Beyond Shared Hierarchies: Deep Multitask Learning through Soft Layer Ordering

Elliot Meyerson, Risto Miikkulainen

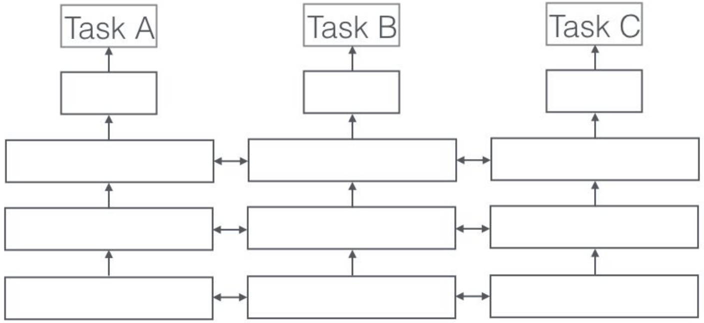
MACHINE
INTELLIGENCE
COMMUNITY

Devin de Hueck
May 17th, 2018

Multitask Learning

- Shares information between related tasks
- Learn from seemingly unrelated tasks
- How does it work:
  - Implicit data augmentation
  - Attention focusing
  - Eavesdropping
  - Regularization

[1]



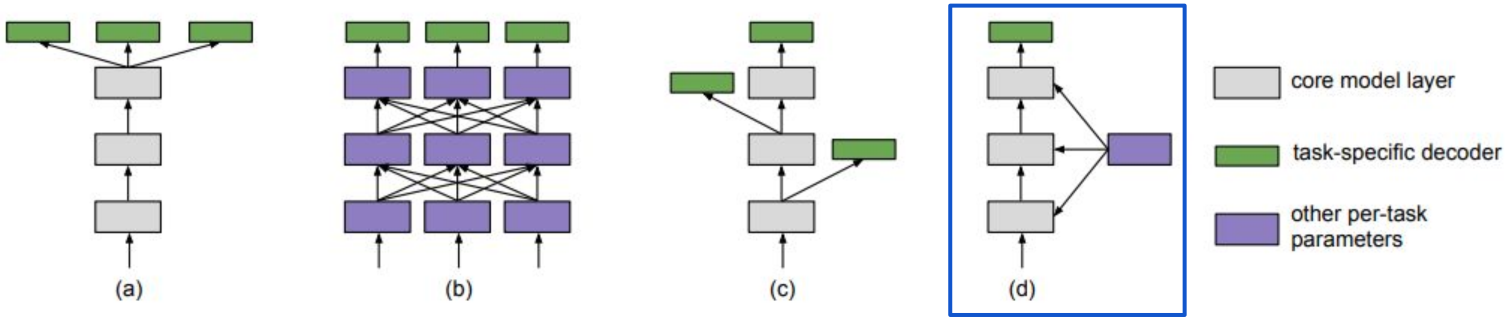
The Problem

- Past Multitask Learning (MTL) approaches have been constrained to training few or closely related tasks
- Two assumptions
    - Learned information can be shared across tasks
    - This sharing takes place only at aligned layers - *parallel ordering assumption*

The Parallel Ordering Examples

A common assumption is that layers within a deep network extract progressively higher level features at further depths.

a. Classical Approach **b.** Column Based approach **c.** Supervision at custom depths **d.** Universal representations



Breaking Down the Parallel Ordering Assumption

The Assumption:

$$y_i = (\mathcal{D}_i \circ \phi_D \circ W_D^i \circ \phi_{D-1} \circ W_{D-1}^i \circ \ldots \circ \phi_1 \circ W_1^i \circ \mathcal{E}_i)(x_i), \text{ with } \boxed{W_k^i \approx W_k^j \,\forall\, (i, j, k)}.$$

Where:

$$\left.\begin{array}{ll} W_D^i & \text{- Weight tensor at depth } D \\ \phi_D & \text{- Nonlinearity/Activation at depth } D \\ \mathcal{E}_i & \text{- Encoder for task } i \\ \mathcal{D}_i & \text{- Decoder for task } i \end{array}\right\} \text{Multitask Network}$$

Permuting Shared Layers

Standard Multitask Network:

$$y_i = (\mathcal{D}_i \circ \phi_D \circ W_D \circ \phi_{D-1} \circ W_{D-1} \circ \ldots \circ \phi_1 \circ W_1 \circ \mathcal{E}_i)(x_i)$$

$$\Downarrow$$

Permuted Multitask Network:

$$y_i = (\mathcal{D}_i \circ \phi_D \circ W_{\underline{\rho_i(D)}} \circ \phi_{D-1} \circ W_{\underline{\rho_i(D-1)}} \circ \ldots \circ \phi_1 \circ W_{\underline{\rho_i(1)}} \circ \mathcal{E}_i)(x_i)$$

This results in a set of **layers that are assembled in different ways for different tasks**.

