Reinforcement Learning: A Review and Its Applications in IoT for Robotics

1.Kondanna Kanamaneni, Asst.Professor, Dept of CSE, Srinivasa Ramanujan Institute of Technology

2.K.Sudhakar, Asst.Professor, Dept of CSE, Srinivasa Ramanujan Institute of Technology

3.Annavaram Kiran Kumar, Asst.Professor, Dept of CSE, Srinivasa Ramanujan Institute of Technology

4.Ejaru Naga Prabhakar, Asst.Professor, Dept of CSE, Srinivasa Ramanujan Institute of Technology

ABSTRACT

Reinforcement Learning (RL) is a branch of machine learning where agents learn to make decisions through interactions with an environment, receiving feedback in the form of rewards or penalties. Over the past few decades, RL has gained significant attention due to its ability to solve complex sequential decision-making problems in dynamic and uncertain environments. This review provides a comprehensive overview of the foundational concepts of RL, including key algorithms such as Q-learning, policy gradient methods, and deep reinforcement learning (DRL). We also discuss the challenges in applying RL, including sample inefficiency, exploration-exploitation trade-offs, and the high computational costs of training. Despite its successes, RL in robotics faces several challenges, particularly in terms of real-time performance, safety, and transferability to real-world environments. In this review, we provide insights into the latest advancements in RL research and discuss promising future directions, including the development of more sample-efficient algorithms, improved safety mechanisms, and the incorporation of human-robot collaboration. The review concludes with a reflection on the future of RL in robotics, emphasizing its potential to revolutionize various industries by enabling intelligent, adaptable, and autonomous robotic systems.

KEYWORDS: deep reinforcement learning (DRL), decision-making problems, human-robot collaboration

1. INTRODUCTION

Reinforcement Learning (RL) is an area of machine learning concerned with how agents should take actions in an environment in order to maximize a cumulative reward. It is inspired by behavioral psychology, where learning is driven by feedback from the environment. Unlike supervised learning, where training data includes input-output pairs, RL agents learn from the consequences of their actions through interaction with the environment, typically by trial and error. This process allows the agent to develop strategies for achieving specific goals, making RL a powerful tool for decision-making problems in dynamic, uncertain settings. Over the past few decades, RL has evolved to solve increasingly complex tasks and has gained significant attention due to its remarkable success in various fields.

The core idea behind RL is simple: an agent interacts with an environment, taking actions, and receiving rewards or penalties. The agent aims to learn an optimal policy, which is a strategy that maps states of the environment to the best possible actions. The learning process is often modeled as a **Markov Decision Process (MDP)**, where each state transition is based on the agent's actions and the rewards received. The agent's goal is to maximize the cumulative reward over time, which is often expressed as a **reward function** or **return**. However, the challenge lies in balancing the exploration of new actions with the exploitation of known actions that lead to high rewards, a dilemma referred to as the **exploration-exploitation trade-off**.

Traditional RL methods, such as **Q-learning** and **temporal difference (TD) learning**, have been foundational in developing solutions for discrete, low-dimensional problems. These methods rely on value-based approaches where the agent estimates the expected reward for each action and chooses the one that maximizes this expected reward. However, these methods have limitations when dealing with complex, high-dimensional data, such as images or continuous action spaces, which is a common scenario in robotics. To address these challenges, the field has evolved with the introduction of **Deep Reinforcement Learning (DRL)**, which leverages deep neural networks to approximate complex value functions and policies, allowing RL to scale to more complex problems.

