Hindawi
International Journal of Intelligent Systems
Volume 2023, Article ID 5588877, 11 pages
https://doi.org/10.1155/2023/5588877

Research Article

Predicting Potential Risk: Cerebral Stroke via Regret Minimization

Jie Zhang, Hao Chen, Cheng Jin, Qiangqiang He, Wei He, and Chongjun Wang

Department of Computer Science and Technology, Nanjing University, 163 Xianlin Avenue, Qixia District, Nanjing 210023, China

Correspondence should be addressed to Jie Zhang; iip_zhangjie@smail.nju.edu.cn

Received 5 April 2023; Revised 29 September 2023; Accepted 3 October 2023; Published 2 December 2023

Academic Editor: Alexander Hošovský

Objective. The data processing of medical test report has always been one of the important contents in biological information domain, especially the process of extracting the effective information from the report so as to assist doctors with the correct medical plan. Usual methods neglect the implicit relationship between features. More features are generally not a better choice because more noise is generated between feature combinations. We propose a practical feature selection strategy RMFS, which aims to select the optimal combination of features. *Materials and Methods*. Based on the above situation, in this paper, 64 features are extracted from a real medical test report dataset for stroke and feature selection is defined as a reinforcement learning problem to optimize the feature combination by minimizing regret. We select three current mainstream feature selection methods and conduct comparative experiments. *Results*. We processed and completed a dataset derived from real medical test reports of stroke. We redefine the feature selection problem as a reinforcement learning problem and propose an optimization strategy based on regret minimization and train weight parameters in a DQN network. Experimental results demonstrate that our method can identify feature combinations with higher prediction accuracy. *Discussion*. RMFS shows a strong robustness to the randomness of the environment and has high computational efficiency and accuracy. Compared with the previous feature selection methods, our method yields superior results. *Conclusion*. The experimental results demonstrate that our method can obtain a more accurate prediction rate under the same feature scale and we can achieve baseline performance with fewer features.

1. Introduction

Cerebral stroke is known as apoplexia or cerebral vascular accident (CVA). It is an acute cerebrovascular disease, including ischemic stroke and hemorrhagic stroke, which is caused by a sudden rupture of a blood vessel in the brain or a blockage of a blood vessel that prevents blood from flowing to the brain and results in brain tissue damage. The incidence of ischemic stroke is higher than that of hemorrhagic stroke, accounting for 60% to 70% of all strokes. According to the newly published Global Burden of Disease Study data, the number of stroke patients worldwide is estimated to exceed 100 million. In China, for example, the prevalence of stroke has shown a rapid growth trend from 1.89% since 2012, with an annual growth rate of more than 7%, according to the National Cerebrovascular Disease Data Platform. Data from the Global Burden of Disease Study show that stroke is one of the leading causes of death and disability among adults in China [1]. China is the largest developing country, with a population of about one-fifth of the world's total, and the number of current stroke patients ranks first in the world. As one of the important components of stroke, more than 20 million people around the world have the potential risk of cerebral stroke, so how to predict the incidence of stroke has become a daunting task. A medical examination report (MER) includes the patient's personal data in a medical institution as well as examination data, such as identification information, drug allergy history, and medical history. MER not only raises efficiency for doctors and healthcare professionals but also provides a valuable source of data for researchers. The current situation of the prediction of potential risk of stroke involves the use of various clinical risk

prediction models [2], such as the Framingham Stroke Risk Profile and the CHA2DS2-VASc Score [3], which take into account various risk factors such as age, gender, blood pressure, diabetes, smoking, and previous history of stroke or heart disease [4]. These models are used by healthcare professionals to identify individuals who may be at a high risk of experiencing a stroke and to guide preventative interventions such as lifestyle modifications or medication. However, there are several existing problems with the current methods of predicting stroke risk. One major issue is that these models may not be accurate enough in certain populations, such as younger individuals or those from different ethnic backgrounds. Additionally, there may be other risk factors that are not yet included in these models, such as genetic factors or lifestyle factors that are difficult to measure. Another issue is that even when high-risk individuals are identified, there may be barriers to accessing preventative interventions, such as lack of resources or inadequate healthcare infrastructure.

To address these problems, further research is needed to develop more accurate and comprehensive risk prediction models and to better understand the underlying mechanisms of stroke risk. A crucial question in our research is how to improve predictive performance by learning the features of patients and diseases so as to perform a better risk control and treatment for the disease [5]. Deep reinforcement learning has some research on this issue, such as attention-based mechanisms [1], but there are still some challenges in effectively utilizing data and model interpretation:

(1) Neglected edge information

Due to the numerous examinations in the medical domain, the data sources for predicting a single disease are relatively complex. Only key data are selected as the benchmark for model learning because the sampling probability of edge information in the traditional definition is low and even the edge information might be abandoned during model learning. The approach that uses a graph structure to classify diseases on different levels into different types of graphs is adopted, but it ignores the help of information such as complications for future diagnosis prediction.

(2) Optimal solution of sequential decision-making

In traditional reinforcement learning, sequential decision-making has always been one of the significant research problems. On how to influence the future reward by changing the current strategy, the paper [6] selects continuous partially observable Markov decision processes (POMDP) scenarios and uses approximate solution to infer the potential state, but it neglects the relationship between the solution of the optimal decision sequence and the environmental information.

(3) Lack of model generalization

Due to the lack of data, the data sources of different hospitals lead to the mismatch between data features and distribution. Therefore, it may be difficult to learn an accurate model using the data of one hospital and the feature selection of the data is required to select the common data with high importance as the reference index. Many models do not make full use of data, which leads to the unsatisfactory result by lack of generalization.

In view of the abovementioned points, we can conclude that in the current medical environment, reinforcement learning still has some problems in disease prediction. How to choose the optimal combination of features as the input to calculate the optimal decision-making is the problem that this paper studies. Based on it, we will introduce the concept of regret value [7], rank features by minimizing regret values, and learn the optimal combination of features with DQN. This paper has the following major contributions:

(i) We redefine feature selection as a reinforcement learning problem, propose an optimization strategy based on regret minimization, and train the weight parameters in DQN network.