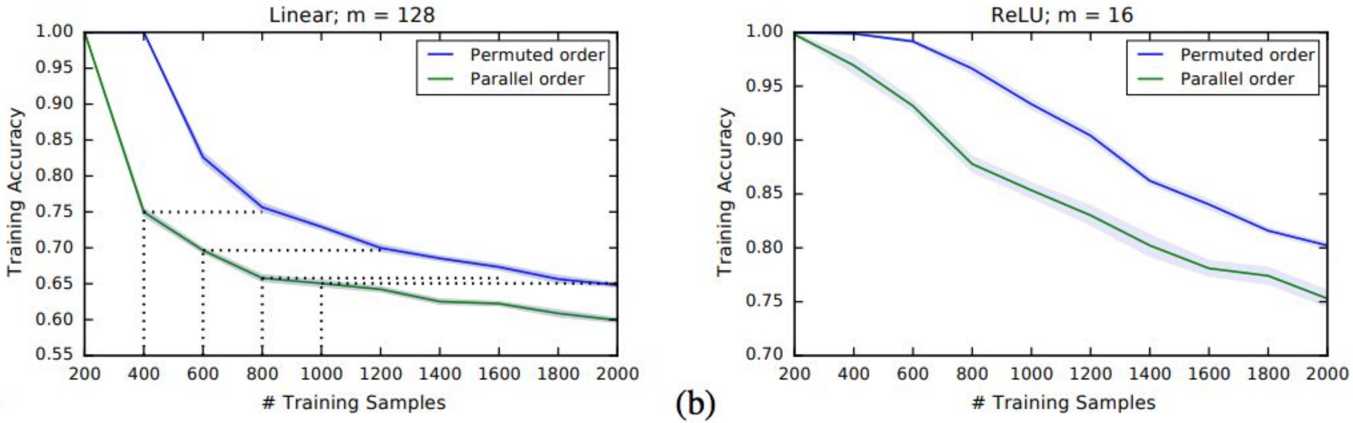
Expressivity of Permuted Ordering

- Tasks fitted to randomly generated patterns

Permuted Multitask Network:

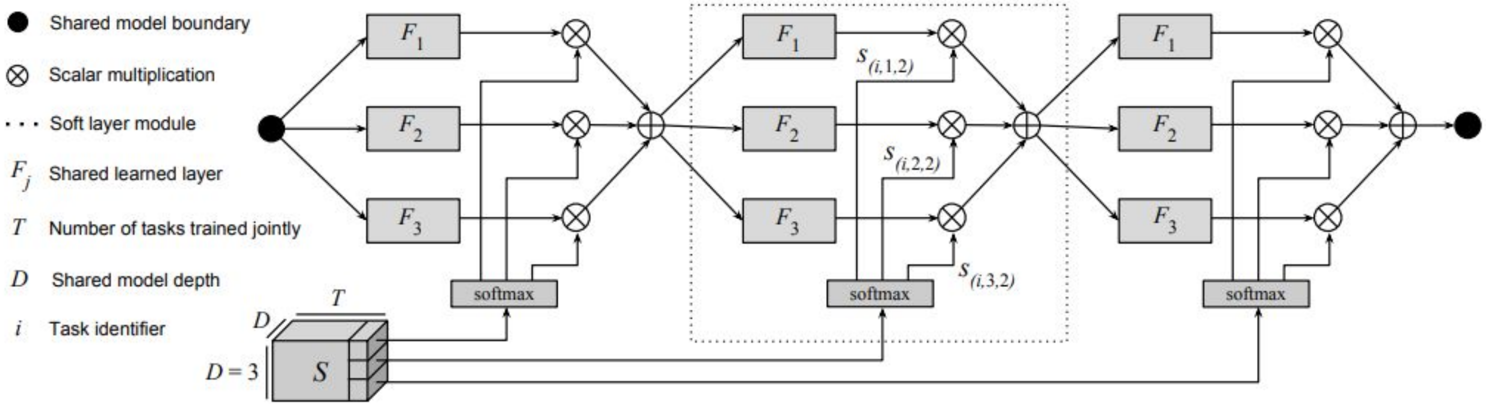
$$y_1 = (O \circ \phi \circ \underline{W_2} \circ \phi \circ \underline{W_1})(x_1) \text{ and } y_2 = (O \circ \phi \circ \underline{W_1} \circ \phi \circ \underline{W_2})(x_2)$$

