Enhancing Diabetic Retinopathy Management Through Personalized Recommendations with Reinforcement Learning

H.M.A.I.B Herath, P.V.Thushan, T. F. Samoon, H.G.D.Sandakalum

BSc (Hons) in Information Technology, Faculty of Computing, Sri Lanka Institute of Information Technology, Malabe, Sri Lanka

ABSTRACT

Diabetic retinopathy (DR) is one of the main causes of vision impairment among diabetes patients, which leads to the requirement of effective management strategies to stop its progression. In traditional management methods, this process is often less personalized, which results in poor dietary and exercise practices among patients. This paper presents a mobile application, "Retina-Care," created to give out personalized dietary and exercise recommendations using Q-learning – a method of reinforcement learning. This proposed system dynamically learns and adapts to individual user behaviors, which leads to users engaging with the system more and adhering to the recommendations given. The paper addresses the cold start problem which can be seen in reinforcement learning systems, through a combination of epsilon-greedy methods, initialized Q-tables, and integration of an expert system. The results received through simulations and user testing indicate that the proposed strategy was successful in giving recommendations that were suitable to the unique medical conditions and preferences of the diabetic retinopathy patients. The said findings show that reinforcement learning can be used to enhance the effectiveness of diabetic retinopathy management systems, offering a promising solution for broader healthcare applications.

Keywords : - *Recommendation system, Q learning, Diabetic-Retinopathy, Reinforcement-learning, machine learning*

---

1. INTRODUCTION

Diabetic retinopathy stands as a prominent threat to vision health among diabetic patients, constituting a leading cause of vision loss globally. The prevalence of diabetes is escalating, and with it, the incidence of diabetic retinopathy is on the rise, affecting millions of individuals worldwide. The ocular complications associated with diabetes, specifically retinopathy, underscore the critical importance of early detection and intervention to prevent irreversible vision impairment.

When diabetic patients are Diagnosed with Diabetic Retinopathy, management of Diabetic Retinopathy so that it is ensured that the condition doesn’t rapidly worsen is essential for the patients. On most occasions, we can see Diabetic Retinopathy patients without proper knowledge engaging in bad dietary and exercise habits which results in patients facing vision impairment and requiring surgery in an untimely manner. Signifying the importance of diabetic Retinopathy Management.

Diabetic Retinopathy management usually involves three aspects: Medications, Diet and Exercise. First and foremost, Diabetic retinopathy patients are required to obtain the medication signed by the doctors. Additionally, retinopathy patients are also required to adhere to a dietary plan which ensures that their blood sugar levels stay low. Consuming foods with low glycemic index is encouraged. Finally, diabetic retinopathy patients are advised to get the required exercise to make their body more sensitive to insulin which is the hormone that allows cells in your body to use blood sugar for energy.

Traditional management methods of diabetic retinopathy can often be less effective. Practitioners usually give out generic instructions and pointers towards healthy dietary practices and exercises. With limited screening time it is challenging to the doctors to guide patients personally towards their dietary restrictions and exercise goals. These limitations have led to a pressing need of innovative, efficient, and reliable solutions to manage Diabetic Retinopathy and give out personalized recommendations to diabetic retinopathy patients. The proposed solution plans to use reinforcement learning to present personalized recommendations to diabetic retinopathy patients.

This paper is focused on the recommendation system of the software we developed named “Retina-Care” where diabetic retinopathy patients would receive personalized dietary and exercise recommendations through the system.

Thereby, this research paper introduces a Dietary and Exercise recommendation system using Q learning. Q learning is a prominent method in reinforcement learning that estimates action – values. This paper focuses on adapting Q learning on the said context, solving the cold start problem and development of the proposed solution. This work addresses the recommendation challenge as a step-by-step decision problem, offering a comprehensive and effective way to give out personalized recommendations.

The remainder of this paper is organized as follows: Background and literature review section dives into the theoretical aspects employed in this study, and then examines existing recommendation systems while comparing them with our own system. The Methodology section explains the development process of the recommendation systems of the Retina-care application, along with a detailed explanation of how the proposed solution effectively addresses the research problem. The results and discussion section shows the results obtained by user testing and discuss them. Finally, the Conclusion provides a brief explanation on the findings and insights obtained within the paper.

2. BACKGROUND & LITERETURE REVIEW

In the past, recommendation problems were viewed as prediction or classification tasks. However, there is now a widely accepted perspective that framing them as sequential decision problems can better capture the interactions between users and systems. This approach rephrases the problem as a Markov decision process (MDP), making it suitable for reinforcement learning (RL) techniques.

