Meta Reinforcement Learning

Manav Choudhary

April 4, 2019

Abstract

We present a brief review of the literature on metalearning, for the requirement of thesis work in Master's of Artificial Intelligence, Università della Svizzera Italiana.

1 Introduction

Definition of metalearning: Metalearning has been defined in several ways.
1. Schmidhuber et al. (1996) proposed that metalearners should be able to a) compare different learning methods, b)evaluate the impact of early learning on subsequent learning c) reason about learning strategies and select "useful" ones.
2. Thrun & Pratt (1998) proposed that given a family of tasks, training experience for each of these tasks, a family of performance measures, a metalearner's performance at each task is expected to improve with experience and with number of tasks.
3. Lemke et al. (2015) proposed that a metalearning system must include a learning subsystem, which adapts with experience, where experience is gained by exploiting meta-knowledge extracted either in a previous learning episode on a single dataset, and/or from different domains or problems. Also, the learning bias must be chosen dynamically.

We look at metalearning from the perspective of connectionist networks. A learning system has 4 major components:

1. computational architecture,
2. performance measure,
3. optimization algorithm and
4. initial values for the parameters of computational architecture.

For a given distribution of tasks, a metalearner, hence, should be able to learn the optimal configuration for each of these components.

Following work has been done recently with respect to the above 4 components:
1. Learning computational architecture: Zoph & Le (2016) and Real et al. (2017) used reinforcement learning (RL) and evolutionary algorithms.
2. Learning performance measure: Hochreiter et al. (2001) used a recurrent neural network (RNN) to learn the learning algorithm. Xu et al. (2018) used gradient updates through the meta parameters of the performance measure.
3. Learning optimization algorithm: Andrychowicz et al. (2016) used a RNN to calculate the gradients of the learner.
4. Learning initial values of parameters: Finn et al. (2017) used gradient descent to learn good initial weights. Rusu et al. (2018) used Latent Embedding Optimization to generate parameter distribution. Fernando et al. (2018) used Baldwin effect.

Hochreiter et al. (2001) implements a fixed-weight RNN which can improve its performance on new tasks even without modifying network weights, in supervised learning setting. Such fixed-weight RNN metalearner [Cotter & Conwell (1990), Younger et al. (1999, 2001), Prokhorov et al. (2002), Santoro et al. (2016)], are of much interest to us as they can learn both the performance measure and the optimizer algorithm based on the task distribution. Also, for such a metalearner, the frozen weights of the RNN are effectively the best suited initial values of parameters. Hence such an architecture addresses learning 3 of the 4 primary components.

Schmidhuber (1992, 1993) present a more general approach to metalearning, building upon the work from Schmidhuber (1987), with networks that can modify their own weights. Schmidhuber et al. (1996) presents the success-story algorithm implementation of meta-RL and self-referential policy of meta-RL.

We believe that all 4 components of a neural network architecture should be learnt optimally, specific to the task distribution, to achieve better performance, as is demonstrated by these works on individual components. We now focus on metalearners in the domain of reinforcement learning.

2 Research work in meta-reinforcement learning

The goal of meta-reinforcement learning [Schmidhuber et al. (1996), Schweighofer & Doya (2003)] is to learn a full-fledged reinforcement learning algorithm which is suited for a given distribution of tasks. The learner which learns such a specialized reinforcement learning algorithm can itself be a reinforcement learning algorithm. The learnt RL algorithm is independent of the meta RL algorithm.

Recently, fixed weight RNN metalearning has been explored in reinforcement learning domain. Wang et al. (2016) use a LSTM [Hochreiter & Schmidhuber (1997)] as a RNN for meta-reinforcement learning. In this work, r_{t-1} (reward) and a_{t-1} (action) is fed to the RNN as input along with the o_t (observation). This architecture [Fig. 1] is then trained using a standard reinforcement learning algorithm, advantage actor critic as detailed in Mnih et al. (2016), Mirowski et al. (2016). This work shows that the learnt RL algorithm can behave like a model based RL algorithm even when the meta-RL algorithm used to learn it is a model free algorithm, suggesting that the learnt RL algorithm is independent of the meta-RL algorithm. They perform experiments on bandits, two-step task, and maze navigation. This work is inspired by Hochreiter et al. (2001) which used the same principles in supervised learning and demonstrated that the learnt regression algorithm is much faster than gradient descent algorithm, which was used as the metalearning algorithm. The resultant effect that the model-free meta-RL algorithm learns a model-based RL algorithm, where both the model(M) and the controller(C) are implemented in the activations of the RNN, is related to the M-C architecture proposed by Schmidhuber (2015). Duan et al. (2016) is closely related to this work, they perform experiments on relatively unstructured task distribution, whereas this work focuses on structured task distributions.

