Resource Allocation Reinforcement Learning for Quality of Service Maintenance in Cloud-Based Services

Dupyo Hong
Department of Computing
Soongsil University
Seoul, Korea(South)
envydp99@soongsil.ac.kr

DongWan Kim
Department of Computing
Soongsil University
Seoul, Korea(South)
kk5339@soongsil.ac.kr

Oh Jung Min
Department of Computing
Soongsil University
Seoul, Korea(South)
jmin980317@soongsil.ac.kr

Yongtae Shin
Department of Computing
Soongsil University
Seoul, Korea(South)
shin@ssu.ac.kr

2023 International Conference on Information Networking (ICOIN) | 978-1-6654-6268-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICOIN56518.2023.10048905

***Abstract*—Recently, in order to improve the service quality of cloud-based services, research on a reinforcement learning model that predicts an appropriate amount of cloud resources by identifying patterns of user demands is being conducted. Reinforcement learning Q-learning algorithms rely on building a table (Q-table) for Q values, so if the state space and action space are vastly larger, they do not obtain optimal policies. In addition, learning errors in false experiences from the correlation of successive data in reinforcement learning may exist. In this paper, we study reinforcement learning modeling techniques that achieve higher accuracy than existing models by reducing the state definition space of hardware resources arising from services. It is possible to maximize service quality by allocating cloud resources or returning unnecessary resources with accurate resource demand prediction. For performance analysis, the service request prediction results according to the number of learnings were confirmed, and the service request prediction accuracy of three different models according to the neural network was compared. In the experiment, the model applying the proposed Convolutional Neural Network(CNN) neural network modeling technique is found to predict the amount of cloud resources in close proximity to the actual service request as the number of learning increases. We also compare the average of service request prediction accuracy of different models applying three neural networks, Deep Neural Network(DNN), Long Short-Term Memory(LTSM), and CNN, and find that the proposed technique has 3.36% higher prediction accuracy than LSTM-based models, and 40.2% higher than DNN-based models. In the future, additional research is needed, such as building various learning datasets or applying other reinforcement learning algorithms. Further research is also needed on cloud resource rental costs and provisioning latency.**