Traditional recommendation systems filter their contents and products based on three main filtering techniques [2]. Those three different approaches are as follows: Rule-based filtering- This approach is much close to customization. User needs to identify themselves and configure their individual settings to maintain their environment on time. [2] , Content-based filtering- This approach creates and maintains a user profile based on the content description of the item which was previously rated by user. A Major disadvantage of this approach is that the recommended items are similar to the items seen previously by the user. [2] , Collaborative filtering- One of the most successful and widely used recommender system. In this approach analyze similarities of users’ previous rates and generate new recommendations based on analyzed similarities[2].

Hybrid approaches (combining content-based and collaborative filtering) has also been implemented in recommender system. Unlike these approaches, RL can manage dynamic user-system interactions and consider long-term user engagement.

In the reinforcement learning context, the cold start problem is referred to the initial phase of the reinforcement learning system where it lacks sufficient data to make decisions, since reinforcement learns through user feedback, and in the context of personalized recommendations, the system has no data on the user to give a suitable recommendation, the proposed system is to solve the cold start problem in the given context [9].

Proposed Personalized recommendation system can learn dynamically about user behavior against presented recommendations by using Reinforcement Learning. The proposed System is to use the trial-and-error method to learn about user interest with the help of Reinforcement Learning algorithms.

The literature surrounding personalized recommendation for diabetic retinopathy using reinforcement Learning varies on different aspects of the research topic. Initially, on the topic of choosing a proper technique for automating the personalized recommendation system, a research paper done on the publication of Deep Learning Techniques for Biomedical and Health Informatics [3], explores reinforcement learning on personalized recommendation systems and the efficiency of reinforcement models on said context[8].

A study was done on the significance of using Reinforcement Learning for Personalized recommendations. In this study, extensive research is done on the usage of machine learning methods such as supervised learning, unsupervised learning, semi supervised learning in addition to reinforcement learning. It signifies how reinforcement learning is the most beneficial machine learning method when it comes to personalized recommendations. Furthermore, the research presents the approaches of reinforcement learning in personalized recommendation systems [4].

Research was done on a personalized risk-based screening approach for Diabetic retinopathy patients, supervised learning was used on this approach which signified personalization aspects for DR patients [5]. Furthermore, an expert system is proposed by the research on developing a reinforcement learning-based expanded treatment recommendation model using the health records of South Korean citizens to assist physicians, the system was developed for physicians operating on diabetic patients [6].

A system that uses deep ensemble learning to predict diabetic retinopathy and implements a recommendation system was developed in South Korea. The overall performance of the said recommendation system was 99.6% and research is done on a fully automated prediction and recommendation system [7].

In a study done by S.V Bartholomeusz et al, an approach for a similar recommender system was proposed to manage stress levels of IT professionals using Q learning [1]

The proposed personalized recommendation system advances previous solutions by offering a fully per-user personalized approach tailored specifically to diabetic patients while considering the the diabetic retinopathy aspect.
In comparison,

- The study conducted by Edirisinghe, Kelum [4] used reinforcement learning but did not tailor the solution to diabetic patients or consider diabetic retinopathy.
- The study conducted by Edirisinghe, Kelum [4] used reinforcement learning but did not tailor the solution to diabetic patients or consider diabetic retinopathy.
- The approach presented by García-Fiñana M et al [5] provided personalized recommendations for diabetic retinopathy patients using supervised learning, but it lacked a recommendation system.
- The expert system developed by Sang Ho Oh et al [6] used reinforcement learning for diabetic patients but didn't focus on diabetic retinopathy or include a recommendation system.
- The system proposed by Baha Ihnaini et al [7] uses deep ensemble learning for predicting diabetic retinopathy and included a recommendation system, but it didn't use reinforcement learning or focus on personalized recommendations.

The proposed solution uniquely integrates all these aspects, providing a comprehensive system that is personalized, tailored for diabetic patients, uses reinforcement learning, and focuses on diabetic retinopathy.

3. METHODOLOGY

As mentioned before, we created a cross platform mobile application named "Retina-care' using React native. Within this application, a key component is diabetic retinopathy management through the dietary and exercise recommender system. Q learning with an integrated expert system was implemented to generate recommendations personalized to each user according to their medical condition and preferences. The reinforcement learning aspect was mainly implemented on dietary recommendations since exercise recommendations were found to be more generic comparatively. This section elaborates on how we implemented this recommendation system.

First and Foremost we needed medical expertise on diabetic retinopathy management. Medical journals, and meetings with medical practitioners and specialists were used to gather information on diabetic retinopathy management. Initially, we started to implement the personalized dietary recommendation system.