(ii) We process and complete a data set about stroke, which is derived from the real medical test report of stroke, and the experimental work in this paper is also completed based on this data set. The process is shown in Figure 1.

(iii) We selected three mainstream feature selection methods for comparison in the experiment, and the experimental results demonstrate that our method can find feature combinations with a more accurate prediction rate.

2. Related Work

Recent work [8, 9] suggests that reinforcement learning has a wide range of applications in medical information processing [10]. By selecting features from medical test reports, extracting feature combinations and learning strategies are two important tasks in this process for different prediction task scenarios. For brevity, we discuss only the medical reinforcement learning literature relevant to our work [11]. This can be roughly divided into three categories.

Feature selection can effectively prepare high-dimensional data for various learning tasks such as classification, clustering, and anomaly detection. In healthcare [12], we need to capture patient heterogeneity for personalized predictive modeling, which can be characterized by a subset of instance-specific features. Reference [13] proposed a novel unsupervised personalized feature selection (UPFS) framework to find shared features of all instances and unique features of individual instances. Feature selection can be applied to case diagnosis, and the authors in [14] explored a nonnegative generalized fusion lasso model for stable feature selection in the diagnosis of Alzheimer's disease. As technology advances, artificial intelligence (AI) models become critical in the medical domain, and the ability to interpret predictions to clinical end users is essential to harness the power of artificial intelligence models

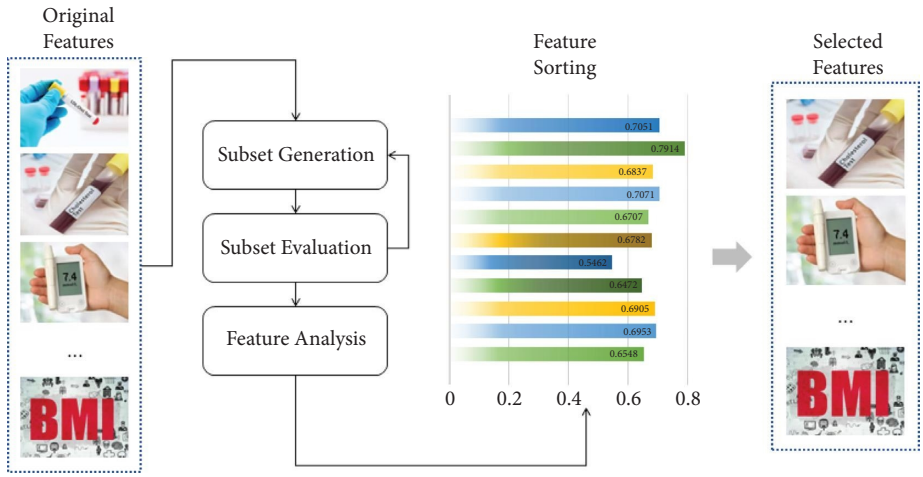


Figure 1: Process of feature selection. Different subsets of features are continuously selected from initial dataset for performance evaluation and then the contribution of each subset to model is calculated based on performance differences, which is used as a reference indicator for feature selection.

for clinical decision support. Extracted more information from the predictor through an Information Calibration method [15] and used an adversarial-based technique to calibrate the information extracted by the two models.

Feature sources in medical scenes are usually multimodal data of text or images. For medical images [16, 17], feature attribution maps or heat maps are the most common form of interpretation. The Mode-Specific Feature Importance (MSFI) index [18] encodes clinical requirements for prioritizing and localizing specific features within treatment modalities. The study demonstrated that the results produced by MSFI satisfy clinicians' needs for multimodal interpretation. The authors in [19] described the application of deep learning to multimodal medical imaging analysis.

Reinforcement learning can be used to analyze medical imaging reports and improve accuracy [20], where different modalities of image information have their own characteristics and differences in contrast and resolution due to different imaging principles. Integrated reinforcement learning with MR image manipulation can reconstruct damaged images [21]. Reference [8] proposed and optimized the Stochastic Planner-Actor-Critic (SPAC) method for medical image alignment. Nonindependent and homogeneously distributed (non-iid) data in medical images remain a prominent challenge in real practice. Reference [22] proposed a framework, HarmoFL, where perturbations helped global models converge to optimal solutions by aggregating multiple locally flat optimal solutions without additional communication costs [23]. Low resource medical dialogue generation [24] used the general knowledge graph to characterize the relationship between previous symptoms of the disease. Model-based reinforcement learning can be applied to biological sequence settings, such as DyNA-PPO. A model-based PPO variant was proposed in the paper [25], Model-based Reinforcement Learning for Biological Sequence Design, which had a good performance in biological sequence setting. Off-policy evaluation in reinforcement learning provides the feasibility for using observational data to improve the future medical and educational fields. Gottesman et al. [26] introduced a method as a hybrid artificial intelligence system, enabling human experts to analyze the accuracy of policy evaluation.

In summary, feature selection is an important technique for preparing high-dimensional data in healthcare for various learning tasks. Personalized predictive modeling requires the capture of patient heterogeneity using a subset of instance-specific features. Interpretation of predictions to clinical end-users is essential for clinical decision support. Multimodal data of text or images is common in medical scenes, and deep learning techniques can be applied to analyze them. Reinforcement learning can be used to improve accuracy in medical imaging analysis, considering the different modalities of image information. Non-iid data in medical images remains a challenge, and off-policy evaluation in reinforcement learning [27] provides the feasibility for using observational data to improve the future of medical and educational fields.

3. Preliminaries

In this section, we will introduce some preliminary knowledge for the work in this paper.

3.1. Markov Decision Process (MDP). Let us assume a standard reinforcement learning scenario, where the purpose is to learn a policy that maximizes the expected cumulative discount reward in a Markov decision process [28], which is defined by a tuple (S, A, P, R, γ). S denotes state, and A represents a set of actions. π is the strategy for the state transition, and P: $S \times A \times S \longrightarrow R$ is the reward gained during the state transition, where R is the function of reward. $\gamma \in (0, 1)$ is the discount factor. P represents the probability of taking action a in a certain state S to transfer to the next state $S^{'}$, denoted as $P_s^a = P[S_{t+1} = S^{'}|S_t = S, A_t = a]$, where a is the selected policy action in the current state transition.

