Smart Media Journal / Vol.10, No.2 / ISSN:2287-1322
https://dx.doi.org/10.30693/SMJ.2021.10.2.48

Application of Deep Recurrent Q Network with Dueling Architecture for Optimal Sepsis Treatment Policy

Thanh-Cong Do*, Hyung Jeong Yang**, Ngoc-Huynh Ho*

Abstract

Sepsis is one of the leading causes of mortality globally, and it costs billions of dollars annually. However, treating septic patients is currently highly challenging, and more research is needed into a general treatment method for sepsis. Therefore, in this work, we propose a reinforcement learning method for learning the optimal treatment strategies for septic patients. We model the patient physiological time series data as the input for a deep recurrent Q-network that learns reliable treatment policies. We evaluate our model using an off-policy evaluation method, and the experimental results indicate that it outperforms the physicians' policy, reducing patient mortality up to 3.04%. Thus, our model can be used as a tool to reduce patient mortality by supporting clinicians in making dynamic decisions.

Keywords : Sepsis | Reinforcement Learning|Deep Learning | Deep Recurrent Q Network

Ⅰ. INTRODUCTION

Sepsis is a serious infection that leads to life-threatening acute organ dysfunction [1]. It is one of the leading causes of patient mortality globally, and it costs billions in hospital care annually. However, there are some factors that make it difficult to decide among the available treatment methods for sepsis: First, it is a medical emergency that requires rapid intervention from a physician [2]. Secondly, there is no exact definition of the effective treatment for sepsis in every case, because each patient can respond differently to the medication. In practice, each individual physician administers treatment differently. Recent research has shown that in addition to antibiotics and infection source control, the specific administration of intravenous fluids and vasopressors also plays a key role in clinical interventions [3]. However, the specific amounts of these two medicines given to a patient are mostly based on the experience of the clinician, and there are no reliable guides. This can lead to negative side effects in some portion of patients [4].

Over the past decade, considerable advancements have been made in machine learning for use in the medical field, particularly in diagnosing conditions and forecasting the outcomes of patients. To date, there has not been much research focusing on treatment. For some specific diseases, such as sepsis, finding the optimal treatment is extremely important but highly challenging. Reinforcement learning (RL) can be considered the best candidate to address this problem because of its effectiveness in many sequential decision-making problems. However, in many real-world problems, it is costly or impossible to create an environment in which to train RL agents. In such cases, it is necessary to learn from observational historical data, which is the idea behind off-policy reinforcement learning. "Off-policy" means that the agent learns the optimal policy from another available policy. Our agent is trained to learn the optimal treatment policy from the

*Student Member, Graduate Student, **Member, Professor, Dept. of AI Convergence, Chonnam National University
* This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT). (NRF-2020R1A2B5B01002085).
* This work was supported by the Bio & Medical Technology Development Program of the National Research Foundation (NRF)& funded by the Korean government (MSIT) (NRF-2019M3E5D1A02067961).

Manuscript : 2021. 03. 04
Confirmation of Publication: 2021. 05. 25
Corresponding Author: Hyung-Jeong Yang
e-mail: hjyang@jnu.ac.kr

physicians, then improves upon it.

In this work, we propose an approach for discovering the optimal treatment policies for sepsis based on a historical time-series dataset of sepsis patients. Our method involves combining a variational auto-encoder (VAE) with a deep recurrent Q-network with dueling architecture (DDRQN), which is an extension of the dueling deep Q-network (DDQN) [5], which can memorize past states and actions. Considering that the previous state-action pairs can affect the decision at the current time-step, we apply LSTM layers instead of fully connected neural network layers in the DDQN architecture to capture the long-term memories, thereby improving the performance of the RL agent.

The rest of this paper is organized into five sections. Section II discusses the background and related work. Section III describes how to set up the RL environment for the sepsis treatment problem. Next, our proposed method for finding an optimal treatment policy is shown in Section IV. The experimental results are provided in Section V. Finally, the conclusions and future work are presented in Section VI.

