Ensemble Reinforcement Learning: A Survey

Yanjie Song[1], P. N. Suganthan[2,3], Witold Pedrycz[4], Junwei Ou[1], Yongming He[1], Yingwu Chen[1], Yutong Wu[5]

Abstract

Reinforcement Learning (RL) has emerged as a highly effective technique for addressing various scientific and applied problems. Despite its success, certain complex tasks remain challenging to be addressed solely with a single model and algorithm. In response, ensemble reinforcement learning (ERL), a promising approach that combines the benefits of both RL and ensemble learning (EL), has gained widespread popularity. ERL leverages multiple models or training algorithms to comprehensively explore the problem space and possesses strong generalization capabilities. In this study, we present a comprehensive survey on ERL to provide readers with an overview of recent advances and challenges in the field. First, we introduce the background and motivation for ERL. Second, we analyze in detail the strategies that have been successfully applied in ERL, including model averaging, model selection, and model combination. Subsequently, we summarize the datasets and analyze algorithms used in relevant studies. Finally, we outline several open questions and discuss future research directions of ERL. By providing a guide for future scientific research and engineering applications, this survey contributes to the advancement of ERL.

Keywords: ensemble reinforcement learning, reinforcement learning, ensemble learning, artificial neural network, ensemble strategy

1. Introduction

Over the past several decades, reinforcement learning (RL) methods have proven to be highly effective in solving complex problems across various fields, including gaming, robotics, and computer vision. With the advent of breakthroughs such as deep Q neural networks [1], AlphaGo [2], video games [3, 4], and robotic control tasks [5], RL has witnessed a revitalization that outperforms human performance. The success of this approach is attributed to the agent's ability to automate feature acquisition and complete end-to-end learning. Artificial neural networks (ANN) and gradient descent further enhance RL's exploration and exploitation capabilities, making it suitable for handling time-consuming manual work or challenging tasks.

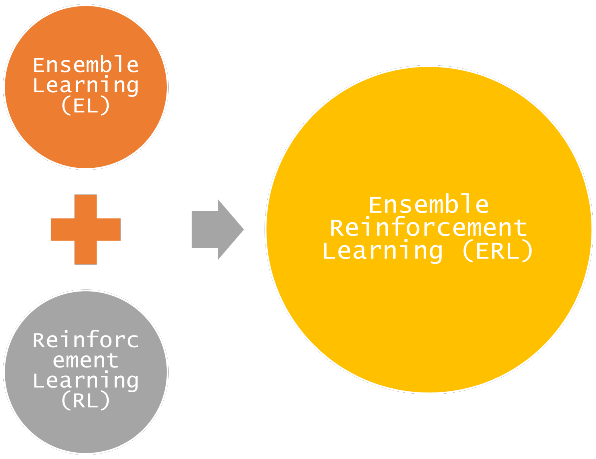

Ensemble Learning (EL)
Reinforc ement Learning (RL)
Ensemble Reinforcement Learning (ERL)

Figure 1: Components of the ERL method

Nevertheless, each type of RL has unique advantages and limitations. For instance, deep reinforcement learning (DRL) requires extensive training to obtain a policy [4], leading to

Email addresses: songyj_2017@163.com (Yanjie Song), p.n.suganthan@qu.edu.qa,EPNSugan@ntu.edu.sg (P. N. Suganthan), wpedrycz@ualberta.ca (Witold Pedrycz), junweiou@163.com (Junwei Ou), heyongming10@hotmail.com (Yongming He), ywchen@nudt.edu.cn (Yingwu Chen), Y.Wu42@newcastle.ac.uk (Yutong Wu)

[1] College of Systems Engineering, National University of Defense Technology, Changsha, China
[2] KINDI Center for Computing Research, College of Engineering, Qatar University, Doha, Qatar
[3] School of Electrical & Electronic Engineering, Nanyang Technological University, Singapore
[4] Department of Electrical & Computer Engineering, University of Alberta, Edmonton AB, Canada
[5] Newcastle University Business School, UK

additional challenges such as overfitting [6], error propagation [7], and imbalance between exploration and exploitation [8]. These challenges serve as motivation for researchers to design their models or training algorithms. One such approach is the implementation of ensemble learning into the RL framework, which presents a novel way to enhance the learning and representation ability of algorithms (see Figure 1). This method, called ensemble reinforcement learning (ERL), has shown excellent performance in various applications. The idea of ensemble learning was first demonstrated by Marquis de Condorcet [9], who showed that average voting outperforms individual model decisions. Subsequent studies by Krogh and Vedelsby [10], Breiman [11], and others have theoretically demonstrated the significant advantages of ensemble methods from different perspectives. The success of ensemble methods in the field of deep learning and reinforcement learning is attributed to the decomposition of datasets [12], powerful learning capabilities [13], and diverse ensemble methods [11].

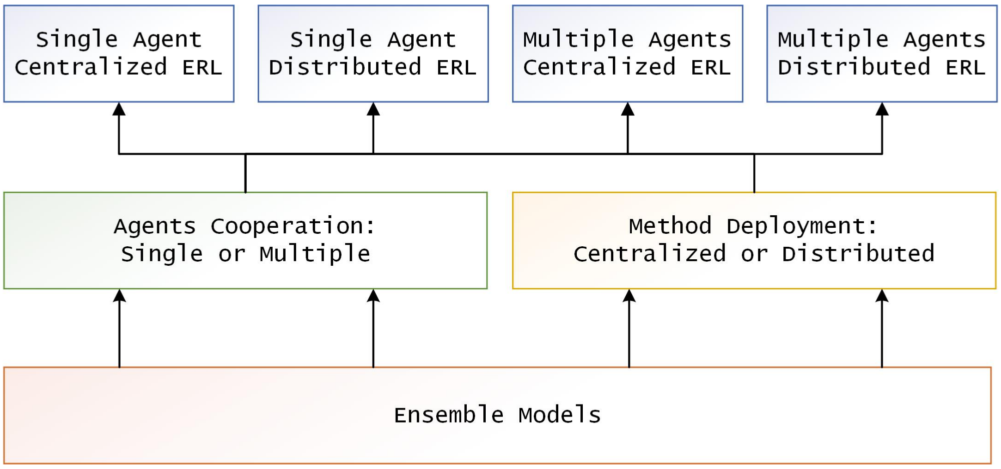

Single Agent Centralized ERL
Single Agent Distributed ERL
Multiple Agents Centralized ERL
Multiple Agents Distributed ERL
Agents Cooperation: Single or Multiple
Method Deployment: Centralized or Distributed
Ensemble Models

Figure 2: A taxonomy of ERL according to agent cooperation and method deployment

The ERL method can be classified according to different criteria. It can be classified into high-level ensembles [14] and low-level ensembles [15] based on the constituent elements. It can also be classified into single-agent ERL [16] and multi-agent ERL [17] based on the number of agents. Furthermore, centralized ERL [18] and distributed ERL [19] are classifications of ERL based on how the agents work. Figure 2 gives a taxonomy according to agent cooperation and method deployment as criteria. All these taxonomies are reasonable and can be used as reference frameworks for designing new ERL methods. Designing new ERL methods can be done quickly based on the existing framework, while understanding the effects of the strategies used in the framework allows for more focused design. In this paper, we provide a detailed description of ERL methods according to the improvement strategies used and discuss their applications to guide the design of new methods.

The existing literature on ERL encompasses a wide range of related work, covering training algorithms, ensemble strategies, and application areas. The motivation of this paper is to provide readers with a systematic overview of the existing related research, the current research progress, and the valuable conclusions achieved. **To the best of our knowledge, this is the first survey focusing solely on ensemble reinforcement learning.** In this survey, we present the strategies used in ERL and related applications, discuss several open questions, and provide a guide for future exploration in the ERL area.

The remainder of this paper is structured as follows. Section 2 presents the background of ensemble reinforcement learning methods. Section 3 introduces implementation strategies in ERL. Section 4 discusses the application of ERL to different domains. Section 5 discusses the datasets and compared methods used in the ERL-related studies. Section 6 discusses several open questions and possible future research directions. Section 7 gives the conclusion of this paper. (See Figure 3).

2. Background

To aid readers in comprehending ensemble reinforcement learning methods, this section provides a brief overview of reinforcement learning (RL) and ensemble learning (EL).

2.1. Reinforcement Learning

Reinforcement learning is an artificial intelligence method in which an agent interacts with an environment and makes decisions to continuously correct errors to obtain optimal