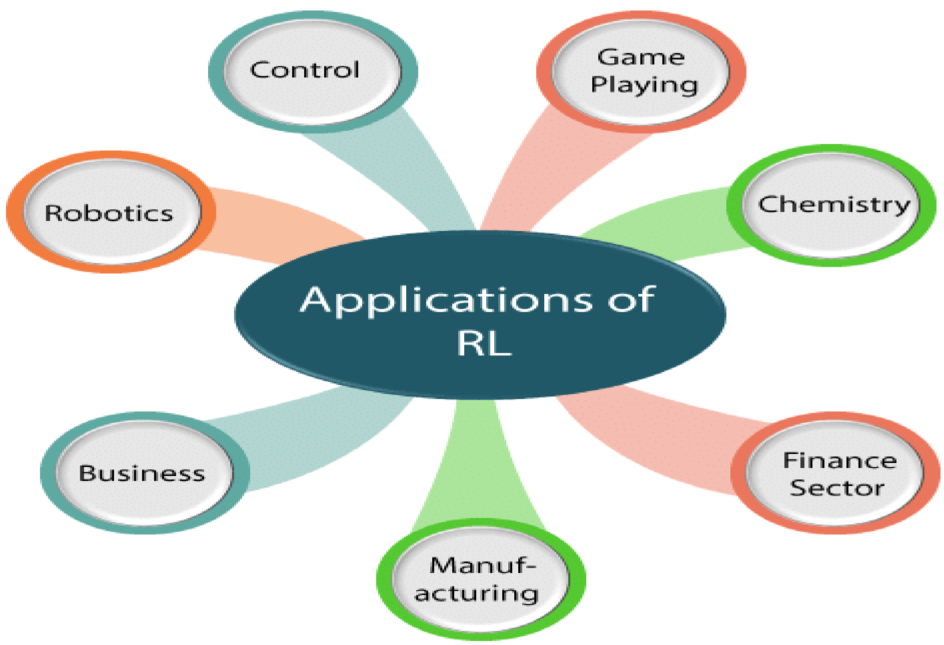

Fig 1: Applications of Reinforcement Learning

One of the most notable achievements in the field of RL came with the success of **Deep Q-Networks (DQN)**, a method that combines Q-learning with deep learning. DQN was able to achieve human-level performance in classic video games, marking a breakthrough in the application of RL to high-dimensional spaces. Since then, DRL has rapidly advanced, with techniques such as **Actor-Critic methods**, **Proximal Policy Optimization (PPO)**, and **Trust Region Policy Optimization (TRPO)** becoming popular for solving continuous control problems. These advancements have paved the way for applying RL to a wide range of real-world applications, particularly in fields that require sequential decision-making, such as robotics.

Robotics, with its diverse challenges and high-dimensional nature, provides an ideal application domain for RL. The ability of RL to adapt and learn from experience is particularly valuable for robots that need to perform tasks in dynamic environments, where pre-programmed rules are insufficient, and flexibility and adaptability are key. In robotics, RL has been used for tasks such as robot navigation, manipulation, control, and planning. For example, robots can use RL to learn how to grasp objects, navigate through unfamiliar terrain, or even perform complex assembly tasks. The dynamic and often unstructured nature of real-world environments makes RL particularly suitable, as it enables robots to improve their performance over time through feedback and interaction.

Despite its potential, RL in robotics faces several challenges. One major obstacle is **sample inefficiency**, which refers to the high number of interactions required to learn an effective policy. In robotics, where real-world interactions are costly and time-consuming, collecting sufficient data can be a significant limitation. This problem is especially pronounced when training robots on physical hardware, as each trial might require substantial time and resources. **Simulation-to-real transfer** is another challenge, as robots trained in simulated environments often struggle to generalize to the real world due to discrepancies between the simulation and the actual environment.

To address these challenges, various techniques have been proposed. **Sim2Real** methods aim to bridge the gap between simulated training and real-world deployment by using domain adaptation or domain randomization techniques. This allows a robot trained in a simulated environment to transfer its learned policies to real-world tasks more effectively. Additionally, approaches such as **meta-learning** and **few-shot learning** have been explored to improve the sample efficiency of RL algorithms, enabling robots to learn from fewer interactions and adapt quickly to new tasks.

RL's integration with **deep learning** has been particularly transformative for robotics, allowing robots to learn directly from high-dimensional sensor data such as images, videos, and sensory inputs. This combination of RL and deep learning, known as **Deep Reinforcement Learning (DRL)**, has led to major advances in enabling robots to learn from raw data without explicit feature engineering. For example, robots can use convolutional neural networks (CNNs) to process visual information and use this information in their decision-making processes, greatly enhancing their ability to interact with and navigate through complex environments.

