The World Of Reinforcement Learning

Harini Anand
harini.anand@gmail.com

ABSTRACT

Reinforcement learning is an interesting area of machine learning which deals with how intelligent agents take actions in an environment to maximize the notion of gaining a cumulative reward. It is one of three basic machine learning paradigms. In Reinforcement Learning (RL), the agents are trained on the basis of a reward and punishment mechanism where it is rewarded for performing correct moves and punished for the incorrect ones. In doing so, the agent tries to minimize incorrect moves and maximize the correct ones. This paper focuses on exhaustive examples of RL being applied in real-life scenarios and solving global problems in a resource and time efficient manner.

1 Introduction

Reinforcement learning is a subset of Machine[1] Learning. It involves taking the right action to maximize reward and is employed by many machines to find the best possible path that it should take in a particular situation. Reinforcement learning differs from supervised learning as it refers to the training data which contains the answer key with it and thus, the model is trained with the correct answer itself. Whereas in reinforcement learning, there is no answer but the reinforcement agent decides what to do to perform the given task. In the absence of a training data set, it is bound to learn from its experience. In essence, Reinforcement Learning is the science of decision-making and learning about the optimal behavior in an environment to obtain the maximum reward.[2]

2 Applications

2.1 Autonomous Navigation

Some of the autonomous driving[3] tasks where reinforcement learning could be applied include trajectory optimization, controller optimization, motion planning and scenario-based learning[4] policies for highways. Lane changing can be achieved using Q-Learning while overtaking can be implemented by learning an overtaking policy while avoiding collision and maintaining a steady speed thereafter. Q-learning is a model-free method of reinforcement learning that will find the best course of action, based on the current state of the agent. AWS Deep Racer is an example of an autonomous racing car that has been designed to test out RL on a physical track. It uses cameras to visualize the runway and a reinforcement learning model to control the throttle and direction.[5]

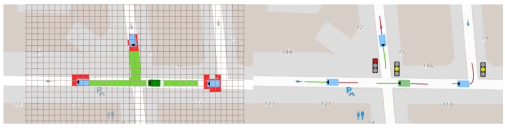

Fig. 4. Bird Eye View (BEV) 2D representation of a driving scene. Left demonstrates an occupancy grid. Right shows the combination of semantic information (traffic lights) with past (red) and projected (green) trajectories. The ego car is represented by a green rectangle in both images.

2.2 Trading and Finance

Supervised time series models are certain sequence of data observations that a system collects within specific periods of time, which can be used for predicting future sales as well as predicting stock prices.[6] An RL agent can decide on this task; whether to hold, buy, or sell, and is evaluated using market benchmark standards to ensure that it's performing

optimally and maximise profits. This automation brings consistency into the process, unlike previous manual methods where analysts would have to make every single[7] decision that would prove to be inconsistent due to human borne biases.

2.3 News Recommendation

User preferences are changing frequently, therefore recommending news to users based on reviews and likes could become obsolete quite quickly. With reinforcement learning, the RL system can track the reader's return behaviors, and optimise future suggestions. Construction of such a system would involve obtaining news[8] features, reader features, context features, and reader news features. News features include but are not limited to the content, headline, and publisher. Reader features refer to how the reader interacts with the content e.g clicks and shares. Context features include news aspects such as timing and freshness of the news. [9]A reward is then defined based on the users' behavior and how frequently they revisit the same genre of current affair topics.