The experience replay buffer is used in the off-policy agent, which is denoted as B. At each time step t, the agent interacts with the environment and sores (S_t, a_t, r_t, S_{t+1}) into B [29] and B is defined as B_i at a certain position i. Next, the agent uses B_i obtained from buffer sampling to update for each step during training. Based on the described above, the off-line replay policy learning problem is redefined as follows.

Let the task T, off-policy agent Λ, and experience replay buffer B. The goal is to learn the replay policy φ as each training step batch of transitions B_i from B to train agent Λ, which is to learn a mapping ϕ in order to train the agent to obtain better performance on task $B \longrightarrow B_i$.

3.2. Deep Q-Networks (DQN). We consider the standard reinforcement learning paradigm, including an agent interacting with the environment, and for the convenience of introduction, we assume that the environment is fully observable. Deep Q-Network [30] is a model-free RL algorithm applied in discrete action spaces. Keeping a neural network Q in DQN approximates Q^*. $\pi_Q(s) = \text{argmax}_{a\in A} Q(s,a)$ represents the greedy strategy w.r.t. Q. A is a random behavior with probability ε (uniformly sampled from $\mathscr{A}$) that has probability $1-\varepsilon$ of the action $\pi_Q(s)$.

During training, we generate episodes by using an approximation of the current action value function Q of the ε – greedy policy. The transition tuples (s_t, a_t, r_t, s_{t+1}) encountered during training are stored in the replay buffer. The generation of new episodes is interspersed with neural network training. The network is trained using small batch gradient descent on loss $\mathscr{L}$ so that the approximate Q function satisfies the Bellman equation: $\mathscr{L} = \mathbb{E}(Q(s_t, a_t) - y_t)^2$, where $y_t = r_t + \max_{a'\in A} Q(s, a')$ and the tuple (s_t, a_t, r_t, s_{t+1}) is sampled from the replay buffer.

The targets y_t are usually computed using a separate target network in order to make the optimization process more stable and the target network takes a slower rate change than the main network. It is common to regularly set the weight of the target network to weights of the main network (e.g., [30]) or to use the Polyak and Juditsky averaged [31] version of the main network [32].

3.3. Regret Minimization. Regret value is an important tool for computer to solve approximate Nash equilibrium [33]. The most widely used method in extended game is to minimize the regret value as much as possible to solve an approximate Nash equilibrium [34]. Based on the concept of MDP, it is formally defined as follows:

$$\begin{aligned} R^T(s) &= \textstyle\sum_t s(t) v^t - \sum_t \sigma^t v^t, \\ R_S^T &= \max_{s\in S} R^T(s). \end{aligned} \tag{1}$$

Here, we specify the action sequence in MDP as h. Suppose that player i replaces the actual policy σ with policy s, and the part of revenue generated by the new policy s over the original policy is the value of regret value. In particular, reward v in the regret value can be any mapping from the legal action set A to the real number R. The regret value can be minimized as long as the total cumulative regret value is sublinear. When the regret values of all actions are sufficiently small, we can consider that our policy is close enough to the Nash equilibrium to solve the problem. Here, we present the procedure of how to update the policy using regret values. When policy σ is adopted, the virtual value of the corresponding action sequence h is calculated as follows:

$$\underset{i}{v}(\sigma, h) = \sum_{z\in Z} \underset{-i}{\overset{\sigma}{\pi}}(h) \overset{\sigma}{\pi}(h, z) \underset{i}{u}(z). \tag{2}$$

We first calculate the probability value of the other players in producing the action sequence h, multiply the probability of entering the ending situation z from the action sequence h under this policy, and finally multiply the probability of player i in the ending situation z. After completing the iteration of the final situation, we add up the products. Therefore, when taking action a, the virtual regret value obtained by player i is $r(h,a) = v_i(\sigma_{I\longrightarrow a}, h) - v_i(\sigma, h)$ and the regret value of information set I corresponding to action sequence h is $r(I,a) = \sum r(h,a)$. The regret value of player i, when taking action a in round T is $\text{Regret}_t^T(I,a) = \sum_{t=1}^T r_i^t(I,a)$. Similarly, the negative regret value is not considered and is denoted as $\text{Regret}_t^{T,+}(I,a) = \max(R_i^T(I,a), 0)$. In $(T+1)$ round, the probability of player i choosing action a is calculated as follows:

$$\underset{i}{\sigma}^{T+1}(I,a) = \begin{cases} \dfrac{\text{Regret}_i^{T,+}(I,a)}{\sum_{a\in A(I)} \text{Regret}_i^{T,+}(I,a)}, \\ \text{if } \displaystyle\sum_{a\in A(I)} \text{Regret}_i^{T,+}(I,a) > 0, \\ \dfrac{1}{|A(I)|} \text{ otherwise,} \end{cases} \tag{3}$$

i chooses the next behavior according to the regret value, and if the regret value is negative, one behavior is randomly selected for the game.

4. Optimal Feature Selection Strategy via Regret Minimization

We propose a deep reinforcement learning feature selection algorithm based on the minimum regret value as Regret Minimization Feature Selection (RMFS) to learn the optimal feature combination. RMFS captures the data dependencies between features, enhances features through reward changes between action sequences, and updates buffer by minimizing regret to improve policy learning, as shown in Figure 2. Theoretically, the ideal sampling policy is to sample to the transition with higher value. Therefore, methods such as uniform sampling as well as priority sampling are derived. In general, the policy

is uniform sampling, which neglects the significance of experience. The regret minimization framework proposed in this paper increases the probability of reward samples being sampled because we believe that targeted optimization for transition with small immediate reward is essential to improve the performance of the policy. We recommend using the immediate rewards in transition as a reasonable proxy so that the state can be sampled frequently and updating action in transition to improve the immediate rewards. The off-policy algorithm uses deep neural networks as value function approximators and stores past experience in buffer B to calculate updated gradients. We assume that B_i is a transition in Buffer B and define a reward priority instant reward function φ as sampling policy. The smaller the value of φ, the higher the probability of being replay.