***Index Terms*—Resource Allocation, Reinforcement Learning, Deep Q Network, Cloud Computing**

I. Introduction

As non-face-to-face activities increased due to COVID-19, the content industry developed and the demand for online services surged. In order to provide a service with high satisfaction, the service providing server needs to satisfy an appropriate technical condition according to an increasing demand. This means that cloud infrastructure resources are needed to have computing resources suitable for large-scale traffic and provide smooth services. As a result, the size of the cloud market is expected to grow. In 2021, the private distribution sector dominated the global cloud computing market with a sales share of more than 46%. Private clouds provide computing services to a small number of non-public users over private networks or the Internet. In addition, the service-based Software as a Service (SaaS) sector accounted for more than 54% of sales share in 2021. The main features of SaaS that have allowed this occupancy are cost flexibility, ease of maintenance, and deployment [1]. The server that provides the service builds the system by renting a large number of servers, storage, and network equipment provided by cloud operators. When deployed in this way, service delivery servers can be provided with the necessary computing resources to reduce the cost of building a service infrastructure and increase efficiency and productivity by paying only as much as they use [2]. It also enables agile and flexible responses to rapidly changing business environments [3]. On the other hand, such cloud-based services can cause several problems if appropriate resource distribution is not achieved. When servers suddenly face unexpected traffic and workload spikes, the Quality of Service(QoS) can deteriorate if they do not have sufficient resources required accordingly [4]. Conversely, if there are more resources left than the demand, it may cause waste of resources, and the service provider may not benefit from the operation. In order to minimize this problem, research is being conducted to predict the appropriate amount of resources by learning reinforcement learning models by receiving hardware resource information such as CPU, storage, memory, and network bandwidth. Reinforcement Learning (RL) [5] can automatically make decisions by interacting with the environment without the support of historical data. Therefore, reinforcement learning has recently been used to solve the complex problem of resource allocation with low complexity. However, because traditional RL-based methods target static environments with workloads, this decision-making model must be retrained when workloads change [6]. Thus, these existing approaches cannot efficiently adapt to real-world scenarios of cloud-based software services with varying workloads and service requests [7]. In addition, if the data format used in reinforcement learning is continuous data, the correlation between the data is large, so when learning with continuous data such as time series, it is difficult to find an optimal action pattern because it is focused on a specific learning pattern at the beginning of the learning. Since the Q-learning algorithm relies on building a table (Q-table) for Q values, it will not obtain optimal policies if the state space and the action space become enormous. This case is common in Radio Resource Allocation and Management(RRAM) problems in modern wireless systems [8]. To address these problems, the Reinforcement Learning Deep Q-Network (DQN) algorithm uses learning methods using Experience-replay techniques. The experience-replay technique is not to learn immediately in the order in which input data comes in, but to store enough experience in memory and randomly select sample data to learn. This reduces the correlation between experiences. In this paper, we propose a modeling technique applying a reinforcement learning-based DQN algorithm that achieves higher accuracy than conventional RL methods by reducing the state definition space of hardware resources arising from cloud-based services. The proposed method allows for maximizing the QoS by allocating cloud resources or returning unnecessary resources with accurate resource demand prediction. For performance analysis, the service request prediction results according to the number of learnings were confirmed, and the service request prediction accuracy of three different models according to the neural network was compared. In the experiment, it is confirmed that the model applying the proposed technique predicts the amount of cloud resources close to the actual service request as the number of learning increases. In addition, we compare the average of service request prediction accuracy of different models applying deep neural network (DNN), long short-term memory (LTSM), and convolutional neural network (CNN) to find that the proposed technique has 3.36% higher prediction accuracy than LSTM-based models, and 40.2% higher than DNN-based models. The rest of this paper consists of the following. Chapter 2 examines reinforcement learning algorithms and reinforcement learning-based resource management technologies and studies related to dynamic action space adjustment reinforcement learning. Chapter 3 proposes a reinforcement learning modeling technique applying neural networks. Chapter 4 analyzes the performance of the proposed technique through experiments. Chapter 5 concludes this paper and proposes directions for future research.

II. Related Work

A. Reinforcement Learning Deep Q-Network(DQN) Algorithm

Q-Learning is a representative reinforcement learning algorithm that finds optimal policies that maximize overall rewards by determining actions that can be done in a particular state and granting corresponding rewards. The Q-Learning algorithm has several problems. In an environment where numerous actions exist for the current state, a massive increase in the Q-table to fit all possible actions would not result in optimal policy [8]. In addition, if the data format used for learning is continuous data, the correlation between the data is large, so when learning with data such as time series, it is difficult to find optimal action patterns because it is focused on specific learning patterns at the beginning of learning [9]. To address these problems, the Reinforcement Learning Deep Q-Network (DQN) algorithm uses a learning method using Experience-replay. The experience-replay technique is not to learn immediately in the order in which input data comes in, but to store enough experience in memory and randomly select sample data to learn. It replaces the Q-table with a DNN that attempts to approximate the Q value. In reinforcement learning, the proposed virtual environment-based DQN model [9] improves performance by about 63% over the results of the underlying DQN model to address learning errors in false experiences from the correlation of continuous data. Therefore, it is possible to find the best policy by reducing the correlation between experiences. In addition, to accurately predict the amount of requested resources for cloud-based services, research is being conducted to learn machine learning models by receiving hardware resource information such as CPU, storage, memory, and network bandwidth to identify patterns of user demand and predict the appropriate amount of resources [10][11].