Results:



(a) (b)

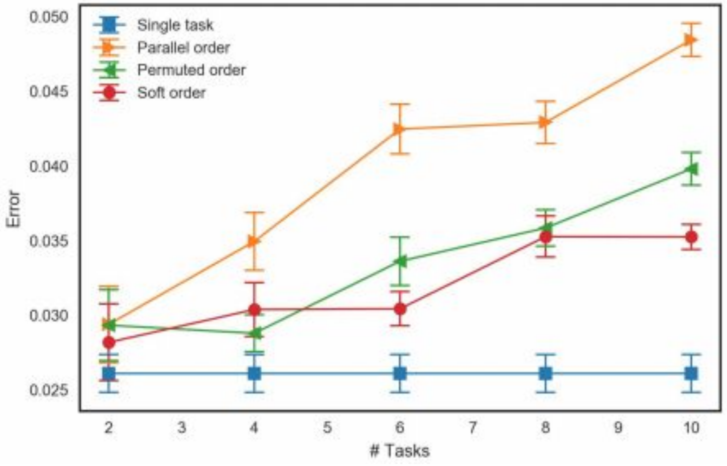
Soft Ordering of Shared Layers



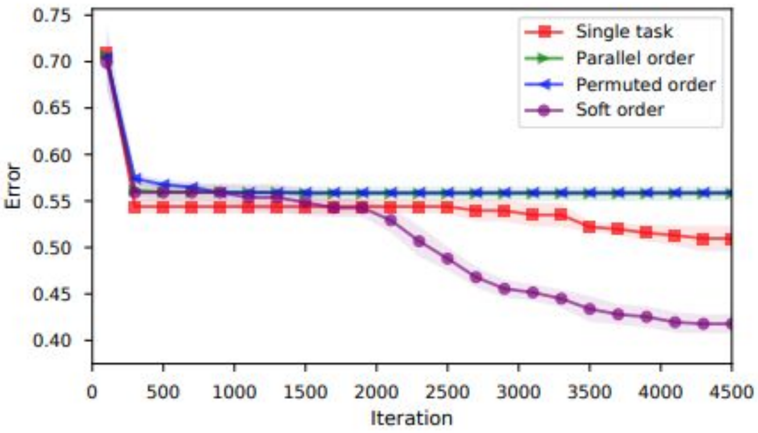
Allows jointly trained models to learn ***how*** layers are applied while simultaneously learning the layers themselves