2.4 Healthcare

RL can find optimal policies using prior experiences without the need for previous information on the mathematical model of biological systems.RL applications in healthcare domains range from dynamic treatment regimes in chronic diseases and[10] critical care to automated medical diagnosis from both unstructured and structured clinical data. This approach is more applicable than other control-based systems in healthcare and has now taken over medical report generation, identification of nodules/tumors and blood vessel blockage and analysis of medical reports.[11]

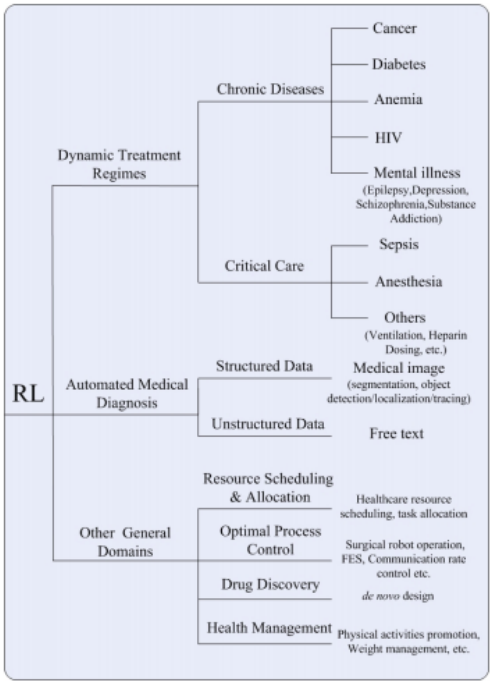

Fig. 2. The outline of application domains of RL in healthcare.

2.5 Traffic Control

For adaptive traffic signal control, RL models are trained with the objective of learning a policy using a value function that optimally controls the traffic light based on the current status of the traffic and the stationary vehicles at the junction. The decision-making needs to be dynamic depending upon the arrival rate of traffic from different directions, which ought to vary at different times of the day. [12].The conventional way of handling traffic seems to be limited due to this non-stationary behavior.

2.6 Marketing

Reinforcement Learning is used in various marketing spheres to develop techniques that maximize customer growth and strive for a balance between long-term and short-term rewards. Personalized product suggestions give customers what they want. The Reinforcement Learning bot is trained to handle situations where challenging barriers like reputation, limited customer [13]data, and consumers evolving mindset are dealt with. It learns the customer's requirements dynamically and analyses the behavior to serve high-quality recommendations. This increases the Return Of Interest and profit margins for the company. Analyzing which advertisement would suit the need at a given scenario is very hard by manual methods. The algorithm meets associated user preferences and chooses the perfect frequency for buyers. As a result, increased online conversions are transforming browsing into business thus improving net profit margins.[14]

2.7 Optimizing Contextual Bandits for recommendation systems

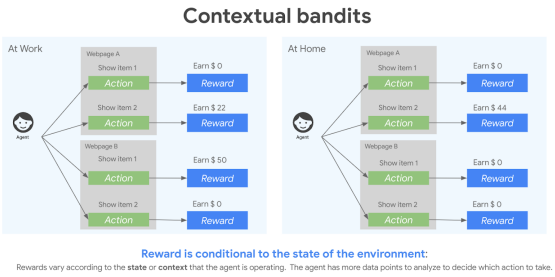

Figure 1: Auto ML Tables

The contextual bandit algorithm is an extension of the multi-armed bandit approach where the customer's environment is factored in, or in other words, context when choosing a bandit. The context affects how a reward is associated with each bandit, so as contexts change, the model learns to adapt its bandit choice. Recent work has demonstrated a meta learning approach (Auto ML Tables) to learn the Contextual bandit architecture and optimal set of parameters on recommendation system tasks.

The contextual bandit approach needs to find the maximum reward as well as reduce the reward loss when[15] exploring different contextual bandits. When judging the performance of a model, the metric that measures reward loss is regret—the difference between the cumulative reward from the optimal policy and the model's cumulative sum of rewards over time. The lower the regret, the better the model.

Contextual bandits is an exciting method for solving the complex problems businesses face today, and meta learning makes it accessible for a wide range of organizations and performs extremely well.