B. Dynamic Action Space Adjustment Enhancement Learning Study

Traditional Q-learning requires a large amount of state space for learning. Although the impact of state space problems has recently been reduced through deep learning, there is still a problem with state space and training time limited, so a way to solve this problem is needed [12]. Q-learning learns the optimal policy through the action value function, which computes the cost incurred when a particular action is performed in the current state. In Q-learning, the set of all possible states is called the state space, and the set of all possible actions is called the action space. As mentioned earlier in Section 2.1, if the table of Q-learning becomes large, the optimal policy may not be obtained. Therefore, it is necessary to reduce the state space and the action space to learn with a minimum Q-table. The action space may be divided into a space used for learning and a space not used, thereby reducing an unused space. Actions that are not performed are not learned, so there is no need for space. State space can also be removed from the workspace when unused. It can also eliminate spaces that are not related to optimal action. Therefore, there is a need for a method of applying it to reinforcement learning and constructing rules by considering only actions related to optimal action [13]. Unperformed or unlikely optimal actions are not learned and state space is not allocated, so learning time can be shortened and state space can be reduced. Since the dynamic action space adjustment method requires only a small state space, high accuracy can be obtained. Woo and Sung [12] proposes a reinforcement learning method that dynamically adjusts the action space. According to the paper, even when the state space is reduced to about 0.33%, we show similar results to conventional Q-learning, showing that the cost and time required for the same amount of learning can be reduced.

III. The Proposed Reinforcement Learning Modeling Using Neural Network

This chapter defines the cloud-based service framework to which the proposed reinforcement learning-based modeling techniques will be applied and the parameter elements required for reinforcement learning (Section 3.1, Section 3.2).

A. Cloud-based service architecture for resource allocation

This paper aims to achieve higher accuracy than conventional RL methods by reducing the state definition space of hardware resources arising from cloud-based services. We construct a system as shown in Fig. 1 using a reinforcement learning model with DQN and dynamic action space adjustment techniques.

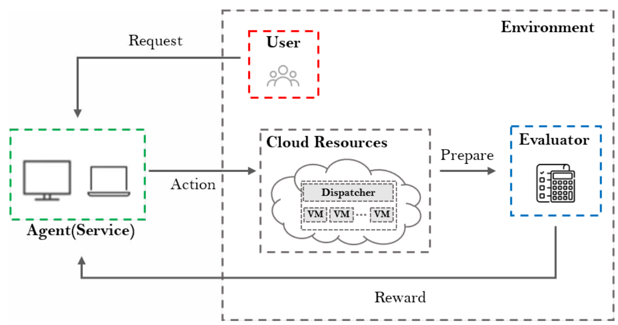

Fig. 1. Cloud-based service framework for resource allocation with reinforcement learning

The environment consists of an evaluator that evaluates whether the amount of cloud and calculated required resources is appropriate. A cloud-based service (Agent) is a subject that determines what action to take depending on the state. Action is a cloud resource management function. Request refers to the amount of resources required due to service use. The resources (Prepare) provided by the cloud include CPU, network bandwidth, storage, memory, etc. Reward refers to the gain gained when the Agent performs an action. Based on the determined action, the amount of resources held by the Agent and the amount of resources actually requested are compared to determine the success of provisioning and reward it.

B. DQN-based resource prediction model

Reinforcement learning trains the model by applying the Q-Learning algorithm. The algorithm is defined as an MDP element, and the MDP element includes state (S), action (A), compensation (R), and reduction rate (γ). Table 1 defines the MDP elements for reinforcement learning.

The operating hours of cloud-based services are t=0, 1, 2, ..., s_t is defined as a state(S) over time t, and S is a set of all possible states. The action taken in a state of s_t is called a_t, A is the set of all possible actions. Compensation(R) means a reward obtained when an action a_t is performed and is expressed as r_t.

TABLE I
Define of MDP Elements for Reinforcement Learning

Parameters	Definition
S	Set of all possible states {amounts of resource demanded by users }
A	Set of all possible actions {Alloc, Stay, Return}
R	Compansation to evaluate only current status and action
γ	The value of reward awarded

$$Q(s_t, a_t) = r_t + maxQ(s_{t+1}, a_{t+1}) \tag{1}$$

$Q(s_t, a_t)$, Eq. (1), refers to the value of an action when an action is taken in a state a_t over a specific time t, and $maxQ(s_{t+1}, a_{t+1})$ means the greatest value that obtains among the actions that can be taken in the next state. In Eq. (2), during reinforcement learning, the value of a_t changes dynamically. Since DQN algorithms are mainly used to learn optimal policies, the optimal Q function is derived from the following recursive Bellman equation.