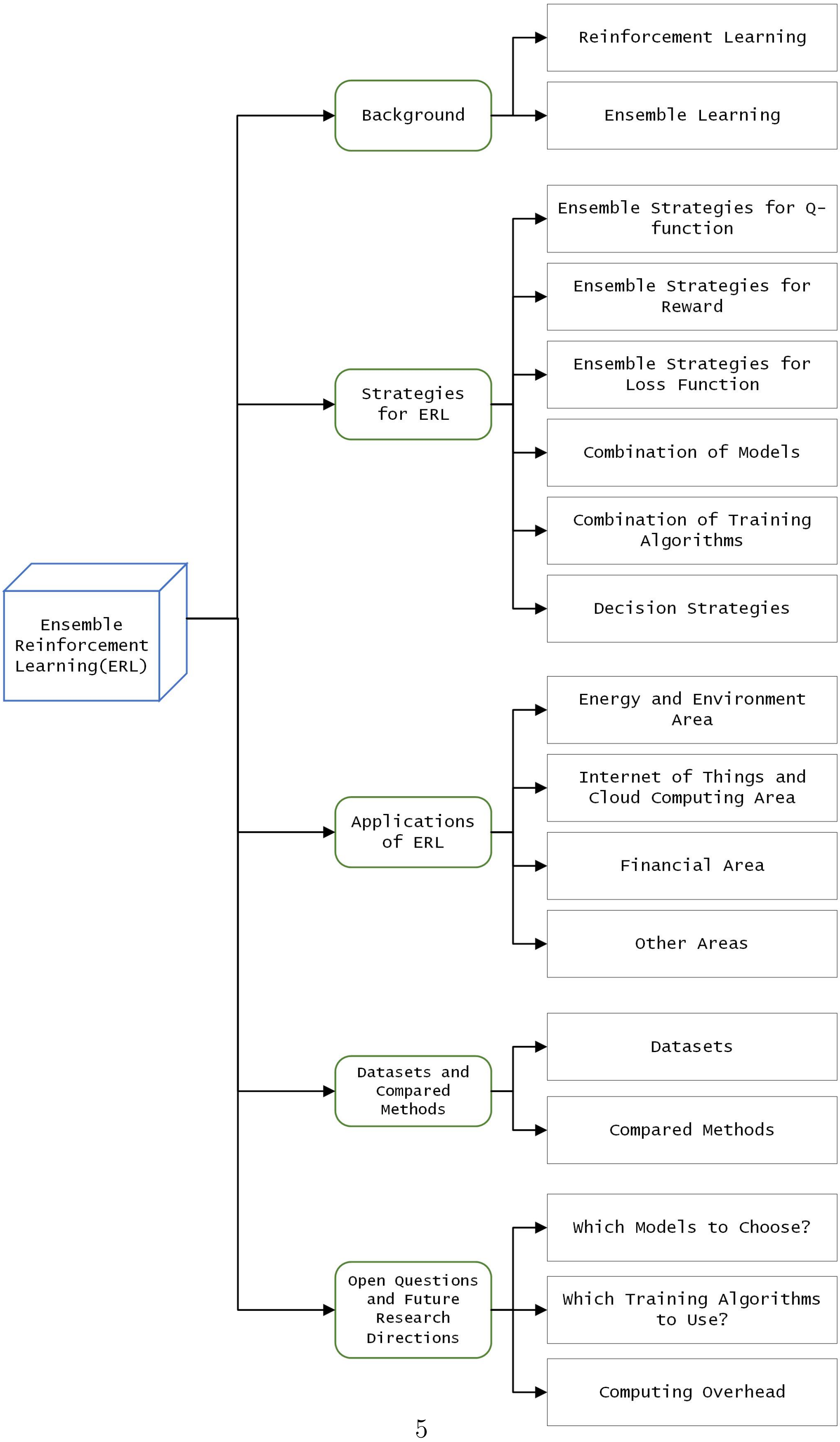

Ensemble Reinforcement Learning(ERL)
Background
Reinforcement Learning
Ensemble Learning
Strategies for ERL
Ensemble Strategies for Q-function
Ensemble Strategies for Reward
Ensemble Strategies for Loss Function
Combination of Models
Combination of Training Algorithms
Decision Strategies
Applications of ERL
Energy and Environment Area
Internet of Things and Cloud Computing Area
Financial Area
Other Areas
Datasets and Compared Methods
Datasets
Compared Methods
Open Questions and Future Research Directions
Which Models to Choose?
Which Training Algorithms to Use?
Computing Overhead

Figure 3: Structure of the paper

decisions. Markov Decision Process (MDP) forms the foundation for using RL to solve problems [20]. RL can be used when an agent's decision is only related to the current state and not to the previous state. Figure 4 illustrates the agent-environment interaction process. A tuple $\langle S, A, P, R, \gamma \rangle$ can represent the MDP, where S denotes the state, A denotes the action, $P : S \times A \rightarrow P(S)$ denotes the state transfer matrix with the probability value $p(s' \mid s) = p(S_{t+1} = s' \mid S_t = s)$, $R : S \times A \rightarrow \mathbb{R}$ denotes the reward function, and $\gamma \in [0, 1]$ denotes the discount factor. The agent's state at time step t is s_t, and it will take action a_t. The combination of all states and actions defines a policy π. Here, the Q-value evaluates the expected return obtained by the agent following policy π.

$$Q^{\pi}(s, a) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \gamma^{t} R(s_t, a_t) | s_0 = s, a_0 = a\right] \tag{1}$$

The aim of using RL methods is to find an optimal policy π that maximizes Q^{π}. For finite-state MDPs, Q-learning is the most typical RL method [21], which uses a Q-table to record the combinations of $\langle$state,action$\rangle$. Subsequently, a series of RL methods using artificial neural networks were proposed to cope with the infinite state space.

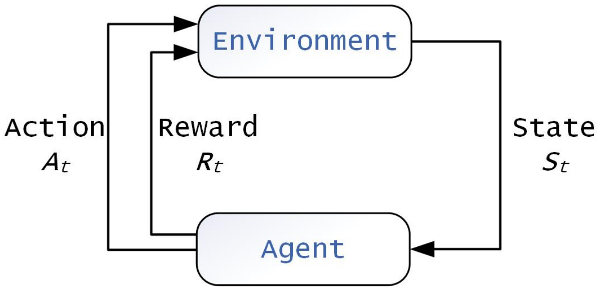


Figure 4: Interaction process between agent and environment

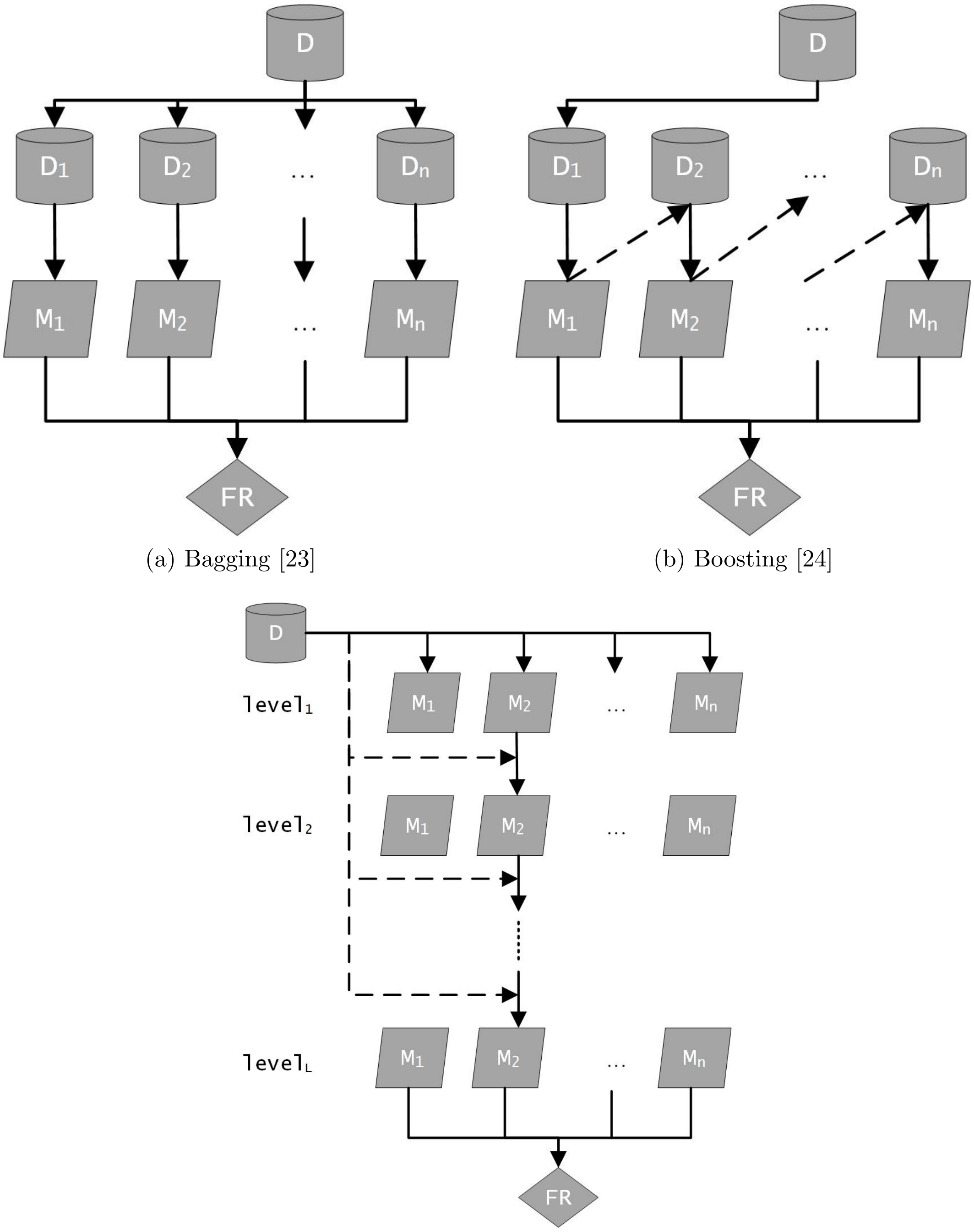
Training algorithms can be divided into model-based RL and model-free RL according to whether the environment model in RL is given in advance or can be obtained through learning. These training algorithms can also be classified according to state-based or policy-based, or state-policy combination-based. More detailed research progress on RL can be found in ref. [22].

2.2. Ensemble Learning

Ensemble learning (EL) is a mainstream approach in the field of machine learning (ML). The core idea of EL methods is to train multiple predictors, combine them, and make a decision from all predictions as the final result of an ensemble model. Compared to individual basic models, this EL method can harness the characteristics of various types of models to improve the predictive performance of EL models and obtain more robust results. The main types of ensemble learning methods include bagging [23], boosting [24], and stacking [25]. Figure 5 gives a schematic diagram of these three types of EL methods, where D denotes the dataset, D_1 to D_n denote the sample selection from the dataset, M_1 to M_n denote the model, and FR denotes the final result. The dotted line in Figure 5-(b) indicates that the weights of samples in the dataset change with the next round of the dataset. The dotted line in Figure 5-(c) indicates that all datasets are used for model prediction from level$_2$ to level$_L$. The main difference between these three types of methods is the way of sample selection. These original and improved EL methods have been applied in various areas, and domain knowledge implemented in the improved EL method achieves outstanding performance. In summary, the EL method has been proven to be advantageous in the following three ways.

- **Bias–variance Decomposition**

The bias-variance decomposition has been widely employed to demonstrate the efficacy of ensemble learning (EL) methods over individual learning methods. While bagging reduces variance among base learners, other EL methods reduce both bias and variance. Krogh and Vedelsby initially demonstrated the effectiveness of EL for problems with a single data set, utilizing the idea of ambiguity decomposition to decrease variance [10]. Subsequently, Brown et al. [26] and Geman et al. [27] verified the effectiveness of EL methods for problems with



(a) Bagging [23]

(b) Boosting [24]

(c) Stacking [25]

Figure 5: Schematic diagram of three types of EL methods

multiple data sets. The decomposition equation can be expressed as follows [12]:

$$E[s-t]^2 = bias^2 + \frac{1}{N}var + (1-\frac{1}{N})covar \tag{2}$$

$$bias = \frac{1}{N}\sum_i (E[s_i]-t) \tag{3}$$

$$var = \frac{1}{N}\sum_i E[s_i - E[s_i]]^2 \tag{4}$$

$$covar = \frac{1}{N(N-1)}\sum_i \sum_{j\neq i} E[s_i - E[s_i]][s_j - E[s_j]] \tag{5}$$

where i denotes the i-th model of EL, s denotes the solution of the problem, and N denotes the number of models in EL. The bias and variance are obtained using the average of the differences between multiple models, while *covar* measures the pairwise difference between models in the EL method.

For a single model, a decrease in bias leads to an increase in variance. However, the ensemble model can be used used for prediction and reduce the variance without increasing bias.

- **Statistical Perspective**

The advantages of EL from the statistical perspective are supported by the work of Dietterich [13]. From a statistical point of view, a machine learning problem exists in a search space with multiple hypotheses. The target of the prediction model is to find the optimal hypothesis. The size of the data used for training is generally only a fraction of the size within the search space, increasing the risk of making incorrect inferences. The use of an EL method can reasonably combine these hypotheses to obtain a better understanding of the search space features and reduce the chance of incorrect classification or invalid prediction.

- **Diversity Perspective**

The advantages of EL from the diversity perspective are intuitive and easy to understand.