In common supervised learning, the training data are assumed to be independent and identically distributed and one or several data will be randomly sampled from the training data for gradient descent every time the neural network is trained. As learning continues, each transition will be used several times. Based on the original Q-learning, a replay buffer is maintained and some data are randomly sampled from the replay buffer to train the Q-network, which can play the following roles: the samples meet the independence assumption. The data obtained by interactively sampling in MDP do not satisfy the independence assumption by itself since s_t is related to s_{t-1}. The nonindependently distributed data have a great influence on the training of neural network, which will adapt the neural network to the latest data. ER can break the sample correlation, make it meet the independence assumption, and improve sample efficiency. In the deep Q-network (DQN) algorithm [30], a deep neural network is used to approximate the optimal value function:

$$Q^*(s,a) = \max_{\pi} Q^{\pi}(s,a), \tag{4}$$

after experiencing a state s and taking an action a. Deep Q network $Q(s,a;\theta)$ is parameterized by deep neural network, where θ is a parameter. During training, the DQN agent stores its experience $e_t = (s_t, a_t, r_t, s_{t+1})$ into the replay buffer $\mathscr{D} = \{e_1, e_2, \cdots\}$ at each time step t, which deposits the last million transitions. When implementing the update, by minimizing the loss, small batches of experiences $(s,a,r,s') \sim \mathscr{D}$ are sampled uniformly from the replay buffer to optimize the deep Q-network with stochastic gradient descent:

$$L(\theta) = \mathbb{E}_{(s,a,r,s')\sim U(\mathscr{D})}\left[y - Q(s,a;\theta)^2\right], \tag{5}$$

where $y = r + \gamma \max_{a'} Q(s',a';\theta^-)$ represents the bootstrapping target, θ^- denotes the parameters of the target network $Q^-(s,a;\theta^-)$, and $Q(s,a;\theta)$ is a periodic copy of the deep Q-network. Due to the advantages of combining the deep RL algorithm with the empirical replay algorithm, DQN and its variants [35] demonstrate exceptional performance on our dataset. The specific algorithm is as follows Algorithm 1.

5. Experiment

5.1. Experimental Setting and Baseline. In this section, our study was based on data from hospital's Brain Infarction Screening Program for high-risk populations. The data mainly include demographic information, medical history information, personal history, family history information, and blood index information. In order to better analyze the risk factors of stroke, we fully consider three aspects in the data stage preprocessing: (1) how to fill in the missing data; (2) how to deal with categorical features; (3) how to deal with continuous features. Afterwards, we obtained 64 features in each sample of 6527 patients. The three aspects of data preprocessing are described in detail as follows:

(i) *Filling in missing data*: Because our study was based on regular follow-up of community residents, residents could drop out or be lost to follow-up, resulting in data loss. In the original data set, most attribute values are greater than or equal to "0," and we can uniformly fill the missing values with "−1," which makes it more accessible to distinguishing the missing values from the normal values.

(ii) *Classification feature processing*: We adopted one-hot coding for classification features (without the difference between the correlation feature value and its actual meaning, such as PayStyle and Job) to obtain the effect of different attributes of stroke, which can make the data distribution more sparse and expand the feature space.

(iii) *Continuous feature processing*: In order to simplify the model and reduce the risk of model overfitting, some continuous features such as age and height are discretized. We map features from different intervals to different buckets.

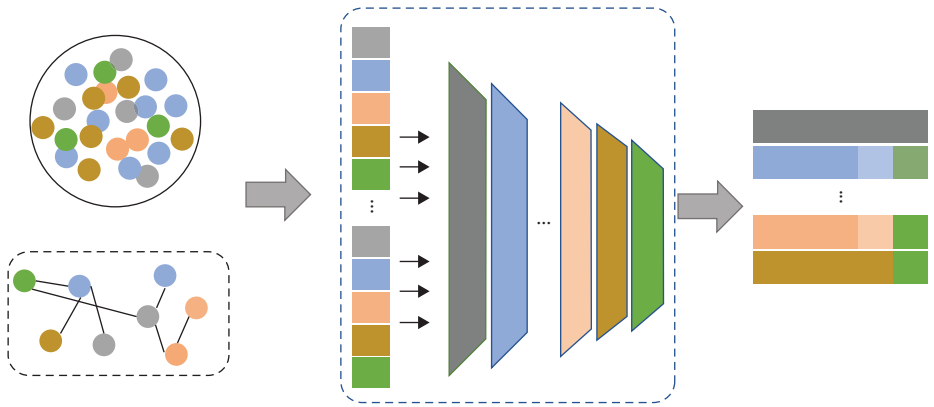
In order to make a fair comparison and prove the effectiveness of our algorithm, we use the following common feature selection methods as comparison:

(i) *Chi-square test*: It uses the idea of commonly used hypothesis testing in probability theory and mathematical statistics and aims to measure the correlation between two variables.

(ii) *F-test*: It is a hypothesis test method based on F-distribution; that is, it is applied to capture the linear relationship between each feature and label.

(iii) *Mutual information*: A variable that measures the relationship between two random numbers sampled at the same time.

We use DQN to explore feature selection. In DQN, the buffer size of the experience pool is set to ten thousand and each batch size is set to 16. The network structure of DQN is an MLP network with a hidden layer [256, 256], and the update frequency of target net is once every 100 training



(a) Implicit Relationship between the Feature Distributions (b) Feature Filter (c) Feature Combination

Figure 2: Mainly composed of three parts: (a) can be observed that there are implicit correlations between features. By filtering and sorting the features of (b), the ordered feature expression in (c) can be obtained according to the importance and other indicators.

(1) Initialize feature $(f_1, f_2, \ldots, f_n)$ as F.
(2) Calculate the accuracy of a single feature.
(3) Initialize replay memory D to capacity N.
(4) Initialize action-value function Q with random weights.
(5) for episode = 1, M do
(6) Initialize sequence $s_1 = F$ and processed sequenced $\varnothing_1 = \varnothing(s_1)$.
(7) for $t = 1, T$ do
(8) Select a feature f into F_t as action a_t.
(9) Execute action a_t in prediction p_t and reward r_t.
(10) Set $s_{t+1} = s_t, a_t, x_{t+1}$ and $\varnothing_{t+1} = \varnothing(s_{t+1})$.
(11) Store transition (s_t, a_t, r_t, s_{t+1}) in D.
(12) Sample random minibatch of transitions
(13) $(\varnothing_j, a_j, r_j, \varnothing_{j+1})$ in D.
(14) end for
(15) end for

Algorithm 1: Regret minimization experience replay with DQN.

times, for a total of 1000 epochs. In the experiment, the learning rate of DQN network was set as 0.01 and 0.001, and the reward discount factor was set as 0.9 and 0.99, respectively, for network training. The parameters of chi-square test, F-test, and mutual information are set the same as DQN.

