energies

MDPI

Review

A Systematic Study on Reinforcement Learning Based Applications

Keerthana Sivamayil [1], Elakkiya Rajasekar [1,2], Belqasem Aljafari [3], Srete Nikolovski [4,*], Subramaniyaswamy Vairavasundaram [1,*] and Indragandhi Vairavasundaram [5]

1 School of Computing, SASTRA Deemed University, Thanjavur 613401, India
2 Department of Computer Science, BITS Pilani, Dubai Campus, Dubai 345055, United Arab Emirates
3 Department of Electrical Engineering, Najran University, Najran 11001, Saudi Arabia
4 Power Engineering Department, Faculty of Electrical Engineering, Computer Science and Information Technology, J. J. Strossmayer University of Osijek, K. Trpimira 2B, HR-31000 Osijek, Croatia
5 School of Electrical Engineering, Vellore Institute of Technology, Vellore 632014, India
* Correspondence: srete.nikolovski@ferit.hr (S.N.); vsubramaniyaswamy@gmail.com (S.V.)

Abstract: We have analyzed 127 publications for this review paper, which discuss applications of Reinforcement Learning (RL) in marketing, robotics, gaming, automated cars, natural language processing (NLP), internet of things security, recommendation systems, finance, and energy management. The optimization of energy use is critical in today's environment. We mainly focus on the RL application for energy management. Traditional rule-based systems have a set of predefined rules. As a result, they may become rigid and unable to adjust to changing situations or unforeseen events. RL can overcome these drawbacks. RL learns by exploring the environment randomly and based on experience, it continues to expand its knowledge. Many researchers are working on RL-based energy management systems (EMS). RL is utilized in energy applications such as optimizing energy use in smart buildings, hybrid automobiles, smart grids, and managing renewable energy resources. RL-based energy management in renewable energy contributes to achieving net zero carbon emissions and a sustainable environment. In the context of energy management technology, RL can be utilized to optimize the regulation of energy systems, such as building heating, ventilation, and air conditioning (HVAC) systems, to reduce energy consumption while maintaining a comfortable atmosphere. EMS can be accomplished by teaching an RL agent to make judgments based on sensor data, such as temperature and occupancy, to modify the HVAC system settings. RL has proven beneficial in lowering energy usage in buildings and is an active research area in smart buildings. RL can be used to optimize energy management in hybrid electric vehicles (HEVs) by learning an optimal control policy to maximize battery life and fuel efficiency. RL has acquired a remarkable position in robotics, automated cars, and gaming applications. The majority of security-related applications operate in a simulated environment. The RL-based recommender systems provide good suggestions accuracy and diversity. This article assists the novice in comprehending the foundations of reinforcement learning and its applications.

Keywords: machine learning; reinforcement learning; deep reinforcement learning; Markov decision process; contextual bandits; inverse reinforcement learning; multi-agent RL; energy management system

Citation: Sivamayil, K.; Rajasekar, E.; Aljafari, B.; Nikolovski, S.; Vairavasundaram, S.; Vairavasundaram, I. A Systematic Study on Reinforcement Learning Based Applications. *Energies* **2023**, *16*, 1512. https://doi.org/10.3390/en16031512

Academic Editor: Gerardo Maria Mauro

Received: 10 December 2022
Revised: 30 January 2023
Accepted: 1 February 2023
Published: 3 February 2023

1. Introduction

Machine learning supports various applications, such as data prediction, classification, regression analysis, and clustering. In supervised learning (SL) [1], input and output data (labeled data) are fed into the system that generates a model. Unsupervised learning (UL) [2] finds associations, similarities, and variations in unlabeled data and extracts hidden information. SL, UL, and RL [3] are three types of machine learning. The generated model can predict the output for new unknown data. Spam detection [4], image and object

detection [5], predictive analytics [6], text classification, and other applications make use of SL. SL algorithms include logistic regression, linear regression, decision tree, random forest, and support vector machine (SVM) [7]. Clustering with the k-means clustering algorithm and dimension reduction with principal component analysis (PCA) are two UL applications. Deep learning has been able to overcome the shortcomings of machine learning. Machine learning extracts features separately, whereas deep learning extracts features automatically and without human intervention. Machine learning cannot handle large amounts of data, but deep learning can take millions of data points. Deep learning can process unstructured data such as images and audio. Because of their dynamic nature, specific real-time scenarios, such as recommendation systems and self-driving cars, do not have sufficient datasets for training. Reinforcement learning is introduced to address this issue. The third type of machine learning is RL [8]. RL is appropriate for sequential decision-making processes and learns through direct interaction with the environment to achieve long-term goals without external motivation or complete knowledge of the environment. Figure 1 depicts the RL framework.

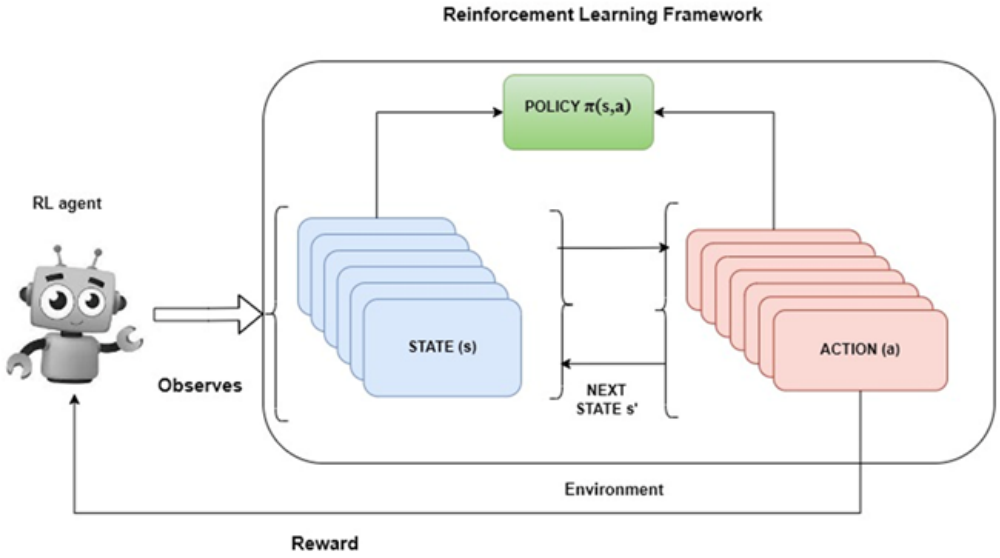

Figure 1. Reinforcement learning framework.

The RL agent examines the condition of the environment and chooses an appropriate action. If the RL agent performs the correct action, it receives a positive reward. If the agent makes the wrong move, a negative is generated. RL must balance exploration and exploitation. Exploitation occurs when an agent attempts to maximize the reward based on a previously established route. If the agent always tries to explore a new way to reach the destination, it is called exploration. There are several applications for RL. RL The model does not need a large dataset and learns by trial and error.

A recommendation system is designed to suggest items valuable to the user. Recommendation systems (RS) are used by most e-commerce [9] websites to increase revenue. Different types of recommendation systems are content-based, collaborative-based, and hybrid-based. Another type of RS is location-based recommendation [10,11]. Traditional RS systems suffer from a cold start, a warm start, a long tail, data sparsity, and scalability. To address these shortcomings, RL is introduced in RS to provide recommendations that are accurate, relevant, and diverse. Reinforcement learning-based RS [12] is used in various applications, including news article recommendations, course recommendations, product recommendations, movie recommendations, and so on. In conjunction with deep learning, RL handles high-dimensional sensory input in applications such as Atari games.

The deep Q network (DQN) is the most widely used algorithm in deep reinforcement learning. Multi-armed bandit comprises multiple arms, each with a different chance of winning. Contextual bandit solves the exploration-exploitation problem. For personalized

recommendations, the contextual bandit is used. The most widely used algorithms in contextual bandit are upper confidence bound (UCB), Thomson sampling, and LinUCB. In IoT applications, RL is used to protect users from security threats. Most RL works in security applications are simulation-based, making the transition into the real world expensive. The most efficient robots are those built with RL. Social robots are primarily used in healthcare to take care of the elderly. Cognitive empathy is also used in the social robot to understand human emotions better and make them comfortable with the response. Natural language processing works much better with RL. Inverse reinforcement learning (IRL) [13] and the multi-agent RL approach are considered different forms of RL. In some instances, the rewards and transition probability of the systems are unknown. In such cases, IRL has been applied to solve this problem. IRL, on its own, learns the reward function. IRL observes the domain expert's behaviours to implement the observed behaviour.

One of the critical applications of RL are EMSs. Energy demand continues to increase, and optimizing energy consumption while maintaining comfort is challenging. Most EMSs are implemented with the help of RL. In this review, we briefly discuss how the RL algorithm is built for EMSs. The RL algorithm learns through trial and error, adjusting the control policy based on the HEV's performance feedback. The algorithm can consider various factors, including the battery's present condition, the vehicle's speed and power demands, and the expected driving conditions. This method can result in more effective use of the HEV's power sources, leading to longer battery life and better fuel economy. RL can be used in HVAC control to teach an agent the optimal control approach for the HVAC system by interacting with a simulated environment that models the building and its thermal dynamics. The agent can perform actions such as adjusting the temperature setpoint or changing the airflow and is rewarded or punished based on energy consumption and thermal comfort level. The agent understands how to make decisions that conserve energy while sustaining thermal comfort over time. Eco-driving, also referred to as fuel-efficient driving, is the driving practice aimed at lowering fuel consumption and emissions. RL can be used to optimize the control of a vehicle's powertrain to achieve eco-driving. The vehicle's powertrain and its surroundings, such as the traffic and road conditions, are monitored closely in an RL-based eco-driving system. The RL algorithm then uses this information to determine the best actions for the driver to take, such as adjusting the speed, gear selection, and engine power to reduce fuel consumption. The algorithm also considers the driver's behaviour and preferences, such as the driver's comfort level and the urgency of the trip, to provide personalized eco-driving recommendations.

Section 2 of this review paper defines reinforcement learning algorithms. We discuss the application of reinforcement learning in various fields in Section 3.

2. Reinforcement Learning Algorithms

An RL process that satisfies the Markov property (MP) is called a Markov decision process (MDP) [14]. Different RL algorithms are available in the literature, and the hierarchy of algorithms is given in Figure 2.

RL algorithms can be value-based [15] or policy-based [16]. The value-based algorithm begins with a random starting value. Initially, the RL agent chooses a spontaneous action for a specific state and computes the value function. The RL agent can find the best policy using the value function. It does not save the policy but preserves the value function. Value-based RL requires a long time to compute because the q table has more states and actions. In policy-based RL, the agent begins with a random policy. It selects an action for a specific state based on policy. Then it computes the value function for the randomly chosen policy. The policy-based algorithm has a higher rate of convergence. Policy-based RL is appropriate for large state-space applications. It is possible to learn stochastic policies using policy-based RL. However, policy-based RL suffers from high variance. It requires few iterations to converge, but the algorithm is complex. An actor-critic algorithm is a combination of both value-based and policy-based algorithms.