Dietterich points out that the combination of different base models can enhance diversity [13]. Some typical EL methods, such as adaboost and random forest, show the importance of diversity from the perspective of training data. And the use of random noise can enhance the richness of the output. In other words, diversity allows decision-makers to combine the model output with usage requirements to obtain a more reasonable final result.

3. Strategies for Ensemble Reinforcement Learning

Previous studies have shown that the Ensemble Reinforcement Learning (ERL) method demonstrates superior average performance and sampling efficiency compared to Reinforcement Learning (RL) methods, as evidenced by results obtained from public RL test sets and practical tasks [15, 28]. By using ERL, the performance improvement can reach up to 20% [29, 30, 31]. Moreover, for classification tasks, the ERL method achieves the best accuracy scores across multiple benchmarks in the UCI online data repository [32, 33].

The strategies that ERL employed to perform better than other solution methods in numerous problems are closely connected. Due to the varied improvements made to the composition of ERL, these strategies can be categorized accordingly. The ensemble strategies for ERL are diverse and include strategies for the Q-function, reward, and loss function ensemble, as well as the combination of models, combination training algorithms, and decision strategies. In this section, we introduce these strategies separately.

3.1. Ensemble Strategies for Q-functions

In most Reinforcement Learning (RL) methods, the Q-function reflects how good the agent is in any given state [20]. A ”good state” here means that the agent can obtain a high expected return, which depends on the action taken by the agent. The classical Q-function formula, applicable to Ensemble Reinforcement Learning (ERL) methods, is provided in the background section. Besides, designing Q-functions specifically for the ERL method can further improve the search performance of the algorithm [15, 34, 35, 36].

A Maxmin Q-learning algorithm using multiple Q-functions was proposed by Lan et al. [37] to evaluate the performance. The maxmin mechanism integrates multiple predicted values as a reference for agent decision-making. Specifically, the prediction term in the original Q-value calculation formula is determined by the smallest of the multiple Q functions. The performance of this algorithm is demonstrated by using the Mountain Car environment. It can be seen from the results that the improvement of the Q-function has a positive effect on both the convergence performance and the search performance of the algorithm.

Bayesian optimization is also utilized to improve the Q-function. Chen et al. applied this idea in the design of the ERL method to update Q^* using Bayesian optimization. This method is mainly suitable for solving high-dimensional ERL problems, especially useful when the solution space is ultra-large [8]. In this study, an upper-confidence bounds (UCB) based solution space exploration strategy is used for the agent's action selection. In the experiment part, the Atari game is used to test the performance of the proposed method. And the experimental results verify the effectiveness of the proposed strategy.

Some Q-value approximation methods from related work of RL can also be used in ERL methods. Ghosh et al. used an ERL method based on a multi-agent framework to solve the air traffic control problem [38]. A kernel-based Q-value approximation via sample transitions is used to speed up the convergence [39].

3.2. Ensemble Strategies for Reward

Reward reflects the agent's performance in actions taken based on the state. A good decision generally corresponds to a high reward, while a problematic decision prompts the agent to find and correct the error through the reward. Based on this, Yao et al. designed an averaging reward calculation method for the ERL method, which allows the ERL method to take into account the relationship between exploration and exploitation [40]. Then, a soft actor-critic method is used to train Artificial Neural Network (ANN) models. This ERL method is well suited to solve the problem of exploring unknown regions.

The combination of reward functions in ERL can also be used with weight aggregation. Lin et al. proposed an adaptive adjustment method for reward function weights combining Upper Confidence Bounds (UCB) and error [41]. The weight update strategy allows the ERL method to evaluate the correctness of previous policies and improve generalizability. Qi et al. also used an ERL method with weighted reward aggregated functions to solve the traffic signal control problem [42].

Although the traditional calculation method of reward is widely used, it has the shortcoming of a complex process. Compared to traditional methods, fuzzy-based methods can reduce computational costs. A fuzzy set can affect the reward value obtained by agents by measuring dissimilarity. Pan et al. proposed a dissimilarity evaluation metric for deciding the weight value of each agent's reward in ERL [43]. In this way, ERL can achieve a good training effect with fewer iterations.

3.3. Ensemble Strategies for Loss Function

The loss function is an essential basis for Ensemble Reinforcement Learning (ERL) to update the network parameters and improve the performance of agent decisions. A smaller loss value indicates that the predicted value of the ERL model is closer to the actual value. However, gradient explosion and gradient disappearance are two fatal problems that often occur in the training process of Reinforcement Learning (RL) models. Some studies of ERL have attempted to improve the accuracy of RL model decisions by improving the loss function [15, 44, 45, 46]. Kumar et al. theoretically analyzed the bootstrapping error and proposed an error accumulation reduction method to enhance the stability of ensemble Q-learning algorithms [7].

Designing a global loss function for all models used is another approach specific to ERL. Adebola et al. proposed an improved global loss function with each member model included in the function [47]. Moreover, an interpolation method is used to control the difference between policies that the algorithm needs to train. This ERL method can also optimize the agent's policy selection using fine-tuning techniques. Based on this idea, Jiang et al. added the training data error between models to the overall loss calculation formula for improving prediction accuracy [19]. It can be seen from the experiment that this method is applicable in mobile edge computing (MEC) systems for rational resource scheduling.

In addition, considering uncertainty is an effective way to improve the loss function. Sun et al. used the uncertainty reduction technique to design an ensemble loss function [48], which effectively prevents the RL model from falling into a local optimum. Within this work, a distillation method is used to select the training data for the model. After that, the performance of the proposed ERL method is verified by the Atari game.

3.4. Combination of Models

The ensemble of different types of models is a common and simple strategy in Ensemble Reinforcement Learning (ERL). These models can be either Machine Learning (ML) models or Artificial Neural Network (ANN) models. The structure of the model combination is determined according to the specific problem and the solution target. A single type or a combination of multiple types of models are both popular. For using only one type of model, ANNs with different depths can be considered. While other studies use different random initialization strategies [16] or ANN in different training stages [49]. Table 1 provides a summary of related work ensemble model combination strategies. There are three models mainly in relevant works, including ML models only, ANN models only, and ML&ANN models hybrid. Such an ERL framework is easy to implement and achieves better performance than individual RL methods. A large-scale ensemble of numerous ML models and ANN models (more than three) implemented in the ERL framework has also been attempted by some researchers. Saadallah et al. [50], Li et al. [51], and Sharma et al. [52] are some examples in this regard.

ERL can be divided into parallel ERL and sequential ERL according to the relationship between base learners in ERL. Figure 6 and Figure 7 give schematic diagrams of these two

Table 1: Combination of models

Year	Author	Combination of Models
2019	Dong et al. [53]	long short-term memory network, gated recurrent unit network
2019	Goyal et al. [46]	convolution neural network, gated recursive unit
2020	Liu et al. [54]	long short-term memory network, deep belief network, echo state network
2020	Perepu et al. [55]	a linear regression model, long short-term memory model, artificial neural network, random forest
2021	Liu et al. [56]	graph convolutional network, long short-term memory networks, gated recursive unit
2021	Saadallah et al. [50]	autoregressive integrated moving average, exponential smoothing, gradient boosting machines, gaussian processes, support vector regression, random forest, projection pursuit regression, MARS, principal component regression, decision tree regression, partial least squares regression, multilayer perceptron, long short-term memory network (LSTM), Bi-LSTM: Bidirectional LSTM, CNN-based LSTM, convolutional LSTM
2021	Daniel L. Elliott and Charles Anderson [57]	convolution neural network, gated recursive unit, artificial neural network
2022	Shang et al. [30]	gated recursive unit, graph convolutional network, graph attention network
2022	Tan et al. [31]	graph attention network, long short-term memory networks, temporal convolutional network
2022	Li et al. [58]	gated recurrent unit, deep belief network, temporal convolutional network
2022	Zijie Cao and Hui Liu [59]	temporal convolutional network, Bidirectional long short-term memory network, kernel extreme learning machine
2022	Birman et al. [60]	machine learning models, artificial neural network
2022	Li et al. [51]	naive bayes, support vector machine with stochastic gradient descent, FastText, Bi-directional long short-term memory
2022	Sharma et al. [52]	support vector regressor (SVR), eXtreme gradient boosting (XGBoost), random Forest (RF), artificial neural network (ANN), long short-term memory (LSTM), convolution neural network (CNN), CNN-LSTM, CNN-XGB, CNN-SVR, and CNN-RF
2022	Shi Yin and Hui Liu [61]	group method of data handling, echo state network, extreme learning machine
2023	Yu et al. [29]	graph attention network, gated recursive unit, temporal convolutional network

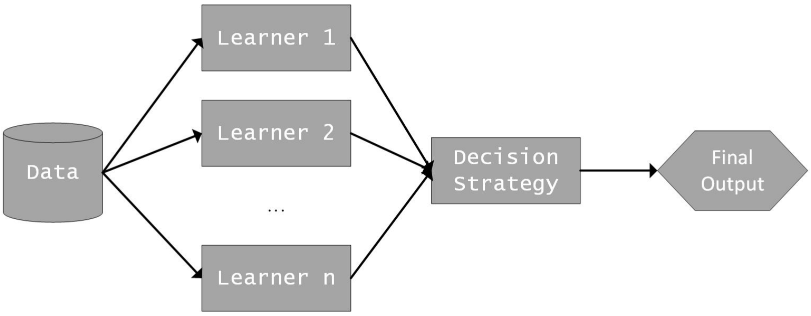


Figure 6: Parallel ensemble reinforcement learning

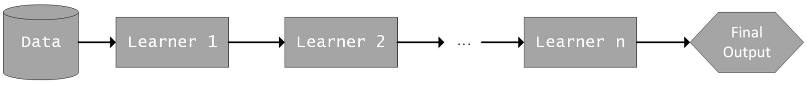


Figure 7: Sequential ensemble reinforcement learning

ERL methods. In most ERL studies, such as Liu et al. [33], Schubert et al. [62], and Shen et al. [63], base learners are constructed in a parallel framework. These ML or ANN models are responsible for the same task. After each model processing, the final prediction result will be generated by a certain strategy. There are also some studies, such as Qin et al. and Ferreira et al., that try to construct the ERL method in the sequential framework [64, 65]. In this framework, the base learner completes the final prediction step by step in a certain order [65].