A set of conditions (such as activeness level and whether the patient is vegetarian or not), which were used to differentiate patients was initially chosen with collaboration of medical practitioners, and all unique combinations of these conditions were identified. In a Reinforcement Learning point of view, these combinations of conditions are called "States". Then, according to the set of states, we chose a list of Diet plans. These diet plans were created with the help of dieticians and general medical practitioners, adhering to the restrictions of diabetes patients. When creating this list of diet plans, it was made sure that at least a few diet plans from the list is suitable for patients in each state (The conditions chosen before). In a RL point of view, these diet plans are called the "Actions". Additionally, In a RL point of view, an "episode" is one complete iteration of the RL system. In our context, an episode is completed when the system reads the state of the user, gives out the action (Recommendation) and then receives the reinforcement from the user.

Initially, when setting up the recommender system, user handling was developed first. The system was developed to create a new Q table each time a user registers. In Reinforcement Learning, Q tables are tables which include a value named Q value for each state-action pair. Initially, the Q table was set in a pre-defined size, according to the actions and states chosen. Upon creation all Q values in the Q table was set to null.

The cold start problem was a major challenge faced in the development process of the reinforcement learning model. After extensive research these were the three solutions presented

i. Epsilon-Greedy Method

ii. Initialized Q tables

iii. Integrating an expert system

All three solutions were implemented and tested out as elaborated in the following sections.

Epsilon greedy method is a common method used in reinforcement learning, especially with Q learning to balance the “Exploration” and “Exploitation” [10] aspects. Initially, a parameter called “Epsilon Parameter” (denoted by “ ϵ”) was set up which ranges from 0 to 1 , and initially was set to 1. Upon each episode, the method of choosing an action was set to a probability based on the epsilon parameter (ϵ) , with the probability of ‘ϵ’ , the system was to choose a random action from the action list , and with probability ‘1- ϵ’ , the system was to choose the action with the highest Q value from the Q table. The epsilon value was set to gradually decrease after each episode.

As a solution for the “Cold start” problem which doesn’t use “Explore and Exploit Method”, the Q table initializing method was proposed. In this method , when a new user signs up to the system and a new Q table is created, rather than initially setting all Q values to 0 , All Q values of the Q table were assigned pre-defined Q values, so when choosing an action even in the first episode, the system is able to give out the action with higher Q values, Since the Q values will be updated according to user reinforcement with each episode. With time, the Q table would be personalized to each user.

The initial Q values in this method were decided after continued sessions with medical experts on diabetes, each meal recommendation (action) was compared with each state, and a Q value (ranging from 0 to 1) was assigned to each state-action pair. A higher Q value was assigned if the experts highly recommended the patients with given state consuming the meal plan corresponding to the given action, while a lower Q value was assigned if experts did not recommend the action to the given state. For the implementation of this solution, a sample Q table was initialized which contained only 4 states and 9 actions (Shown in fig. 1).

q(a,S)	s-->			
a	1	2	3	4
1	0.3	0.6	0.2	0.9
2	0.2	0.9	0.4	0.8
3	0.4	0.9	0.2	0.5
4	0.3	0.8	0.2	0.7
5	0.7	0	0.8	0
6	0.7	0	0.7	0
7	0.9	0	0.4	0
8	0.8	0	0.7	0
9	0.6	0.4	0.4	0.8

Fig-1: Sample Initialized Q table.

Following the practicality issues of the initialized Q table method, the cold start issue was successfully resolved by integrating an expert system. In this solution, the episode count of the user was tracked alongside other user data upon registration. An expert system was then implemented where the expert system would read the user state, and give out a list of meal recommendations (actions) that are medically acceptable according to the state of the user. The recommender system would then randomly choose one of the actions given by expert system, and present it to the user. The Q table would then be updated based on the reinforcement the user gives out to the given action.

This expert system was developed with collaboration with medical practitioners, abiding to the medical restrictions of meal plans for diabetic retinopathy patients. For a set no of user episodes, the expert system would be responsible for choosing actions, and after the no. of episodes are exceeded and the Q table is initialized, the system would use the Q table to give out recommendations to users.

The reinforcement aspect of the system was developed as follows. After the user was shown the recommendation, the user is prompted to give out a response about the dietary recommendation. A Slider UI component was used for this and the amount of likeness the user has for the recommendation was measured using this response. A value mapped from -1 to 1 (-1 if the slider value was lowest and 1 if highest) was then returned to the system. In a RL point of view, this value is called the “reward”.

After the system receives the reward value, it would use the Q function to update the relevant Q value to the correspondent state-action pair of the reward. The Q function is a crucial aspect of Q learning and can be depicted as follows: [11]

$$Q(s,a) \leftarrow Q(s,a)+\alpha[R+\gamma \max Q(s',a')-Q(s,a)] \qquad (1)$$

Where α is the learning rate and has a value between 0 and 1. R is the reward and γ is the discount factor which also ranges between 0 and 1. It represents how much future rewards are valuable compared to immediate rewards. The learning rate was set to gradually decrease as the no of episodes grow and the discount factor was kept at an constant state. The system would use this Q function to replace the corresponding Q value with the new Q value obtained from the Q function.