Real-world applications of RL in robotics have demonstrated its potential in a wide range of industries. **Autonomous vehicles**, for example, rely heavily on RL for decision-making in complex driving scenarios, where real-time adjustments are necessary based on unpredictable traffic conditions. **Industrial robots** have successfully used RL to optimize their task execution, such as assembly line robots learning to perform pick-and-place operations with minimal supervision. Additionally, **assistive robots**, such as those used in healthcare and rehabilitation, have been improved through RL, allowing them to adapt to individual user needs and environments.

Despite its success, the application of RL in robotics is still an evolving field with many open challenges. Key issues include the difficulty of ensuring safety and stability during learning, as agents might take harmful or undesirable actions during exploration. Additionally, ensuring the

robustness of learned policies in real-world, unstructured environments is a critical area of research. New strategies, such as **safe RL**, which incorporates constraints into the learning process to prevent unsafe actions, are being actively explored to mitigate these risks.

In conclusion, reinforcement learning has the potential to revolutionize the field of robotics by enabling robots to autonomously learn complex tasks in dynamic and uncertain environments. While significant progress has been made in improving RL algorithms, addressing challenges such as sample efficiency, real-world applicability, and safety remains crucial for the widespread deployment of RL in robotics. Future research is likely to focus on making RL more sample-efficient, robust, and scalable, bringing us closer to the vision of intelligent, adaptable robots capable of performing a wide range of tasks across industries.

2. LITERATURE SURVEY

Algorithm	**Key Contributions**	**Applications**	**Limitations/Challenges**
Q-Learning	Introduced model-free RL; off-policy learning using Q-values.	Used for discrete control tasks, such as navigation and path planning in robotic systems.	Struggles with high-dimensional spaces and continuous actions. Sample inefficiency.
Deep Q-Network (DQN)	Combines Q-learning with deep neural networks to handle high-dimensional sensory input (e.g., images).	Robotic control, video game AI (e.g., Atari games), autonomous driving.	Training instability, high computational cost, sample inefficiency.
Policy Gradient Methods	Directly optimizes the policy by estimating gradients to maximize rewards.	Complex robotic tasks such as in-hand manipulation, grasping, and robotics control.	High variance in gradient estimates, slow convergence.
Proximal Policy Optimization (PPO)	Uses clipped objective function to improve training stability in policy optimization.	Robotic locomotion, manipulation tasks, industrial robots, and robotic arms.	Sensitive to hyperparameters, may not work well for highly complex tasks.

Trust Region Policy Optimization (TRPO)	Optimizes the policy by limiting the step size to avoid large updates that destabilize learning.	Robot control tasks like walking, balancing, and robotic manipulation.	Computationally expensive, challenging to implement in real-time applications.
Asynchronous Advantage Actor-Critic (A3C)	Uses multiple agents to train asynchronously, improving training speed and stability.	Robotic navigation, robotic manipulation, and multi-robot coordination.	Requires multiple workers, higher resource consumption for parallel training.
Deep Deterministic Policy Gradient (DDPG)	Adaptation of DQN for continuous action spaces, combining deterministic policy with Q-learning.	Robotic arms, drone control, continuous control tasks in robotics.	Sensitive to hyperparameters, slow convergence.
Soft Actor-Critic (SAC)	Maximum entropy reinforcement learning approach that encourages exploration while learning optimal policies.	Robotic locomotion, continuous control tasks like biped walking, quadruped locomotion.	High computational cost, requires fine-tuning.
A3C (Asynchronous Advantage Actor-Critic)	Parallelizes experience collection across multiple agents, leading to improved exploration.	Complex robotic manipulation tasks, navigation tasks in dynamic environments.	Requires large computational resources, and sensitive to training environments.
Monte Carlo Tree Search (MCTS)	Uses a tree structure to simulate possible outcomes and make decisions based on value propagation.	Used in combination with RL for complex planning tasks in robotics (e.g., motion planning).	Can be slow for high-dimensional problems; may not scale well in real-time tasks.
Q-Prop (Q-Propagation)	Combines model-free RL with value propagation across an extended state space for high-	Used in high-dimensional continuous control tasks like robotic arm manipulation and walking.	High sample inefficiency, need for large datasets to generalize effectively.