$$\begin{aligned} Q(s_t, a_t) \leftarrow\ & (1-\alpha)Q(s_t, a_t) \\ & +\alpha(r_t + maxQ(s_{t+1}, a_{t+1})) \end{aligned} \tag{2}$$

Both α and γ are constants, values between 0 and 1, α is a learning rate, which means how the new value for the Q value weighs more than the previous value, and γ is a depreciation rate, which means the consumption rate of value. The proposed method not only reduces the state space but also achieves higher accuracy by reducing the number of selectable actions.

$$\begin{aligned} & Loss(\theta_t) = r_{t+1} \in R[r_t + \\ & \gamma \max Q(s_{t+1}, a_t|\theta') - Q(s_t, a_t|\theta)]^2 \end{aligned} \tag{3}$$

And the DQN algorithm is optimized by iteratively updating the weights θ of the DNN to minimize the next Bellman loss function (Eq. (3)). θ' is the weights of the target Q network. The Agent stores enough experience in memory and randomly selects it to learn.

IV. Experiments

This chapter introduces a series of processes for verifying the proposed method. The proposed method is applied in a simulation environment that imitates a real-world environment. We also examine the application process of the proposed method and analyze the experimental results.

A. Environment Introduction

Experiments utilize RLtrader, an open source related to reinforcement learning, to implement a simulator that learns data and predicts cloud resource volumes of cloud resources. Table. 2 shows an environment constituting a simulator.

TABLE II
DEVELOPMENT ENVIRONMENT OF REINFORCEMENT LEARNING SIMULATOR

Type	Environment
Operating System	Windows 10
CPU	Intel(R) Core(TM) i7-9700K 3.60 GHz
GPU	NVIDIA GeForce RTX 2060
RAM	32GB
Programming Language	Python
Library	Tensorflow Keras

B. Experimental Results

In this paper, service request data of real users generated from cloud-based services were used to analyze the performance of the proposed technique. The proposed technique stores values for the following states over time, and compares the proposed models using DNN, LTSM, and CNN for neural networks.

As described in Chapter 3, the proposed model aims to predict close to the actual service request by assigning and releasing the amount of cloud resources according to the user's service request based on the reinforcement learning DQN algorithm. Fig. 2 shows service request data generated in an actual cloud-based service environment. The X-axis means time, and the Y-axis means service request amount.

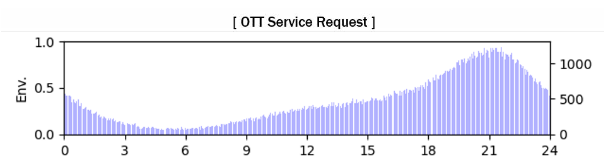

Fig. 2. Cloud-based service request data

Fig. 3 shows the results of the model applying the reinforcement learning DQN algorithm allocating and releasing cloud resources while repeating the learning process using the actual service request dataset. The X-axis means time, and the Y-axis means the amount of allocated cloud resources.

The prediction result of cloud resource allocation changes according to the number of learnings, and it can be seen that the prediction result does not meet the actual cloud-based service request at the beginning. However, as the number of epochs increased, it was confirmed that cloud resources were allocated in a form similar to cloud service requests. It was confirmed that they were allocated and released in the form