Soft Ordering Evaluation - Related and Unrelated Tasks

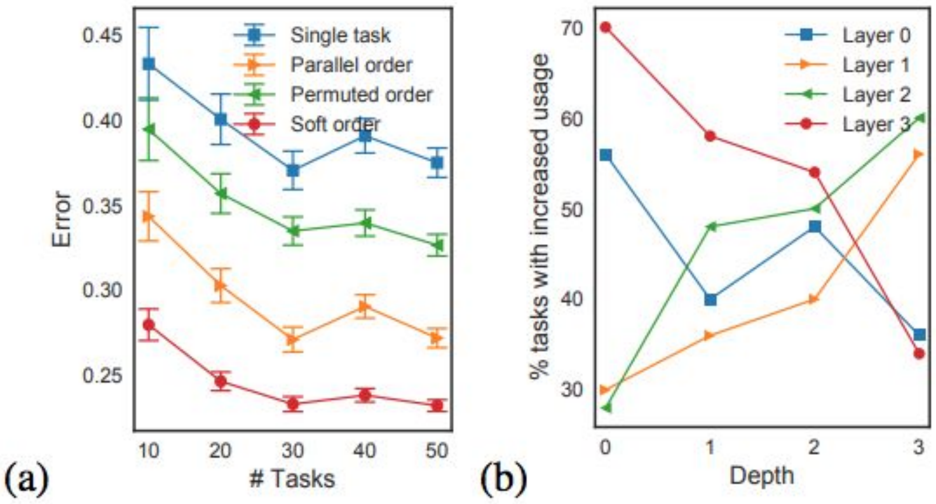
MNIST - digit vs digit classification



10 UCI dataset tasks

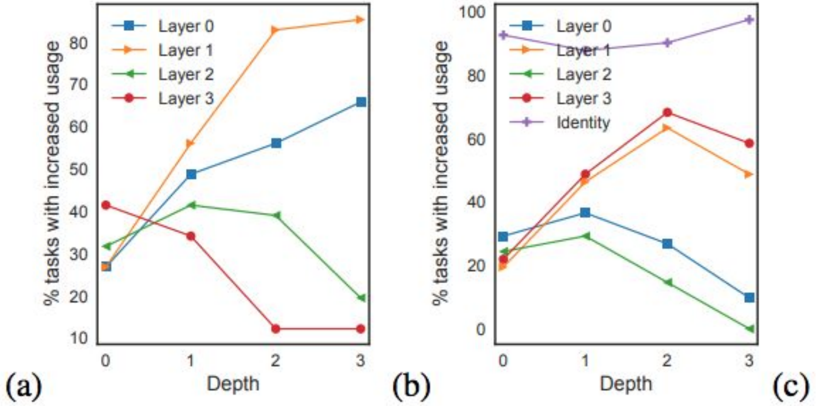


Soft Ordering Evaluation - Omniglot



- A "hierarchy" of layers is discovered with soft ordering

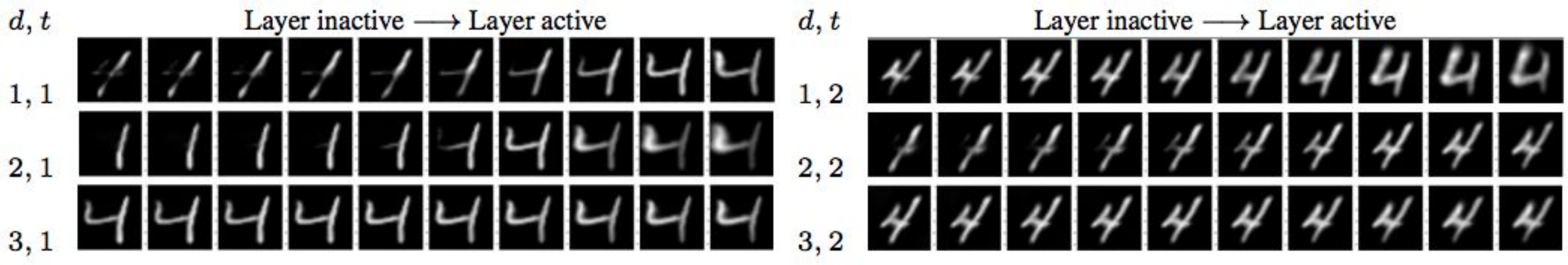
Soft Order Evaluation - Facial Attribute Recognition



Deep MTL method	Test Error %
Single Task (He et al., 2017)	10.37
MTL Baseline (He et al., 2017)	9.58
Parallel Order	10.21
Parallel Order + Landmarks	10.29
Soft Order	8.79
Soft Order + Landmarks	8.75
Soft Order + Identity	**8.64**
Soft Order + Landmarks + Identity	8.68

Visualizing the Behavior of Soft Ordered Layers

The success of soft layer ordering suggests that layers learn ***functional primitives*** that can be applied in different contexts.



Conclusion

- Future Work
  - Connections to recurrent architectures
  - Generalizing the structure of shared layers
  - Training generalizable building blocks
- Aligning closer to our understanding of real-world processes

References and Further Reading

1. Ruder, Sebastian. ***"An Overview of Multi-Task Learning in Deep Neural Networks"*** arXiv preprint arXiv:1706.05098 2017.
2. Liang, Jason, et al. "***Evolutionary Architecture Search For Deep Multitask Networks.***" arXiv preprint arXiv:1803.03745. 2018.