5.2. Performance Comparison and Effect of Parameters. In Figure 3, the abscissa represents the number of selected features and the ordinate represents the accuracy of using the selected features on the test set. From the experimental results, our DQN method had achieved better results than the other three methods although we adjusted the vector and a reward in DQN discount factor, the final selected feature on the test set arrived the highest accuracy than the other three methods, and in most cases, the accuracy is the same, but our approach requires a smaller number of features. In Figure 4, we also provide the performance comparison of these algorithms in terms of F1 score, precision, and recall metrics. It can be observed that due to the complexity of the F1 method, it exhibits significantly higher computational resource consumption compared to other methods, which may result in certain performance advantages. However, this does not align with practical requirements. In contrast, our method achieves a more balanced trade-off between performance and resource consumption.

It can be observed from Table 1, and achieved a higher accuracy of our method in both the number of features is less than the other three methods, including vector in 0.01, reward the discount factor for 0.9 DQN method selected nearly half, less number of features can be shown using DQN feature selection achieved very good result. At the same time, it can also be found that in this experiment, the smaller the learning rate, the higher the final accuracy, indicating that the DQN network fully and stably learned better experience. However, the smaller the reward discount factor, the DQN network will pay more attention to the current reward, and the number of features selected will be smaller, but the highest possible accuracy rate will be lower.

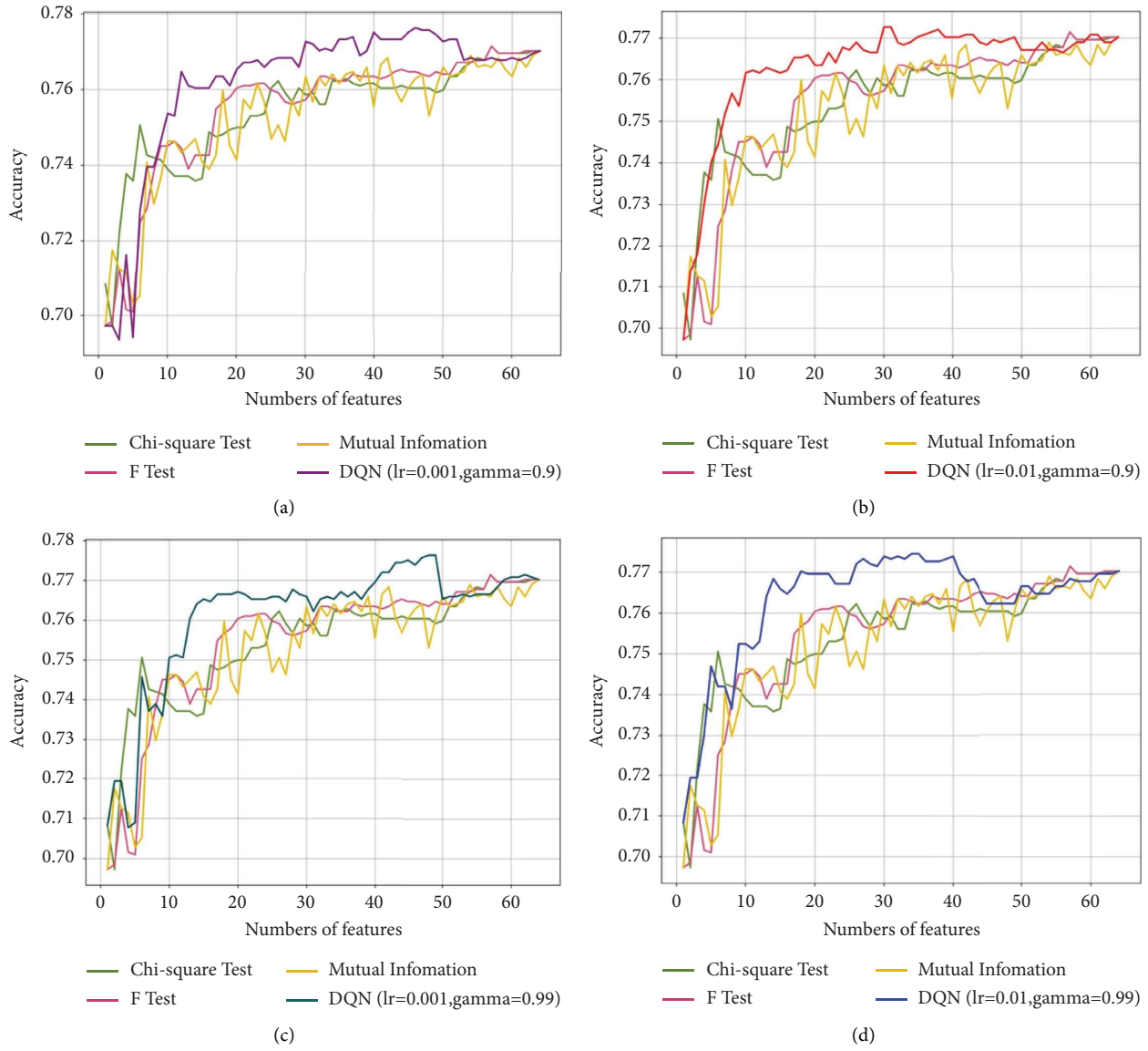


Figure 3: Accuracy of performance characteristics of different methods. (a–d) show the accuracy comparison between DQN and chi-square test, *F* test, and mutual information methods with different feature parameters modified, respectively.