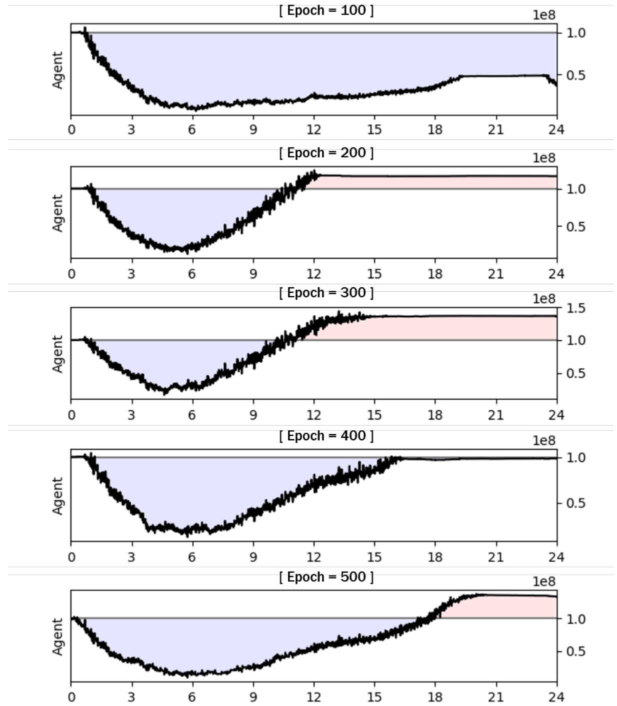

Fig. 3. Cloud Resource Allocation Prediction Performance Applied Reinforcement Learning DQN Algorithm

closest to the actual service request at 500 due to sufficient epoch.

Fig. 4 shows the results of comparing the service request prediction accuracy of different models according to the neural network.

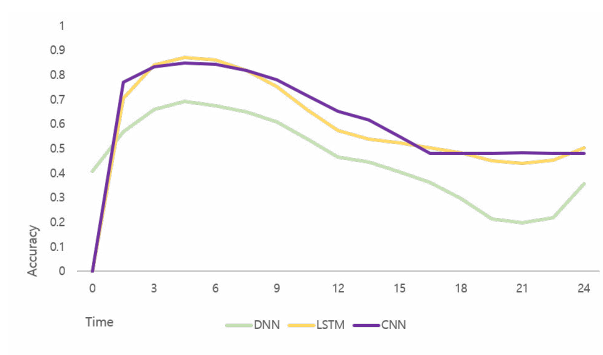

Fig. 4. Accuracy of predicting service requests of models according to neural networks

Different models according to the neural network used a set of service request data to derive service request prediction results after a learning process, and measure the prediction accuracy over time. The X-axis represents time and the Y-axis represents prediction accuracy as values between 0 and 1. While high prediction accuracy was measured in all models during times of low service request, models with CNN neural networks were measured to have the highest accuracy during times of rapid service request increase. The average of the prediction accuracy of the model is 0.645, 0.624, and 0.460, respectively, in the order of CNN, LSTM, and DNN, and it is confirmed that the model applying the CNN neural network has a higher prediction accuracy average by 3.36% than the model of LSTM and 40.2% than the model of DNN.

V. Conclusion

In this paper, we propose a modeling technique that applies a reinforcement learning-based DQN algorithm that achieves higher accuracy than conventional RL methods by reducing the state definition space of hardware resources arising from cloud-based services.

In order to verify the performance of the proposed technique, the performance was compared with the existing method. In the experiment, the resource prediction performance between the proposed techniques in the cloud environment was analyzed using a simulator configured based on Python. For performance analysis, the service request prediction results according to the number of learnings were confirmed, and the service request prediction accuracy of three different models according to the neural network was compared.

The first experiment, cloud resource allocation prediction based on the number of learnings, initially shows that cloud resources are allocated and deallocated in a form close to the actual service request when the epoch is sufficiently advanced. In the second experiment, comparing service request prediction accuracy of three different models according to the neural network, high prediction accuracy was measured in all models at low service request times, while the CNN neural network was the most accurate at times of rapid service request. Based on these experimental results, it was confirmed that the technique proposed in this paper is a resource allocation method that meets user requirements and guarantees service quality.

Depending on the nature of the reinforcement learning method, the predicted cloud resources and actual required resources will not match, but it is expected that the predicted performance of the service request will be improved by reducing the number of selectable actions based on accumulated data and feedback information over time.

Finally, we propose a direction for future research. It was confirmed that the reinforcement learning modeling technique proposed in this paper has limitations in obtaining high prediction accuracy during times when service requests increase rapidly. Therefore, further research is needed, such as building various learning datasets or applying other reinforcement learning algorithms. Further research is also needed on cloud resource rental costs and provisioning latency.