	dimensional control.		
DDPG + Hindsight Experience Replay	Uses Hindsight Experience Replay (HER) to improve exploration efficiency and handle sparse rewards.	Robotic manipulation tasks, in particular, for tasks with sparse rewards like object reorientation.	Sample inefficiency, especially in environments with very sparse rewards.
Meta-Reinforcement Learning (Meta-RL)	Uses prior experiences to quickly adapt to new tasks, optimizing transfer learning.	Applications in multi-task robotics, where robots need to generalize across various tasks.	Computationally intensive, challenges in generalizing across very diverse tasks.
Hierarchical Reinforcement Learning (HRL)	Decomposes a task into a hierarchy of sub-tasks to improve learning efficiency.	Long-horizon robotic tasks, where tasks can be broken into sub-tasks, such as cooking or assembly.	Difficulty in designing hierarchies and handling large task spaces.
Evolution Strategies (ES)	Optimization of policies through the use of evolutionary algorithms, instead of gradients.	Applications in continuous control tasks, such as locomotion and robotic navigation.	May require a large number of evaluations; difficult to fine-tune hyperparameters.
Multi-Agent Reinforcement Learning (MARL)	Models interactions between multiple agents to learn coordinated behaviors.	Collaborative tasks in robotics, such as multi-robot exploration, warehouse robots, and drones.	Communication overhead, difficulty in ensuring stability in highly dynamic multi-agent environments.
Sim2Real Transfer Learning	Transfers policies learned in simulations to real-world robotic tasks.	Simulation-based training of robotic arms, navigation in unstructured environments.	Large gap between simulation and reality, leading to poor real-world performance.

Inverse Reinforcement Learning (IRL)	Learns the reward function from expert demonstrations, and then uses RL to learn optimal policies.	Imitation learning for robotics, teaching robots from expert demonstrations in tasks like cooking or assembly.	Requires high-quality expert data, limited scalability for complex tasks.
Learning from Demonstration (LfD)	Leverages expert demonstrations as input for RL-based robot learning.	Applications in robot learning by imitation, especially for complex or high-precision tasks.	High dependency on expert data, limited generalization to unseen situations.
Deep Recurrent Q-Learning (DRQN)	Combines deep learning with recurrent networks to handle partially observable environments.	Navigation and decision-making in partially observable environments (e.g., robots with vision).	Requires substantial computational power, challenging to implement efficiently.
Distributed RL (DRL)	Allows multiple agents to learn simultaneously, accelerating the learning process.	Large-scale robotics applications with multiple agents, like coordinated drone control or warehouse robots.	Difficulty in synchronizing agents, communication latency.
Q-Learning with Hindsight Experience Replay	Uses experiences from unsuccessful trials to retroactively adjust the reward function.	Robotic manipulation in environments with sparse rewards, such as grasping objects.	Sample inefficiency, limited generalization in complex environments.
Neuro-Inspired Reinforcement Learning	Emulates the brain's reward system to enhance the efficiency of RL algorithms.	Applications in cognitive robotics and adaptive systems.	High computational cost, not well suited for real-time applications.

3. CONCLUSION

The application of RL in robotics has expanded into diverse areas, including robotic locomotion, manipulation, grasping, and collaborative multi-robot systems. Methods like Inverse Reinforcement Learning (IRL) and Learning from Demonstration (LfD) have enabled robots to learn complex tasks by imitating human behavior or expert demonstrations. However, issues such as the simulation-to-reality gap, high training time, and the need for large amounts of data still present significant barriers to the widespread deployment of RL-based robotic systems. Recent advancements, including techniques like Meta-RL, Hierarchical Reinforcement Learning (HRL), and Distributed RL, have shown promise in overcoming some of these challenges, especially when applied to more dynamic and multi-task environments. These methods allow robots to learn faster, adapt to new tasks with minimal retraining, and operate efficiently in real-world conditions. Moreover, the integration of RL with deep learning and recurrent networks has enabled robots to handle complex sensory input, such as visual and auditory data, which is critical for tasks in unstructured or partially observable environments. Despite these advancements, the field of RL for robotics is still evolving. Future research is likely to focus on improving the efficiency and scalability of these methods, enabling robots to learn in more natural, flexible, and real-time ways. Moreover, as RL methods continue to evolve, it will be essential to address their limitations, particularly in terms of real-time decision-making, exploration strategies, and safe learning in high-stakes environments.