3.5. Combination of Training Algorithms

Ensemble Reinforcement Learning (ERL) can not only use model combinations to obtain diverse prediction results but can also use different training algorithms to achieve full exploration of the solution space. Training algorithms can be classified into three categories: state-based, policy-based, and state-policy combination-based. Each of these training algorithms has its unique sampling strategy and output data characteristics. Researchers can quickly use the training algorithms according to the application scenarios without focusing on data sampling technology, which is similar to the EL method. Table 2 provides information about studies using the combination strategy of training algorithms. There exists a

Table 2: Combination of training algorithms

Year	Author	Combination of Training Algorithms
2008	Marco A. Wiering and Hado van Hasselt [66]	Q-learning, Sarsa, actor-critic, QV-learning, ACLA
2018	Chen et al. [67]	deep Q-networks, deep Sarsa networks, double deep Q-networks
2020	Yang et al. [14]	proximal policy optimization, advantage actor-critic, deep deterministic policy gradient
2020	Saphal et al. [68]	advantage actor-critic, sample efficient actor-critic with experience replay, actor-critic using Kronecker-factored trust region, deep deterministic policy gradient, soft actor-critic, trust region policy optimization
2021	Smit et al. [28]	double deep Q-Learning, soft actor-critic
2022	Eriksson et al. [69]	residual gradient, TD, TD(λ)
2022	Németh, Marcell and Szűcs, Gábor [70]	deep deterministic policy gradient, advantage actor-critic, proximal policy optimization

new method of combining online and offline training algorithms or using training algorithms based on different optimization strategies, which can take advantage of their respective strengths to handle complex tasks. Accordingly, the complexity of ERL methods using such improved strategies increases. For this reason, the training process of the ERL method takes more time. Moreover, the ensemble model obtained also requires the design of a decision strategy to select the prediction results that are closest to the actual situation.

Currently, the research related to the combination of training algorithms is not deep enough and simply combines multiple typical algorithms. However, there are some similarities between training algorithms. Transfer learning can be considered to transfer sampled data to reduce the training time. In addition, the termination conditions of multiple training algorithms also deserve in-depth analysis. Using the same number of iterations may result in some models finding the best policy long ago, while some models still need further training. Therefore, more research is required to investigate the combination of training algorithms in ERL.

3.6. Decision Strategies

In ERL, multiple base learners are implemented to obtain one result each, which may lead to differences between the multiple results. Therefore, the ERL algorithm needs to adopt certain decision strategies to determine the final model and output. In existing ERL-based research, the common decision strategies include voting, optimal combination, binning, aggregation, weighted aggregation, stacking, and Boltzmann multiplication (see Figure 8).

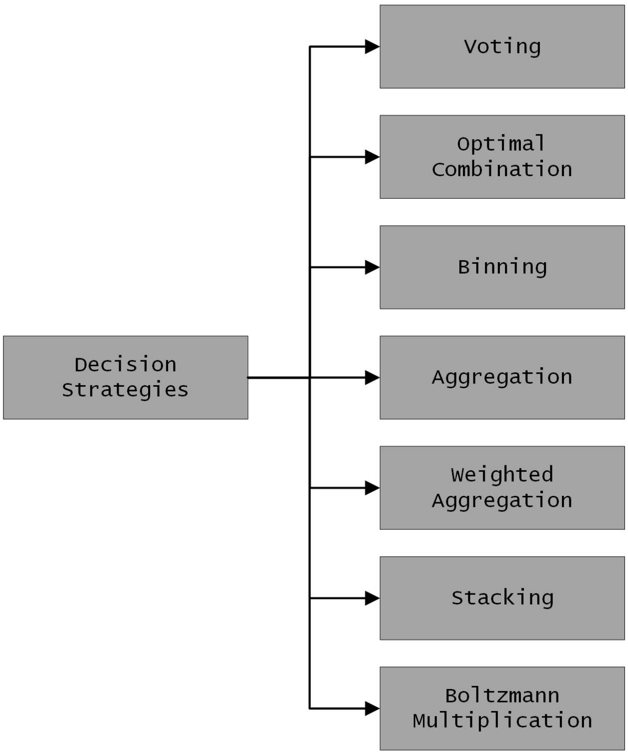


Figure 8: Decision strategies

Voting: Voting, as a common ERL decision strategy, records the number of occurrences of each prediction at first [71]. Then, the final prediction can be selected from the results according to the principle of majority or ranking.

Optimal Combination: For classification problems, this is a commonly used decision strategy [72]. Multiple base classifiers are trained separately, from which the optimal subset of models is selected to form an ensemble to classify the test set.

Binning: Binning is a majority voting decision strategy with continuous action space [68]. First, the action space is discretized into multiple intervals. After that, the number of occurrences of the actions in each interval is recorded. Finally, the average value of the action within the interval with the highest number of occurrences is selected as the final prediction result.

Aggregation: The prediction results of all the models in ERL are summed to produce an overall evaluation value, which is taken as the final result [51]. In the aggregation method, each model is considered to be equally reliable.

Weighted Aggregation: The prediction results obtained from different models are summed according to their weights [55]. A high value of weight is used for aggregation models with high prediction accuracy.

Stacking: First, an additional machine learning model is involved to further predict the results of base learners. Then, the output of this machine learning model is used as the final prediction result [60].

Boltzmann Multiplication: Boltzmann distribution is the basis for decision making [66]. The probability value of each action can be calculated according to the Boltzmann distribution. The outcome with the highest probability value is selected and will not be changed.

In summary, the selection of decision strategies in ERL depends on the specific application scenario and the characteristics of the base learners. At present, there is no related study on the in-depth analysis of the application scenarios of these strategies. This study is helpful to improve the prediction accuracy of ERL methods. It will also promote the process of ERL research.

3.7. Discussions

Strategies play a crucial role in designing ERL methods. Among the strategies for improving the ERL element, the model combination is the easiest to implement. The model combination allows the ERL method to have multiple machine learning models or ANN search strategies simultaneously, which is sufficient for some practical application problems. The training algorithm combination strategy is slightly more complicated than the model combination. If researchers use this strategy, they need to understand the conditions of use of each training algorithm and design the combination of algorithm and model accordingly. When ERL models are used in a multi-agent or distributed framework, some new framework-related issues arise, such as how base learners interact with the environment and whether training is done independently or jointly.

Other strategies, such as improved Q-functions, reward, and loss functions are considered for the generalization of ERL methods. This requires researchers to propose new methods, which can be applied to various types of practical application problems. What's more, the decision strategy is another aspect that can improve the performance of ERL. The outputs from multiple base learners may be similar or significantly different. These outputs can be adopted in their entirety, or only one or more of them can be selected as the result. There are few studies related to the combined use of decision strategies, and using this method allows the ERL to search the solution space comprehensively. The design of which strategy to use under which conditions becomes an important factor for the strategy combination to work. The idea of integration can also be used in decision strategies, where a new model evaluates the performance of the results by using various decision strategies.

4. Applications of Ensemble Reinforcement Learning

A significant portion of existing ERL-based research involves discrete/continuous control actions [41, 73, 74, 75] and game environments [76, 77, 78] to verify the effectiveness of proposed algorithms. Additionally, researchers have attempted to utilize ERL methods to solve practical application problems. Figure 9 illustrates the main application areas of ERL, which encompass energy and environment, IoT and cloud computing, finance, and other areas. Among these, energy and environment is the most widely studied application area of ERL. In this section, we discuss the application of ERL in various domains.

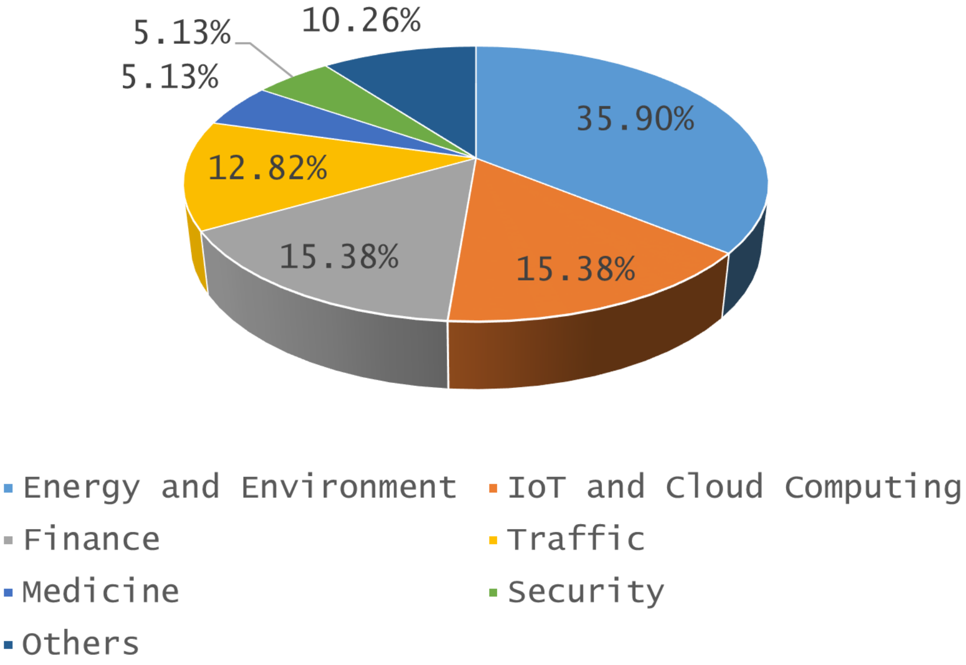


Figure 9: The summary of the different application areas of ERL

4.1. Energy and Environment Area

As the global economy continues to grow, energy and environmental issues have garnered increasing attention worldwide. The use of neural network methods to predict future conditions using historical data has become the basis for policy formulation. In this type of prediction problem, there exists a spatiotemporal relationship between data, which makes the recurrent neural network (RNN) preferred. In these related studies, ERL methods combine multiple classical RNN models to obtain more reliable conclusions. Table 3 shows some recent applications of ERL in the energy and environment domain. It can be seen that wind power and PM 2.5 prediction are hot research topics. From a statistical point of view, sixty-four percent of ERL-based studies used Q-learning as the training algorithm, while the remaining studies used Sarsa, deep Q-network, and deep deterministic policy gradient algorithms. Overall, the application of ERL methods in the energy and environment area focuses on ensemble models. In these studies, reinforcement learning algorithms are mainly used directly. Compared with traditional studies using machine learning (ML) or artificial neural network (ANN) prediction methods [79, 80], there is a significant gap between the prediction results obtained by these methods and ERL.

Table 3: Application in energy and environment area

Year	Authors	Problem	Training Algorithm
2020	Liu et al. [54]	wind speed short term forecasting	Q-learning
2021	Jalali et al. [81]	solar irradiance forecasting	Q-learning
2021	Liu et al. [56]	PM2.5 forecasting	Q-learning
2021	Li et al. [82]	PM2.5 forecasting	Q-learning
2021	Chao Chen and Hui Liu [83]	wind speed prediction	deep Q-Network
2021	Jalali et al. [84]	wind power forecasting	Q-learning
2022	Tan et al. [31]	PM2.5 prediction	Sarsa
2022	Qin et al. [85]	unit commitment problem	deep Q-Network
2022	Sogabe et al. [86]	smart energy optimization and risk evaluation	Q-learning
2022	Sharma et al. [52]	estimating reference evapotranspiration	Q-learning
2022	He et al. [87]	wind farm control	deep deterministic policy gradient
2022	Jalali et al. [88]	solar irradiance forecasting	Q-learning
2022	Shi Yin and Hui Liu [61]	wind power prediction	Q-learning
2023	Yu et al. [29]	wind power prediction	deep deterministic policy gradient

4.2. Internet of Things and Cloud Computing Area

In the area of the Internet of Things (IoT) and cloud computing, ERL is widely used to optimize system performance and business processing capabilities. The IoT connects various devices such as sensors, smart terminals, and industrial systems to form a globally interconnected system. Optimizing the efficiency of these devices and facilities has a positive impact on improving the overall performance of IoT systems. Cloud computing is another technology closely related to the IoT. Users can access computing resources or services in this distributed system provided by a cloud platform over the network on demand. Resource allocation and optimization have been the focus of research in the IoT and cloud computing area. Table 4 presents the applications of ERL methods in this area. Here, most studies use the offline algorithm, except for Polyzos et al. [89], who used an online algorithm. The performance of the ERL method has been verified on simulation platforms [90]. It can be seen from experimental results that the use of ensemble models makes the ERL method schedule significantly better than compared RL methods. When applying ERL in this area, matching the application requirements of multi-agent and distributed architecture becomes a core point. This system architecture allows the ensemble models in ERL to handle the same or different tasks.