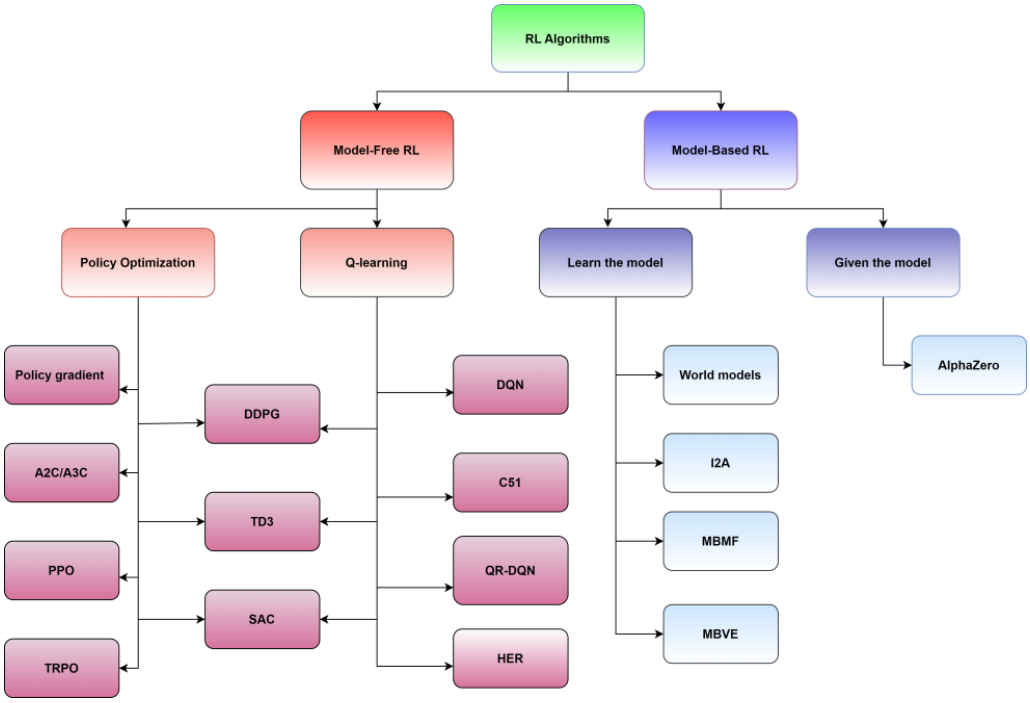

Figure 2. Reinforcement learning algorithm hierarchy.

2.1. Three Approaches to RL

The other three approaches of RL are Monte Carlo [17,18], temporal difference (TD) [19], and dynamic programming (DP) [20]. Dynamic programming requires the model to learn and follow a recursive approach. The model has to define states, actions, rewards, and transition probability. Dynamic programming is used for solving complex problems. It resolves a problem by breaking it into sub-tasks/sub-problems after determining the solution and combining the sub-tasks to form the solution. It can solve interlaced sub-problems, optimal substructure, and MDP. Monte Carlo (MC) and TD require experience and can learn by exploration (trial and error) without the model. MC follows the learning approach and is used for uncertain environments. TD determines the value estimates based on the estimates of other values, a process called bootstrapping. TD works for continuous tasks (non-terminating) and does not require a model. After every step, it updates the values estimates of the value. It does not have to wait till the entire trial and only updates the values for the visited path. TD is used in online and incremental learning and has low variance and some bias.

2.2. Different Types of RL Algorithms

Monte Carlo (MC) follows a learning approach based on interacting with the environment. MC is suitable for episodic tasks and has a starting and ending state (destination state). No matter what action it takes, the episode will terminate. MC does not require a model. It learns from the complete episode, and no bootstrapping is performed. When an agent visits a state for the first time in an episode, it is called the first visit. The episode's return can be calculated based on the average of the samples concerning the first visit. In every visit, the return can be calculated based on the average of state values concerning every visit. A state is visited multiple times in every episode. The average of every visit is the sum of all the samples divided by the number of visits. Two types of approaches are employed: MC Control and MC prediction. MC control is in charge of calculating the state value function. MC prediction is responsible for finding the optimal policy based on the value estimates. MC has high variance and zero bias. MC and TD use sampling, but DP does not perform sampling.

Q-learning is one of the most widely used algorithms and follows a model-free approach and off-policy. The RL agent tries to find the maximum Q value. Q-learning is a value-based algorithm. Initially, all the q values are zero or random. The states and actions are represented as the rows and columns of the Q table. The Bellman equation is used to derive the Q value. The best action is selected based on the max Q value for a particular state using the ϵ-greedy policy. It starts exploring during ϵ times and (1-ϵ) times it finds the best possible q value (optimal policy). Q-learning has been used in gaming [21] and robotics [22]. State action reward state action (SARSA) is identical to Q-learning except that it does not find the max q value and uses on-policy. Utilizing the current set of actions carried out in the current state enhances the agent's learning process. Previous states and rewards are not considered for a new set of states. Electric vehicle route optimization has been implemented using the SARSA algorithm [23]. It is used in non-stationary applications. Actor-critic (A2C/A3C) is a combination of both policy-based and value-based approaches. An actor performs specific actions, and the critic analyzes the value function of the corresponding action. By using TD error, the critic evaluates the action taken by the RL agent. Q-learning is not sufficient for vast state spaces and action spaces. For many states and actions, more computations and time are required. Deep neural networks are used in deep Q networks (DQN). The Q value is determined using a neural network. DQN is suitable for significant state space problems.

Proximal policy optimization (PPO) is an on-policy algorithm that generates data and updates the policy using the existing policy. PPO comprises two parts: a policy network that maps states to actions and a value network that computes the value of a state or a state-action combination. The policy network is trained to maximize the predicted cumulative reward, whereas the value network is taught to evaluate the worth of states and actions. It is reasonably straightforward to deploy and robust to hyperparameter selection. It has been employed in various applications, such as robots, gaming, and continuous control. Trust region policy optimization (TRPO) is another type of RL algorithm. It is similar to PPO. It is an on-policy algorithm that generates data and updates a policy using the existing policy. The trust region is used to update the policy settings, and the algorithm uses a natural gradient rather than a standard gradient. It outperforms the other RL algorithms in terms of performance. TPRO is utilized in various applications, including gaming, robotics, and continuous control. Deep deterministic policy gradient (DDPG) is another type of RL algorithm. The DDPG approach combines Q-learning with policy gradient methods. It employs the actor-critic network, which comprises two neural networks. The actor network is in charge of choosing an action, and the critic network examines the action selected by the actor network. The actor network seeks to maximize the total payoff. It is suitable for use in environments with continuous action spaces and is utilized in various applications, including robotics, gaming, and self-driving cars.

Soft actor critic (SAC) is another type of RL algorithm. SAC is a model-based reinforcement learning (RL) technique that integrates actor-critic architecture with entropy regularization. The actor-network chooses actions depending on the current state, whereas the critic network is employed to assess the worth of those actions. The entropy regularization element is introduced to the objective function to stimulate exploration and prevent premature convergence. SAC has been demonstrated to be highly successful on a vast scope of continuous control tasks and has been utilized in various applications, including robotics, gaming, and simulated physical systems. Twin delayed deep deterministic policy gradient (TD3) is the next step in the evolution of DDPG. It carries out the postponed policy update. It employs two criteria. The critic network computes the current state-activity pair's value, whereas the actor-network selects the best course of action given the current scenario. TD3 reduces the overestimation of the Q value and increases its accuracy. The delayed policy update can lessen the overfitting of the network.

C51 (Categorical DQN) is a well-known Q-learning algorithm for discrete action spaces. In classic Q-learning, the Q value is expressed as a single value for each state-action pair, which tackles the problem of classical Q-learning discretization. Instead of employing a

single Q value for each action, C51 extends the DQN by including a categorical distribution across the Q values. QR-DQN (quantile regression DQN) is an enhancement to the DQN algorithm. QR-DQN updates the network's parameters using a quantile regression loss function, which differs from the standard mean squared error loss function used in DQN. It is defined as the sum of the Huber losses for each quantile of the action-value distribution. This loss function enables the agent to learn various possible action values rather than just one optimum action. Hindsight experience replay (HER) is another type of RL algorithm. It enables the agent to learn from previous failures and overcome them by focusing on the present objective. Additionally, the same experience is replayed with the new aim in mind. It is beneficial for tasks with sparse rewards and is appropriate for various suboptimal goal states. HER uses experience replay, which stores all previous experiences and is compatible with algorithms such as DDPG, DQN, and A3C.

World models are unsupervised algorithms consisting of three parts: a variational auto encoder (VAE), a recurrent neural network (RNN), and a decoder. The variational auto encoder encodes environmental observations. The RNN simulates the dynamics of the environment, and the decoder provides observations based on the VAE and RNN output. The RNN may be used to predict future environmental conditions. It is employed in various applications, namely robotics and computer vision. Model-based meta-learning with flexible computation (MBMF) was proposed in 2021 by a researcher from OpenAI. The agent learns from a collection of tasks. It quickly learns the new task by sharing computation across the tasks. It takes advantage of a computing model which is made up of a collection of previously learned activities. The agent is instructed on a series of tasks using an environment model called the task model, and the neural network serves as the task model. This network predicts the following state environment based on the present state and action. It contributes to the RL algorithm's sampling efficiency. Model-based value expansion (MBVE) is a technique for improving the sample efficiency of model-based algorithms. It forecasts the future using the model environment. This algorithm calculates the value estimates. Based on the value estimates, it predicts the next state of the environment and can be combined with model predictive control (MPC) or guided policy search (GPS) to improve the sample efficiency. Imagination-augmented agent (I2A) consists of two components: RNN and DQN. RNN acts as the imagination module, and DQN acts as the control module. The imagination module generates the imaged scenarios by simulating the environment. The control module forecasts the future based on the generated scenarios. The control module estimates the return of state and action using the Q-learning algorithm. AlphaZero is a computer program developed by DeepMind that can play chess, shogi, and the game of go. It combines deep neural networks and Monte Carlo tree search algorithms.

3. Reinforcement Learning Applications

RL has been used in many applications in recent years. In this paper, we list several domains and their applications, which are shown in Figure 3.

3.1. Reinforcement Learning Applications in Recommendations

In course recommendations, user interest belongs to different categories of subjects. A novel method for course recommendations using the dynamic attention model using RL has been proposed [24]. This model represents the interaction between the user's preferences and the profile reviser. To enhance recommendation accuracy, their model adaptively modifies the attention weight of the related course during various sessions to monitor changes in user preferences. They use two real-time datasets from massive open online courses. It has a high level of recommendation accuracy. In [25], a novel course recommendation model is proposed, which uses a profile constructor with self-learning capabilities to recommend courses specifically for each student. They offer a unique policy gradient method to overcome the exploration-exploitation trade-off problem in generating user profiles. The model uses a context-aware recurrent approach to capitalize

on the available knowledge to investigate the user's potential future preferences. Two real-world datasets were used for extensive tests to validate the effectiveness of the proposed HELAR model. The results prove that HELAR performs better than cutting-edge course recommender systems. Another method providing a personalized learning experience to students using reinforcement learning has been proposed [26]. The authors employ the Q-learning algorithm to recommend educational materials to students based on their present situation rather than their records, building the policies from collected data. At the initial state, the model generates a random policy, and later, the optimal policy is obtained based on the Q Value. To overcome the cold start problems, grey sheep, and sparsity of RS, ref. [27] proposed an RL-based dynamic recommendation. In [28], an RL model is designed that recommends learning objectives (LO) to students based on sensors that capture students' heartbeats, quiz scores, blinks, and facial expressions.