2.8 Industrial Applications

2.8.1 Semi-analytical Industrial Cooling System Model

[16] A hybrid industrial cooling system model that embeds [17] analytical solutions within a multi physics simulation which balances simplicity with simulation fidelity and its ability to be interpreted correctly. It can be leveraged to control and efficiently optimize industrial processes such as cooling and heating due to its wide range of applications in residential and commercial scenarios.[18]

However, applying RL in live facilities comes with its own set of technical challenges, such as limited exploration where the agent must act cautiously to prevent overheating a critical facility and not exploring all combinations of the state which will result in generating low-variance data. Solutions are not robust enough to different equipment conditions, weather patterns, and other variations, as training is limited to experiences with a singular system during a fixed amount of real-time interactions.

Interacting with a simulator rather than a live facility greatly increases the amount of available data and also allows the agent to explore a broader action space safely. A high-level design for an Industrial Task Suite (ITS) is used, that defines highly parameterized facility configurations to enable flexible experimentation and development of suitable agents.

EnergyPlus is a popular open-source library that provides full-featured functionality, but there are fidelity limitations that can be crucial when simulating real-world environments.

An apparatus is used to keep the temperature of a device from exceeding limits for safety concerns. Cooling systems use water because it has a high boiling point and specific heat. The chilled water compartment is responsible for keeping the rooms cool, within the desired temperature range. The condenser water compartment extracts heat from the chilled water side. No water mass is exchanged between the loops, thus being restricted to heat transfer via modes of conduction and convection. The cooling process can be achieved in the cooling tower by the latent heat of evaporation and the chillers through the vapor compression cycle. There are two types of temperatures that are vital in these cooling systems: dry bulb temperature and wet bulb temperature. They are used to calculate the relative humidity of the air. Simulation speed and fidelity are not exhaustive yet and need more data-based work to improve overall accuracy.

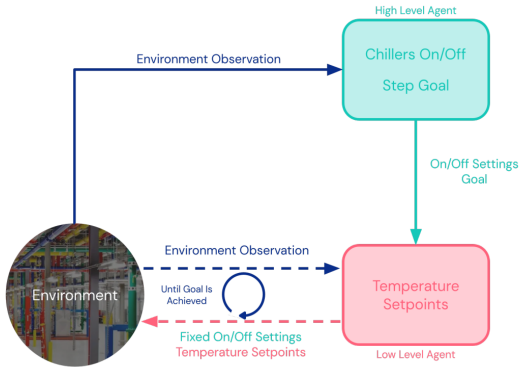

Figure 2: Hierarchical reinforcement learning agent loop

0

2.8.2 Optimizing Industrial HVAC Systems with Hierarchical Reinforcement Learning

Optimizing chiller plants, a component of HVAC systems are responsible for removing heat from the buildings typically via a liquid refrigerant. Chillers should be only turned on and off every few hours and usage should be spread equally among chillers to avoid unnecessary wear and tear. At the same time, building temperature needs to be maintained within specified bounds throughout chiller cycling. Hierarchical reinforcement learning offers the ability to reason across different time scales and avoids the necessity of extensive reward engineering to meet building temperature requirements. Simulation steps take on average 20 to 40 seconds. As a result, simulating and training on a large number of roll outs is cumbersome. Similar to using real-world offline data, agents must be sample efficient and learn with a finite amount of data.[19]

Many real-world problems are multi-objective, making a single reward function to be difficult. Using one chiller at all times or switching between chillers every few hours for equal utilization helps meet building temperature requirements. Real-World Feasibility Controlling HVAC systems is a safety-critical task. Temperature violations can be harmful to occupants and chiller wear and tear can be monetarily expensive. Additionally, simulators often are nonexistent, necessitating agents to learn from limited data.

3 Conclusion

In this paper, real world applications of Reinforcement Learning was presented which focused on the core concepts. Furthermore, the paper provides a good knowledge-base about Reinforcement Learning, their limitations and applications on particular problem domains. Overall, we can conclude that Reinforcement learning is a very active research area. Significant progress has been made to advance the field and apply it in real life. As seen in the above-mentioned applications, it has been crucial in accelerating the pace and exponentially increased the scope of solving these large-scale problems effectively. [20]

References

[1] Yuxi Li. Deep reinforcement learning: An overview. *arXiv preprint arXiv:1701.07274*, 2017.