Table 4: Application in IoT and cloud computing area

Year	Authors	Problem	Training Algorithm
2020	Ashiquzzaman et al. [91]	IoT sensor calibration	deep Q-Network
2021	Polyzos et al. [89]	resource allocation	Sarsa
2021	Jiang et al. [19]	large-scale MEC systems	deep Q-Network
2021	Gu et al. [92]	online cloud task scheduler	deep deterministic policy gradient
2021	Liu et al. [93]	deep reinforcement learning training on GPU cloud platform	actor-critic network
2022	Mahmud et al. [94]	non orthogonal multiple access unmanned aerial network	deep Q-Network

4.3. Financial Area

In the financial area, complex decision-making problems, such as pricing financial products and portfolio optimization, are being tried to be solved by ERL methods. Though single models can make predictions on a specific problem, their generalization is affected by the problem scenario. Compared with the single model, ensemble models are affected less by the problem scenario factors. Table 5 presents the applications of ERL methods in finance. In these studies, 67% used only one training algorithm, while the rest of the studies used multiple training algorithms in an ERL method. Three algorithms, namely proximal policy optimization, advantage actor-critic, and deep deterministic policy gradient, have shown good performance in training.

Table 5: Application in financial area

Year	Authors	Problem	Training Algorithm
2020	Yang et al. [14]	stock trading	proximal policy optimization, advantage actor-critic, deep deterministic policy gradient
2020	Xu et al. [95]	fuel economy improvement	Q-learning
2021	Carta et al. [49]	stock market forecasting	deep Q-Network
2022	Li et al. [58]	regional GDP prediction	deep Q-Network
2022	Zijie Cao and Hui Liu [59]	carbon price forecasting	Q-learning
2022	Németh, Marcell and Szűcs, Gábor [70]	algorithmic trading	proximal policy optimization, advantage actor-critic, deep deterministic policy gradient

4.4. Other Areas

Apart from the previous three classic application areas, Ensemble Reinforcement Learning (ERL) has also been successfully applied in other areas such as transportation, medicine, and security, which will be discussed in this section. Table 6 provides an overview of these applications. ERL methods primarily focus on making predictions. Only a few classification problems such as diagnosis and recognition are solved using ensemble ideas. The work of Eriksson et al. [69], who used ERL methods on autonomous driving problems, is particularly noteworthy. If the ERL method can be successfully applied to the automatic assisted driving of small cars, it is likely to trigger a new research and application boom in this area.

Table 6: Application in other areas

Year	Authors	Problem	Area
2021	Ghosh et al. [38]	air traffic control	traffic
2021	Dong et al. [53]	traffic speed forecasting	traffic
2022	Shang et al. [30]	traffic volume forecasting	traffic
2022	Qi et al. [42]	traffic signal control	traffic
2022	Eriksson et al. [69]	autonomous driving	traffic
2016	Tang et al. [96]	symptom checker	medicine
2021	Jalali et al. [72]	COVID-19 diagnosis	medicine
2022	Birman et al. [60]	malware detection	security
2022	Li et al. [51]	rumor tracking	security
2023	Henna et al. [97]	FSO/RF communication systems	optics
2019	Cuayáhuitl et al. [98]	chatbots	dialogue system
2010	Alexander Hans and Steffen Udluft [34]	pole balancing	engineering control
2018	Ferreira et al. [65]	cognitive satellite communication	aerospace

In the future, we expect to see more areas using ERL methods for complex tasks. Existing research results can provide valuable references for subsequent research, including improving existing algorithms to overcome their limitations, or extending the problem domain to obtain new insights. In the next section, we discuss some potential directions for future research on ERL methods.

5. Datasets and Compared Methods

This section examines the datasets and comparison methods used in various studies related to Ensemble Reinforcement Learning (ERL). As presented in Table 7, experiments are conducted to evaluate the performance of the proposed ERL methods. The datasets used in these experiments mainly include real-world data and publicly available datasets or environments. Real-world data are useful for objectively testing the predictive or classification performance of the method for specific applications. For instance, studies in the field of energy and environment have collected data from multiple cities to predict desired outcomes [29, 54]. In contrast, publicly available datasets or environments such as the OpenAI Gym environment in the field of reinforcement learning are widely used to test the predictive performance of algorithms for continuous/discrete actions [103]. The most commonly used public dataset for classification problems is the UCI machine learning repository [33]. Furthermore, some medical-specific datasets are also utilized in studies of disease diagnosis [72].

To evaluate the effectiveness of the proposed ERL methods, various comparison methods are used in the literature. The single model-based RL method (SM-RL) is one of the simplest ways to reflect the effectiveness of the proposed ERL method [62]. The training algorithm used in SM-RL remains the same as that in the ERL method. However, this compared method has limited convincing power. Therefore, some other studies have used other training algorithms to compare with the proposed algorithm from another perspective [15, 38]. To comprehensively test the effectiveness of algorithms, different models, training algorithms, and integration methods should be separately evaluated [54].

Table 7: Datasets and compared Methods

Year	Authors	Dataset	Compared Methods
2016	Osband et al. [99]	Atari games	DQN
2017	Chen et al. [8]	Atari games	A3C+
2017	Partalas et al. [100]	UCI machine learning repository	classifier combination methods voting (V) and SMT and the forward selection (FS), selective fusion (SF)
2018	Pearce et al. [101]	Cart Pole control problem	Q-learning with different layer NNs
2019	Dong et al. [53]	traffic speed dataset	GRU, LSTM, MLP, RBF, LSTM-GRU-GA
2019	Pan et al. [43]	Maze, Mountain Car, Robotic Soccer Game Simulation	counterpart
2019	Goyal et al. [46]	CATS (Competition on Artificial Time series) dataset	LSTM, ANN, Linear regression, Random Forest, Online NN
2019	Macheng Shen and Jonathan P How [102]	two-player asymmetric game	single model, RNN
2020	Qingfeng Lan et al. [37]	Mountain Car	Q-learning, Double Q-learning, Averaged Q-learning
2020	Liu et al. [54]	three different groups of measured wind speed data from Xinjiang wind farms	Network: LSTM method, the DBN method, the ESN method; Training algorithm: SARSA
2020	Lin et al. [41]	Maze, soccer robot game	orthogonal projection inverse reinforcement learning method (OP-IRL)
2020	Junta Wu and Huiyun Li [73]	2D Robot Arm Open Racing Car Simulator (TORCS)	DDPG
2020	Yang et al. [14]	Dow Jones 30 constituent stocks (at 01/01/2016)	PPO, A2C, DDPG
2020	Liu et al. [33]	UCI online data repository	classifiers combination approaches majority voting (MV), weighted voting (WV), ensemble selection methods forward selection (FS)
2021	Ghosh et al. [38]	open source air traffic simulator	PPO
2021	Jalali et al. [81]	GHI data sets	adaptive hybrid model (AHM), hybrid feature selection method (HFS), Outlier-robust hybrid model (ORHM), novel hybrid deep neural network model (NHDNNM), OHS-LSTM
2021	Liu et al. [56]	data collected from a congested intersection in Changsha	RNN, ENN, ESN, DBN, RBF, GRNN, MLP
2021	Jalali et al. [72]	two well-known open-source image datasets named as Mendely and Kaggle	original version of GSK and eight powerful evolutionary algorithms including grasshopper optimization algorithm (GOA), Slime mold algorithm (SMA), genetic algorithm, gray wolf optimizer (GWO), particle swarm optimization (PSO), differential evolution (DE), biogeography-based optimization (BBO)
2022	Hassam Ullah Sheikh et al. [15]	Mujoco environments, Atari games	TD3, SAC and REDQ
2022	Shang et al. [30]	actual traffic volume data of nine stations of Changsha freeway	Chebnet, CNN, LSTM, DBN, RNN, ESN, multi-layer perceptron (MLP)
2022	Tan et al. [31]	actual data	RNN, the deep belief network (DBN), the echo state network (ESN), the error encoding network (ENN), General Regression Neural Network (GRNN), radial basis function network (RBF), multilayer perceptron (MLP)
2022	Li et al. [58]	three sets of data from three Provinces of China	ESN, ENN, RNN, BPNN, ELM, RBF
2022	Cao et al. [59]	The data for the three carbon trading markets come from the Hubei Carbon Trading Network, Beijing Carbon Emissions Electronic Trading Platform, and International Carbon Action Partnership (ICAP)	Network: TCN, BiLSTM, KELM, BPNN, MLP, echo state network (ESN), Elman neural network (ENN), and gradient boosting decision tree (GBDT); Training algorithm: SARSA
2022	Qin et al. [85]	historical load data of the California Independent System Operator (CASIO) from January 1, 2021 to July 5, 2021	PPO guided tree search, the MIQP algorithm with Gurobi 9.1
2022	Sogabe et al. [86]	optimal energy management in a residential building microgrid	mixed-integer linear programming (MILP)
2022	Birman et al. [60]	a range of real-world scenarios	Aggregation method
2022	Li et al. [51]	PHEME, RumorEval	Naive Bayes, SVM-SGD, Dense, BiLSTM, FastText, TextCNN, VRoC, some combinations of above methods
2022	Sharma et al. [52]	two MEC servers and 30 IoTDs randomly distributed in the squared area with size 50m×50m	Actor-Critic, DDPG
2022	Schubert et al. [62]	SymCat's symptom-disease database	single model-based RL
2023	Yu et al. [29]	actual wind power data of nine wind turbines	GMDH, DBN, ESN, ENN, the extreme learning machine (ELM), the radial basis function (RBF), multi-layer perceptron

6. Open Questions and Future Research Directions

6.1. Open Questions

In this section, we highlight three open questions in the field of Ensemble Reinforcement Learning (ERL) that can contribute to the future development of ERL (see Figure 10).

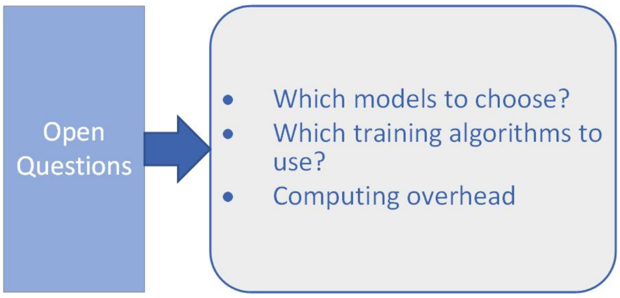


Figure 10: Open Questions

6.1.1. Which Models to Choose?

Models are the basis for constructing ERL methods and have a direct impact on the final output. The most critical aspect of model selection is the ability to feature selection and learning. If a model cannot learn valid information, it is meaningless. Compared to methods using a single model, ensemble methods that use multiple same or different models can reduce the possibility of incorrect inference and improve the overall predictive performance. Accordingly, the models implemented should be beneficial to the overall predictive performance ERL method.

