Novel task learning by distributed and specialized neural computation architectures

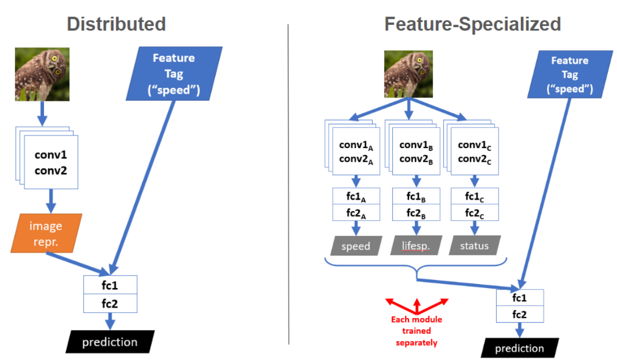
Abstract:

Neural computation in neural networks falls on a spectrum from *distributed*, where each node in a network contributes to many tasks, and *specialized*, where nodes or layers are tailored for specific tasks. One hypothesis claims that distributed networks offer better generalization and flexibility but sacrifice the domain-specific performance that specialized networks can achieve. This raises the question: which paradigm is more effective for novel task learning? To answer this, we created distributed and specialized neural network architectures, trained them on a set of three initial feature extraction tasks, and compared their ability to learn a novel feature extraction task. Initial results with an animal image-feature dataset are unpersuasive and give a marginal edge to the specialized network in learning the three initial tasks. Given these findings and our limited resources, we pivoted to a smaller, analogous experiment using the MNIST numbers dataset. Successfully running our comparative methodology, we found that the specialized network outperformed the distributed network in both initial task learning and novel task learning. Suspecting that the specialized network's greater model complexity conferred an unfair advantage, we repeated the experiment with an identically constructed distributed network and found that following novel task training, the distributed network outperformed the specialized network in some initial and novel tasks.

Additional Details:

In neural computation, the choice between distributed and specialized neural network architectures critically influences their learning and processing capabilities. Distributed networks are akin to the human brain's neural circuits, where each node contributes to many tasks, and are thought to offer adaptability. Conversely, specialized networks feature nodes or layers tailored for specific tasks, leading to efficient processing within their designed domain. Which architecture is more effective for learning novel tasks? Distributed networks, with their comprehensive node involvement, might excel in generalizing learning and adapting to novel inputs. In contrast, specialized networks could potentially offer more efficient learning of tasks closely related to their specialized domain.

FIGURE 1



Our study analyzed animal traits using 58,289 images of 45 animals from ImageNet-1k, enriched with the Animal Information Dataset's numerical (speed, weight, height, gestation period) and categorical (conservation status, social behavior, diet, habitat) data. We normalized and trained both distributed and specialized networks on three features: weight, height, and average speed, encoded as numeric features with images. The distributed network learned from image-feature combinations, while the specialized network initially trained a submodule on each feature separately, later merging these into a unified network for feature-specific processing. **Figure 1** provides schematics of both network structures.

Our training faced hurdles due to the large dataset and task complexity, challenging our computational resources and time. Even after trying various hyperparameters, the distributed network's results, shown in **Figure 2**, were underwhelming. The network often settled on values minimizing loss per feature, mostly overlooking the image content. The specialized network, however, yielded more promising results. Its three tasks nearly converged, showing adaptability as seen in **Figure 3**. This indicates the network's partial success in integrating image details into its learning, despite not completely mastering the task.

