Report project of RL: A Simple Neural Attentive Meta-Learner

Luca Teodorescu, Mohamed Fyras Telmini

January 28, 2024

Abstract

This report serves to summarize the findings in the paper[1]. We start by the theoretical setting and an overview of the algorithm, to then explain the different experiments conducted by the authors with a particular focus on the reinforcement-learning context. We finally provide our own impression of the paper as a review. The report was written equally by both of us.

1 Paper summary

1.1 Theoretical setting and SNAIL algorithm

The scope of the article is to provide a model of Meta-learning which is a sub-field of machine learning focusing on training on different tasks all at once and generalizing over the tasks rather than over the data. Such a choice is relevant for precise tasks that are poor in data or when being adaptive is at stake. The «Simple Neural AttentIve Learner»'s (SNAIL) objective is to overcome the limitation of meta-learner's use of past experience due to the structure of how these past experiences are referenced. The issue being that sequence to sequence structure - used in several implementation of meta-learning- hinder how the meta-learner could internalize experience from sequences far ago.

The SNAIL algorithm's is expressed with regards to the following formalization :

An episodic task $\mathcal{T}_i$ is defined by inputs x_t, outputs a_t, a loss function $\mathcal{L}_i(x_t, a_t)$, the distribution of the Markov transition $P(x_i|x_{t-1}, a_{t-1})$, and length of an episode H_i.

$$\min_{\theta} E_{\mathcal{T}_i \sim \mathcal{T}}[\sum_{t=0}^{H_i} \mathcal{L}_i(x_t, a_t)]$$

where

$$x_t \sim P_i(x_t|x_{t-1}, \dots, a_{t-1}), a_t \sim \pi(a_t|x_1, \dots, x_t; \theta)$$

To train such a meta-learner, we optimize the above expected loss over tasks we sample from the distribution of tasks $\mathcal{T}$. We test the meta-learner on unseen tasks from another task distribution $\tilde{\mathcal{T}}$ with a similar to the train task distribution $\mathcal{T}$
The SNAIL algorithm is constructed with regards to two methods combined in one architecture. First, the principle of 1D-convolutions of the temporal (Episodic) distribution increasing the power to access past experiences. The limits of using only such an architecture is that the more episodes it will see the more layers it needs. Such a property raises the issue of complicated access to a large spectrum of experiences. Secondly, the principle of attentive learning allowing a model to highlight precise piece of information from its large context to enhance its effect and importance overall in the model. Such a technique is generally used on multi-dimensional data. Combining both ideas complements each other by making use of temporal convolution's good access to past experience and attention's ability to focus on information despite a large context.
In the end the overall structure of such a SNAIL algorithm is constructed with the following blocks:

1. A dense block applying a 1D-convolution
2. A temporal convolution block
3. An attention block (built after the self-attention mechanism of Vaswani et al. 2017 [2])

1.2 Related works

We chose to include this part in our summary since, especially in the experiments part, the proposed model was benchmarked against several other approaches in the same context of meta-learning. So to better interpret the power (or lackthereof) of this model, it is interesting to rapidly explain the other approaches involved. Two main approaches considered by the authors were:

LSTM (Long Short Term Memory networks [3]) are a kind of RNN (recurrent neural network) designed to avoid long-term depedency. As a RNN they have the form of a chain of repeating modules of neural network. LSTM's particularity is that its cell is accompanied by an input gate, an output gate and a forget gate controlling the flow of information within the cell. It is a recognized to be a good solution when dealing with sequences or time series data.

MAML (Model-Agnostic Meta-Learning) is a class of meta-learning algorithm that relies on good initialization over a broad distribution of tasks then fine tuning on specific tasks. The goal being to have a model that performs well with scarce data points. [4]
A few other approaches are used for the later experiments in reinforcement-learning (in the tabular MDP's experiments) such as :

PSRL (Posterior Sampling for Reinforcement Learning) is the idea to use the Bayesian frame of work [5] for reinforcement learning, meaning that we sample from a distribution hypothesis.

OPSRL (Optimistic Posterior Sampling for Reinforcement Learning) [6] adds on the previous class of algorithm with an optimistic exploration strategy to enhance learning efficiency.

UCRL2 [7] is a reinforcement learning inspired by Auer and Ortner, 2007's UCRL which utilises the upper confidence bound to choose an optimistic policy. It is applied in the multi-armed bandit paradigm.

ϵ-greedy also from the MAB problem, is a straight-forward strategy in which the best lever known is selected for a proportion of 1-ϵ.

1.3 Experiments

The authors conducted multiple experiments using the proposed SNAIL meta-learner and benchmarked it against other meta-learning models proposed by other papers. The experiments were designed to investigate three main questions:

1. How SNAIL's generality affects its performance on a range of meta-learning tasks

2. How its performance compares to existing approaches that are specialized to a particular task domain or have elements of a high-level strategy already built-in

3. How SNAIL scales with high-dimensional inputs and long-term temporal dependencies

The first experiments involved supervised learning. The model was evaluated against multiple benchmarks in the context of few-shot image classification. This means that the model was trained to correctly classify images from classes it never saw, while being trained on few examples of multiple other classes.
The most interesting experiments for us were the reinforcement learning ones, this paper included a total of 4 experiments in this context. The context of reinforcement learning was particularly interesting here since it has inherent long term temporal dependencies. All experiments were benchmarked against two other meta-learning approaches (LSTM and MAML).