The models implemented in ERL mainly include ML and ANN models, which have a simple structure and strong generalization ability and are suitable for some practical applications. If the data scale is large and there are many features, ML models may be difficult to handle. In this case, it is easier for the ANN model to get features from the dataset and get the prediction results close to the actual value. However, some tasks that need to be understood are difficult for ANN models. Some studies such as [50], [60] have considered implementing ML models and ANN models together into ERL methods.

Using models with different training stages simultaneously is much easier for ERL design [49]. There are differences in the parameters within these models, and the predicted bias and variance are also distinctive. If such an ensemble model is used, it is worthwhile to deeply investigate what conditions are chosen to save each ensemble element in the training process.

Some studies have used proposed ensemble pruning methods like forward selection (FS) [104], and selective fusion (SF) [105] to automate the optimization of model structures. In these studies, a model library is used to select models, which can reduce the human selection effort to a certain extent. The model library can also be continuously extended based on the latest research to optimize the overall performance of the ERL.

6.1.2. Which Training Algorithms to Use?

Model training is a well-known challenge in the ERL method. For model-free based ERL, the agent updates model parameters based on state transfer or trajectory after a certain number of iterations. But there is no guarantee that all the data obtained by sampling are useful. The classic strategy is to use experience delay technology, which can improve the sampling efficiency. While Schaul et al. [106] found that the sampling inefficiency problem occurs when the sample in the relay buffer is useless. Selected experience relay makes the algorithm training more efficient by selectively choosing the adopted data into the replay buffer [107]. It is worth noting that the experience relay buffer is only applicable to the off-policy RL method. Besides, the model-based RL method can guarantee sampling efficiency by learning the environment model. However, such a training algorithm has to face a huge action space. In this case, approximating the environment model becomes an extremely difficult task.

Good training algorithms should balance exploration and exploitation, whereas, it is difficult to achieve by using only one. Therefore, using multiple types of training algorithms simultaneously should be the general trend for future development. This ERL method requires designing the sampling technique of the solution space according to the training strategy separately. How to use sampled data from the same type of strategy for model training processes is also worth considering. In addition, such algorithm training will have high requirements on the machine's CPU and GPU computing ability.

6.1.3. Computing Overhead

Computing overhead is another issue that is closely related to the first two problems and has to be considered for ERL. Compared to a single model, multiple ML or ANN models implemented in ERL make the number of parameters ultra-large. Especially when each ANN model has a complex structure, memory consumption becomes a factor that cannot be ignored. Similarly, multiple training algorithms can complicate the training process. Even when using techniques related to computational acceleration, the time consumed by a large number of computations can be significantly longer than that of the individual training. Many studies have found that ERL methods can complete sampling efficiently, but are also accompanied by an increase in computation time [15, 89]. In the testing stage, complex decisions then easily lead to longer computation time than other methods [100]. So, some researchers tried to design strategies based on scenarios, which reduce the computational overhead to some extent. An et al. achieved a reduction in model training time along with a reduction in memory consumption by taking uncertainty into account in the ERL method [36]. Pan et al. reduced the time consumed by the algorithm for each iteration of training by fuzzifying the reward [43]. Up to now, the number of computationally cost-reducing models is still small. So, it is difficult to show that the improvement strategy is still applicable to large-scale models. In addition, the cost of data interaction needs extra attention when the models are deployed on multiple machines, which will affect the efficiency of the system.

The cost-effectiveness of ERL using complex structures and training processes is a fundamental basis for measuring method design and algorithm training. Increasing the number of models can improve the ensemble prediction performance but is also accompanied by an increase in computing overhead. After a certain number of models are implemented, the computing overhead of using more models can be significantly greater than the improvement in method performance. In such cases, increasing the size of the ensemble model is not advisable.

In some practical application problems, feasible solutions obtained by ERL methods can serve as problem-solving results. Controlling the number of iterations of the training process is an alternative if the computing overhead of searching for the optimal strategy is much greater than the contribution it can make. Thus, addressing the issue of computing overhead is essential for the successful application of ERL in various fields.

6.2. Future Research Directions

There has been a lot of valuable research that uses ERL well to solve problems in science and multiple application areas. Based on the analysis of relevant literature mentioned in the survey, it can be seen that most of the research is concentrated within the last decade. There are still many directions waiting to be explored by researchers. Here, some potential research directions are given.

1. **Randomized models**: Randomized models, such as random vector functional link networks [108], random initialized implicit layers [109] are an effective strategy to reduce training. In addition, implicit/explicit ensembles [110] can improve the model training efficiency from the perspective of diversity. Ensuring diversity among base learners is a core problem that needs to be solved in the ERL method and deserves further study.

2. **Effect of decision strategy**: decision strategy is used to obtain the final output based on the prediction results of individual base learners. Although existing studies have tried to use various types of decision strategies, there is a lack of systematic research on the effect of strategies. Accordingly, the decision strategies applicable in the case of different integrated models and the number of training algorithms are worth analyzing in detail.

3. **Hierarchical ensemble**: Hierarchical reinforcement learning methods can be used to solve some challenging problems. For example, Qin et al. and Ferreira et al. respectively tried to use multiple RL models to complete different tasks separately in order [31, 64, 65]. The current model structure is designed from experience and lacks systematic theoretical validation. The performance of the single RL model and the ensemble model should be weighed in hierarchical ensemble reinforcement learning. The role of each element in the hierarchical framework also deserves a detailed design according to specific problems.

4. **Large-scale ensemble**: The number of base learners used in existing ERL methods is generally about three [55, 57]. From the diversity perspective, a larger number of models constituting a new ERL method can explore a large amount of feature information and make accurate predictions. From the statistical point of view, the increase of ensemble components can make more hypotheses and increase the chances of finding the optimal hypothesis. If a large-scale ERL is used, the information-sharing mechanism between base learners can be designed, which reduces the total training cost of the model.

5. **Distributed approach**: Ensemble reinforcement learning can also be trained or used in a distributed manner. Existing distributed ERL-based research only uses ERL in a distributed framework and lacks methodological improvements [19, 69]. Therefore, how to take advantage of both ERL and distributed reinforcement learning deserves further analysis. The implementation of ERL into a distributed framework will inevitably lead to increased model training and communication costs. In a distributed framework, it is necessary to focus on low-cost training methods and controlled training time to ensure that ERL is applied to practical scenarios.

6. **Online model training**: Currently, ERL adopts offline training and direct online use. This model training algorithm will make it difficult for the model to grasp the latest situation. Accordingly, the strategy adopted by the agent is not optimal. If online or near-online model training methods can be used, new information will be added to the training

dataset in time to ensure that the model can respond to the latest situation. Online training needs to focus on the triggering mechanism of model training, over-training or under-training can have a negative impact on the performance of the ERL method. Moreover, forgetting history memory can also help ERL find new optimal strategies.

7. **Efficient training**: The sampling efficiency of DRL also deserves attention, as this problem is still prevalent in ERL methods. Therefore, this raises many related problems, such as how to split the data set for training, initialize the model parameters, set the hyperparameters, and update the strategy. Models belonging to different training stages can also be used together to find the optimal combination of model configurations [49].

8. **Embedded into big data platform**: Most of the current ERL-related studies are based on simulation environments, which is still a certain gap from practical application. For some practical forecasting tasks, integrating ERL methods into a big data platform can make timely inferences based on the data obtained from the system. For short-term forecasting and long-term forecasting, diverse ERL methods can be deployed in the big data platform according to the forecasting objectives.

Future research can be carried out in the above but not limited to these aspects. Certainly, several new situations will also be encountered when designing methods and solving problems. It should be understood that no free lunch theorem applies to any ERL method [20]. So, the complexity and the training time need to be taken into account in the design process of ERL methods.

7. Conclusion

This paper has provided a comprehensive review of the research progress on ensemble reinforcement learning (ERL) methods from the background, strategies, applications, and other aspects. First, the description of reinforcement learning methods and ensemble learning methods has enhanced the understanding of ERL. Then, various strategies, such as

Q-function ensemble, model combination, and decision strategies, are introduced and discussed. After that, the application of ERL methods, datasets, and compared methods are described. Moreover, we have discussed future research directions that can further improve ERL's performance.

ERL's powerful predictive or classification capability makes it a promising framework for solving complex problems. ERL has been successfully applied in various fields, including finance, robotics, and healthcare. However, there is still considerable potential for future research. The potential research directions highlighted in this paper include randomized models, the effect of decision strategy, hierarchical ensemble, large-scale ensemble, distributed approach, online model training, efficient training, and embedding ERL into big data platforms.

We believe that ERL can achieve satisfactory performance in more application areas in the future. Therefore, researchers need to continue exploring and developing new ERL methods to address the challenges encountered in practical applications.

Acknowledgement. This work is supported by the National Natural Science Foundation of China (72201273,72001212), the Science and Technology Innovation Team of Shanxi Province (2023-CX-TD-07), the Special Project in Major Fields of Guangdong Universities (2021ZDZX1019), and the Hunan Key Laboratory of Intelligent Decision-making Technology for Emergency Management (2020TP1013).

References

[1] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, et al., Human-level control through deep reinforcement learning, nature 518 (7540) (2015) 529–533.

[2] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, et al., Mastering the game of go with deep neural networks and tree search, nature 529 (7587) (2016) 484–489.

[3] O. Vinyals, I. Babuschkin, W. M. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, D. H. Choi, R. Powell, T. Ewalds, P. Georgiev, et al., Grandmaster level in starcraft ii using multi-agent reinforcement learning, Nature 575 (7782) (2019) 350–354.

[4] Ł. Kaiser, M. Babaeizadeh, P. Miłos, B. Osiński, R. H. Campbell, K. Czechowski, D. Erhan, C. Finn, P. Kozakowski, S. Levine, et al., Model based reinforcement learning for atari, in: International Conference on Learning Representations, 2020.

[5] R. Liu, F. Nageotte, P. Zanne, M. de Mathelin, B. Dresp-Langley, Deep reinforcement learning for the control of robotic manipulation: a focussed mini-review, Robotics 10 (1) (2021) 22.

[6] S. Fujimoto, H. Hoof, D. Meger, Addressing function approximation error in actor-critic methods, in: International conference on machine learning, PMLR, 2018, pp. 1587–1596.

[7] A. Kumar, J. Fu, M. Soh, G. Tucker, S. Levine, Stabilizing off-policy q-learning via bootstrapping error reduction, Advances in Neural Information Processing Systems 32 (2019).