II. BACKGROUND AND RELATED WORK

To learn the optimal policy from the observational historical data, one popular algorithm in RL is Q-learning. Recently, with the growth of deep learning (DL) techniques, a lot of research has successfully combined RL with DL into a method called deep reinforcement learning. The approach which applies the Q-learning algorithm along with DL is known as the Deep Q Network (DQN). The idea of this method is to use a deep neural network that takes the states as input, then estimates the action-value function (Q-value function) for every possible action. The Q-value of a state-action pair (s, a) corresponds to the expected cumulative reward from taking an action "a" at state "s":

$$Q_{(s,a)} = E[\sum_{t\geq 0} \gamma^t r_t] \quad (1)$$

Here, γ is the discount factor, which captures the trade-off between immediate and future rewards. Then, the optimal policy is specified by choosing the action with the highest Q-value at each state. Figure 1 shows the simple idea behind DQN.

The original DQN algorithm is known to have some

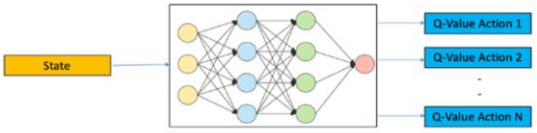

Fig. 1. A Deep Q Network

weaknesses, which are overcome in the Dueling Deep Q Network (DDQN). In this model, the Q-value is split into two streams: the value function (which determines how good the current state is) and the advantage function (which represents the quality of the chosen action). The advantage of this splitting strategy is that it can evaluate the potential of the current state without having to learn the effect of each action on it. In some specific cases, actions might not always affect the environment in meaningful ways. For example, for states in which the patient is almost dead no matter what action is taken, trying to find the optimal action may waste time and make the model difficult to converge. A recent research has applied DDQN to sepsis treatment and achieved positive results [6]. However, it is shown that sometimes the DDQN model can select a dangerous action which rarely or never performed by clinicians [7]. Therefore, in [7], they proposed a method of combining DRL-based method with kernel-based which keeps the model stay close to the data, removing some dangerous actions.

As our dataset in this research included a lot of information without a clear structure, it is difficult for machine learning models to learn it. To overcome this difficulty, we apply a variational autoencoder (VAE) [8] that generates the latent state representation for the patients states. One limitation of a common autoencoder is that it is trained to ensure that the encoding and decoding process involves as little loss as possible without giving any consideration to how the latent space is organized. To overcome this limitation, VAE introduces some regularization terms during the training process to avoid over-fitting and ensure that the latent space has good properties for the generative process.

Similar to a standard autoencoder, a VAE is an architecture that includes both an encoder and a decoder that are trained to minimize the reconstruction error. However, rather than encoding an input as a single point, VAE encodes it as a

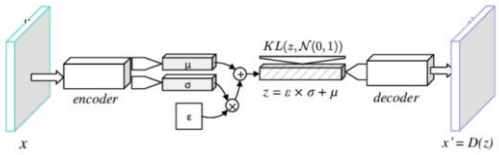

Fig. 2. A Variational Autoencoder

distribution over the latent space. Next, from that distribution, a point from the latent space is sampled and then decoded to compute the reconstruction loss. Finally, the reconstruction error is back propagated through the network. The VAE architecture is shown in Figure 2. The regularization term here is that it enforces the distributions returned by the encoder to be close to a standard normal distribution. Therefore, the loss function is composed of common reconstruction loss and a regularization term, which is the Kulback–Leibler divergence between the returned distribution and a standard Gaussian distribution:

$$L = |x - x'|^2 + KL[N(\mu_x, \sigma_x), N(0, I)] \quad (2)$$

III. SETTING UP ENVIRONMENT FOR REINFORCEMENT LEARNING

In this work, we use observational historical data extracted from the Multiparameter Intelligent Monitoring in Intensive Care Unit (MIMIC–III v1.4) database [9]. This is a large, public, critical care database that contains health data of more than 60,000 Intensive Care Unit (ICU) admissions. The database includes demographics, laboratory test values, vital signs, medications, and more. For our research purposes, we focus on patients who fulfill the sepsis–3 criteria [1]; this is defined as a suspected infection combined with organ dysfunction, which is specified by a Sequential Organ Failure Assessment (SOFA) score greater than or equal to 2.