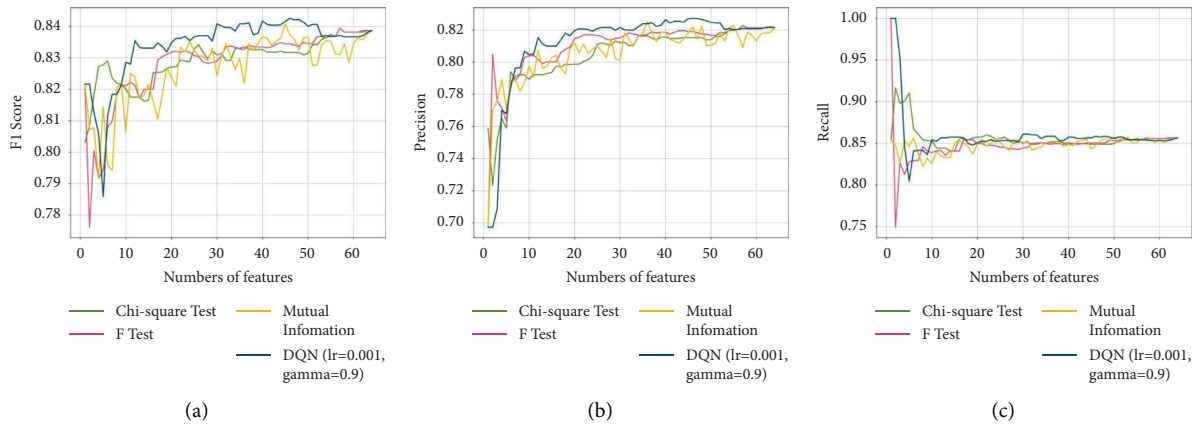


Figure 4: Continued.

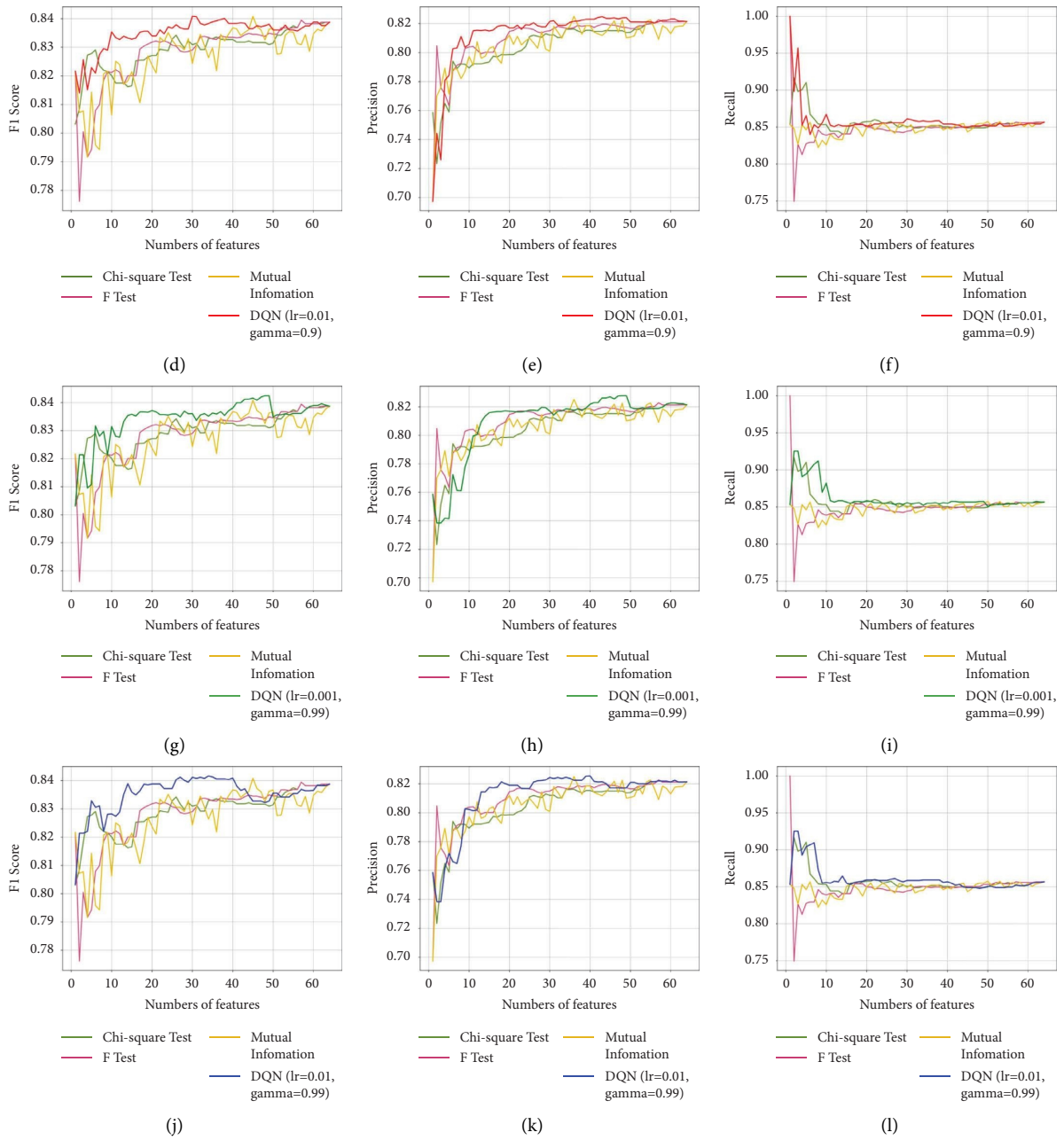


Figure 4: Performance of different algorithms in terms of F1 score, precision, and recall. The x-axis represents the number of selected features, while the y-axis represents the metrics. From top to bottom, the first row represents the performance results of four algorithms under different metrics when the parameter settings are $\lambda = 0.001$ and $\gamma = 0.9$; the second row represents the performance results when the parameter settings are $\lambda = 0.01$ and $\gamma = 0.9$; the third row represents the performance results when the parameter settings are $\lambda = 0.001$ and $\gamma = 0.99$; and the fourth row represents the performance results when the parameter settings are $\lambda = 0.01$ and $\gamma = 0.99$.

Table 1: Comparison of optimal performance on different methods.