FIGURE 2

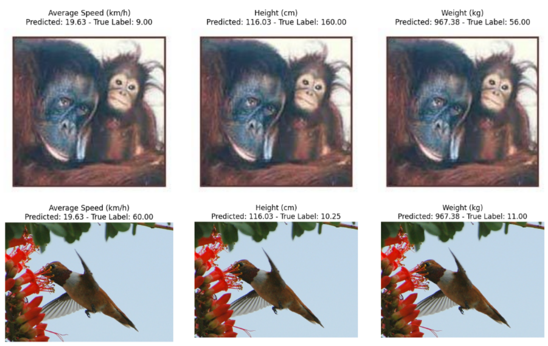
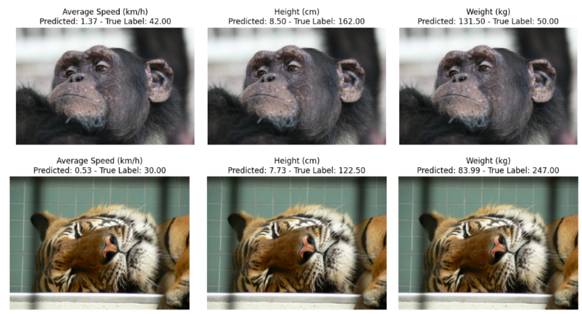
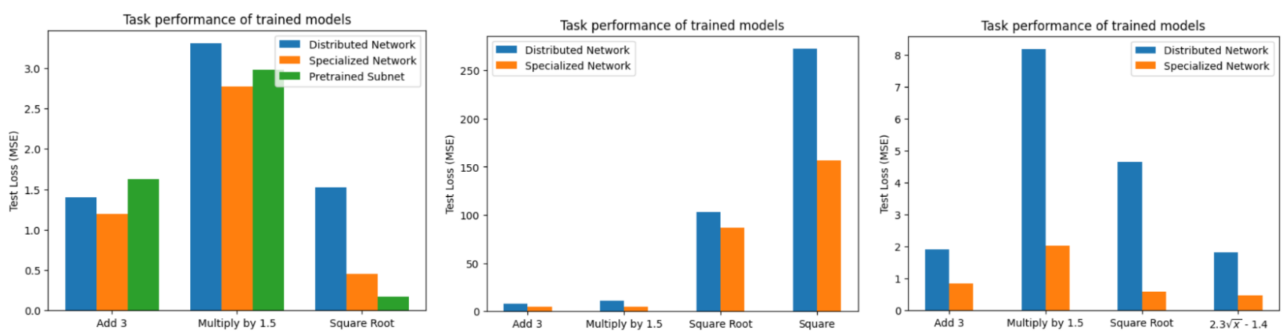


FIGURE 3



Our inconclusive results may be due to the difficulty of learning abstract features like “conservation status” from only 45 classes of animals. Thus, we opted to run our methodology on the MNIST dataset, where the images represent 10 classes of numbers; the three initial “features” or tasks are to Add 3, Multiply by 1.5, and Square Root; and for the novel task, we tested both Square and 2.3*sqrt(x)-1.4 (which requires knowledge of all the initial tasks). This instance of novel task learning is similar to a student learning new mathematics operations. In **Figure 4**, we first see the similarly good post-training performance on the three initial tasks by the distributed and specialized networks along with the pretrained submodule. After adding the square operation and training for 1 epoch, the specialized network is able to achieve a much lower loss on squaring and maintain a lower loss on the initial tasks. If we instead add 2.3*sqrt(x)-1.4 as the fourth task, the specialized network outperforms the distributed network across all tasks.

FIGURE 4



We then hypothesized that the specialized network may have an unfair advantage because it has over three times the total free parameters of the distributed network. Thus, we created a “distributed” network with an identical architecture as the specialized network, but trained end to end instead of task-optimized. The results are in **Figure 5**. This time, the specialized network still far outperformed the distributed network on the square task, but the distributed network fared better on ¾ tasks if 2.3*sqrt(x) - 1.4 was the additional task. Our results are not strong enough to conclude that distributed or specialized architectures are superior for novel task learning, but we show that novel task learning ability is dependent not only on neural computation paradigm and network architecture, but also on task and data. Future work would explore variations on models, datasets, tasks, and training strategies, investigate the case of multi- novel task learning, and dig into learning dynamics to explain the observed trends.

FIGURE 5