The exclusion criteria for our final dataset include patients younger than 18 years old as well as patients with medicine intake or mortality that were not documented. For each patient, data are aggregated into multidimensional discrete windows with 4–hour time steps. Some specific features with multiple measurements within 4 hours are either summed (such as urine output) or averaged (such as heart rate). After the extraction and preprocessing steps, the final dataset includes 17,928 patients.

The MIMIC–III v1.4 dataset is stored in Postgres SQL and queried using pgAdmin 4. Some information about the resulting cohort is listed in Table 1. The data are divided into 70% for the training set, 10% for validation, and 20% for the testing set.

Table 1: Some information about the final dataset

	% Female	Mean Age	Total Population
Survivors	42.7	62.5	15,463
Non–Survivors	48.1	69.2	2,465

To define the states, for each patient, we extracted 42 physiological features which we believed to be the important factors that should be examined by physicians when deciding upon the dosage for patients. These features contain information on demographics, vital signs, laboratory test results, and intake/output events. The features were converted into multidimensional time series data with a time interval of 4 hours. The list of used features is presented in Table 2.

Table 2. 42 Physiological features used to represent each patient's state.

Group	Features
Demographics	Shock Index, Elixhauser, SIRS, Gender, GCS, SOFA, Age
Lab Values	Albumin, Arterial pH, Calcium, Glucose, PTT, Potassium, SGPT, Arterial Blood Gas, BUN, Chloride, INR, Sodium, Arterial Lactate, CO2, Creatinine, Ionised Calcium, PT, Platelets Count, SGOT, Total bilirubin, White Blood Cell Count
Vital Signs	Diastolic Blood Pressure, Systolic Blood Pressure, Mean Blood Pressure, PaCO2, PaO2, FiO2, PaO2/FiO2 ratio, Respiratory Rate, Temperature, Heart Rate, SpO2, HCO3
Intake and Output Events	Fluid Output – 4 hourly period, Total Fluid Output

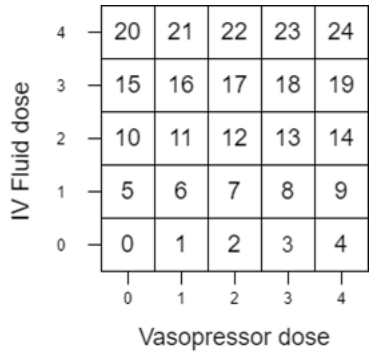

Fig. 3: An action space

In this work, we follow a previous study that defined a discrete action space based on a combination of two popular drugs for sepsis: intravenous fluids and vasopressors [10]. For each medicine, we divided the dosage into five levels, from 0 to 4; note that we included the case in which no drug was given as 0. The action space included all possible dosage combinations of these two kinds of drugs. Therefore, there are 25 actions in total, which we converted into integers ranging from 0 to 24, as shown in Figure 3.

In many real-world problems, it is extremely difficult to build a reward system for an RL model. In some contexts, such as video and board games, reward is based mostly on the terminal state, which corresponds to winning or losing the game. In these cases, the RL agent can play the game again and again many times, where the only target is winning. Thus, there is not much attention given to feedback of intermediate states. However, for some specific fields, such as the healthcare and medical field, setting only one reward at the end (survival or death) can lead to a lot of potential danger for patients, because a bad action in any step can lead to significant harmful effects. Further, in the context of treating septic patients, it is difficult to find a reward function that balances immediate improvements with future long-term success. The task of defining reward function in such cases typically requires advice from domain experts.

In our work, for the final time steps, we issued a positive reward of +15 if the patients survived during their stay in the hospital, and a penalty of −15 if they did not. In terms of the intermediate states, we followed prior research to define the reward based on changes in SOFA score (measuring patients' organ failure) and lactate levels (measuring the cell-hypoxia that is often higher in sepsis patients) [6]. If there were decreases in these metrics between two consecutive states, we gave a positive score, while we applied a penalty if both metrics increased. Our reward function for intermediate time steps is shown as:

$$r(s_t, s_{t+1}) = C_1(s_{t+1}^{SOFA} - s_t^{SOFA}) + C_2 \tanh\,(s_{t+1}^{lactate} - s_t^{lactate}) \quad (3)$$

We opted to use C1 = −0.125 and C2 = −2; these values were selected to ensure that the values of the intermediate rewards did not exceed the final reward (±15). It should be noted that we also gave a reward of −0.025 if the SOFA scores did not change after moving to the next state. The final reward value at the end of each patient's trajectory was selected as ±15, because the highest SOFA score value is 15. Patients who reach that score are almost dead.