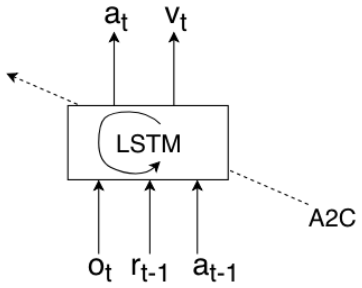

Figure 1: Fixed weight Meta-reinforcement learning

Building upon Wang et al. (2016), Wang et al. (2018) then presented a framework which highlights the roles of phasic dopamine and pre-frontal cortex in learning. They demonstrate the similarities between fixed weight meta-RL with learning in brains of monkeys and humans. They conduct further experiments detailing the model based RL behaviour of the learnt RL algorithm.

Ritter, Wang, Kurth-Nelson, Jayakumar, Blundell, Pascanu & Botvinick (2018) then demonstrates a fundamental limitation of meta-reinforcement learning framework presented in Wang et al. (2016) for repetitious tasks. They propose using a neural episodic memory in addition to the metalearning framework, which they refer to as episodic meta-reinforcement learning. In natural environments, tasks reoccur frequently, a meta-RL system without a long term episodic memory will have to relearn a task encountered before, albeit it will be faster at learning it than a hand-designed reinforcement learning algorithm. With episodic recall the agent can "pick up where it left off". They propose to save the cell state of LSTM to a differentiable neural dictionary (DND) at the end of an episode, with the embedding of task

"context" as the key. When the tasks reoccur, these cell states are retrieved based on the similarity of the context of the current task with these keys. To incorporate the retrieved cell state into the current cell state they use a reinstatement gate. This reinstatement gate is parameterized by weights and uses current input and previous hidden state to compute its value, similar to input/forget gates. They perform experiments on bandits and two-step task. However an episode after which the cell state is written to the DND is explicitly defined in this setup and the contextual cues are very well defined rather than discovering them, as is necessary in natural environments.

Building on the previous work, Ritter, Wang, Kurth-Nelson & Botvinick (2018) conduct further experiments analyzing the behaviour of episodic meta-reinforcement learning. They analyze the components of incremental model-free, incremental model-based, episodic model-free and episodic model-based behaviour in the policy learnt by episodic meta-reinforcement learning. Importantly, similar to Wang et al. (2016) they demonstrate that an episodic model-free meta-RL algorithm can give rise to learnt episodic model-based as well as incremental model-based algorithm. They also do further analysis of reinstatement gate activations to demonstrate that the reinstatement gate is more active on the cued trials rather than on uncued trials.

This line of work is exciting as it demonstrates how to learn specialized reinforcement learning algorithm in neural networks which can outperform hand-designed reinforcement learning algorithms.
r_{t-1}

References

Andrychowicz, M., Denil, M., Colmenarejo, S. G., Hoffman, M. W., Pfau, D., Schaul, T. & de Freitas, N. (2016), 'Learning to learn by gradient descent by gradient descent', *CoRR* **abs/1606.04474**.
URL: *http://arxiv.org/abs/1606.04474*

Cotter, N. E. & Conwell, P. R. (1990), Fixed-weight networks can learn, *in* '1990 IJCNN International Joint Conference on Neural Networks', pp. 553–559 vol.3.

Duan, Y., Schulman, J., Chen, X., Bartlett, P. L., Sutskever, I. & Abbeel, P. (2016), 'Rl$ˆ2$: Fast reinforcement learning via slow reinforcement learning', *CoRR* **abs/1611.02779**.
URL: *http://arxiv.org/abs/1611.02779*

Fernando, C., Sygnowski, J., Osindero, S., Wang, J., Schaul, T., Teplyashin, D., Sprechmann, P., Pritzel, A. & Rusu, A. (2018), Meta-learning by the baldwin effect, *in* 'Proceedings of the Genetic and Evolutionary Computation Conference Companion', GECCO '18, ACM, New York, NY, USA, pp. 1313–1320.
URL: *http://doi.acm.org/10.1145/3205651.3208249*

Finn, C., Abbeel, P. & Levine, S. (2017), 'Model-agnostic meta-learning for fast adaptation of deep networks', *CoRR* **abs/1703.03400**.
URL: *http://arxiv.org/abs/1703.03400*

Hochreiter, S. & Schmidhuber, J. (1997), 'Long short-term memory', *Neural Comput.* **9**(8), 1735–1780.
URL: *http://dx.doi.org/10.1162/neco.1997.9.8.1735*

Hochreiter, S., Younger, A. S. & Conwell, P. R. (2001), Learning to learn using gradient descent, *in* 'IN LECTURE NOTES ON COMP. SCI. 2130, PROC. INTL. CONF. ON ARTI NEURAL NETWORKS (ICANN-2001', Springer, pp. 87–94.

Lemke, C., Budka, M. & Gabrys, B. (2015), 'Metalearning: a survey of trends and technologies', *Artificial Intelligence Review* .