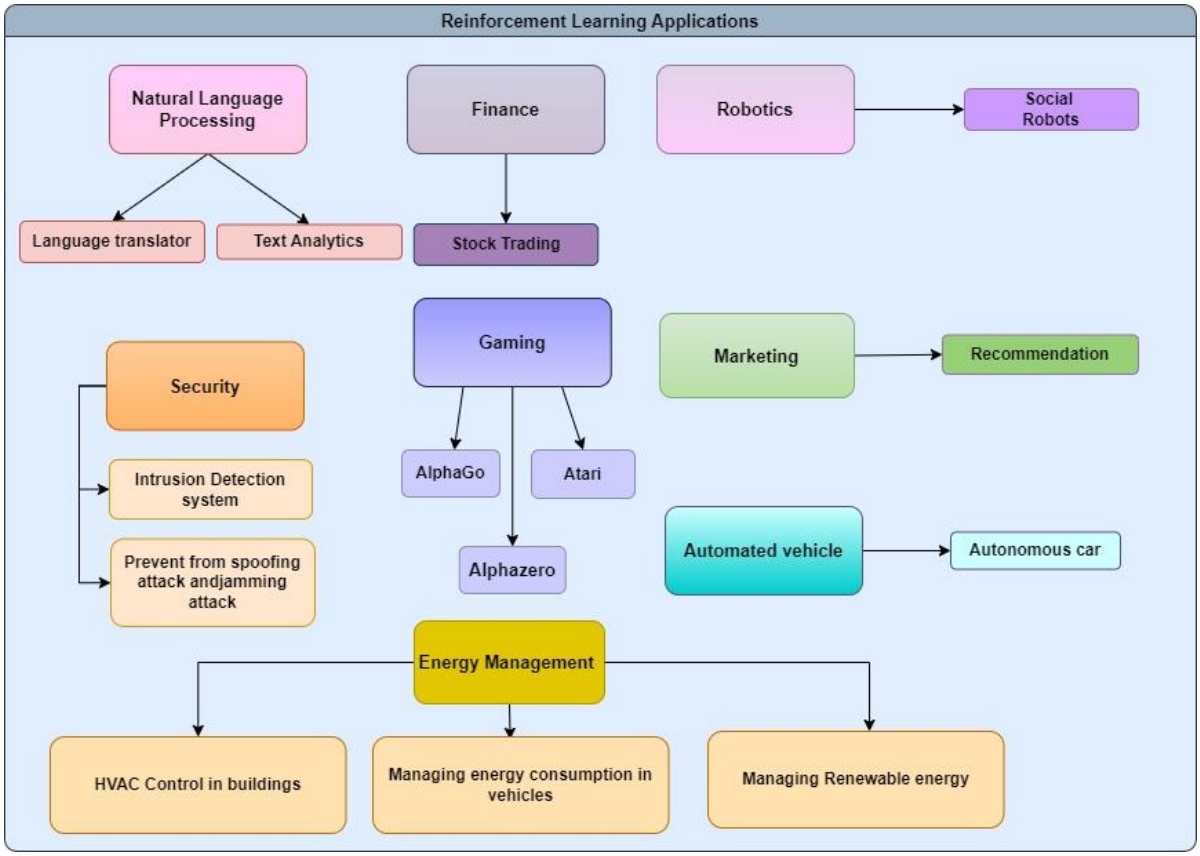

Figure 3. Overview of reinforcement learning applications.

3.1.1. Deep Recommender Systems in Recommendations

Deep reinforcement learning (DRL)-based RS was developed to provide more accuracy than traditional RS. DRL can manage more states and actions in a dynamic environment. A recent study proposed a DRL with user-commodity state representation called UCSR-DRL [29]. They interpret RS as a sequential process and use the actor-critic algorithm, which includes long and short-term incentives. The cold start problem and data sparsity were overcome by their proposed DRL based on a prioritized experience replay network for capturing user interest changes [30]. Promotion of user engagement using deep reinforcement learning-based RS has been proposed [31]. To handle massive action sequences, a deep hierarchical category-based RS was implemented, and a two-layered DQN was developed. The first level selects the item category, and the second recommends the item to the user. Four real-time datasets are used, including Netflix and, Movielens (ML) (20 M, 10 M, and 1 M). The recommendation accuracy is higher based on the hit ratio and normalized

discounted cumulative gain. To solve the cold and warm start problem, ref. [32] proposed a long-term recommendation model using a recurrent neural network with reinforcement learning. This model was evaluated using the hit ratio and normalized discounted cumulative gain. The recommendations can be fine-tuned based on the feedback of the user. A DQN for analyzing both negative and positive feedback of users was proposed by [33]. Positive feedback is given whenever a user clicks or orders an item; negative feedback is given when the user skips the item. A DRL framework for the interactive suggestion was developed, comprising convolution neural network and GAN [34]. The proposed combinatorial product recommendation system includes a consumer behaviour simulator and utilizes DRL to find appropriate product combinations that can improve the platform's sales [35]. A user-specific and biased user-specific DQN for interactive recommendation problems for explicit feedback has been developed [36]. Matrix factorization is used to create user-specific states to learn the best recommendation policies. A biased user-specific DQN was developed to model the user-specific information by including the bias to analyze the Q values of every user.

3.1.2. Recommendation Using Contextual Bandits

The contextual bandit is an extension of the multi-armed bandit. Contextual bandit is used for solving the exploration-exploitation dilemma. The RS considers items or users as one of the arms in the contextual bandit framework. Contextual bandit aims to minimize regret. Based on previous experience, the bandit chooses the arm which maximizes the reward. The newly trained RS tries to exploit the products that the users are already engaged with. To overcome this problem, ref. [37] proposed an advertisement recommendation system using contextual bandits that uses two exploration algorithms: upper confidence bound (UCB) and Thomson sampling. They built a hybrid method combination of bootstrap and dropout. Based on the CTR, the model recommends an item to the users, and instead of training the RS, the model learns by dynamic interaction with users. Ke-LinUCB is a model proposed for personalized recommendations in a changing domain [38]. The dynamic preferences of the users are captured by this proposed system. The intention-selection method is used to analyze the behaviour of users. LinUCB uses an exploitation and exploration strategy to recommend the items to the users. Amazon-book and Yelp2018 were benchmark datasets used in this work. The recommendation accuracy of this model depends on the information of items and users. In [39], a news recommendation system is proposed that uses three different algorithms: CoLin, LinUCB, and Hybrid-LinUCB. Each algorithm is implemented on different datasets, such as MovieLens20M, Yahoo FrontPage Today Module, Synthetic, and LastFM. Each news article is considered an arm, and the reward is different for different users. The confidence bound is calculated based on the three algorithms. This system makes more accurate and diverse recommendations. An implicit feedback-based recommendation system that uses multi-armed bandit (MAB) was suggested by [40]. In this system, clicks and favourites are considered implicit feedback, which is divided into three categories: strong interaction, non-interaction, and weak interaction. The item category is considered an arm, and the number of arms is fixed. Thomson sampling is applied to balance the exploration-exploitation dilemma. To handle the cold start problem, the best-selling or popular items are recommended to new users. For old customers, items are recommended based on their behaviour history. The authors successfully verified the model on three datasets. In most of the recommendations using contextual bandit, either the item or user is used as an arm. The binary upper confidence bound can be employed to regard users and things as arms of each other [41]. This method also solves the cold start problems for items and users. The method was compared using five baselines. Considering precision, BiUCB provides a high precision value when compared with the five baselines. In another work, ref. [42] applied a contextual bandit algorithm to recommend personalized online learning objectives. The algorithm makes recommendations based on the student's history and current state. Using a dataset from e-learning systems, which contained 365 students, 2519 events, and 78 Learning Outcomes(LO), they found the

correlation of two actions by conditional probability. The RL agent recommends the LO, and the reward is generated based on the click through rate (CRT). Contextual bandit algorithms resulted in a higher CRT than the ϵ-greedy, greedy optimal, and upper confidence bound algorithms. Table 1 provides an overview of RL recommendation applications using different datasets and models.

Table 1. Reinforcement learning applications for recommendation systems.

S. No	References	Product	Datasets	Model
1	[24]	Courses	MOOC	Dynamic attention and hierarchical reinforcement Learning (DARL)
2	[25]	Courses	MOOCCourse and MOOCCube	Hierarchical reinforcement learning with a dynamic recurrent mechanism
3	[29]	E-commerce items	Item-info, trainset, and track2_testset	Actor-critic with state representation
4	[30]	Movies	MovieLens	Deep Q network
5	[31]	Movies	MovieLens and Netflix	Deep Q network
6	[32]	Movies	MovieLens 100K, MovieLens 1M and Steam	Recurrent neural network
7	[33]	E-commerce items	JD.com(E-commerce website)	Deep Q network
8	[34]	E-commerce items	E-commerce website	Deep Q network with CNN and GAN
9	[35]	E-commerce items	E-commerce website	LSTM and DDPG
10	[36]	Movies and Music	ML100K, ML1M, and YMusic	User-specific deep Q-learning
11	[37]	Advertisements	ADS-16	Deep bayesian bandits
12	[38]	E-commerce items	Amazon book and Yelp2018	Knowledge-enhanced Ke-LinUCB
13	[39]	News and Movies	Yahoo Front Page Today Module, Lastfm, and MovieLens20M	LinUCB, Hybrid-LinUCB, and CoLin
14	[40]	Item	Yoochoose, IJCAI-15 Retailrocket	Thompson sampling
15	[41]	Movies	MovieLens	BiUCB (Binary upper confidence bound)

3.2. Reinforcement Learning in Gaming

RL is widely used in many gaming applications. RL learns the game through trial and error based on short- and long-term rewards. Sometimes the RL agent has to sacrifice the immediate reward for the long-term reward, which yields good results. An RL model to learn the Othello game without the intervention of human knowledge has been proposed [21]. In [43], the authors implemented a DRL to play Atari games. They used Q-learning with convolutional neural network (CNN). Image pixels were used as input, and value functions were the outputs. They developed the algorithm for seven Atari games with this CNN and Q-learning and found that it performs better than the previous methods. In other research, the DDPG algorithm was proposed to track the pursuer and seize the evader quickly [44]. RL can also provide game-based learning in which students can efficiently learn through the interactive RL environment. In their article, ref. [45] presented game-based learning in a reinforcement learning environment. Many gaming scenarios have successfully deployed self-play RL, where the RL agents learn by interacting with themselves [46]. The game of go is a classical game involving two players, and one player has to surround more places than the other. The AlphaGo program, which comprises neural networks and Monte Carlo, was developed by [47]. AlphaGo performed better than the other Go programs. A general reinforcement learning framework called AlphaZero has been developed for three games: go, chess, and shogi (Japanese chess) [48]. The game rules

were provided to the RL model without domain knowledge, and the RL model learned by self-play. AlphaZero defeated the world champions in these games. To prevent time delays during online execution, automated decision-making with DQN has been developed in the Boulder Dash game [49]. Self-play is one of the applications of RL games. The model plays against itself and learns the environment by interacting, and experience replay has been added to improve the effectiveness of the actor-critic algorithm [50]. This model was implemented with benchmark Atari and Mujoco. The value function is calculated without using Q-learning, and the model can also be used for both continuous and discrete jobs. Table 2 lists the gaming applications implemented using RL.

Table 2. Reinforcement learning applications in gaming.

S. No	References	Application	Algorithm
1	[21]	Othello game	Q-learning
2	[43]	Atari 2600 games	Convolutional neural network with Q-learning
3	[44]	Pursuit-Evasion differential game	Deep deterministic policy gradient
4	[47]	AlphaGo	Deep neural network and Monte Carlo
5	[48]	AlphaZero	Monte Carlo tree search
6	[49]	Boulder Dash game	Deep Q network

3.3. Reinforcement Learning in Automated Vehicles

Autonomous cars use technology to replace human drivers to reduce road accidents and ensure road safety. These cars work based on rules and imitation. In a model based on rules, the decisions of the cars are based on the rules framed. In an imitation model system, supervised learning methods are used to train the model. Deep learning-based automated vehicles require excessive data to train the model and cannot correct accumulated faults. RL has been introduced into automatic vehicles to learn by trial and error and overcome these issues. Self-driving cars are one of the critical application areas of reinforcement learning. Two tasks of these cars are perception and decision making systems (DMS). The car's state is observed using the sensors, cameras, global positioning system (GPS), etc. [51]. Decision-making systems using perception systems are in charge of moving the car from the starting position to the desired goal. A model for self-driving cars using confidence-aware reinforcement learning has been proposed [52]. These researchers created an RL policy as well as a benchmark rule-based policy. The RL policy deals with the situation when the classic rule-based approach fails. Sometimes RL agents produce the wrong decision that is not encountered during the learning, leading to failure of the systems. To handle such uncertainty, ref. [53] proposed a model for uncertain environments using DeepSet-Q with a Gaussian mixture (DwGM-Q). Their experiments showed that uncertainty was well detected in the simulation environment, and the computation time was shorter than the existing ensemble method.