IV. PROPOSED METHOD

In this paper, we proposed a method of using deep reinforcement learning to find the optimal treatment policies for septic patients. First, we apply a VAE to preprocess and denoise the physiological states. Then, the latent states space from VAE is used as the input for our model, the Deep Recurrent Q Network with dueling architecture, to estimate the Q-value for every state-action pair. The optimal policy is specified by following the action with the highest Q-value at each state. The overall workflow of our proposed method is illustrated in Figure 4.

In this work, VAE was used to encode 42 patient features into a smaller, more compact state representation of 30 features which are used as input for the RL model. By doing so, we can simplify the learning problem and exclude redundant information from the input data. We implemented the VAE with three dense down-sampling layers in the encoder, sampling via re-parameterization, and three up-sampling layers in the decoder.

In this research, we use an extension of DQN, the deep recurrent Q-network (DRQN) [11]. DQN has proven to be unable to master the problems that require agents to remember events that are far from the past. By contrast, DRQN, which combines a Long Short Term Memory (LSTM) with a DQN, handles the loss information better than the standard DQN. In our model, we apply the DRQN with the dueling structure (DDRQN) while using two LSTM layers

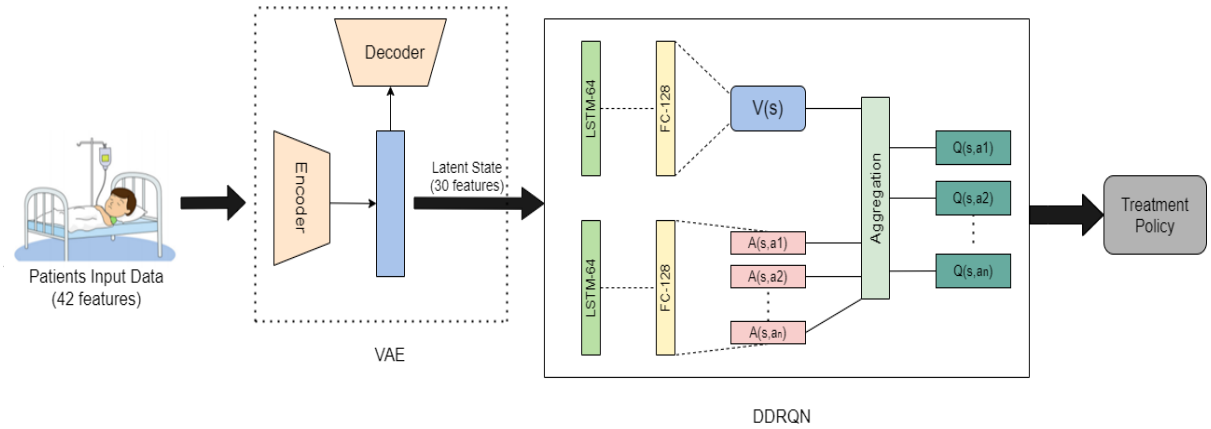

Fig. 4. Overall workflow of our proposed method

that feed to two fully connected layers, before splitting into the value stream and the advantage stream, which are finally aggregated to obtain the Q-value. The overall workflow of our proposed method is shown in Figure 4. The final loss function is similar to the common loss function of DDQN:

$$L(\theta) = (Q_{double-target} - Q(s,a;\theta))^2 \quad (4)$$

With:

$$Q_{double-target} = r + \gamma Q(s', argmax_{a'} Q(s', a'; \theta); \theta') \quad (5)$$

where θ are the weights used to parameterize the main network and θ' are the weights used to parameterize the target network.

V. EXPERIMENTAL RESULTS

Our proposed model uses two LSTM layers with 64 hidden nodes before going through two fully connected layers with 128 nodes. We implemented DDRQN in Tensorflow while using an Adam optimizer, batch normalization, and the Leaky-ReLU activation function. The model was trained for 200,000 steps with a batch size of 32. We opted to use the discount factor γ =0.99 to capture the influence of far future rewards.