REFERENCES

[1] V. Mnih, K. Kavukcuoglu, D. Silver, et al., "Human-level control through deep reinforcement learning," Nature, vol. 518, no. 7540, pp. 529-533, 2015.

[2] T. P. Lillicrap, J. J. Hunt, A. Pritzel, et al., "Continuous control with deep reinforcement learning," in Proc. Int. Conf. on Learning Representations (ICLR), 2016.

[3] S. Khaleelullah, K. S. Reddy, A. S. Reddy, D. Kedhar, M. Bhavana and P. Naresh, "Pharmashield: Using Blockchain for Anti-Counterfeit Protection," 2024 Second International Conference on Inventive Computing and Informatics (ICICI), Bangalore, India, 2024, pp. 529-534, doi: 10.1109/ICICI62254.2024.00092.

[4] T. Aruna, P. Naresh, B. A. Kumar, B. K. Prakash, K. M. Mohan and P. M. Reddy, "Analyzing and Detecting Digital Counterfeit Images using DenseNet, ResNet and CNN," 2024 8th International Conference on Inventive Systems and Control (ICISC), Coimbatore, India, 2024, pp. 248-252, doi: 10.1109/ICISC62624.2024.00049.

[5] G. Chanakya, N. Bhargavee, V. N. Kumar, V. Namitha, P. Naresh and S. Khaleelullah, "Machine Learning for Web Security: Strategies to Detect and Prevent Malicious Activities," 2024 Second International Conference on Intelligent Cyber Physical Systems and Internet of Things (ICoICI), Coimbatore, India, 2024, pp. 59-64, doi: 10.1109/ICoICI62503.2024.10696229.

[6] Balakrishna, C. ., Sapkal, A. ., Chowdary, B., Rajyalakshmi, P., Kumar, V. S. ., & Gupta, K. G. . (2023). Addressing the IoT Schemes for Securing the Modern Healthcare Systems with Block chain Neural Networks. International Journal on Recent and Innovation Trends in Computing and Communication, 11(7s), 347–352. https://doi.org/10.17762/ijritcc.v11i7s.7009

[7] Ravi, C., Raghavendran, C. V., Satish, G. N., Reddy, K. V. R., Reddy, G. K., & Balakrishna, C. (2023). ANN and RSM based Modeling of Moringa Stenopetala Seed Oil Extraction: Process Optimization and Oil Characterization. International Journal on Recent and Innovation Trends in Computing and Communication, 11(7s), 329–338. https://doi.org/10.17762/ijritcc.v11i7s.7007.

[8] P. Rajyalakshmi, C. Balakrishna, E. Swarnalatha, B. S. Swapna Shanthi and K. Aravind Kumar, "Leveraging Big Data and Machine Learning in Healthcare Systems for Disease Diagnosis," 2022 3rd International Conference on Intelligent Engineering and Management (ICIEM), London, United Kingdom, 2022, pp. 930-934, doi: 10.1109/ICIEM54221.2022.9853149.

[9] C. Nagesh, B. Divyasree, K. Madhu, T. Allisha, S. Datta Koushlk and P. Naresh, "Enhancing E-Government through Sentiment Analysis: A Dual Approach Using Text and Facial Expression Recognition," 2024 International Conference on Science Technology Engineering and Management (ICSTEM), Coimbatore, India, 2024, pp. 1-6, doi: 10.1109/ICSTEM61137.2024.10560678.