3.4. Reinforcement Learning in Natural Language Processing (NLP)

NLP is a notable AI application in which the computer recognizes human language and responds to its queries. Some examples of NLP are intelligent assistants, language translation, text analytics, etc. In their review article, ref. [54] discuss NLP applications such as syntactic parsing, language understanding, text generation systems, and machine translation using RL. In syntactic parsing, states are considered as parse trees of all possible combinations. Grammar rules are considered as actions, and the reward depends on the number of arcs identified correctly in the final parse tree. In text generation, states are formed with the feature vector and by adding/deleting the word as an action. Rewards are generated based on feedback from the user. Machine translation takes input in one language and converts it into another language with a similar meaning. States are the set of all possible input strings, actions are added/deleted, and rewards are generated based

on a match between the input and output meanings. The complexity of natural language and its constant change make it challenging to build an NLP model. DRL can store many grammatical structures as a neural model. RL models are suitable for problems which continue changing. RL models jointly applied with DRL can provide better results. There are two types of text summarization: abstract and extract. Extract summarization highlights the critical sentence. Paraphrasing is performed in abstract summarization, which is complex. In their review article, ref. [55] discuss papers on automatic text summarization using RL and transfer learning in terms of algorithms, datasets, challenges, solutions, and performance metrics. Chatbots can perform conversations with customers to answer queries. DRL has been used with an ensemble-based model to build a chatbot [56].

3.5. Reinforcement Learning in the Internet of Things Security

Security is a significant problem in IoT systems due to the many smart devices linked to the internet, and protecting users' data from security threats is a challenge. Supervised machine learning algorithms can detect only trained threats. RL can learn by trial and error without training and does not need a massive volume of data. In their review article, ref. [57] discuss RL methods to overcome security threats such as jamming, spoofing, and denial of service attacks. Other authors [58] have reviewed the DRL methods for identifying the intrusions in the system and their challenges. Most of the studies implemented the IDS in a simulated environment. Training the DRL for IDS in a real-world environment is costlier and more complex.

3.6. Reinforcement Learning in Finance

The dynamic nature of the financial systems can cause difficulty in framing equations, a problem called the curse of modeling. RL learns by trial and error to overcome the curse of modeling by interacting with the environment. The authors of [59] proposed portfolio management in finance using DRL. They studied how the total rewards were influenced by earlier states and actions using RNN (recurrent neural network). The optimal portfolio management policy was obtained by combining RL with DRL.

3.6.1. Trading

Framing adaptive stock trading policies is difficult due to the dynamic properties of stocks. An adaptive stock trading technique using DRL has been suggested [60]. To extract the characteristics of the financial data, a gated recurrent unit (GRU) is used. The model designers developed strategies for performing quantitative stock trading using GRU along with a deterministic policy gradient and gated DQN. They solved the disadvantages of traditional trading strategies which are constrained to single market patterns. In the ever-changing stock market, a critic-only network with GDQN is less stable than GDPG with an actor-critic structure. A multi-model RL for trading, which considers price fluctuations and sentiment analysis on news articles, has been proposed [61]. Other authors have suggested algorithmic trading based on historical data, correlation of features, and technical analysis indicators [62]. A deep neural network (DNN) model was proposed to forecast the number of shares to trade [63]. These authors used DQN to find suitable action techniques for improving the profit in the market. A long short term memory (LSTM) model along a with proximal policy optimization algorithm for constructing a Bitcoin trading strategy has been proposed [64]. ResNet and LSTM provide outperforming results for automatic trading compared with RL algorithms [65]. A random neural network based on DRL was proposed to forecast market data's upward, downward, and similar trends [66]. Short-term memory is required in stock market analysis rather than long-term history. A multi-agent DQN was proposed for automatic trading [67], and the DQN was fine-tuned by adjusting the hyper-parameters, such as activation function, number of q networks, learning rate, discount factor, etc., using a Forex (EUR/USD) dataset. The model uses performance metrics such as sharp ratio, average cumulative return, maximum cumulative return, minimum cumulative return, etc. A DQN model for automated trading, trained on the same dataset

multiple times to predict future market conditions was suggested [68]. This model was applied to intraday trading. The majority of research in trading applications employs value-based RL. An asynchronous advantage actor-critic (A3C) algorithm is proposed for stock selection [69], which makes use of both policy-based and value-based models. Table 3 shows the references and their corresponding datasets, Modes, and performance metrics for trading applications.

Table 3. Reinforcement learning applications in finance.

S. No	References	Datasets Source	Model	Performance Measures
1	[62]	Yahoo Finance	Time-driven feature-aware jointly deep reinforcement learning model	Total profit, transaction times, the annualized rate of return, and sharp ratio
2	[63]	Thomson Reuters and Yahoo Finance	Deep neural network regressor and DQN (Deep Q network)	Total profit
3	[64]	Cryptodatadownload	LSTM and PPO	Profit rate
4	[66]	UK house prices, Gold, Bitcoin, FTSE, and Brent oil Market Validation	DRL with random neural network	Accuracy, RSME, MAE, MAPE
5	[67]	Forex	Multi-agent DQN	Sharp ratio, average cumulative return, maximum cumulative return, minimum cumulative return, etc.
6	[68]	Standard & Poor's 500 (S&P500) and the German stock index (DAX)	Q-learning	Equity curve, accuracy, coverage, maximum drawdown, and Sortino ratio

3.6.2. Comparison between ML and RL in Credit Risk

Plenty of research work has been published about using machine learning to analyze credit risk. In their research, ref. [70] categorized credit card defaulters by combining DALEX and XGBoost. The performance metrics employed were sensitivity, specificity, and accuracy. In other research [71], a deep learning and machine learning strategy for categorizing credit risk was proposed. They used machine-learning approaches such as random forest, logistic regression, and gradient boosting. In addition, the four DL model vary in the neurons, hidden layer, and regularization methods. In their review of credit risk analysis and its limitations, ref. [72] concluded that DL models provide better prediction accuracy than ML and statistical methods. Credit, market, and operational risks can be assessed using ML, as reviewed by [73]. In their work, ref. [74] provided a suggestion for banking risk management utilizing ML and AI. These suggestions apply to small and mid-sized banks in developed and developing markets. They evaluated the client risk based on the credit score before granting a loan to a customer. A credit risk analysis using an RL model was recommended by [75]. The credit score can be used to analyze the customer risk category. Another paper calculated credit scores using DQN and the changing reward function [76]. They tested the model on five different datasets including Australia (AU), Chongqing (CH), credit card fraud (CR), German (GE), and Leadingclub (LE). The evaluation measures used by them were true negative rate (TNR), area under the ROC curve, percentage correctly classified (PCC), true positive rate (TPR), precision, and F1 score. The specific threshold was defined to differentiate between good and bad credit.

3.7. Reinforcement Learning in Robotics

The most popular application field of RL is robotics. The authors of [77] examined the use of social robots, which are designed with a different form of reward mechanism. They addressed three types of reward mechanisms: interactive reinforcement learning, intrinsic

motivation, and task performance. In classical RL, an agent obtains the reward from the environment by utilizing a predetermined reward function. Interactive RL interacts with humans to obtain explicit and implicit feedback. Explicit feedback was direct based on ratings and labels. Implicit feedback was indirect and based on non-verbal cues such as emotions, speech, and gestures. The third form of reward was based on the robot's performance while interacting with a human. Nowadays, Robots are used in various applications such as manufacturing, packing, disaster management, healthcare, logistics warehouses, space, etc. Robots can also be used to understand human emotions and make them feel comfortable. The authors of [78] suggested a framework to introduce cognitive empathy in social robots. This model identifies the user's affective state based on their facial expression and sends them an empathic behaviour. If the user's affective state changes from a negative to a positive or neutral state, the robot receives a positive reward; otherwise, it receives a negative reward. In this experiment, the researchers investigated four basic emotions and three types of people. RL robots have been deployed in airways, waterways, and on land. The authors of [3] reviewed the applications of RL in different domains. A DRL-based algorithm was proposed to explore underwater [79]. With the help of RL, mobile robots that are moved from one location to another to perform specific tasks have become popular. Such robots face a slew of navigational challenges. The authors of [80] review listed mobile robots' challenges and solutions to these challenges. The authors of [81] discuss the challenges of motion planning for mobile robots. Finding the optimal route for the mobile robot without colliding with an obstacle is difficult. In their work, ref. [82] proposed a method to optimize the indoor path for mobile robots. Other researchers proposed development of a DQN for planning the most efficient route for the mobile robot [83]. They made a comparison of conventional DQN and enhanced DQN. Excellent outcomes were achieved by increasing the reward value and decreasing the loss function. Mobile robots suffer from the problem of deadlock and redundant paths. To overcome this problem, ref. [84] proposed a fusion model which consists of fuzzy logic, long short term memory (LSTM), and RL algorithms. In their work, ref. [85] proposed continual learning with RL to optimize a mobile robot's trajectory and reach the current destination in the real world and simulation environment. Another review discusses the advantages and disadvantages of robotics in RL [86].

3.8. Reinforcement Learning in Healthcare

A large number of people have lung cancer, but early diagnosis and discovery can lower the death rate. In their work, ref. [87] suggested a computer-aided diagnosis system for detecting lung cancer using deep reinforcement learning. They used value-based algorithms such as DQN, hierarchical DQN, and deep successor Q-network. In other research, supervised learning and RL gave a dynamic treatment recommendation [88]. These researchers used two signals: an indicator signal and an evaluation signal. The indicator signal matches the signal with the doctor's prescription, and the evaluation signal is the overall reward obtained from the survival rate. They also used the actor-critic algorithm and recurrent neural network for treatment recommendations. The authors of [89] reviewed methods to analyze disease detection and recommended the medications to the patients using RL. In their article, ref. [90] suggested context-aware RL for analyzing human health using sort retention double DQN. They compared DQN and sort retention double DQN and concluded that this proposed method achieves better results. Precision medicine is one of the essential applications of RL. It provides personalized treatment recommendations based on the disease's symptoms. In their work, ref. [91] proposed a precision medicine model using RL. They clustered the patients based on their same states and recommended treatments to them. Diabetes affects a large number of people and necessitates lifelong medication. A treatment recommendation for diabetic patients is given using an RL-based approach [92].