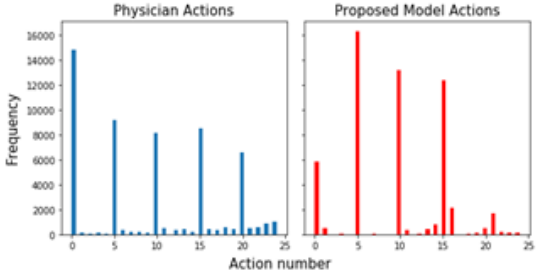

Fig. 5. Action distribution on the test set

To ensure the confidence of our proposed method, we examined the frequency of each action on the testing set following our model's policy, then compared it with that of the physicians' policy. Figure 5 shows the action distribution of the physicians' policy and our model's policy on the testing data. As shown in the figure, clinicians most often use actions 0, 5, 10, and 15. This trend is also reflected in the policies learned by our model. This proves that the policy derived from our proposed method is clinically interpretable and reliable. The main difference between our model and the physicians' policy is that our model suggested action 5 most of the time, while physicians most often prefer action 0 (meaning no drug is given).

In this paper, we applied a method of Doubly Robust Off-policy Evaluation to estimate the expected return of each policy [12]. For each patient trajectory H, we estimated the value of the learned policy V_{DR}^{H} using the following recursion:

$$V_{DR}^{H-t+1} = V(s_t) + \rho_t(r_t + \gamma V_{DR}^{H-t} - Q(s_t, a_t)) \quad (6)$$

then, we averaged the value obtained across all trajectories to obtain the results. Note that H is the length of a trajectory and that $V(s_t)$ is the value function at state s_t, which measures the expected cumulative reward from state s_t.

In this work, the target of our model is to find the optimal treatment policy that can reduce the risk of mortality of septic patients. To understand how the expected discounted returns are related to mortality, we obtain the Q-values of physicians' actions to

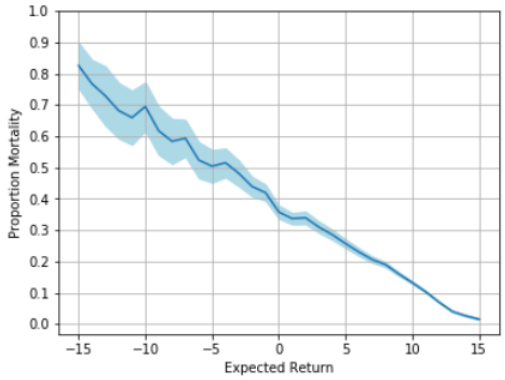

Fig. 6: Expected return versus mortality

produce an empirically derived function of mortality versus expected return, as we know the real mortality of patients when following the physicians' policy. Then, we compute the proportions of mortality among the different methods to compare the performance of our model with those of others. Figure 6 shows the inverse relationship between them, which indicates that our proposed method can decrease patient mortality. Table 3 presents the comparison of the performance of our proposed model (DDRQN) with those of other methods. Note that we test our model with two options: with and without VAE. The result indicates that our proposed method outperforms others, ultimately reducing patient mortality by 3.04% compared to that of the physicians', and that VAE also slightly improves the DDRQN.

Table 3. Comparison among different methods

Policy	Expected return	Estimated percent of mortality
Physician	9.67	**14.08 ± 0.5%**
DDQN [10]	10.16	**12.57 ± 0.4%**
MoE [12]	11.04	**12.14 ± 0.5%**
DDRQN	11.23	**11.68 ± 0.4%**
VAE-DDRQN	**11.87**	**11.04 ± 0.4%**

V. CONCLUSION

In this paper, we proposed a method of improving sepsis treatment policies by using the deep recurrent Q-network with dueling architecture technique. Our model was proven to be clinically interpretable, and it outperforms the physicians' policy in some extents based on the Doubly Robust estimator. The proposed framework can be used as a clinical decision-supporting tool for physicians to deal with the problem of rapid intervention for septic patients. However, off-policy evaluation is a complex problem that requires more research and guarantees. For future work, we plan to focus on building an environment that predicts the next state of a patient when given the current state and an action. As off-policy evaluation methods have some specific limitations, building a simulator of patient trajectories is necessary and important for evaluating the performance of RL models.

