Week 12: Final Summary of RLVR

BEH Chuen Yang

Abstract

This final report summarizes the work done throughout the past weeks on Reinforcement Learning (RL), Large Language Models (LLMs), and RL with Verifiable Rewards (RLVR) in LLMs. Leaving the intermediate work aside, this report presents most of the key results and findings from all weeks of the project. We hope the report serves as a good roadmap for understanding RLVR and how it can be implemented in LLMs.

1 Introduction

Large Language Models (LLMs) have long been demonstrated capable of performing a staggering variety of tasks in the Natural Language Processing (NLP) domain, ranging from text generation, sentiment analysis, to machine translation, question answering (Brown et al. (2020)), and in recent years, even complex reasoning tasks (Lambert et al. (2025); Shao et al. (2024); DeepSeek-AI et al. (2025)).

It goes without saying, then, that being able to peer behind the curtain of such a powerful and apparently generalized model will be a fascinating endeavor. By gaining a deeper and more minute understanding of how LLMs work, what they are capable of, and what their limitations are, we will be able to harness their power in more suitable and effective ways, and even expand the boundaries of what is possible with deep learning in general.

In this brief paper, we present an executive summary of the prerequisite background in Reinforcement Learning (RL) and LLMs, highlighting key concepts and findings from our work [1], and progressively *build up to a high-level overview of RL with Verifiable Rewards (RLVR) in LLMs*, which is the main focus of this report. We supplement our descriptions and theorywork with experimental results from past weeks, in order to scaffold a more comprehensive understanding of how RL, LLM training, and RLVR work in practice, as well as how and where they may fall short.

2 Background

2.1 LLMs

2.2 RL

In the general case, RL is a machine learning paradigm where an agent learns to make a sequence of *actions* or *decisions* in an *environment* in order to maximize an *objective* (whose utility is defined by a cumulative reward signal) (Sutton & Barto (2018)).

2.2.1 RL Problems as Markov Decision Processes (MDPs)

Most commonly, RL problems are formulated as MDPs (Achiam (2018); Levine et al. (2023); Sutton & Barto (2018)) by *assuming the Markov property*, where environment dynamics can only depend on the current state and action, and not on past states or actions (Sutton & Barto (2018)).

[1] These works will not be cited in-line for brevity, but can be found in the references section at the end of this report. The reader should keep in mind that the previous works serve as the foundation for the current report, and can be referenced for more in-depth treatments of their respective topics.

This *Markov assumption* is **key** to enabling the theoretical performance guarantees of many RL algorithms (Sutton & Barto (2018)). Formally speaking, an MDP is defined as a tuple $(\mathcal{S}, \mathcal{A}, \tau, r, \gamma)$, where

- $\mathcal{S}$ is the set of *states* in the environment,
- $\mathcal{A}$ is the set of *actions* the agent can take,
- $\tau : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ (**guaranteed by the Markov property**) is the *transition function* that maps a state and an action to the next state,
- $r : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ is the *reward function* that maps a state, action, and next-state tuple to a real-valued reward, and
- $\gamma \in [0, 1]$ is (an optional) *discount factor* determining how "important" future rewards are compared to immediate rewards.

Typically, an agent interacts with the environment in an *episodic* manner, where it:

- starts in an initial state $s_0 \sim \rho_0$ (where ρ_0 is the initial state distribution),
- takes an action $a_t \in \mathcal{A}$,
- receives a next state $s_{t+1} \sim \tau(s_t, a_t)$,
- repeats the above until it reaches a terminal state $s_n \in \mathcal{S}$, whereupon the episode concludes.

This sequence of interactions $T = ((s_0, a_0), (s_1, a_1), \ldots, (s_n, a_n))$ is termed a *trajectory*, and it should be regarded analogously to one of many games of Tic-Tac-Toe, or a single playthrough of a video game.

Rehashing what was mentioned earlier, the goal of the agent is to learn an optimal policy $\pi^* : \mathcal{S} \to \mathcal{A}$ which allows it to *maximize the cumulative reward* it receives across all possible trajectories. Formally:

$$\begin{aligned}
\pi^* &= \arg\max_{\pi} \mathop{\mathbb{E}}_{T \sim \pi} \left[\sum_{t=0}^{n} \gamma^t r(s_t, a_t, s_{t+1}) \right] \\
&= \arg\max_{\pi} \mathop{\mathbb{E}}_{T \sim \pi} \left[\sum_{t=0}^{n} \gamma^t r(s_t, a_t) \right] \qquad \left(r(s_t, a_t) = \mathop{\mathbb{E}}_{s_{t+1} \sim \tau(s_t, a_t)} [r(s_t, a_t, s_{t+1})] \right)
\end{aligned} \tag{1}$$

References

Joshua Achiam. Spinning Up in Deep Reinforcement Learning. 2018. URL https://spinningup.openai.com/en/latest/.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020. URL https://arxiv.org/abs/2005.14165.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin

Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V. Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, Yuling Gu, Saumya Malik, Victoria Graf, Jena D. Hwang, Jiangjiang Yang, Ronan Le Bras, Oyvind Tafjord, Chris Wilhelm, Luca Soldaini, Noah A. Smith, Yizhong Wang, Pradeep Dasigi, and Hannaneh Hajishirzi. Tulu 3: Pushing frontiers in open language model post-training, 2025. URL https://arxiv.org/abs/2411.15124.

Sergey Levine, Kyle Stachowicz, Vivek Myers, Joey Hong, and Kevin Black, 2023. URL https://rail.eecs.berkeley.edu/deeprlcourse/.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018. URL http://incompleteideas.net/book/the-book-2nd.html.