3.9. Inverse Reinforcement Learning (IRL)

MDP and reward functions are not known in some applications. In such cases, inverse reinforcement learning is used. IRL learns the reward function on its own. There is no need to specify the reward function explicitly. IRL is a demonstration learning in which the learner tries to learn the reward function of the instructor giving the lessons. In earlier forms of IRL, the expert provided demonstrations, whereas in later IRL, demonstrations are given as trajectories (state-action pairs). Different categories of IRL algorithms are max margin, Bayesian method, and maximum entropy [13]. In specific environments, the transition probability is unknown, a situation called model-free. IRL can be applied to the model-free approach. IRL is applied in dynamic route recommendations [93]. In the model-based approach, the reward function is estimated in linear and non-linear systems [94]. When IRL is used in a system where the developer cannot frame the rewards explicitly, expert behaviour is transferred to the RL agent so that it can perform well in the desired task. Inverse RL has been applied in a multiplayer, non-cooperative environment [95]. The authors of [96] proposed data-driven IRL for multiplayer environments. IRL with Dijkstra's algorithm was modelled to optimize the route for food delivery applications [97]. Based on the delivery staff's preferences, it recommends the optimized route. The authors of [98] suggested a model using IRL to predict commenting behaviour among the users and inattentive user groups on YouTube. They considered each user as an individual contextual bandit problem. Based on the commenting behaviour of the user, they grouped the users into different clusters. Their primary research conclusion was that viewers were eager to leave comments on popular videos.

3.10. Multi-Agent Reinforcement Learning (MARL)

In multi-agent RL, multiple agents are involved in decision-making. Each agent is solely accountable for their actions. There are three types of environments for MARL: competition between agents, cooperation between agents, and a combination of both. Agents compete against each other to win the game in a competition situation. In a cooperative situation, agents work together to attain a common goal. Intelligent traffic light control has been implemented with multi-agent DRL [99]. This model solves traffic congestion in less time. The experiments were conducted on three different datasets and achieved good results. Job scheduling has been implemented using multi-agent deep reinforcement learning [100]. Dynamic route optimization for human drivers using multi-agent DQN was implemented by [101]. Predictive maintenance has been performed with multi-agent RL [102]. The agent observed the machine's state and performed the maintenance task based on the prediction, which improved performance by 75%. Multiple agents try to learn simultaneously to improve the cumulative reward. In this case, the agent's policies were difficult to converge. An entropy regularizer is used along with an actor-critic algorithm to overcome this scenario [103].

3.11. Energy Management

EMSs are used for achieving various goals, such as lowering energy consumption, controlling energy supply and demand, enhancing the use of renewable energy and lowering energy expenses. Energy management in a smart grid is a difficult challenge. The authors of [104] proposed a smart grid modelled as a Markov decision process and used Q-learning to reduce energy consumption and cost. The authors of [105] investigated power grid operation and maintenance implemented using a Q-learning-based artificial neural network (ANN). Reinforcement learning has been used in energy management in various contexts, such as building energy management systems, reducing the consumption of electric vehicles, and renewable energy integration. RL has been used in building energy management systems to control heating, ventilation, and air conditioning (HVAC), thereby minimizing energy use while keeping the consumer in their comfort zone. RL has been applied to reduce the fuel consumption of electric vehicles. RL can also be used to optimize the control and management of renewable energy. The authors of [106] reviewed

the advantages and disadvantages of using RL in the energy consumption of intelligent buildings. Real-time deployment of RL in intelligent buildings to reduce energy usage may pose difficulties in real-world scenarios as training takes a long time.

3.11.1. HVAC Control in Buildings

Most people like to spend time inside their homes, and indoor air quality affects human health. Higher levels of CO_2 cause ill health in human beings. It is essential to maintain the level of CO_2. RL-based approaches have been proposed to monitor and manage the level of CO_2 [107,108]. These authors proposed smart home-based energy management using RL. The smart home consisted of thermal storage systems, rooftop photovoltaics, battery storage systems, etc. They designed the energy management with a constrained MDP. A primal-dual deterministic policy gradient algorithm was used to reduce consumption and cost function. Similarly, thermal comfort is necessary for the employees in an office environment to work comfortably and also to avoid health issues. The system has to provide both comfort and consume less energy. The authors of [109] proposed a cooling water system using multi-agent reinforcement learning based on a model-free algorithm. The energy consumption of their proposed method was better than the rule-based model and close to the model-based design. A multi-actor attention critic approach has been proposed for controlling HVAC [110]. These researchers used a model-free algorithm and framed their own Markov decision process, which consists of states, actions, and rewards. They used comfort-related performance metrics such as average temperature deviation and average CO_2 concentration deviation. In another work [111], it was suggested that control systems do not need models since they learn through interaction with their environment. These researchers used a double deep Q network and performed the hyperparameter tuning in the double deep Q network to obtain the desired results. In addition, they compared the results with a model predictive control. Their proposed method outperformed the model predictive control regarding the deviation of temperature and period.

An actor-critic-based model was proposed for regulating ventilation, air conditioning, and heating [112]. These authors investigated the relationship between energy usage and thermal stability and found that the SAC algorithm outperformed the existing algorithms. The authors of [113] framed the occupant behaviour as a Markov decision process. The thermostat was adjusted based on the behaviour of the occupants. In the winter season, the model had to raise the temperature. In this case, the model receives positive rewards. When the occupant feels a neutral temperature, the model increasing or decreasing the temperature results in negative rewards. The researchers performed the simulation for four different seasons with reinforcement learning. In addition, they used transfer learning to transfer the trained RL model to a separate building with only a few pieces of information.

The authors of [114], applied a DRL-based system to minimize energy use while keeping the occupant in their comfort zone using a DDPG algorithm. Another work reviewed the challenges of managing the energy consumption system in buildings [115]. These researchers divided the work into single-building RL, clusters of buildings, and MARL. MDP modifications, off-policy algorithms, and model-based RL can improve the sample efficiency of an RL model. Expert knowledge can reduce the RL's training time. Therefore, they used transfer learning instead of starting from scratch in different buildings. The conclusions of their review were that model-based RL provides good results in terms of sample efficiency. They also mentioned that transfer learning shows promising results. Most of the research work focused on single-variable control.

The authors of [116] proposed a multi-variate control using a branching dueling Q-network. This agent was pre-trained in the simulated environment and then deployed in a real-world environment. They reduced the cooling energy by 14% and improved thermal acceptability by 11%. The authors of [117] proposed an attention-based multi-agent DRL which does not require prior knowledge of an uncertain environment. They promoted the coordination between the personal comfort system and HVAC. In another

work [118], researchers framed a non-stationary MDP. Whenever there is a degradation in the performance, this DRL model relearns to improve the performance. To improve the sample efficiency, they used elastic weight consolidation. They compared it with the classic rule-based model, model predictive control, PPO, and DDPG models. Table 4 provides an overview of HVAC control in buildings using RL algorithms.

Table 4. Reinforcement learning applications for HVAC control.

S. No	References	Applications	Algorithms
1	[109]	Water cooling system	Multi-agent DRL
2	[110]	HVAC control in commercial buildings	Multi-agent DRL with actor attention critic
3	[111]	Control of HVAC considering dynamic occupant patterns	Double deep Q networks
4	[112]	HVAC control by maintaining the thermal stability	Actor-critic
5	[113]	HVAC control based on occupant behaviour for different buildings	Q-learning
6	[114]	Multi-zone HVAC control	Deep deterministic policy gradient
7	[116]	Multi-variant occupant-centric HVAC	Branching dueling Q-network
8	[117]	HVAC control in office buildings	Attention-based multi-agent DRL
9	[118]	HVAC control for non-stationary buildings	A deep reinforcement learning model

3.11.2. Energy Management in Vehicles

Vehicle fuel management has received attention due to the scarcity of energy and oil and environmental issues such as the greenhouse effect. The authors of [119] proposed a reverse RL based on the energy management of hybrid vehicles. In [120], a DRL-based automated guided vehicle (AGV) is presented that controls its speed according to the environment. These researchers designed a deep deterministic policy gradient according to the actor-critic algorithm. This method reduced energy consumption by 4.6%. In hybrid electric vehicles, hydrogen is used to reduce pollution. Fuel cell vehicles typically need a combination of various power sources to satisfy driving demands. Different power sources can be optimized with an EMS. A deep Q-learning system with priority experience replay and DDPG with priority experience replay to reduce consumption has been proposed [121]. Fuel cells have some disadvantages such as delays in response and the inability to recover braking energy. Researchers have proposed a TD3 algorithm for logistic trucks to reduce hydrogen consumption and extend the fuel and battery life span [122]. They compared a deep deterministic policy gradient and a non-linear programming algorithm. Their proposed algorithm minimizes the ageing of core components and hydrogen consumption. Eco-driving is an effective technology for reducing energy consumption in vehicles. An EMS using the Q-learning algorithm has been implemented [123]. These authors compared their algorithm with dynamic programming. The RL-based model adjusts the vehicle's speed based on road conditions and maintains a suitable distance from the leading vehicle. The RL model simulation results provided near-optimal performance compared with dynamic programming. Other researchers have proposed a SAC algorithm [124]. They employed cooperative optimization and implemented the MARL for eco-driving by optimizing energy management. They framed the reward function concerning collision avoidance, driving comfort, and energy efficiency. This model outperformed the hierarchical model used as a predictive control. Table 5 provides the RL algorithms for reducing energy consumption in hybrid vehicles.

Table 5. Reinforcement learning applications of energy management in vehicles.

S. No	References	Applications	Algorithms
1	[120]	Automated guided vehicle	Actor-critic and DDPG
2	[121]	Reducing the fuel consumption in hybrid vehicles	Deep Q-learning and DDPG
3	[122]	Energy management strategy for logistic trucks	TD3
4	[123]	Energy-efficient eco-driving by controlling the speed in electric vehicles	Q-learning
5	[124]	Eco-driving in hybrid electric vehicles	MARL using SAC

3.11.3. Renewable Energy Management

Renewable energy contributes to an eco-friendly environment and sustainable use. Recently, renewable energy has been used in buildings. Solar energy is one of the fastest-growing sources of renewable energy. Traditional rule-based and model predictive controllers manage use for optimizing energy resources. Renewable energy optimization has been examined by [125]. In this study, the Gaussian distribution was used to frame the reward function. They compared the two DRL algorithms TD3 and DDPG. They tested the algorithms under both random and extreme conditions. In another work [126], the authors proposed a DRL-based renewable energy optimization. Their goal was to produce hydrogen and sell it to increase revenue, and they created this model to be used in conjunction with OpenAI Gym and Ray/RLib for deep reinforcement learning applications. They used three algorithms including TD3, PPO, and SAC. PPO provided stable performance over the other two algorithms.

Researchers have proposed a storage system for renewable energy based on hydrogen using DRL [127]. They contrasted this strategy to rule-based and dynamic programming. The DRL model outperformed the rule-based method in simulation results. They employed the PPO method with the stochastic gradient descent (SGD) optimizer and fine-tuned the parameters with hyperparameters. Wind energy is one of the most environmentally friendly sources of electricity. The authors of [128] applied actor-critic RL for wind turbine control. Wind turbines are devices that transform kinetic energy into electrical energy. The simulation was carried out using the OpenFAST simulator. Table 6 shows how renewable energy can be managed using RL algorithms.

Table 6. Reinforcement learning application of renewable energy.

S. No	References	Applications	Algorithms
1	[125]	Off-grid optimization of renewable energy	TD3 and DDPG
2	[126]	Nuclear renewable integrated energy system optimization	TD3, PPO, SAC
3	[127]	Storage systems of renewable energy	PPO
4	[128]	Control of wind turbines	Actor-Critic

4. Conclusions

In this research, we reviewed the literature on reinforcement learning. Because RL can learn independently, it is well suited to dynamic contexts. In both critical and non-critical applications, RL is used. Critical applications include self-driving cars, security, healthcare, energy management systems, and finance. The non-critical applications are those such as Gaming. RL has the potential to revolutionize energy management by allowing systems to adapt and optimize energy use in real time. RL algorithms may use previous data to forecast future energy demand, allowing energy systems to modify energy output and consumption accordingly. RL-based EMS can result in more efficient energy consumption and cost savings.

