AdaShare: Learning What To Share For Efficient Deep Multi-Task Learning

Ximeng Sun¹, Rameswar Panda², Rogerio Feris², Kate Saenko¹,²
¹. Boston University, ². MIT-IBM Watson AI Lab, IBM Research
{sunxm, saenko}@bu.edu, {rpanda, rsferis}@us.ibm.com

Project page: https://cs-people.bu.edu/sunxm/AdaShare/project.html
Multi-task Learning

Multi-task learning (MTL) is a subfield of machine learning in which multiple learning tasks are solved at the same time, while exploiting commonalities and differences across tasks.

MTL improves generalization by leveraging the domain-specific information contained in the training signals of related tasks.
Previous Work – Two prevailing trends for MTL

**Hard parameter sharing**

- Task A
- Task B
- Task C

  Task-specific layers

  Shared layers

Examples: Deep Relationship Network, Fully-Adaptive Feature Sharing

**Soft parameter sharing**

- Task A
- Task B
- Task C

  Constrained layers

Examples: Cross Stitch, Sluice, NDDR

**Hand-crafted** tree structure

**Non-scalable** with the increasing number of tasks
AdaShare – Learn non-handcrafted and scalable sharing patterns

- **Each block**: shared or task-specific
- **Adaptive Computation**: Gumbel-Softmax Sampling
- **Loss**: Task-specific Loss, Sparsity Loss, Sharing Loss
- **Training Strategy**: policy learning stage and retraining stage
Experiments – Quantitative Results

Datasets: NYU v2 (2 or 3 tasks), CityScapes (2 tasks), Tiny-Taskonomy (5 tasks), DomainNet (6 tasks), Text-Recognition (10 tasks)

Table 4: **Tiny-Taskonomy 5-Task Learning.** $T_1$: Semantic Segmentation, $T_2$: Surface Normal Prediction, $T_3$: Depth Prediction, $T_4$: Keypoint Estimation, $T_5$: Edge Estimation.

<table>
<thead>
<tr>
<th>Models</th>
<th># Params ↓</th>
<th>$\Delta T_1$ ↑</th>
<th>$\Delta T_2$ ↑</th>
<th>$\Delta T_3$ ↑</th>
<th>$\Delta T_4$ ↑</th>
<th>$\Delta T_5$ ↑</th>
<th>$\Delta T$ ↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multi-Task</td>
<td>-80.0</td>
<td>-2.1</td>
<td>-0.7</td>
<td>-9.1</td>
<td>+ 1.5</td>
<td>+ 5.2</td>
<td>-1.0</td>
</tr>
<tr>
<td>Cross-Stitch</td>
<td>0.0</td>
<td>+ 2.6</td>
<td>-3.3</td>
<td>0.0</td>
<td>-2.5</td>
<td>-3.3</td>
<td>-1.3</td>
</tr>
<tr>
<td>Slicue</td>
<td>0.0</td>
<td>-6.1</td>
<td>-0.7</td>
<td>-4.6</td>
<td>+ 2.5</td>
<td>+ 6.6</td>
<td>-0.4</td>
</tr>
<tr>
<td>NDDR-CNN</td>
<td>+ 8.2</td>
<td>+ 6.3</td>
<td>-0.3</td>
<td>-11.4</td>
<td>+ 1.5</td>
<td>+ 2.8</td>
<td>-0.2</td>
</tr>
<tr>
<td>MTAN</td>
<td>-9.8</td>
<td>-10.8</td>
<td>-0.7</td>
<td>-4.5</td>
<td>+ 2.0</td>
<td>+ 4.2</td>
<td>-2.0</td>
</tr>
<tr>
<td>DEN</td>
<td>-77.6</td>
<td>-28.2</td>
<td>-3.0</td>
<td>-22.7</td>
<td>+ 2.5</td>
<td>+ 4.2</td>
<td>-9.4</td>
</tr>
<tr>
<td>AdaShare</td>
<td>-80.0</td>
<td>+ 1.6</td>
<td>0.0</td>
<td>-13.6</td>
<td>+ 2.5</td>
<td>+ 9.0</td>
<td>-0.1</td>
</tr>
</tbody>
</table>

**Single-Task Learning:** Seg: 0.575; SN: 0.707; Depth: 0.022; Keypoint: 0.197; Edge: 0.212
Experiments – Policy Visualization

Observations:
1. Not all blocks contribute to the task equally
2. More blocks shared only among a sub-group of tasks in ResNet’s conv3_x layers, where middle/high-level features (more task-specific) are starting to get captured
3. Similar tasks should have similar execution distribution to share knowledge
Thank you and welcome to our poster!