[2] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[3] Derrick Mwiti. Applications of reinforcement learning, 2022.

[4] Shai Shalev-Shwartz, Shaked Shammah, and Amnon Shashua. Safe, multi-agent, reinforcement learning for autonomous driving. *arXiv preprint arXiv:1610.03295*, 2016.

[5] Sen Wang, Daoyuan Jia, and Xinshuo Weng. Deep reinforcement learning for autonomous driving. *arXiv preprint arXiv:1811.11329*, 2018.

[6] Zihao Zhang, Stefan Zohren, and Stephen Roberts. Deep reinforcement learning for trading. *The Journal of Financial Data Science*, 2(2):25–40, 2020.

[7] John Moody and Matthew Saffell. Reinforcement learning for trading. *Advances in Neural Information Processing Systems*, 11, 1998.

[8] M Mehdi Afsar, Trafford Crump, and Behrouz Far. Reinforcement learning based recommender systems: A survey. *ACM Computing Surveys (CSUR)*, 2021.

[9] Guanjie Zheng, Fuzheng Zhang, Zihan Zheng, Yang Xiang, Nicholas Jing Yuan, Xing Xie, and Zhenhui Li. Drn: A deep reinforcement learning framework for news recommendation. In *Proceedings of the 2018 world wide web conference*, pages 167–176, 2018.

[10] Chao Yu, Jiming Liu, Shamim Nemati, and Guosheng Yin. Reinforcement learning in healthcare: A survey. *ACM Computing Surveys (CSUR)*, 55(1):1–36, 2021.

[11] Andre Esteva, Alexandre Robicquet, Bharath Ramsundar, Volodymyr Kuleshov, Mark DePristo, Katherine Chou, Claire Cui, Greg Corrado, Sebastian Thrun, and Jeff Dean. A guide to deep learning in healthcare. *Nature medicine*, 25(1):24–29, 2019.

[12] Ana LC Bazzan. Opportunities for multiagent systems and multiagent reinforcement learning in traffic control. *Autonomous Agents and Multi-Agent Systems*, 18(3):342–375, 2009.

[13] Matthew G Reyes. Reinforcement learning in a marketing game. In *Intelligent Computing-Proceedings of the Computing Conference*, pages 705–724. Springer, 2019.

[14] Vinay Singh, Brijesh Nanavati, Arpan Kumar Kar, and Agam Gupta. How to maximize clicks for display advertisement in digital marketing? a reinforcement learning approach. *Information Systems Frontiers*, pages 1–18, 2022.

[15] Zi Yang. Better bandit building: Advanced personalization the easy way with automl tables, 2022.

[16] Yuri Chervonyi, Praneet Dutta, Piotr Trochim, Octavian Voicu, Cosmin Paduraru, Crystal Qian, Emre Karagozler, Jared Quincy Davis, Richard Chippendale, Gautam Bajaj, et al. Semi-analytical industrial cooling system model for reinforcement learning. *arXiv preprint arXiv:2207.13131*, 2022.

[17] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[18] Richard S Sutton, Andrew G Barto, et al. Reinforcement learning. *Journal of Cognitive Neuroscience*, 11(1):126–134, 1999.

[19] William Wong, Praneet Dutta, Octavian Voicu, Yuri Chervonyi, Cosmin Paduraru, and Jerry Luo. Optimizing industrial hvac systems with hierarchical reinforcement learning. *arXiv preprint arXiv:2209.08112*, 2022.

[20] Peter Dayan and Yael Niv. Reinforcement learning: the good, the bad and the ugly. *Current opinion in neurobiology*, 18(2):185–196, 2008.