Furthermore, by forecasting and compensating for swings in energy output, RL may improve the functioning of renewable energy sources such as wind and solar power. Overall, RL has the potential to increase the efficiency and efficacy of energy management systems significantly. Most EMS applications have been tested in simulation environments. In the future, more EMS should be implemented in real-world world settings. RL has a broad scope in the future with the potential to enhance the energy efficiency of HVAC systems dramatically. Still, additional research is needed to build RL algorithms that can manage the complexity and unpredictability of real-world HVAC systems.

Recommendation systems built with RL work effectively and have high prediction accuracy. In security-related work, RL is used in a simulated environment, but it may be used in the real world in the future. With the help of RL, gaming applications are expanding. RL is also being used in financial applications. More works have been published in trading applications using RL, but fewer papers have been published about credit risk analysis using RL. In the future, RL has a broad scope for applications in all fields

Author Contributions: Conceptualization, K.S., E.R., B.A., S.N., S.V. and I.V.; methodology, K.S., E.R., B.A., S.N., S.V. and I.V.; investigation, K.S., E.R., B.A., S.N., S.V. and I.V.; writing—original draft preparation, K.S., E.R., B.A., S.N., S.V. and I.V.; writing—review and editing, K.S., E.R., B.A., S.N., S.V. and I.V. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Acknowledgments: We would like to thank SASTRA Deemed University for providing infrastructure facilities.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kadhim, A.I. Survey on Supervised Machine Learning Techniques. *Artif. Intell. Rev.* **2019**, *52*, 273–292. [CrossRef]
2. Yau, K.A.; Elkhatib, Y.; Hussain, A.; Al-fuqaha, A.L.A. Unsupervised Machine Learning for Networking: Techniques, Applications and Research Challenges. *IEEE Access* **2019**, *7*, 65579–65615. [CrossRef]
3. Singh, B.; Kumar, R.; Singh, V.P. Reinforcement Learning in Robotic Applications: A Comprehensive Survey. *Artif. Intell. Rev.* **2022**, *55*, 1–46. [CrossRef]
4. Rao, S.; Verma, A.K.; Bhatia, T.A. Review on Social Spam Detection: Challenges, Open Issues, and Future Directions. *Expert Syst. Appl.* **2021**, *186*, 115742. [CrossRef]
5. Sahil, S.; Zaidi, A.; Samar, M.; Aslam, A.; Kanwal, N.; Asghar, M.; Lee, B. A Survey of Modern Deep Learning Based Object Detection Models. *Digit. Signal Process.* **2022**, *126*, 103514. [CrossRef]
6. Bochenek, B.; Ustrnul, Z. Machine Learning in Weather Prediction and Climate Analyses—Applications and Perspectives. *Atmosphere* **2022**, *13*, 180. [CrossRef]
7. Keerthana, S.; Santhi, B. Survey on Applications of Electronic Nose. *J. Comput. Sci.* **2020**, *16*, 314–320. [CrossRef]
8. Razzaghi, P.; Tabrizian, A.; Guo, W.; Chen, S.; Taye, A.; Thompson, E.; Wei, P. A Survey on Reinforcement Learning in Aviation Applications. *arXiv* **2022**, arXiv:2211.02147.
9. Islek, I.; Gunduz, S. A Hierarchical Recommendation System for E-Commerce Using Online User Reviews. *Electron. Commer. Res. Appl.* **2022**, *52*, 101131. [CrossRef]
10. Elangovan, R.; Vairavasundaram, S.; Varadarajan, V.; Ravi, L. Location-Based Social Network Recommendations with Computational Intelligence-Based Similarity Computation and User Check-in Behavior. *Concurr. Comput. Pract. Exp.* **2021**, *33*, 1–16. [CrossRef]
11. Asik Ibrahim, N.; Rajalakshmi, E.; Vijayakumar, V.; Elakkiya, R.; Subramaniyaswamy, V. An Investigation on Personalized Point-of-Interest Recommender System for Location-Based Social Networks in Smart Cities. *Adv. Sci. Technol. Secur. Appl.* **2021**, 275–294. [CrossRef]
12. Afsar, M.M.; Crump, T.; Far, B. Reinforcement Learning Based Recommender Systems: A Survey. *ACM Comput. Surv.* **2022**, *55*, 1–38. [CrossRef]
13. Adams, S.; Cody, T.; Beling, P.A. A Survey of Inverse Reinforcement Learning. *Artif. Intell. Rev.* **2022**, *55*, 4307–4346. [CrossRef]
14. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An introduction*; Sutton Barto Second Book; MIT Press: Cambridge, MA, USA, 1998; Volume 258. [CrossRef]
15. Liu, B.; Xie, Y.; Feng, L.; Fu, P. Engineering Applications of Artificial Intelligence Correcting Biased Value Estimation in Mixing Value-Based Multi-Agent Reinforcement Learning by Multiple Choice Learning. *Eng. Appl. Artif. Intell.* **2022**, *116*, 105329. [CrossRef]

16. Yu, M.; Sun, S. Engineering Applications of Artificial Intelligence Policy-Based Reinforcement Learning for Time Series Anomaly Detection. *Eng. Appl. Artif. Intell.* **2020**, *95*, 103919. [CrossRef]
17. Wei, Q.; Yang, Z.; Su, H.; Wang, L. Monte Carlo-Based Reinforcement Learning Control for Unmanned Aerial Vehicle Systems. *Neurocomputing* **2022**, *507*, 282–291. [CrossRef]
18. Wang, Q.; Hao, Y.; Cao, J. Learning to Traverse over Graphs with a Monte Carlo Tree Search-Based Self-Play Framework. *Eng. Appl. Artif. Intell.* **2021**, *105*, 104422. [CrossRef]
19. Ramicic, M.; Bonarini, A. Correlation Minimizing Replay Memory in Temporal-Difference Reinforcement Learning. *Neurocomputing* **2020**, *393*, 91–100. [CrossRef]
20. Bertsekas, D. Results in Control and Optimization Multi-agent Value Iteration Algorithms in Dynamic Programming and Reinforcement Learning. *Results Control Optim.* **2020**, *1*, 100003. [CrossRef]
21. van Eck, N.J.; van Wezel, M. Application of Reinforcement Learning to the Game of Othello. *Comput. Oper. Res.* **2008**, *35*, 1999–2017. [CrossRef]
22. Maoudj, A.; Hentout, A. Optimal Path Planning Approach Based on Q-Learning Algorithm for Mobile Robots. *Appl. Soft Comput. J.* **2020**, *97*, 106796. [CrossRef]
23. Aljohani, T.M.; Mohammed, O. A Real-Time Energy Consumption Minimization Framework for Electric Vehicles Routing Optimization Based on SARSA Reinforcement Learning. *Vehicles* **2022**, *4*, 1176–1194. [CrossRef]
24. Lin, Y.; Feng, S.; Lin, F.; Zeng, W.; Liu, Y.; Wu, P. Adaptive Course Recommendation in MOOCs. *Knowl.-Based Syst.* **2021**, *224*, 107085. [CrossRef]
25. Lin, Y.; Lin, F.; Zeng, W.; Xiahou, J.; Li, L.; Wu, P.; Liu, Y.; Miao, C. Hierarchical Reinforcement Learning with Dynamic Recurrent Mechanism for Course Recommendation. *Knowl.-Based Syst.* **2022**, *244*, 108546. [CrossRef]
26. Tang, X.; Chen, Y.; Li, X.; Liu, J.; Ying, Z. A Reinforcement Learning Approach to Personalized Learning Recommendation Systems. *Br. J. Math. Stat. Psychol.* **2019**, *72*, 108–135. [CrossRef]
27. Ke, G.; Du, H.L.; Chen, Y.C. Cross-Platform Dynamic Goods Recommendation System Based on Reinforcement Learning and Social Networks. *Appl. Soft Comput.* **2021**, *104*, 107213. [CrossRef]
28. Chen, Y. Towards Smart Educational Recommendations with Reinforcement Learning in Classroom. In Proceedings of the IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE), Wollongong, NSW, Australia, 4–7 December 2018; pp. 1079–1084. [CrossRef]
29. Jiang, P.; Ma, J.; Zhang, J. Deep Reinforcement Learning Based Recommender System with State Representation. In Proceedings of the 2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, 15–18 December 2021; pp. 5703–5707. [CrossRef]
30. Yuyan, Z.; Xiayao, S.; Yong, L. A Novel Movie Recommendation System Based on Deep Reinforcement Learning with Prioritized Experience Replay. In Proceedings of the 2019 IEEE 19th International Conference on Communication Technology (ICCT), Xi'an, China, 16–19 October 2019; pp. 1496–1500. [CrossRef]
31. Fu, M.; Agrawal, A.; Irissappane, A.A.; Zhang, J.; Huang, L.; Qu, H. Deep Reinforcement Learning Framework for Category-Based Item Recommendation. *IEEE Trans. Cybern.* **2021**, *52*, 12028–12041. [CrossRef]
32. Huang, L.; Fu, M.; Li, F.; Qu, H.; Liu, Y.; Chen, W. A Deep Reinforcement Learning Based Long-Term Recommender System. *Knowl.-Based Syst.* **2021**, *213*, 106706. [CrossRef]
33. Zhao, X.; Xia, L.; Zhang, L.; Tang, J.; Ding, Z.; Yin, D. Recommendations with Negative Feedback via Pairwise Deep Reinforcement Learning. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, London, UK, 19–23 August 2018; pp. 1040–1048. [CrossRef]
34. Gao, R.; Xia, H.; Li, J.; Liu, D.; Chen, S.; Chun, G. DRCGR: Deep Reinforcement Learning Framework Incorporating CNN and GAN-Based for Interactive Recommendation. In Proceedings of the 2019 IEEE International Conference on Data Mining (ICDM), Beijing, China, 8–11 November 2019; pp. 1048–1053. [CrossRef]
35. Zhou, F.; Luo, B.; Hu, T.; Chen, Z.; Wen, Y. A Combinatorial Recommendation System Framework Based on Deep Reinforcement Learning. In Proceedings of the 2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, 15–18 December 2021; pp. 5733–5740. [CrossRef]
36. Lei, Y.; Li, W. Interactive Recommendation with User-Specific Deep Reinforcement Learning. *ACM Trans. Knowl. Discov. Data* **2019**, *13*, 1–15. [CrossRef]
37. Guo, D.; Ktena, S.I.; Myana, P.K.; Huszar, F.; Shi, W.; Tejani, A.; Kneier, M.; Das, S. Deep Bayesian Bandits: Exploring in Online Personalized Recommendations. In Proceedings of the 14th ACM Conference on Recommender Systems, Virtual Event, Brazil, 22–26 September 2020; pp. 456–461. [CrossRef]
38. Gan, M.; Kwon, O.C. A Knowledge-Enhanced Contextual Bandit Approach for Personalized Recommendation in Dynamic Domains. *Knowl.-Based Syst.* **2022**, *251*, 109158. [CrossRef]
39. Pilani, A.; Mathur, K.; Agrawal, H.; Chandola, D.; Tikkiwal, V.A.; Kumar, A. Contextual Bandit Approach-Based Recommendation System for Personalized Web-Based Services. *Appl. Artif. Intell.* **2021**, *35*, 489–504. [CrossRef]
40. Yan, C.; Xian, J.; Wan, Y.; Wang, P. Modeling Implicit Feedback Based on Bandit Learning for Recommendation. *Neurocomputting* **2021**, *447*, 244–256. [CrossRef]

