is Omniglot, a multitask image classification dataset (around 30000 105x105 images of characters from different languages)
  - Dataset split into 50% train, 20% val, 30% test
  - We train each assembled soft order network from scratch for 3000 iterations
  - We return average accuracy on 20 tasks evaluated on validation set as fitness
  - Run on cloud with 100 GPUs
  - Each assembled soft order network is evaluated and trained on separate GPU

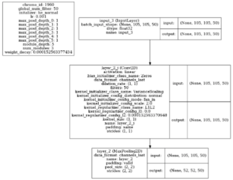


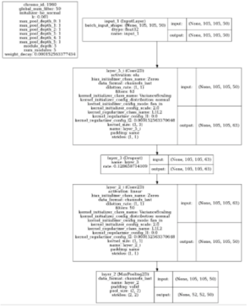


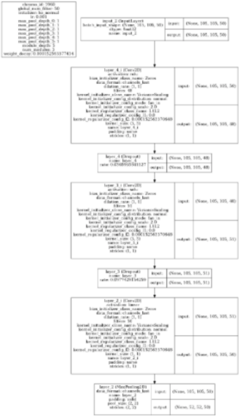
Module Evolution Results



Evolved Modules

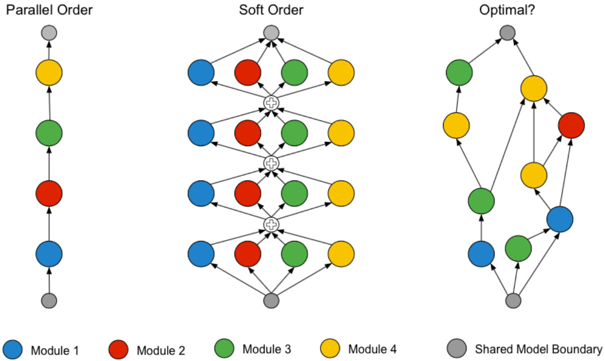






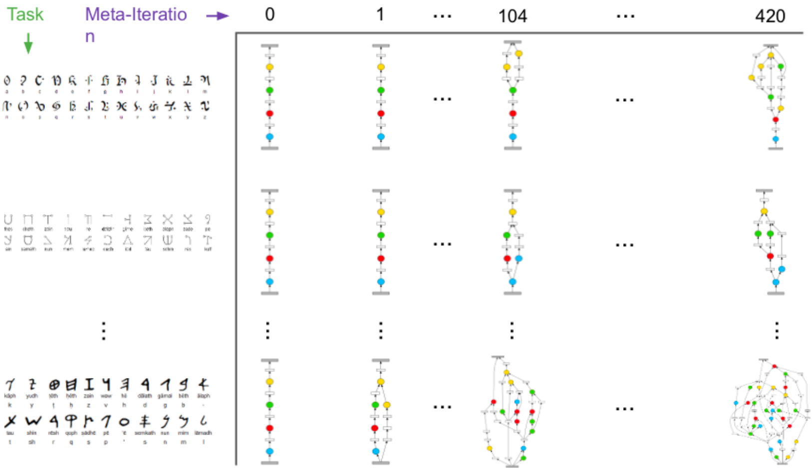
- Three modules with different structure
  - 1, 2, or 3 convolutional layers + maxpooling
  - REUL, SELU, Linear activation functions
  - Fewer params (660k) than original
- Difficult to discover by hand

Topology-level Evolution



- Evolve the topology, choice of modules
- Currently fixed convolutional layer modules
- Evolved with 1+1 evolutionary strategy

Topology-level Evolution (2)



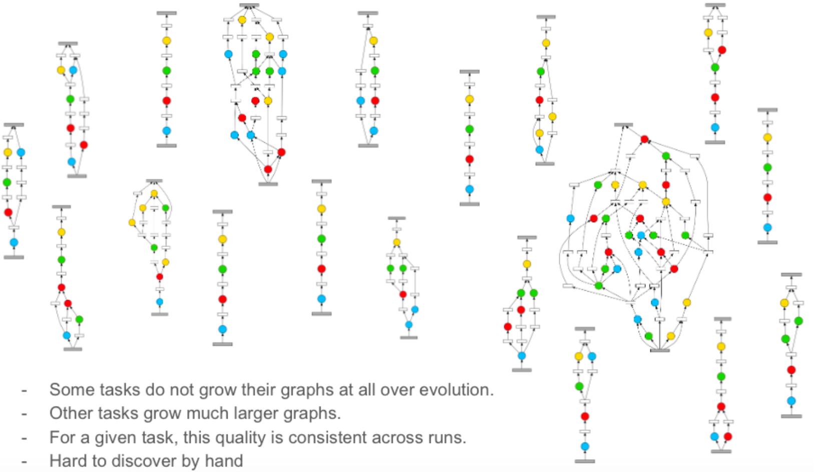
- Topologies for each task diverge over evolution
- Modules trained simultaneously in all tasks

Topology Evolution Results

Method	10 Task Accuracy	20 Task Accuracy	50 Task Accuracy	10 Task % Error	20 Task % Error	50 Task % Error
Single Task	0.642	0.6312	0.6611	35.8	36.88	33.89
Parallel Order	0.6211	0.6715	0.6974	37.89	32.85	30.26
Permuted Order	0.6706	0.6791	0.697	32.94	32.09	30.3
Soft Order	0.6911	0.683	0.704	30.89	31.7	29.6
EMR	**0.8048**	**0.8276**	**0.853**	**19.52**	**17.24**	**14.7**
UD-MTL			0.6726			32.74
DMTRL-LAF			0.6446			35.54
DMTRL-Tucker			0.6716			32.84
DMTRL-TT			0.675			32.5

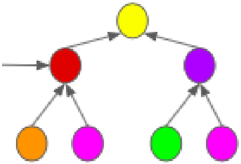
- Significant improvement over state of the art
  - Meyerson et al. 2017; Yang & Hospedales 2017
- Better with more tasks

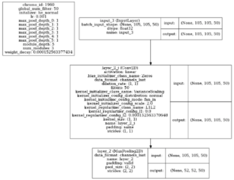
Resulting Topologies

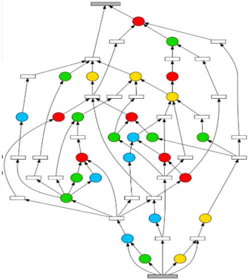


- Some tasks do not grow their graphs at all over evolution.
- Other tasks grow much larger graphs.
- For a given task, this quality is consistent across runs.
- Hard to discover by hand

Conclusion





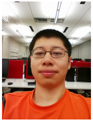


- Evolutionary metalearning is good at discovering structure
  - Node, module, topology levels
  - Multitask learning a particularly good domain
- Well suited for discovering novel solutions
  - RL, gradient descent, Bayesian for refinement
- Future: Co-evolution of the three levels
- Future: Scale-up with extreme compute
  - ENN can scale with more power

Acknowledgments

Collaborators/Students:

Aditya Rawal
(Sentient/UT)



Jason Liang
(Sentient/UT)

Elliot Meyerson
(Sentient/UT)