Multi-arm bandit This is a well known problem in online learning settings, where an agents interacts with "arms" (in reference to casino machines) that have an unknown reward distribution. The agent needs to both explore enough to know which arms better rewards while later on exploiting the best arms to maximize the rewards. This problem has a particularly interesting benchmark which is the Gittins index, that acts as an oracle as N (number of arms) goes to infinity. When compared to the LSTM and MAML banchmarks, the SNAIL achieves state of the art in all the cases where the other two models do so. The authors also say that MAML is too computationally expensive for the $N \geq 500$ cases, which implies that SNAIL is at least as computationally efficient as LSTM and much more than MAML.

Tabular MDP's In the tabular Markov Decision Process (MDP) experiments, each MDP had 10 states and 5 actions. The reward for each state-action pair followed a normal distribution with unit variance where the mean was sampled from N(1,1). The transitions are sampled from a flat Dirichlet distribution with random parameters. Each meta-learner was allowed to interact with an MDP for N episodes of length 10. In addition to a random agent, several human-designed algorithms were considered as baselines. These include PSRL [5], OPSRL [6], UCRL2 [7], and ϵ-greedy. An oracle estimation was also provided by the authors as an upper bound on the performance of every approach. The SNAIL always outperformed LSTM and MAML benchmarks, and achieved state of the art performance when compared to the other proposed methods while being extremely close to the oracle in all instances.

Continuous control This experiment involves the simulation of the movement of two virtual creatures, a 3D ant and a 2D cheetah. Two different goals were studied, the first was the maximizing the velocity of each creature in either the forward or backwards direction. The second was having the creature achieve a randomly selected velocity and direction by rewarding it according to how close its current behaviour is from its goal. The SNAIL algorithm performed incredibly better than MAML and similarely as LSTM on all tasks. The authors interpret this as SNAIL and LSTM being algorithms that are able to specialize very fast in such tasks, while MAML was more general-purpose and specialized slower and worse. Both SNAIL and LSTM were extremely close to the oracle after very few runs.

Visual navigation This task involves the navigation of a 3D maze. SNAIL's performance was compared to LSTM only, since the authors said that MAML scales badly in this kind of tasks. Each maze was generated randomly with a random goal, and the agent had two episodes to reach the goal as fast as it could. The authors noted how the optimal strategy emerges from SNAIL's behavior, as it spends the first episode exploring until finding the goal, then directly goes to it during the second episode. The authors also noted how LSTM has similar behaviour but tends to have a harder time remembering the optimal path by the second episode.

2 Paper review

The paper's presentation of the meta-learning frame of work shines an interesting light on the reinforcement learning frame of work. Its meta-RL application broaden the scope and design needed in RL. The fact we have to train over different tasks over multiple episodes requires a history to be also fed to the model in order for the policy to internalize solution the interactions between rewards, actions and states over different tasks. The precise case of the SNAIL algorithm is in the end rather simple in order to be allegedly as general as possible. A notable point is also the short presentation of attentive learning and its use in reinforcement learning solving some loss of information from the sequential aspect of the algorithms, when the episodes become numerous.

Furthermore, the experiments listed in the paper were very diverse and helped us understand better the use cases for meta-learning. That was especially true in the reinforcement learning context, as the proposed approach almost always outperformed other algorithms. This showed that the intuition behind combining convolutions for contextual width and attention for selectivity within the large context is sound. The experiments also provided an insight over multiple reinforcement learning applications that are relevant to this course.

The official implementation of this paper was not present, instead, there were four different community implementations. The implementations were of different qualities, the Pytorch one was well documented and worked easily, yet it only considered the supervised learning context. The one implementation that considered reinforcement learning was still in active development. Overall, the ideas behind this approach are intuitive. The reviewers on Open-review also made it clear that the results demonstrated by the paper were novel and promising. Even if the implementations are lacking, we're certain that this will change in due time. Going further, a natural extension to the paper's work would be to complete the PyTorch implementation for the RL experiments.

Bibliography

[1] Nikhil Mishra et al. "A Simple Neural Attentive Meta-Learner". In: *International Conference on Learning Representations*. 2018. URL: https://openreview.net/forum?id=B1DmUzWAW.

[2] Ashish Vaswani et al. “Attention Is All You Need”. In: 2017. URL: https://arxiv.org/abs/1706.03762.

[3] Sepp Hochreiter and Jurgen Schmidhuber. “Long Short Term Memory”. In: *Neural Computation*. 1997. URL: http://www.bioinf.jku.at/publications/older/2604.pdf.

[4] Chelsea Finn, Pieter Abbeel, and Sergey Levine. “Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks”. In: 2017. URL: https://arxiv.org/abs/1703.03400.

[5] Malcolm Strens. “A Bayesian Framework for Reinforcement Learning”. In: 2000. URL: https://www.ece.uvic.ca/~bctill/papers/learning/Strens_2000.pdf.

[6] Ian Osband and Benjamin Van Roy. “Why is Posterior Sampling Better than Optimism for Reinforcement Learning?” In: 2017. URL: https://arxiv.org/pdf/1607.00215.pdf.

[7] Thomas Jaksch, Ronald Ortner, and Peter Auer. “Near-optimal Regret Bounds for Reinforcement Learning”. In: 2010. URL: https://www.jmlr.org/papers/volume11/jaksch10a/jaksch10a.pdf.