Acknowledgment

“This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2021-2020-0-01602) supervised by the IITP(Institute for Information & Communications Technology Planning & Evaluation)”

”This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the National Program for Excellence in SW)(2018-0-00209) supervised by the IITP(Institute of Information & communications Technology Planning &Evaluation)”

References

[1] Precedence Research, ”Cloud Services Market Size to Surpass US$ 1630 Billion By 2030”,GlobeNewswire News Room, 2022. [Online]. Available: https://www.globenewswire.com/news-release/2022/06/10/2460529/0/en/Cloud-Services-Market-Size-to-Surpass-US-1630-Billion-By-2030.html. [Accessed: 10- Jun- 2022].

[2] W. Liu, P. Wang, Y. Meng, C. Zhao and Z. Zhang, ”Cloud spot instance price prediction using kNN regression”,Human-centric Computing and Information Sciences, vol. 10, no. 1, 2020. Available: 10.1186/s13673-020-00239-5.

[3] Y. Lee, H. Jang, S. Oh, Y. Lee, Y. Yoon and T. Um, ”Deep Reinforcement Learning-Based Cloud Resource Allocation for Cost-Effective Over-the-Top Services”,Journal of Digital Contents Society, vol. 23, no. 3, pp. 531-540, 2022. Available: 10.9728/dcs.2022.23.3.531.

[4] N. Luong, P. Wang, D. Niyato, Y. Wen and Z. Han, ”Resource Management in Cloud Networking Using Economic Analysis and Pricing Models: A Survey”,IEEE Communications Surveys & Tutorials, vol. 19, no. 2, pp. 954-1001, 2017. Available: 10.1109/comst.2017.2647981.

[5] R. Sutton, F. Bach and A. Barto,Reinforcement Learning. Massachusetts: MIT Press Ltd, 2018.

[6] A. Orhean, F. Pop and I. Raicu, ”New scheduling approach using reinforcement learning for heterogeneous distributed systems”,Journal of Parallel and Distributed Computing, vol. 117, pp. 292-302, 2018. Available: 10.1016/j.jpdc.2017.05.001.

[7] X. Chen, F. Zhu, Z. Chen, G. Min, X. Zheng and C. Rong, ”Resource Allocation for Cloud-Based Software Services Using Prediction-Enabled Feedback Control With Reinforcement Learning”,IEEE Transactions on Cloud Computing, vol. 10, no. 2, pp. 1117-1129, 2022. Available: 10.1109/tcc.2020.2992537.

[8] A.Alwarafy, M. Abdallah, et al. Deep reinforcement learning for radio resource allocation and management in next generation heterogeneous wireless networks: A survey. arXiv preprint arXiv:2106.00574, 2021.

[9] S. Lee, B. On, G. Choi, S. Yi. “DQN model based on virtual environment for improving high score convergence speed in reinforcement learning games.”.Proceedings of KIIT Conference, pp. 472-475. 2020.

[10] Y. Kim, S. Kim and H. Lim, ”Reinforcement Learning Based Resource Management for Network Slicing”,Applied Sciences, vol. 9, no. 11, p. 2361, 2019. Available: 10.3390/app9112361

[11] Y. Kim and J. No, ”Resource Allocation based on Virtual Machine Resource Requirement Estimation in Cloud Computing Environment”,Journal of the Institute of Electronics and Information Engineers, vol. 54, no. 11, pp. 113-121, 2017. Available: 10.5573/ieie.2017.54.11.113.

[12] S. Woo and Y. Sung, ”Dynamic Action Space Handling Method for Reinforcement Learning Models”,Doi.org, 2022. [Online]. Available: https://doi.org/10.3745/JIPS.02.0146.

[13] A. Raghu, M. Komorowski, L. Celi, P. Szolovits and M. Ghassemi, ”Continuous State-Space Models for Optimal Sepsis Treatment: a Deep Reinforcement Learning Approach”,PMLR, 2022. [Online]. Available: https://proceedings.mlr.press/v68/raghu17a.