[10] Nagesh, C., Chaganti, K.R. , Chaganti, S. , Khaleelullah, S., Naresh, P. and Hussan, M. 2023. Leveraging Machine Learning based Ensemble Time Series Prediction Model for Rainfall Using SVM, KNN and

Advanced ARIMA+ E-GARCH. *International Journal on Recent and Innovation Trends in Computing and Communication*. 11, 7s (Jul. 2023), 353–358. DOI:https://doi.org/10.17762/ijritcc.v11i7s.7010.

[11] S. Khaleelullah, P. Marry, P. Naresh, P. Srilatha, G. Sirisha and C. Nagesh, "A Framework for Design and Development of Message sharing using Open-Source Software," 2023 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS), Erode, India, 2023, pp. 639-646, doi: 10.1109/ICSCDS56580.2023.10104679.

[12] P. Naresh, K. Pavan kumar, and D. K. Shareef, 'Implementation of Secure Ranked Keyword Search by Using RSSE,' International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Vol. 2 Issue 3, March – 2013.

[13] M. I. Thariq Hussan, D. Saidulu, P. T. Anitha, A. Manikandan and P. Naresh (2022), Object Detection and Recognition in Real Time Using Deep Learning for Visually Impaired People. IJEER 10(2), 80-86. DOI: 10.37391/IJEER.100205.

[14] J. Schulman, F. Wolski, P. Dhariwal, et al., "Proximal policy optimization algorithms," in Proc. 34th Int. Conf. on Machine Learning (ICML), 2017.

[15] Naresh, P., & Suguna, R. (2019). Association Rule Mining Algorithms on Large and Small Datasets: A Comparative Study. 2019 International Conference on Intelligent Computing and Control Systems (ICCS). DOI:10.1109/iccs45141.2019.9065836.

[16] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "A brief survey of deep reinforcement learning," IEEE Signal Process. Mag., vol. 34, no. 6, pp. 26-38, 2017.

[17] C. Nagesh, B. Divyasree, K. Madhu, T. Allisha, S. Datta Koushlk and P. Naresh, "Enhancing E-Government through Sentiment Analysis: A Dual Approach Using Text and Facial Expression Recognition," 2024 International Conference on Science Technology Engineering and Management (ICSTEM), Coimbatore, India, 2024, pp. 1-6, doi: 10.1109/ICSTEM61137.2024.10560678.

[18] T. Haarnoja, A. Zhou, P. Hartikainen, et al., "Advantage-weighted regression: Shortening the path to deep reinforcement learning," in Proc. 34th Int. Conf. on Machine Learning (ICML), 2017.

[19] Sunder Reddy, K. S. ., Lakshmi, P. R. ., Kumar, D. M. ., Naresh, P. ., Gholap, Y. N. ., & Gupta, K. G. . (2024). A Method for Unsupervised Ensemble Clustering to Examine Student Behavioral Patterns. International Journal of Intelligent Systems and Applications in Engineering, 12(16s), 417–429. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/4854.

[20] Tamar, M. Ghavamzadeh, and S. Mannor, "A survey of multi-objective reinforcement learning," in Proc. 29th Conf. on Neural Information Processing Systems (NeurIPS), 2016.

[21] P. Naresh, B. Akshay, B. Rajasree, G. Ramesh and K. Y. Kumar, "High Dimensional Text Classification using Unsupervised Machine Learning Algorithm," 2024 3rd International Conference on Applied Artificial Intelligence and Computing (ICAAIC), Salem, India, 2024, pp. 368-372, doi: 10.1109/ICAAIC60222.2024.10575444.

[22] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in Proc. 3rd Int. Conf. on Learning Representations (ICLR), 2014.

[23] T. Aruna, P. Naresh, A. Rajeshwari, M. I. T. Hussan and K. G. Guptha, "Visualization and Prediction of Rainfall Using Deep Learning and Machine Learning Techniques," 2022 2nd International Conference on Technological Advancements in Computational Sciences (ICTACS), Tashkent, Uzbekistan, 2022, pp. 910-914, doi: 10.1109/ICTACS56270.2022.9988553.