Methods	Highest accuracy rate (%)	Feature number
Chi-square test	77.14	57
F test	77.14	57
Mutual information	77.02	64
RMFS ($\lambda = 0.01$, $\gamma = 0.9$)	77.27	30
RMFS ($\lambda = 0.01$, $\gamma = 0.99$)	77.45	34
RMFS ($\lambda = 0.001$, $\gamma = 0.9$)	77.63	46
RMFS ($\lambda = 0.001$, $\gamma = 0.99$)	77.63	47

Table 2: Comparison of method accuracy across different feature scales.

Feature number	Method name			
	Chi-square test (%)	*F* test (%)	Mutual information (%)	RMFS (%)
1	(acPayStyle, **70.83**)	(dfHypertension, 69.73)	(dfHypertension, 69.73)	(age_4, 69.73)
2	(acPayStyle, acJob, 69.73)	(dfHypertension, dfSportsLack, 69.85)	(dfHypertension, acPayStyle, **71.75**)	(age_4, dfGlycuresis, 71.38)
3	(acPayStyle, acJob, age_4, 72.18)	(dfHypertension, dfSportsLack, acPayStyle, 71.26)	(dfHypertension, acPayStyle, acJob, 71.45)	(age_4, dfGlycuresis, acPayStyle, **72.92**)
5	(acPayStyle, acJob, age_4, age_6, glu_high, 73.59)	(dfHypertension, dfSportsLack, acPayStyle, acJob, ldlc_low, 70.10)	(dfHypertension, acPayStyle, acJob, dfSportsLack, lsDrink, 70.10)	(age_4, dfGlycuresis, acPayStyle, age_5, ldlc_low, **74.69**)
10	73.90	74.51	74.63	**76.16**
20	75.00	76.04	74.14	**76.35**
30	75.86	75.74	76.35	**77.27**
40	76.16	76.35	75.55	**77.02**
50	75.98	76.41	76.59	**76.72**
60	76.96	76.96	76.35	**77.08**

The bold values indicate the minimum values in each column. The smaller the value is, the better the value is.

The specific computation time of different models is influenced by multiple factors such as the size of the dataset, the number of features, and their complexity. Taking the chi-square test method as a reference with its computation time assumed as 1, the computation time for F Test is approximately 1.2 to 1.5 times that of the chi-square test, while the computation time for mutual information is in the range of 1.5 to 2 times.

We fixed the number of selected features and observed the highest accuracy achieved by different methods given the number of features to be selected. As can be seen from the table, the accuracy performance of our method is higher than the other three methods in almost all cases, especially when the feature number is 30. This indicates that our method can not only select the optimal number and combination of features but also obtain higher accuracy when the number of features to be selected is fixed.

In our work, we listed the feature combinations selected by each method when the number of features to be selected was 1, 2, 3, and 5, respectively, and counted the frequency of each feature to analyze the importance of the feature in Table 2. The top five features, from most to least, were

(i) *AcPayStyle*. This is the most important feature of this experiment, showing that stroke patients have a large proportion of reimbursement from rural cooperative medical insurance, indicating that the prevalence, incidence, and mortality of stroke in rural residents are significantly higher than those in urban residents.

(ii) *DfHypertension*. It is the second most important feature affecting the results of the experiment, which is also in line with the prior knowledge of modern medicine. According to statistics, 70% to 80% of stroke patients have high blood pressure, and hypertension can increase the risk of stroke.

(iii) *Age_4*. This is the third most important feature in the experiment, representing people between the ages of 40 and 50. It is also found in the summary of stroke data in China that the population with stroke tends to be younger in the past 40 years, and the experimental results also reflect this fact to a certain extent.

(iv) *AcJob*. One of the important features affecting the results of the experiment has been shown that if people engage in high-intensity mental work for a long time, they have a significantly higher incidence of high blood pressure, which is an important risk factor for stroke, than the average manual worker.

(v) *DfSportsLack*. It also plays a certain role in influencing the results of the experiment, which is related to the lifestyle of patients. Chronic lack of movement will cause fat and cholesterol to stick to the vessel wall, which in turn narrows the walls, leading to slower blood flow, and over time, such blockages can increase the risk of stroke.

6. Conclusion and Future Work

In this paper, we introduce existing feature selection methods first and point out the drawbacks that these methods may ignore the relationships between features. Based on the requirement of this issue, we analyze the feature selection strategy from the perspective of minimizing regret and model feature selection in terms of RL to train the optimal feature combinations by DQN. Based on the theoretical analysis, we propose a practical feature selection strategy RMFS, which aims to select the optimal combination of features. RMFS shows a strong robust to the randomness of the environment and has high computational efficiency and accuracy. Compared with the previous feature selection methods, our method yields superior results. In future work, we will extend our framework and attempt to adjust the buffer size in different training phases since our framework is general. In addition, we will investigate more about the importance and validity of features such as proxy signals.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This paper was supported by the National Natural Science Foundation of China (Grant nos. 62192783 and 62376117) and the Collaborative Innovation Center of Novel Software Technology and Industrialization at Nanjing University.

References

[1] F. Ma, R. Chitta, J. Zhou, Y. Jing, S. Quanzeng, and G. J. Tong, "Dipole: diagnosis prediction in healthcare via attention-based bidirectional recurrent neural networks," in *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1903–1911, Halifax, Canada, August 2017.

[2] B. Tasci, "Automated ischemic acute infarction detection using pre-trained CNN models' deep features," *Biomedical Signal Processing and Control*, vol. 82, Article ID 104603, 2023.

[3] G. Y. H. Lip, R. Nieuwlaat, R. Pisters, D. A. Lane, and H. J. G. M. Crijns, "Refining clinical risk stratification for predicting stroke and thromboembolism in atrial fibrillation using a novel risk factor-based approach: the euro heart survey on atrial fibrillation," *Chest*, vol. 137, pp. 263–272, 2010.

[4] R. B. D'Agostino, P. A. Wolf, A. J. Belanger, and W. B. Kannel, "Stroke risk profile: adjustment for antihypertensive medication. The Framingham Study," *Stroke*, vol. 25, no. 1, pp. 40–43, 1994.

[5] D. J. Reinkensmeyer, E. Guigon, and M. A. Maier, "A computational model of use-dependent motor recovery following a stroke: optimizing corticospinal activations via reinforcement learning can explain residual capacity and other strength recovery dynamics," *Neural Networks*, vol. 29-30, pp. 60–69, 2012.