41. Wang, L.; Wang, C.; Wang, K.; He, X. BiUCB: A Contextual Bandit Algorithm for Cold-Start and Diversified Recommendation. In Proceedings of the 2017 IEEE International Conference on Big Knowledge (ICBK), Hefei, China, 9–10 August 2017; pp. 248–253. [CrossRef]
42. Intayoad, W.; Kamyod, C.; Temdee, P. Reinforcement Learning Based on Contextual Bandits for Personalized Online Learning Recommendation Systems. *Wirel. Pers. Commun.* **2020**, *115*, 2917–2932. [CrossRef]
43. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Graves, A.; Antonoglou, I.; Wierstra, D.; Riedmiller, M. Playing Atari with Deep Reinforcement Learning. *arXiv* **2013**, arXiv:1312.5602.
44. Wang, M.; Wang, L.; Yue, T. An Application of Continuous Deep Reinforcement Learning Approach to Pursuit-Evasion Differential Game. In Proceedings of the 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chengdu, China, 15–17 March 2019; pp. 1150–1156. [CrossRef]
45. Rajendran, D.; Santhanam, P. Towards Digital Game-Based Learning Content with Multi-Objective Reinforcement Learning. *Mater. Today Proc.* **2021**, 2214–7853. [CrossRef]
46. Liu, S.; Cao, J.; Wang, Y.; Chen, W.; Liu, Y. Self-Play Reinforcement Learning with Comprehensive Critic in Computer Games. *Neurocomputing* **2021**, *449*, 207–213. [CrossRef]
47. Silver, D.; Huang, A.; Maddison, C.J.; Guez, A.; Sifre, L.; Van Den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. Mastering the Game of Go with Deep Neural Networks and Tree Search. *Nature* **2016**, *529*, 484–489. [CrossRef]
48. Silver, D.; Hubert, T.; Schrittwieser, J.; Antonoglou, I.; Lai, M.; Guez, A.; Lanctot, M.; Sifre, L.; Kumaran, D.; Graepel, T.; et al. A General Reinforcement Learning Algorithm That Masters Chess, Shogi, and Go through Self-Play. *Science* **2018**, *362*, 1140–1144. [CrossRef]
49. Núñez-molina, C.; Fernández-olivares, J.; Pérez, R. Learning to Select Goals in Automated Planning with Deep-Q Learning. *Expert Syst. Appl.* **2022**, *202*, 117265. [CrossRef]
50. Gong, X.; Yu, J.; Lü, S.; Lu, H. Actor-Critic with Familiarity-Based Trajectory Experience Replay. *Inf. Sci.* **2022**, *582*, 633–647. [CrossRef]
51. Badue, C.; Guidolini, R.; Carneiro, R.V.; Azevedo, P.; Cardoso, V.B.; Forechi, A.; Jesus, L.; Berriel, R.; Paixão, T.M.; Mutz, F.; et al. Self-Driving Cars: A Survey. *Expert Syst. Appl.* **2021**, *165*, 113816. [CrossRef]
52. Cao, Z.; Xu, S.; Peng, H.; Yang, D.; Zidek, R. Confidence-Aware Reinforcement Learning for Self-Driving Cars. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 7419–7430. [CrossRef]
53. Kim, M.-S.; Eoh, G.; Park, T.-H. Decision Making for Self-Driving Vehicles in Unexpected Environments Using Efficient Reinforcement Learning Methods. *Electronics* **2022**, *11*, 1685. [CrossRef]
54. Uc-Cetina, V.; Navarro-Guerrero, N.; Martin-Gonzalez, A.; Weber, C.; Wermter, S. Survey on Reinforcement Learning for Language Processing. *Artif. Intell. Rev.* **2022**, 1–33. [CrossRef]
55. Alomari, A.; Idris, N.; Qalid, A.; Alsmadi, I. Deep Reinforcement and Transfer Learning for Abstractive Text Summarization: A Review. *Comput. Speech Lang.* **2022**, *71*, 101276. [CrossRef]
56. Cuayáhuitl, H.; Lee, D.; Ryu, S.; Cho, Y.; Choi, S.; Indurthi, S.; Yu, S.; Choi, H.; Hwang, I.; Kim, J. Ensemble-Based Deep Reinforcement Learning for Chatbots. *Neurocomputing* **2019**, *366*, 118–130. [CrossRef]
57. Uprety, A.; Rawat, D.B. Reinforcement Learning for IoT Security: A Comprehensive Survey. *IEEE Internet Things J.* **2021**, *8*, 8693–8706. [CrossRef]
58. Nguyen, T.T.; Reddi, V.J. Deep Reinforcement Learning for Cyber Security. *IEEE Trans. Neural Networks Learn. Syst.* **2021**, 1–17. [CrossRef]
59. Hu, Y.J.; Lin, S.J. Deep Reinforcement Learning for Optimizing Finance Portfolio Management. In Proceedings of the 2019 Amity International Conference on Artificial Intelligence (AICAI), Dubai, United Arab Emirates, 4–6 February 2019; pp. 14–20. [CrossRef]
60. Wu, X.; Chen, H.; Wang, J.; Troiano, L.; Loia, V.; Fujita, H. Adaptive Stock Trading Strategies with Deep Reinforcement Learning Methods. *Inf. Sci.* **2020**, *538*, 142–158. [CrossRef]
61. Chen, Y.F.; Huang, S.H. Sentiment-Influenced Trading System Based on Multimodal Deep Reinforcement Learning. *Appl. Soft Comput.* **2021**, *112*, 107788. [CrossRef]
62. Lei, K.; Zhang, B.; Li, Y.; Yang, M.; Shen, Y. Time-Driven Feature-Aware Jointly Deep Reinforcement Learning for Financial Signal Representation and Algorithmic Trading. *Expert Syst. Appl.* **2020**, *140*, 1–14. [CrossRef]
63. Jeong, G.; Young, H. Improving Financial Trading Decisions Using Deep Q-Learning: Predicting the Number of Shares, Action Strategies, and Transfer Learning. *Expert Syst. Appl.* **2019**, *117*, 125–138. [CrossRef]
64. Liu, F.; Bing, X. Bitcoin Transaction Strategy Construction Based on Deep Reinforcement Learning. *Appl. Soft Comput.* **2021**, *113*, 5–8. [CrossRef]
65. Kanashiro, L.; Caio, F.; Paiva, L.; Vita, C.; De Yathie, E.; Helena, A.; Costa, R.; Del-moral-hernandez, E.; Brandimarte, P. Outperforming Algorithmic Trading Reinforcement Learning Systems: A Supervised Approach to the Cryptocurrency Market. *Expert Syst. Appl.* **2022**, *202*, 117259.
66. Serrano, W. Deep Reinforcement Learning with the Random Neural Network. *Eng. Appl. Artif. Intell.* **2022**, *110*, 104751. [CrossRef]
67. Shavandi, A.; Khedmati, M. A Multi-Agent Deep Reinforcement Learning Framework for Algorithmic Trading in Financial Markets. *Expert Syst. Appl.* **2022**, *208*, 118124. [CrossRef]

68. Carta, S.; Ferreira, A.; Podda, A.S.; Recupero, D.R.; Sanna, A. Multi-DQN: An Ensemble of Deep Q-Learning Agents for Stock Market Forecasting. *Expert Syst. Appl.* **2021**, *164*, 113820. [CrossRef]
69. Kang, Q. An Asynchronous Advantage Actor-Critic Reinforcement Learning Method for Stock Selection and Portfolio Management. In Proceedings of the 2nd International Conference on Big Data Research, Weihai, China, 27–29 October 2018; pp. 141–145. [CrossRef]
70. Srinath, T.; Gururaja, H.S. Explainable Machine Learning in Identifying Credit Card Defaulters. *Glob. Transit. Proc.* **2022**, *3*, 119–126. [CrossRef]
71. Addo, P.M.; Guegan, D.; Hassani, B. Credit Risk Analysis Using Machine and Deep Learning Models. *Risks* **2018**, *6*, 1–20. [CrossRef]
72. Dastile, X.; Celik, T.; Potsane, M. Statistical and Machine Learning Models in Credit Scoring: A Systematic Literature Survey. *Appl. Soft Comput. J.* **2020**, *91*, 106263. [CrossRef]
73. Leo, M.; Sharma, S.; Maddulety, K. Machine Learning in Banking Risk Management: A Literature Review. *Risks* **2019**, *7*, 29. [CrossRef]
74. Milojević, N.; Redzepagic, S. Prospects of Artificial Intelligence and Machine Learning Application in Banking Risk Management. *J. Cent. Bank. Theory Pract.* **2021**, *10*, 41–57. [CrossRef]
75. Sabri, A. Reinforcement Learning on the Credit Risk-Based Pricing. In Proceedings of the 2021 2nd International Conference on Computational Methods in Science & Technology (ICCMST), Mohali, India, 17–18 December 2021; pp. 233–236. [CrossRef]
76. Wang, Y.; Jia, Y.; Tian, Y.; Xiao, J. Deep Reinforcement Learning with the Confusion-Matrix-Based Dynamic Reward Function for Customer Credit Scoring. *Expert Syst. Appl.* **2022**, *200*, 117013. [CrossRef]
77. Akalin, N.; Loutfi, A. Reinforcement Learning Approaches in Social Robotics. *Sensors* **2021**, *21*, 1–37. [CrossRef] [PubMed]
78. Bagheri, E.; Roesler, O.; Cao, H.L.; Vanderborght, B. A Reinforcement Learning Based Cognitive Empathy Framework for Social Robots. *Int. J. Soc. Robot.* **2021**, *13*, 1079–1093. [CrossRef]
79. Cao, X.; Sun, C.; Yan, M. Target Search Control of AUV in Underwater Environment with Deep Reinforcement Learning. *IEEE Access* **2019**, *7*, 96549–96559. [CrossRef]
80. Zhu, K.; Zhang, T. Deep Reinforcement Learning Based Mobile Robot Navigation: A Review. *Tsinghua Sci. Technol.* **2021**, *26*, 674–691. [CrossRef]
81. Sun, H.; Zhang, W.; Yu, R.; Zhang, Y. Motion Planning for Mobile Robots-Focusing on Deep Reinforcement Learning: A Systematic Review. *IEEE Access* **2021**, *9*, 69061–69081. [CrossRef]
82. Gao, J.; Ye, W.; Guo, J.; Li, Z. Deep Reinforcement Learning for Indoor Mobile Robot Path Planning. *Sensors* **2020**, *20*, 1–15. [CrossRef]
83. Wang, W.; Wu, Z.; Luo, H.; Zhang, B. Path Planning Method of Mobile Robot Using Improved Deep Reinforcement Learning. *J. Electr. Comput. Eng.* **2022**, *2022*, 1–7. [CrossRef]
84. Guo, N.; Li, C.; Gao, T.; Liu, G.; Li, Y.; Wang, D. A Fusion Method of Local Path Planning for Mobile Robots Based on LSTM Neural Network and Reinforcement Learning. *Math. Probl. Eng.* **2021**, *2021*, 1–21. [CrossRef]
85. Luong, M.; Pham, C. Incremental Learning for Autonomous Navigation of Mobile Robots Based on Deep Reinforcement Learning. *J. Intell. Robot. Syst.* **2021**, *101*, 1–11. [CrossRef]
86. Manuel, J.; Delgado, D.; Oyedele, L. Advanced Engineering Informatics Robotics in Construction: A Critical Review of the Reinforcement Learning and Imitation Learning Paradigms. *Adv. Eng. Informatics* **2022**, *54*, 101787. [CrossRef]
87. Liu, Z.; Yao, C.; Yu, H.; Wu, T. Deep Reinforcement Learning with Its Application for Lung Cancer Detection in Medical Internet of Things. *Futur. Gener. Comput. Syst.* **2019**, *97*, 1–9. [CrossRef]
88. Wang, L.; He, X.; Zhang, W.; Zha, H. Supervised Reinforcement Learning with Recurrent Neural Network for Dynamic Treatment Recommendation. *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.* **2018**, 2447–2456. [CrossRef]
89. Coronato, A.; Naeem, M.; Pietro, G.; De Paragliola, G. Reinforcement Learning for Intelligent Healthcare Applications: A Survey. *Artif. Intell. Med.* **2020**, *109*, 101964. [CrossRef]
90. Wang, L.; Xi, S.; Qian, Y.; Huang, C. A Context-Aware Sensing Strategy with Deep Reinforcement Learning for Smart Healthcare. *Pervasive Mob. Comput.* **2022**, *83*, 101588. [CrossRef]
91. Ho, S.; Jin, S.; Park, J. Knowledge-Based Systems Effective Data-Driven Precision Medicine by Cluster-Applied Deep Reinforcement Learning. *Knowl.-Based Syst.* **2022**, *256*, 109877. [CrossRef]
92. Ho, S.; Park, J.; Jin, S.; Kang, S.; Mo, J. Reinforcement Learning-Based Expanded Personalized Diabetes Treatment Recommendation Using South Korean Electronic Health Records. *Expert Syst. Appl.* **2022**, *206*, 117932. [CrossRef]
93. Liu, S.; Jiang, H. Personalized Route Recommendation for Ride-Hailing with Deep Inverse Reinforcement Learning and Real-Time Traffic Conditions. *Transp. Res. Part E* **2022**, *164*, 102780. [CrossRef]
94. Self, R.; Abudia, M.; Mahmud, S.M.N.; Kamalapurkar, R. Model-Based Inverse Reinforcement Learning for Deterministic. *Automatica* **2022**, *140*, 110242. [CrossRef]
95. Lian, B.; Xue, W.; Lewis, F.L.; Chai, T. Inverse Reinforcement Learning for Multiplayer Non-cooperative Apprentice Games. *Automatica* **2022**, *145*, 110524. [CrossRef]
96. Lian, B.; Donge, V.S.; Member, G.S.; Lewis, F.L.; Fellow, L.; Chai, T.; Fellow, L.; Davoudi, A.; Member, S. Data-Driven Inverse Reinforcement Learning Control for Linear Multiplayer Games. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, 1–14. [CrossRef] [PubMed]