REFERENCES

[1] M. Singer, et al. "The third international consensus definitions for sepsis and septic shock (sepsis-3)," *J. Am. Med. Assoc.* Vol. 315, No. 8, pp. 801-810, Feb. 2016

[2] Christopher W. Seymour, et al. "Time to treatment and mortality during mandated emergency care for sepsis," *New England Journal of Medicine,* Vol. 376, No. 23, pp. 2235-2244, Jun. 2017

[3] L. Byrne, V. Haren, "Fluid resuscitation in human sepsis: time to rewrite history?" *Ann Intensive Care,* Vol. 7, No. 4, Jan. 2017

[4] P. Marik, R. Bellomo, "A rational approach to fluid therapy in sepsis," *Br. J. Anaesth.* Vol. 116, No. 3, pp. 339-349, Mar. 2016

[5] Z. Wang, N. de Freitas, and M. Lanctot, "Dueling network architectures for deep reinforcement learning," arXiv:1511.06581, Nov. 2015

[6] A. Raghu, M. Komorowski, I. Ahmed, L. Celi, P. Szolovits, and M. Ghassemi, "Deep Reinforcement Learning for Sepsis Treatment," *CoRR,* abs/1711.09602, 2017

[7] X. Peng, Y. Ding, D. Wihl, O. Gottesman, M. Komorowski, L.H. Lehman, Ross, A. Faisal A, F. Doshi-Velez, "Improving Sepsis Treatment

Strategies by Combining Deep and Kernel-Based Reinforcement Learning," *AMIA Annu Symp Proc,* pp. 887-896, Scottsdale, USA, Dec. 2018

[8] Diederik P Kingma, Max Welling, "Auto-encoding variational bayes," *CoRR,* abs/1312.6114, May 2014

[9] E.W. Johnson, Tom J. Pollard, Lu Shen, Li-wei H. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L.A. Celi, and R.G. Mark. "MIMIC-III, a freely accessible critical care database," *Scientific Data*, V0l.3, No.160035, May 2016

[10] A. Raghu, M. Komorowski, I. Ahmed, L. Celi, P. Szolovits, and M. Ghassemi, "Continuous state-space models for optimal sepsis treatment - a deep reinforcement learning approach," *Proceedings of the 2nd Machine Learning for Healthcare Conference*, pp. 147-163, Northeastern University, USA, Aug. 2017

[11] M. Hausknecht and P. Stone, "Deep recurrent q-learning for partially observable mdps," *CoRR,* abs/1507.06527, Jul. 2015

[12] N. Jiang and L. Li, "Doubly Robust Off-policy Evaluation for Reinforcement Learning," *Proceedings of The 33rd International Conference on Machine Learning*, Vol. 48, pp. 652-661, NY, USA, Jun. 2016

Authors

Thanh-Cong Do

He received his B.S. degree in Information Technology from University of Engineering and technology, Vietnam National University (UET-VNU) in 2019. He is currently pursuing the M.S. degree in the School of Artificial Intelligence Convergence at Chonnam National University, South Korea. His research interest includes Reinforcement Learning, Deep Learning and Bioinformatics.

Hyung-Jeong Yang

She received her B.S., M.S. and Ph.D. from Chonbuk National University, Korea. She is a professor at the Department of Artificial Intelligence Convergence, Chonnam National University, Gwangju, Korea. Her main research interests include multimedia data mining, Medical data analysis, Social Network Service data mining and Video data understanding.

Ngoc-Huynh Ho

He received the B.S. degree in the Department of Telecommunication Engineering from Ho Chi Minh City University of Technology, Vietnam, in 2015, the M.S. degree in the School of Electricals & Electronics Engineering from Kookmin University, South Korea, in 2017, and the Ph.D. degree in the Department of Artificial Intelligence Convergence at Chonnam National University, South Korea. He is currently a postdoctoral researcher in Chonnam National University. His research interest includes the multimodal-based emotion recognition, machine learning, deep learning and its applications, and bioinformatics.