Mirowski, P., Pascanu, R., Viola, F., Soyer, H., Ballard, A. J., Banino, A., Denil, M., Goroshin, R., Sifre, L., Kavukcuoglu, K., Kumaran, D. & Hadsell, R. (2016), 'Learning to navigate in complex environments', *CoRR* **abs/1611.03673**.
URL: *http://arxiv.org/abs/1611.03673*

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T. P., Harley, T., Silver, D. & Kavukcuoglu, K. (2016), 'Asynchronous methods for deep reinforcement learning', *CoRR* **abs/1602.01783**.
URL: *http://arxiv.org/abs/1602.01783*

Prokhorov, D. V., Feldkarnp, L. A. & Tyukin, I. Y. (2002), Adaptive behavior with fixed weights in rnn: an overview, *in* 'Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02 (Cat. No.02CH37290)', Vol. 3, pp. 2018–2022 vol.3.

Real, E., Moore, S., Selle, A., Saxena, S., Suematsu, Y. L., Le, Q. V. & Kurakin, A. (2017), 'Large-scale evolution of image classifiers', *CoRR* **abs/1703.01041**.
URL: *http://arxiv.org/abs/1703.01041*

Ritter, S., Wang, J. X., Kurth-Nelson, Z. & Botvinick, M. M. (2018), 'Episodic control as meta-reinforcement learning', *bioRxiv* .
URL: *https://www.biorxiv.org/content/early/2018/07/03/360537*

Ritter, S., Wang, J. X., Kurth-Nelson, Z., Jayakumar, S. M., Blundell, C., Pascanu, R. & Botvinick, M. M. (2018), 'Been There, Done That: Meta-Learning with Episodic Recall', *ArXiv e-prints* .

Rusu, A. A., Rao, D., Sygnowski, J., Vinyals, O., Pascanu, R., Osindero, S. & Hadsell, R. (2018), 'Meta-Learning with Latent Embedding Optimization', *ArXiv e-prints* .

Santoro, A., Bartunov, S., Botvinick, M., Wierstra, D. & Lillicrap, T. (2016), Meta-learning with memory-augmented neural networks, *in* 'Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48', ICML'16, JMLR.org, pp. 1842–1850.
URL: *http://dl.acm.org/citation.cfm?id=3045390.3045585*

Schmidhuber, J. (1987), Evolutionary principles in self-referential learning. on learning now to learn: The meta-meta-meta...-hook, Diploma thesis, Technische Universitat Munchen, Germany.
URL: *http://www.idsia.ch/ juergen/diploma.html*

Schmidhuber, J. (1992), 'Learning to control fast-weight memories: An alternative to dynamic recurrent networks', *Neural Comput.* **4**(1), 131–139.
URL: *http://dx.doi.org/10.1162/neco.1992.4.1.131*

Schmidhuber, J. (1993), A neural network that embeds its own meta-levels, *in* 'IEEE International Conference on Neural Networks', pp. 407–412 vol.1.

Schmidhuber, J. (2015), 'On learning to think: Algorithmic information theory for novel combinations of reinforcement learning controllers and recurrent neural world models', *CoRR* **abs/1511.09249**.
URL: *http://arxiv.org/abs/1511.09249*

Schmidhuber, J., Zhao, J. & Wiering, M. (1996), Simple principles of metalearning, Technical report, SEE.

Schweighofer, N. & Doya, K. (2003), 'Meta-learning in reinforcement learning', *Neural Netw.* **16**(1), 5–9.
URL: *http://dx.doi.org/10.1016/S0893-6080(02)00228-9*

Thrun, S. & Pratt, L. (1998), Learning to learn, Kluwer Academic Publishers, Norwell, MA, USA, chapter Learning to Learn: Introduction and Overview, pp. 3–17.
URL: *http://dl.acm.org/citation.cfm?id=296635.296639*

Wang, J. X., Kurth-Nelson, Z., Kumaran, D., Tirumala, D., Soyer, H., Leibo, J. Z., Hassabis, D. & Botvinick, M. (2018), 'Prefrontal cortex as a meta-reinforcement learning system', *bioRxiv* .
URL: *https://www.biorxiv.org/content/early/2018/04/06/295964*

Wang, J. X., Kurth-Nelson, Z., Tirumala, D., Soyer, H., Leibo, J. Z., Munos, R., Blundell, C., Kumaran, D. & Botvinick, M. (2016), 'Learning to reinforcement learn', *CoRR* **abs/1611.05763**.
URL: *http://arxiv.org/abs/1611.05763*

Xu, Z., van Hasselt, H. & Silver, D. (2018), 'Meta-gradient reinforcement learning', *CoRR* **abs/1805.09801**.
URL: *http://arxiv.org/abs/1805.09801*

Younger, A. S., Conwell, P. R. & Cotter, N. E. (1999), 'Fixed-weight on-line learning', *IEEE Transactions on Neural Networks* **10**(2), 272–283.

Younger, A. S., Hochreiter, S. & Conwell, P. R. (2001), Meta-learning with backpropagation, *in* 'IJCNN'01. International Joint Conference on Neural Networks. Proceedings (Cat. No.01CH37222)', Vol. 3, pp. 2001–2006 vol.3.

Zoph, B. & Le, Q. V. (2016), 'Neural architecture search with reinforcement learning', *CoRR* **abs/1611.01578**.
URL: *http://arxiv.org/abs/1611.01578*