[8] R. Y. Chen, S. Sidor, P. Abbeel, J. Schulman, Ucb exploration via q-ensembles, arXiv preprint arXiv:1706.01502 (2017).

[9] M. d. Condorcet, Essay on the application of analysis to the probability of majority decisions, Paris: Imprimerie Royale (1785).

[10] A. Krogh, J. Vedelsby, Neural network ensembles, cross validation, and active learning, Advances in neural information processing systems 7 (1995) 231–238.

[11] L. Breiman, Random forests, Machine learning 45 (2001) 5–32.

[12] G. Brown, J. Wyatt, R. Harris, X. Yao, Diversity creation methods: a survey and categorisation, Information fusion 6 (1) (2005) 5–20.

[13] T. G. Dietterich, Ensemble methods in machine learning, in: Multiple Classifier Systems: First International Workshop, MCS 2000 Cagliari, Italy, June 21–23, 2000 Proceedings 1, Springer, 2000, pp. 1–15.

[14] H. Yang, X.-Y. Liu, S. Zhong, A. Walid, Deep reinforcement learning for automated stock trading: An ensemble strategy, in: Proceedings of the first ACM international conference on AI in finance, 2020, pp. 1–8.

[15] H. Sheikh, M. Phielipp, L. Boloni, Maximizing ensemble diversity in deep reinforcement learning, in: International Conference on Learning Representations, 2022.

[16] X. Chen, C. Wang, Z. Zhou, K. W. Ross, Randomized ensembled double q-learning: Learning fast without a model, in: International Conference on Learning Representations, 2021.

[17] S. Faußer, F. Schwenker, Ensemble methods for reinforcement learning with function approximation, in: Multiple Classifier Systems: 10th International Workshop, MCS 2011, Naples, Italy, June 15-17, 2011. Proceedings 10, Springer, 2011, pp. 56–65.

[18] O. Anschel, N. Baram, N. Shimkin, Averaged-dqn: Variance reduction and stabilization for deep reinforcement learning, in: International conference on machine learning, PMLR, 2017, pp. 176–185.

[19] F. Jiang, L. Dong, K. Wang, K. Yang, C. Pan, Distributed resource scheduling for large-scale mec systems: A multiagent ensemble deep reinforcement learning with imitation acceleration, IEEE Internet of Things Journal 9 (9) (2021) 6597–6610.

[20] R. S. Sutton, A. G. Barto, Reinforcement learning: An introduction, MIT press, 2018.

[21] C. J. Watkins, P. Dayan, Q-learning, Machine learning 8 (1992) 279–292.

[22] K. Arulkumaran, M. P. Deisenroth, M. Brundage, A. A. Bharath, Deep reinforcement learning: A brief survey, IEEE Signal Processing Magazine 34 (6) (2017) 26–38.

[23] L. Breiman, Bagging predictors, Machine learning 24 (1996) 123–140.

[24] R. E. Schapire, The boosting approach to machine learning: An overview, Nonlinear estimation and classification (2003) 149–171.

[25] D. H. Wolpert, Stacked generalization, Neural networks 5 (2) (1992) 241–259.

[26] G. Brown, J. L. Wyatt, P. Tino, Y. Bengio, Managing diversity in regression ensembles., Journal of machine learning research 6 (9) (2005).

[27] S. Geman, E. Bienenstock, R. Doursat, Neural networks and the bias/variance dilemma, Neural computation 4 (1) (1992) 1–58.

[28] J. Smit, C. T. Ponnambalam, M. T. Spaan, F. A. Oliehoek, Pebl: Pessimistic ensembles for offline deep reinforcement learning, in: Robust and Reliable Autonomy in the Wild Workshop at the 30th International Joint Conference of Artificial Intelligence, 2021.

[29] Y. Chengqing, Y. Guangxi, Y. Chengming, Z. Yu, M. Xiwei, A multi-factor driven spatiotemporal wind power prediction model based on ensemble deep graph attention reinforcement learning networks, Energy 263 (2023) 126034.

[30] P. Shang, X. Liu, C. Yu, G. Yan, Q. Xiang, X. Mi, A new ensemble deep graph reinforcement learning network for spatio-temporal traffic volume forecasting in a freeway network, Digital Signal Processing 123 (2022) 103419.

[31] J. Tan, H. Liu, Y. Li, S. Yin, C. Yu, A new ensemble spatio-temporal pm2.5 prediction method based on graph attention recursive networks and reinforcement learning, Chaos, Solitons & Fractals 162

(2022) 112405.

[32] I. Partalas, G. Tsoumakas, I. Katakis, I. Vlahavas, Ensemble pruning using reinforcement learning, in: Advances in Artificial Intelligence: 4th Helenic Conference on AI, SETN 2006, Heraklion, Crete, Greece, May 18-20, 2006. Proceedings 4, Springer, 2006, pp. 301–310.

[33] Z. Liu, K. Ramamohanarao, Instance-based ensemble selection using deep reinforcement learning, in: 2020 International Joint Conference on Neural Networks (IJCNN), IEEE, 2020, pp. 1–7.

[34] A. Hans, S. Udluft, Ensembles of neural networks for robust reinforcement learning, in: 2010 Ninth International Conference on Machine Learning and Applications, IEEE, 2010, pp. 401–406.

[35] Q. He, H. Su, C. Gong, X. Hou, Mepg: A minimalist ensemble policy gradient framework for deep reinforcement learning, arXiv preprint arXiv:2109.10552 (2021).

[36] G. An, S. Moon, J.-H. Kim, H. O. Song, Uncertainty-based offline reinforcement learning with diversified q-ensemble, Advances in neural information processing systems 34 (2021) 7436–7447.

[37] Q. Lan, Y. Pan, A. Fyshe, M. White, Maxmin q-learning: Controlling the estimation bias of q-learning, arXiv preprint arXiv:2002.06487 (2020).

[38] S. Ghosh, S. Laguna, S. H. Lim, L. Wynter, H. Poonawala, A deep ensemble method for multi-agent reinforcement learning: A case study on air traffic control, in: Proceedings of the International Conference on Automated Planning and Scheduling, Vol. 31, 2021, pp. 468–476.

[39] D. Ormoneit, A. Sen, Kernel-based reinforcement learning, Machine learning 49 (2-3) (2002) 161.

[40] Y. Yao, L. Xiao, Z. An, W. Zhang, D. Luo, Sample efficient reinforcement learning via model-ensemble exploration and exploitation, in: 2021 IEEE International Conference on Robotics and Automation (ICRA), IEEE, 2021, pp. 4202–4208.

[41] J.-L. Lin, K.-S. Hwang, H. Shi, W. Pan, An ensemble method for inverse reinforcement learning, Information Sciences 512 (2020) 518–532.

[42] R. Qi, J. Huang, H. Li, Q. Tan, L. Huang, J. Cui, Random ensemble reinforcement learning for traffic signal control, arXiv preprint arXiv:2203.05961 (2022).

[43] W. Pan, R. Qu, K.-S. Hwang, H.-S. Lin, An ensemble fuzzy approach for inverse reinforcement learning, International Journal of Fuzzy Systems 21 (2019) 95–103.

[44] S. Lee, Y. Seo, K. Lee, P. Abbeel, J. Shin, Offline-to-online reinforcement learning via balanced replay and pessimistic q-ensemble, in: Conference on Robot Learning, PMLR, 2022, pp. 1702–1712.

[45] Z. Yang, K. Ren, X. Luo, M. Liu, W. Liu, J. Bian, W. Zhang, D. Li, Towards applicable reinforcement learning: Improving the generalization and sample efficiency with policy ensemble, arXiv preprint

arXiv:2205.09284 (2022).

[46] A. Goyal, S. Sodhani, J. Binas, X. B. Peng, S. Levine, Y. Bengio, Reinforcement learning with competitive ensembles of information-constrained primitives, arXiv preprint arXiv:1906.10667 (2019).

[47] S. Adebola, S. Sharma, K. Shivakumar, Deft: Diverse ensembles for fast transfer in reinforcement learning, arXiv preprint arXiv:2209.12412 (2022).

[48] Y. Sun, P. Fazli, Ensemble policy distillation in deep reinforcement learning, in: Workshop on Reinforcement Learning in Games, 2020, pp. 1–9.

[49] S. Carta, A. Ferreira, A. S. Podda, D. R. Recupero, A. Sanna, Multi-dqn: An ensemble of deep q-learning agents for stock market forecasting, Expert systems with applications 164 (2021) 113820.

[50] A. Saadallah, K. Morik, Online ensemble aggregation using deep reinforcement learning for time series forecasting, in: 2021 IEEE 8th International Conference on Data Science and Advanced Analytics (DSAA), IEEE, 2021, pp. 1–8.

[51] G. Li, M. Dong, L. Ming, C. Luo, H. Yu, X. Hu, B. Zheng, Deep reinforcement learning based ensemble model for rumor tracking, Information Systems 103 (2022) 101772.

[52] G. Sharma, A. Singh, S. Jain, Deepevap: Deep reinforcement learning based ensemble approach for estimating reference evapotranspiration, Applied Soft Computing 125 (2022) 109113.

[53] S. Dong, C. Yu, G. Yan, J. Zhu, H. Hu, A novel ensemble reinforcement learning gated recursive network for traffic speed forecasting, in: 2021 Workshop on Algorithm and Big Data, 2021, pp. 55–60.

[54] H. Liu, C. Yu, H. Wu, Z. Duan, G. Yan, A new hybrid ensemble deep reinforcement learning model for wind speed short term forecasting, Energy 202 (2020) 117794.

[55] S. K. Perepu, B. S. Balaji, H. K. Tanneru, S. Kathari, V. S. Pinnamaraju, Reinforcement learning based dynamic weighing of ensemble models for time series forecasting, arXiv preprint arXiv:2008.08878 (2020).

[56] X. Liu, M. Qin, Y. He, X. Mi, C. Yu, A new multi-data-driven spatiotemporal pm2.5 forecasting model based on an ensemble graph reinforcement learning convolutional network, Atmospheric Pollution Research 12 (10) (2021) 101197.

[57] D. L. Elliott, C. Anderson, The wisdom of the crowd: Reliable deep reinforcement learning through ensembles of q-functions, IEEE transactions on neural networks and learning systems (2021).

[58] Q. Li, C. Yu, G. Yan, A new multipredictor ensemble decision framework based on deep reinforcement learning for regional gdp prediction, IEEE Access 10 (2022) 45266–45279.

[59] Z. Cao, H. Liu, A novel carbon price forecasting method based on model matching, adaptive decompo-

sition, and reinforcement learning ensemble strategy, Environmental Science and Pollution Research (2022) 1–24.

[60] Y. Birman, S. Hindi, G. Katz, A. Shabtai, Cost-effective ensemble models selection using deep reinforcement learning, Information Fusion 77 (2022) 133–148.

[61] S. Yin, H. Liu, Wind power prediction based on outlier correction, ensemble reinforcement learning, and residual correction, Energy 250 (2022) 123857.

[62] F. Schubert, C. Benjamins, S. Döhler, B. Rosenhahn, M. Lindauer, Polter: Policy trajectory ensemble regularization for unsupervised reinforcement learning, arXiv preprint arXiv:2205.11357 (2022).