[6] H.J. A. Nam, F. Scott, and E. Brunskill, "Reinforcement learning with state observation costs in action-contingent noiselessly observable Markov decision processes," *Advances in Neural Information Processing Systems*, vol. 34, pp. 15650–15666, 2021.

[7] N. Burch, *Time and Space: Why Imperfect Information Games Are Hard*, University of Alberta, Edmonton, Canada, 2018.

[8] Z. Luo, J. Hu, X. Wang et al., "Stochastic planner-actor-critic for unsupervised deformable image registration," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 2, pp. 1917–1925, 2022.

[9] S. K. Zhou, H. N. Le, K. Luu, H. V Nguyen, and N. Ayache, "Deep reinforcement learning in medical imaging: a literature review," *Medical Image Analysis*, vol. 73, Article ID 102193, 2021.

[10] R. Parr, L. Li, G. Taylor, W. Painter, L. Christopher, and L. Michael, "An analysis of linear models, linear value-function approximation, and feature selection for reinforcement learning," in *Proceedings of the 25th international conference on Machine learning*, pp. 752–759, Helsinki, Finland, July 2008.

[11] G. Lim, Z. W. Lim, D. Xu et al., "Feature Isolation for Hypothesis Testing in Retinal Imaging: An Ischemic Stroke Prediction Case Study," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 1, pp. 9510–9515, 2019.

[12] C. Yu, J. Liu, S. Nemati, and G. Yin, "Reinforcement learning in healthcare: a survey," *ACM Computing Surveys*, vol. 55, no. 1, pp. 1–36, 2021.

[13] J. Li, L. Wu, H. Dani, and H. Liu, "Unsupervised personalized feature selection," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[14] B. Xin, L. Hu, Y. Wang, and W. Gao, "Stable feature selection from brain sMRI," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, no. 1, 2015.

[15] L. Sha, O. M. Camburu, and T. Lukasiewicz, "Learning from the best: rationalizing predictions by adversarial information calibration," in *Proceedings of the The 35th AAAI Conference on Artificial Intelligence (AAAI-21)*, pp. 13771–13779, London, UK, May 2021.

[16] T. Rückstieß, C. Osendorfer, and P. Smagt, "Sequential feature selection for classification," *Australasian Joint Conference on Artificial Intelligence*, pp. 132–141, Springer, Berlin, Heidelberg, 2011.

[17] B. Taşcı and I. Tasci, "Deep feature extraction based brain image classification model using preprocessed images: PDRNet," *Biomedical Signal Processing and Control*, vol. 78, Article ID 103948, 2022.

[18] W. Jin, X. Li, and G. Hamarneh, "Evaluating explainable AI on a multi-modal medical imaging task: can existing algorithms fulfill clinical requirements?" in *Proceedings of the Association for the Advancement of Artificial Intelligence Conference (AAAI)*, London, UK, August 2022.

[19] Y. Xu, "Deep learning in multimodal medical image analysis," *International Conference on Health Information Science*, pp. 193200, Springer, Cham, Switzerland, 2019.

[20] Y. Li, M. Murias, S. Major, G. Dawson, and D. E. Carlson, "Extracting relationships by multi-domain matching," *Advances in Neural Information Processing Systems*, vol. 31, pp. 6799–6810, 2018.

[21] W. Li, X. Feng, H. An, X. Y. Ng, and Y. J. Zhang, "Mri reconstruction with interpretable pixel-wise operations using reinforcement learning," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 01, pp. 792–799, 2020.

[22] M. Jiang, Z. Wang, and Q. Dou, "Harmofl: harmonizing local and global drifts in federated learning on heterogeneous medical images," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 1, pp. 1087–1095, 2022.

[23] Y. Bai, T. Xie, N. Jiang, and Y. X. Wang, "Provably efficient q-learning with low switching cost," 2019, https://arxiv.org/abs/1905.12849.

[24] S. Lin, P. Zhou, X. Liang et al., "Graph-evolving meta-learning for low-resource medical dialogue generation," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 15, pp. 13362–13370, 2021.

[25] C. Angermueller, D. Dohan, D. Belanger, D. Ramya, M. Kevin, and C. Lucy, "Model-based reinforcement learning for biological sequence design," *International Conference on Learning Representations*, University of Waterloo, Waterloo, Canada, 2019.

[26] O. Gottesman, J. Futoma, Y. Liu et al., "Interpretable off-policy evaluation in reinforcement learning by highlighting influential transitions," in *Proceedings of the 37 th International Conference on Machine Learning*, pp. 3658–3667, Vienna, Austria, August 2020.

[27] G. Tennenholtz, U. Shalit, and S. Mannor, "Off-policy evaluation in partially observable environments," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 06, pp. 10276–10283, 2020.

[28] A. Rosenberg and Y. Mansour, "Oracle-efficient regret minimization in factored mdps with unknown structure," *Advances in Neural Information Processing Systems*, vol. 34, pp. 11148–11159, 2021.

[29] X. H. Liu, Z. Xue, and J. Pang, "Regret minimization experience replay in off-policy reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 34, pp. 17604–17615, 2021.

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

[31] B. T. Polyak and A. B. Juditsky, "Acceleration of stochastic approximation by averaging," *SIAM Journal on Control and Optimization*, vol. 30, no. 4, pp. 838–855, 1992.

[32] T. P. Lillicrap, J. J. Hunt, A. Pritzel et al., "Continuous control with deep reinforcement learning," 2015, https://arxiv.org/abs/1509.02971.

[33] M. Zinkevich, M. Johanson, and M. Bowling, "Regret minimization in games with incomplete information," *Advances in Neural Information Processing Systems*, vol. 20, 2007.

[34] N. Brown, L. Adam, S. Gross, and T. Sandholm, "Deep Counterfactual Regret Minimization," 2019, https://arxiv.org/abs/1811.00164.

[35] M. Hessel, J. Modayil, and H. Van Hasselt, "Rainbow: combining improvements in deep reinforcement learning," in *Proceedings of the 32nd AAAI conference on artificial intelligence*, Vienna, Austria, July 2018.