[24] V. Krishna, Y. D. Solomon Raju, C. V. Raghavendran, P. Naresh and A. Rajesh, "Identification of Nutritional Deficiencies in Crops Using Machine Learning and Image Processing Techniques," 2022 3rd International Conference on Intelligent Engineering and Management (ICIEM), London, United Kingdom, 2022, pp. 925-929, doi: 10.1109/ICIEM54221.2022.9853072.

[25] P, N., & R Suguna. (2022). Enhancing the Performance of Association Rule Generation over Dynamic Data using Incremental Tree Structures. International Journal of Next-Generation Computing, 13(3). https://doi.org/10.47164/ijngc.v13i3.806.

[26] X. Huang and P. Stone, "A survey of deep reinforcement learning for multi-agent systems," IEEE Trans. Neural Netw. Learn. Syst., vol. 28, no. 6, pp. 1371-1385, 2017.
[27] S. Gu, Y. Li, and S. Levine, "Deep reinforcement learning for robotic manipulation with asynchronous policy updates," in Proc. IEEE Int. Conf. on Robotics and Automation (ICRA), 2017.
[28] B. Narsimha, Ch V Raghavendran, Pannangi Rajyalakshmi, G Kasi Reddy, M. Bhargavi and P. Naresh (2022), Cyber Defense in the Age of Artificial Intelligence and Machine Learning for Financial Fraud Detection Application. IJEER 10(2), 87-92. DOI: 10.37391/IJEER.100206.
[29] P. Naresh, P. Srinath, K. Akshit, M. S. S. Raju and P. VenkataTeja, "Decoding Network Anomalies using Supervised Machine Learning and Deep Learning Approaches," 2023 2nd International Conference on Automation, Computing and Renewable Systems (ICACRS), Pudukkottai, India, 2023, pp. 1598-1603, doi: 10.1109/ICACRS58579.2023.10404866.
[30] T. Schaul, J. Quan, I. Antonoglou, and D. Silver, "Prioritized experience replay," in Proc. 31st Int. Conf. on Machine Learning (ICML), 2015.
[31] Hussan, M.I. & Reddy, G. & Anitha, P. & Kanagaraj, A. & Pannangi, Naresh. (2023). DDoS attack detection in IoT environment using optimized Elman recurrent neural networks based on chaotic bacterial colony optimization. Cluster Computing. 1-22. 10.1007/s10586-023-04187-4.
[32] J. Zhao and S. Gu, "Learning policies for complex robotic tasks using deep reinforcement learning," IEEE Trans. Robotics, vol. 36, no. 1, pp. 1-15, 2020.
[33] Naresh, P., & Suguna, R. (2021). IPOC: An efficient approach for dynamic association rule generation using incremental data with updating supports. Indonesian Journal of Electrical Engineering and Computer Science, 24(2), 1084. https://doi.org/10.11591/ijeecs.v24.i2.pp1084-1090.
[34] E. Parisotto and R. Salakhutdinov, "Neural map: Structured memory for deep reinforcement learning," in Proc. 30th Int. Conf. on Machine Learning (ICML), 2017.
[35] P. Naresh, S. V. N. Pavan, A. R. Mohammed, N. Chanti and M. Tharun, "Comparative Study of Machine Learning Algorithms for Fake Review Detection with Emphasis on SVM," 2023 International Conference on Sustainable Computing and Smart Systems (ICSCSS), Coimbatore, India, 2023, pp. 170-176, doi: 10.1109/ICSCSS57650.2023.10169190.
[36] J. Bohg and A. Billard, "Data-driven approaches to robot manipulation," in Proc. 24th IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2015.
[37] M. Kalakrishnan, F. Moser, and P. Pastor, "Learning robot forward models for manipulation," in Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2011.
[38] P. Kormushev, D. N. Nenchev, and D. G. Caldwell, "Upper-body kinesthetic teaching of a free-standing humanoid robot," in Proc. IEEE Int. Conf. on Robotics and Automation (ICRA), 2013.
[39] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-end training of deep visuomotor policies," in Proc. 30th Int. Conf. on Machine Learning (ICML), 2016.