[63] M. Shen, J. P. How, Robust opponent modeling via adversarial ensemble reinforcement learning in asymmetric imperfect-information games, arXiv preprint arXiv:1909.08735 (2019).

[64] Y. Qin, Z. Wang, C. Chen, Hrl2e: Hierarchical reinforcement learning with low-level ensemble, in: 2022 International Joint Conference on Neural Networks (IJCNN), IEEE, 2022, pp. 1–7.

[65] P. V. R. Ferreira, R. Paffenroth, A. M. Wyglinski, T. M. Hackett, S. G. Bilén, R. C. Reinhart, D. J. Mortensen, Multiobjective reinforcement learning for cognitive satellite communications using deep neural network ensembles, IEEE Journal on Selected Areas in Communications 36 (5) (2018) 1030–1041.

[66] M. A. Wiering, H. Van Hasselt, Ensemble algorithms in reinforcement learning, IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 38 (4) (2008) 930–936.

[67] X. Chen, L. Cao, C. Li, Z. Xu, J. Lai, Ensemble network architecture for deep reinforcement learning, Mathematical Problems in Engineering 2018 (2018).

[68] R. Saphal, B. Ravindran, D. Mudigere, S. Avancha, B. Kaul, Seerl: Sample efficient ensemble reinforcement learning, arXiv preprint arXiv:2001.05209 (2020).

[69] H. Eriksson, D. Basu, M. Alibeigi, C. Dimitrakakis, Sentinel: taming uncertainty with ensemble based distributional reinforcement learning, in: Uncertainty in Artificial Intelligence, PMLR, 2022, pp. 631–640.

[70] M. Németh, G. Szűcs, Split feature space ensemble method using deep reinforcement learning for algorithmic trading, in: Proceedings of the 2022 8th International Conference on Computer Technology Applications, 2022, pp. 188–194.

[71] S. Faußer, F. Schwenker, Neural network ensembles in reinforcement learning, Neural Processing Letters 41 (2015) 55–69.

[72] S. M. J. Jalali, M. Ahmadian, S. Ahmadian, A. Khosravi, M. Alazab, S. Nahavandi, An oppositional-

cauchy based gsk evolutionary algorithm with a novel deep ensemble reinforcement learning strategy for covid-19 diagnosis, Applied Soft Computing 111 (2021) 107675.

[73] J. Wu, H. Li, Deep ensemble reinforcement learning with multiple deep deterministic policy gradient algorithm, Mathematical Problems in Engineering 2020 (2020) 1–12.

[74] H. Sheikh, K. Frisbee, M. Phielipp, Dns: Determinantal point process based neural network sampler for ensemble reinforcement learning, in: International Conference on Machine Learning, PMLR, 2022, pp. 19731–19746.

[75] J. Buckman, D. Hafner, G. Tucker, E. Brevdo, H. Lee, Sample-efficient reinforcement learning with stochastic ensemble value expansion, Advances in neural information processing systems 31 (2018).

[76] G. Chen, Y. Peng, M. Zhang, Effective exploration for deep reinforcement learning via bootstrapped q-ensembles under tsallis entropy regularization, arXiv preprint arXiv:1809.00403 (2018).

[77] O. Peer, C. Tessler, N. Merlis, R. Meir, Ensemble bootstrapping for q-learning, in: International Conference on Machine Learning, PMLR, 2021, pp. 8454–8463.

[78] A. Brown, M. Petrik, Interpretable reinforcement learning with ensemble methods, arXiv preprint arXiv:1809.06995 (2018).

[79] U. Pak, J. Ma, U. Ryu, K. Ryom, U. Juhyok, K. Pak, C. Pak, Deep learning-based pm2. 5 prediction considering the spatiotemporal correlations: A case study of beijing, china, Science of The Total Environment 699 (2020) 133561.

[80] J. Ma, Z. Yu, Y. Qu, J. Xu, Y. Cao, et al., Application of the xgboost machine learning method in pm2. 5 prediction: A case study of shanghai, Aerosol and Air Quality Research 20 (1) (2020) 128–138.

[81] S. M. J. Jalali, M. Khodayar, S. Ahmadian, M. Shafie-Khah, A. Khosravi, S. M. S. Islam, S. Nahavandi, J. P. Catalão, A new ensemble reinforcement learning strategy for solar irradiance forecasting using deep optimized convolutional neural network models, in: 2021 International Conference on Smart Energy Systems and Technologies (SEST), IEEE, 2021, pp. 1–6.

[82] Y. Li, Z. Liu, H. Liu, A novel ensemble reinforcement learning gated unit model for daily pm2. 5 forecasting, Air Quality, Atmosphere & Health 14 (2021) 443–453.

[83] C. Chen, H. Liu, Dynamic ensemble wind speed prediction model based on hybrid deep reinforcement learning, Advanced Engineering Informatics 48 (2021) 101290.

[84] S. M. J. Jalali, G. J. Osório, S. Ahmadian, M. Lotfi, V. M. Campos, M. Shafie-khah, A. Khosravi, J. P. Catalão, New hybrid deep neural architectural search-based ensemble reinforcement learning strategy for wind power forecasting, IEEE Transactions on Industry Applications 58 (1) (2021) 15–27.

[85] J. Qin, Y. Gao, M. Bragin, N. Yu, An optimization method-assisted ensemble deep reinforcement learning algorithm to solve unit commitment problems, arXiv preprint arXiv:2206.04249 (2022).

[86] T. Sogabe, D. B. Malla, C.-C. Chen, K. Sakamoto, Attention and masking embedded ensemble reinforcement learning for smart energy optimization and risk evaluation under uncertainties, Journal of Renewable and Sustainable Energy 14 (4) (2022) 045501.

[87] B. He, H. Zhao, G. Liang, J. Zhao, J. Qiu, Z. Y. Dong, Ensemble-based deep reinforcement learning for robust cooperative wind farm control, International Journal of Electrical Power & Energy Systems 143 (2022) 108406.

[88] S. M. J. Jalali, S. Ahmadian, B. Nakisa, M. Khodayar, A. Khosravi, S. Nahavandi, S. M. S. Islam, M. Shafie-khah, J. P. Catalão, Solar irradiance forecasting using a novel hybrid deep ensemble reinforcement learning algorithm, Sustainable Energy, Grids and Networks 32 (2022) 100903.

[89] K. D. Polyzos, Q. Lu, A. Sadeghi, G. B. Giannakis, On-policy reinforcement learning via ensemble gaussian processes with application to resource allocation, in: 2021 55th Asilomar Conference on Signals, Systems, and Computers, IEEE, 2021, pp. 1018–1022.

[90] A. Sadeghi, F. Sheikholeslami, G. B. Giannakis, Optimal and scalable caching for 5g using reinforcement learning of space-time popularities, IEEE Journal of Selected Topics in Signal Processing 12 (1) (2017) 180–190.

[91] A. Ashiquzzaman, H. Lee, T.-W. Um, J. Kim, Energy-efficient iot sensor calibration with deep reinforcement learning, IEEE Access 8 (2020) 97045–97055.

[92] D. Gu, J. Chen, X. Shi, L. Ran, Y. Zhang, M. Shang, Heterogeneous-aware online cloud task scheduler based on clustering and deep reinforcement learning ensemble, in: Advances in Natural Computation, Fuzzy Systems and Knowledge Discovery, Springer, 2021, pp. 152–159.

[93] X.-Y. Liu, Z. Li, Z. Yang, J. Zheng, Z. Wang, A. Walid, J. Guo, M. I. Jordan, Elegantrl-podracer: Scalable and elastic library for cloud-native deep reinforcement learning, arXiv preprint arXiv:2112.05923 (2021).

[94] S. K. Mahmud, Y. Chen, K. K. Chai, Ensemble reinforcement learning framework for sum rate optimization in noma-uav network, in: 2022 IEEE World AI IoT Congress (AIIoT), IEEE, 2022, pp. 032–038.

[95] B. Xu, X. Hu, X. Tang, X. Lin, H. Li, D. Rathod, Z. Filipi, Ensemble reinforcement learning-based supervisory control of hybrid electric vehicle for fuel economy improvement, IEEE Transactions on Transportation Electrification 6 (2) (2020) 717–727.

[96] K.-F. Tang, H.-C. Kao, C.-N. Chou, E. Y. Chang, Inquire and diagnose: Neural symptom checking ensemble using deep reinforcement learning, in: NIPS workshop on deep reinforcement learning, 2016.

[97] S. Henna, A. A. Minhas, M. S. Khan, M. S. Iqbal, Ensemble consensus representation deep reinforcement learning for hybrid fso/rf communication systems, Optics Communications 530 (2023) 129186.

[98] H. Cuayáhuitl, D. Lee, S. Ryu, Y. Cho, S. Choi, S. Indurthi, S. Yu, H. Choi, I. Hwang, J. Kim, Ensemble-based deep reinforcement learning for chatbots, Neurocomputing 366 (2019) 118–130.

[99] I. Osband, C. Blundell, A. Pritzel, B. Van Roy, Deep exploration via bootstrapped dqn, Advances in neural information processing systems 29 (2016).

[100] I. Partalas, G. Tsoumakas, I. Vlahavas, Pruning an ensemble of classifiers via reinforcement learning, Neurocomputing 72 (7-9) (2009) 1900–1909.

[101] T. Pearce, N. Anastassacos, M. Zaki, A. Neely, Bayesian inference with anchored ensembles of neural networks, and application to exploration in reinforcement learning, arXiv preprint arXiv:1805.11324 (2018).

[102] M. Shen, J. P. How, Robust opponent modeling via adversarial ensemble reinforcement learning, in: Proceedings of the International Conference on Automated Planning and Scheduling, Vol. 31, 2021, pp. 578–587.

[103] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, W. Zaremba, Openai gym, arXiv preprint arXiv:1606.01540 (2016).

[104] R. Caruana, A. Niculescu-Mizil, G. Crew, A. Ksikes, Ensemble selection from libraries of models, in: Proceedings of the twenty-first international conference on Machine learning, 2004, p. 18.

[105] G. Tsoumakas, L. Angelis, I. Vlahavas, Selective fusion of heterogeneous classifiers, Intelligent Data Analysis 9 (6) (2005) 511–525.

[106] T. Schaul, J. Quan, I. Antonoglou, D. Silver, Prioritized experience replay, arXiv preprint arXiv:1511.05952 (2015).

[107] D. Isele, A. Cosgun, Selective experience replay for lifelong learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 32, 2018.

[108] Y.-H. Pao, G.-H. Park, D. J. Sobajic, Learning and generalization characteristics of the random vector functional-link net, Neurocomputing 6 (2) (1994) 163–180.

[109] Q. Shi, R. Katuwal, P. N. Suganthan, M. Tanveer, Random vector functional link neural network based ensemble deep learning, Pattern Recognition 117 (2021) 107978.

[110] B. Han, J. Sim, H. Adam, Branchout: Regularization for online ensemble tracking with convolutional

neural networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 3356–3365.