Sequential learning?

Sheng Zhong

February 27, 2019

One phenomenon that's strange and unexplained is that the initial training with EWC and GD is never faster than training from scratch [1]. I expected it to train faster because of the hypothesis that these two tasks share commonalities (an assumption required for EWC to work, and is demonstrated by the final performance being better than random). Recent work on **metalearning** [2] explores how to learn a model that is easily adaptable to a class of tasks. For example, learning a locomotion model that's robust to changing environments (incline, crippling the robot, adding obstacles). This research also uses the assumption of common information and capabilities amongst the tasks, but asserts that learning this enables rapid adaptation to other similar tasks. This is illustrated in the figure below, and was demonstrated experimentally in their paper (5 gradient descend steps to go from learning about periodic functions to adapting to a specific sine wave)[3].

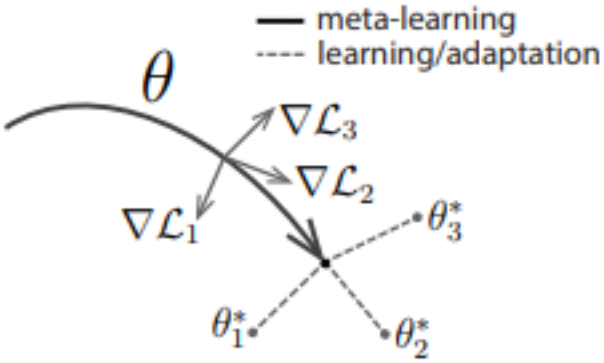

Figure 1: Intuition about metalearning - parameters are learned to be where it could quickly adapt to optimal parameters for any specific task.

It's interesting that a similar assumption could lead to very different results with different approaches. The metalearning approach is to train on all the tasks together (intuitively: find the gradient for each task, then step in their average direction). I think the sequential learning could try something similar where instead of learning each task sequentially, it learns a set of tasks sequentially. Humans are never solely performing a single task - we

[1] page 5 of supplementary methods and experiment details
[2] https://arxiv.org/pdf/1803.11347.pdf
[3] https://arxiv.org/pdf/1703.03400.pdf

often perform similar tasks and variations of the original when trying to learn something. An example would be someone learning to play chess: they would practice tactics (starting from a given position, make the best moves), play quick online games, and over the table games simultaneously during their training.

My other major concern is the usefulness of sequential learning. They claim in the paper that it's necessary to develop towards general AI, but I don't see why. From their supplementary material, we see that training just for a single task either outperforms or does equivalently to their sequential learning models. This is especially true for the stargunner game, which sequential methods all fail at. We also see in their paper that their forget-me-not unsupervised clustering algorithm allows the sequential learning to almost perform as if it had task labels. This suggests the clustering algorithm could be used to detect when a new task is being performed and restart the training of a new model without too much trouble. I expect this kind of ensemble model would outperform any sequential model.

The failure of all sequential learning on the stargunner game also illustrates a glaring weakness of sequential learning. Because adjusting the parameters is restricted to where the tasks share information (in the Fischer matrix sense), **learning a new task is impossible if it doesn't share information with all previous tasks**. I want to emphasize that it's required to share information with all, not any, previous tasks. The more tasks you try to sequentially learn, the more likely that the new task will not share information with all previous tasks. Each new task narrows down the valid parameter space since it's an intersection operation rather than a union. This also means learning any new task will degrade performance on all previous tasks because the previous parameters were "optimal" in some sense for all those tasks. Thus with this assumption of moving towards shared information, there will always be a tradeoff of how many tasks can be sequentially learned and performance on old and new tasks.