97. Liu, S.; Jiang, H.; Chen, S.; Ye, J.; He, R.; Sun, Z. Integrating Dijkstra' s Algorithm into Deep Inverse Reinforcement Learning for Food Delivery Route Planning. *Transp. Res. Part E* **2020**, *142*, 102070. [CrossRef]
98. Hoiles, W.; Krishnamurthy, V.; Pattanayak, K. Rationally Inattentive Inverse Reinforcement Learning Explains Youtube Commenting Behavior. *J. Mach. Learn. Res.* **2020**, *21*, 1–39.
99. Zhu, R.; Li, L.; Wu, S.; Lv, P.; Li, Y.; Xu, M. Multi-Agent Broad Reinforcement Learning for Intelligent Traffic Light Control. *Inf. Sci.* **2023**, *619*, 509–525. [CrossRef]
100. Zhang, J.; He, Z.; Chan, W.; Chow, C. Knowledge-Based Systems DeepMAG: Deep Reinforcement Learning with Multi-Agent Graphs for Flexible Job Shop Scheduling. *Knowl.-Based Syst.* **2023**, *259*, 110083. [CrossRef]
101. Shou, Z.; Chen, X.; Fu, Y.; Di, X. Multi-Agent Reinforcement Learning for Markov Routing Games: A New Modeling Paradigm for Dynamic Traffic Assignment. *Transp. Res. Part C* **2022**, *137*, 103560. [CrossRef]
102. Luis, M.; Rodríguez, R.; Kubler, S.; Giorgio, A.D.; Cordy, M.; Robert, J.; Le, Y. Robotics and Computer-Integrated Manufacturing Multi-Agent Deep Reinforcement Learning Based Predictive Maintenance on Parallel Machines. *Robot. Comput. Integr. Manuf.* **2022**, *78*, 102406. [CrossRef]
103. Hao, D.; Zhang, D.; Shi, Q.; Li, K. Entropy Regularized Actor-Critic Based Multi-Agent Deep Reinforcement Learning for Stochastic Games. *Inf. Sci.* **2022**, *617*, 17–40. [CrossRef]
104. Kim, S.; Lim, H. Reinforcement Learning Based Energy Management Algorithm for Smart Energy Buildings. *Energies* **2019**, *11*, 2010. [CrossRef]
105. Rocchetta, R.; Bellani, L.; Compare, M.; Zio, E.; Patelli, E. A Reinforcement Learning Framework for Optimal Operation and Maintenance of Power Grids. *Appl. Energy* **2019**, *241*, 291–301. [CrossRef]
106. Fu, Q.; Han, Z.; Chen, J.; Lu, Y.; Wu, H.; Wang, Y. Applications of Reinforcement Learning for Building Energy Efficiency Control: A Review. *J. Build. Eng.* **2022**, *50*, 104165. [CrossRef]
107. Duhirwe, P.N.; Ngarambe, J.; Yun, G.Y. ScienceDirect Energy-Efficient Virtual Sensor-Based Deep Reinforcement Learning Control of Indoor CO_2 in a Kindergarten. *Front. Archit. Res.* **2022**. [CrossRef]
108. Ding, H.; Xu, Y.; Chew, B.; Hao, S.; Li, Q.; Lentzakis, A. A Safe Reinforcement Learning Approach for Multi-Energy Management of Smart Home. *Electr. Power Syst. Res.* **2022**, *210*, 108120. [CrossRef]
109. Fu, Q.; Chen, X.; Ma, S.; Fang, N.; Xing, B.; Chen, J. Optimal Control Method of HVAC Based on Multi-Agent Deep Reinforcement Learning. *Energy Build.* **2022**, *270*, 112284. [CrossRef]
110. Yu, L.; Sun, Y.; Xu, Z.; Shen, C.; Member, S. Multi-Agent Deep Reinforcement Learning for HVAC Control in Commercial Buildings. *IEEE Trans. Smart Grid* **2021**, *12*, 407–419. [CrossRef]
111. Esrafilian-Najafabadi, M.; Haghighat, F. Towards Self-Learning Control of HVAC Systems with the Consideration of Dynamic Occupancy Patterns: Application of Model-Free Deep Reinforcement Learning. *Build. Environ.* **2022**, *226*, 109747. [CrossRef]
112. Biemann, M.; Scheller, F.; Liu, X.; Huang, L. Experimental Evaluation of Model-Free Reinforcement Learning Algorithms for Continuous HVAC Control. *Appl. Energy* **2021**, *298*, 117164. [CrossRef]
113. Deng, Z.; Chen, Q. Reinforcement Learning of Occupant Behavior Model for Cross-Building Transfer Learning to Various HVAC Control Systems. *Energy Build.* **2021**, *238*, 110860. [CrossRef]
114. Du, Y.; Li, F.; Munk, J.; Kurte, K.; Kotevska, O.; Amasyali, K.; Zandi, H. Multi-Task Deep Reinforcement Learning for Intelligent Multi-Zone Residential HVAC Control. *Electr. Power Syst. Res.* **2021**, *192*, 106959. [CrossRef]
115. Weinberg, D.; Wang, Q.; Timoudas, T.O.; Fischione, C. A Review of Reinforcement Learning for Controlling Building Energy Systems From a Computer Science Perspective. *Sustain. Cities Soc.* **2023**, *89*, 104351. [CrossRef]
116. Lei, Y.; Zhan, S.; Ono, E.; Peng, Y.; Zhang, Z.; Hasama, T.; Chong, A. A Practical Deep Reinforcement Learning Framework for Multi-variate Occupant-Centric Control in Buildings. *Appl. Energy* **2022**, *324*, 119742. [CrossRef]
117. Yu, L.; Xu, Z.; Zhang, T.; Guan, X.; Yue, D. Energy-Efficient Personalized Thermal Comfort Control in Office Buildings Based on Multi-Agent Deep Reinforcement Learning. *Build. Environ.* **2022**, *223*, 109458. [CrossRef]
118. Naug, A.; Quinones-Grueiro, M.; Biswas, G. Deep Reinforcement Learning Control for Non-Stationary Building Energy Management. *Energy Build.* **2022**, *277*, 112584. [CrossRef]
119. Lv, H.; Qi, C.; Song, C.; Song, S.; Zhang, R.; Xiao, F. Energy Management of Hybrid Electric Vehicles Based on Inverse Reinforcement Learning. *Energy Rep.* **2022**, *8*, 5215–5224. [CrossRef]
120. Drungilas, D.; Kurmis, M.; Senulis, A.; Lukosius, Z.; Andziulis, A.; Januteniene, J.; Bogdevicius, M.; Jankunas, V.; Voznak, M. Deep Reinforcement Learning Based Optimization of Automated Guided Vehicle Time and Energy Consumption in a Container Terminal. *Alexandria Eng. J.* **2023**, *67*, 397–407. [CrossRef]
121. Huo, W.; Chen, D.; Tian, S.; Li, J.; Zhao, T.; Liu, B. Lifespan-Consciousness and Minimum-Consumption Coupled Energy Management Strategy for Fuel Cell Hybrid Vehicles via Deep Reinforcement Learning. *Int. J. Hydrog. Energy* **2022**, *47*, 24026–24041. [CrossRef]
122. Wang, J.; Zhou, J.; Zhao, W. Deep Reinforcement Learning Based Energy Management Strategy for Fuel Cell/Battery/Supercapacitor Powered Electric Vehicle. *Green Energy Intell. Transp.* **2022**, *1*, 100028. [CrossRef]
123. Lee, H.; Kim, K.; Kim, N.; Cha, S.W. Energy Efficient Speed Planning of Electric Vehicles for Car-Following Scenario Using Model-Based Reinforcement Learning. *Appl. Energy* **2022**, *313*, 118460. [CrossRef]
124. Wang, Y.; Wu, Y.; Tang, Y.; Li, Q.; He, H. Cooperative Energy Management and Eco-Driving of Plug-in Hybrid Electric Vehicle via Multi-Agent Reinforcement Learning. *Appl. Energy* **2023**, *332*, 120563. [CrossRef]

125. Gao, Y.; Matsunami, Y.; Miyata, S.; Akashi, Y. Operational Optimization for Off-Grid Renewable Building Energy System Using Deep Reinforcement Learning. *Appl. Energy* **2022**, *325*, 119783. [CrossRef]
126. Yi, Z.; Luo, Y.; Westover, T.; Katikaneni, S.; Ponkiya, B.; Sah, S.; Mahmud, S.; Raker, D.; Javaid, A.; Heben, M.J.; et al. Deep Reinforcement Learning Based Optimization for a Tightly Coupled Nuclear Renewable Integrated Energy System. *Appl. Energy* **2022**, *328*, 120113. [CrossRef]
127. Dreher, A.; Bexten, T.; Sieker, T.; Lehna, M.; Schütt, J.; Scholz, C.; Wirsum, M. AI Agents Envisioning the Future: Forecast-Based Operation of Renewable Energy Storage Systems Using Hydrogen with Deep Reinforcement Learning. *Energy Convers. Manag.* **2022**, *258*, 115401. [CrossRef]
128. Fernandez-gauna, B.; Graña, M.; Osa-amilibia, J.; Larrucea, X. Actor-Critic Continuous State Reinforcement Learning for Wind-Turbine Control Robust Optimization. *Inf. Sci.* **2022**, *591*, 365–380. [CrossRef]