A component was developed and added to the system to test the viability of the SARSA (State-Action-Reward-State-Action) principles in the proposed system. SARSA is an on-policy reinforcement learning algorithm compared to off-policy Q learning, which updates the Q-values based on the state-action pairs encountered by following the current policy. The SARSA function can be depicted as follows:

$$Q(s,a) \leftarrow Q(s,a)+\alpha[R+\gamma Q(s',a')-Q(s,a)] \qquad (2)$$

Compared with Q learning which always uses the highest Q value to choose the action, SARSA chooses the action with the highest Q-value most of the time, but also explores other actions with the probability of ‘ϵ’, and when updating the Q value, an estimated Q value of the next state-action pair is used, while the max Q value of next state is used in Q learning.

The Dietary recommender system of the Retina-care application can be explained as follows: as the first step in the above diagram, the user state and the episode number are the initial inputs for the recommender system, the episode number is tracked in user management of the application.

In the second and third steps, if the episode number exceeds the set limit, the system sends the state to the reinforcement agent, which gets the action with the highest Q value from the Q table and return to the system, and the system shows the action (recommendation) to the user.

If the episode number does not exceed the set limit, the system send the state to an expert system which then send a list of actions corresponding to the state, the system then randomly choose an action from the list and shows to the user.

The user can select the given action or a different action from the action set and then rate the likeness of the recommendation chosen using the application UI, which is considered as the feedback. This feedback is then returned to the system.

The system would then send a reward value to the reinforcement agent as shown in step five , which would then calculate the new Q value for the specific state-action pair, and finally update the Q value in the Q table in the final step

The diagram below shows an overview of how the Recommender System works.

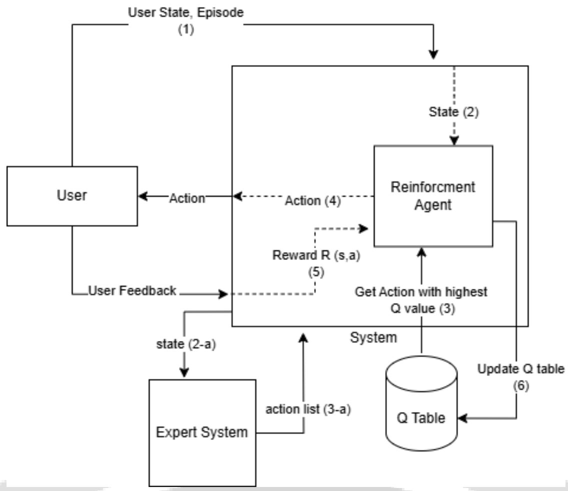

Fig-2: Recommendation System Workflow

After the recommendation system was implemented, a feature was developed for doctors to add new recommendations to the recommender system. In order to implement this feature, the following changes were done to the system.

- Storing actions dynamically on a database
- Managing Expert system dynamically
- Adding method to update current Q tables
- Updating current methods which handles actions

The updated recommender system would store all actions on a database, and when a doctor adds an action, this new action would be added to the action database, and furthermore, all Q tables which are currently active would be updated and this new action would be added. All the Q values of the new action would be set to 0 upon addition.

The below diagram shows an overview of the process of adding a new recommendation.

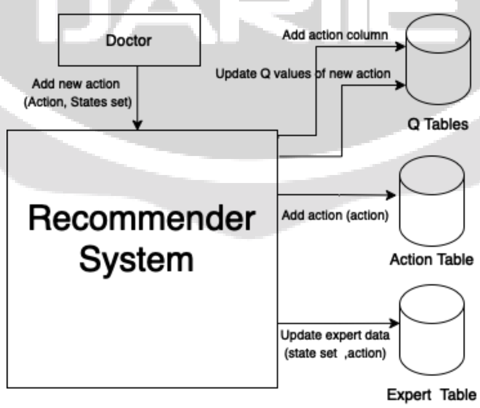

Fig-3: Add recommendation workflow

However, the add action functionality was developed in a way so that the doctor was to specify which conditions (states) of patients were these recommendations valid to. These set of states are then used by the system to update the Q values of the new action of all Q tables, a relatively higher Q value is set for the states specified by the doctor so that users get the new recommendation if they have conditions specified by the doctor. Furthermore, the expert system was updated in a way that a database was used to store which actions belonged to each state, upon the doctor adding a new state, the new state was set to be added to the expert system database according to the states specified by the medical professional upon addition.

The personalized exercise recommendation system was implemented in a different approach. After continued discussions with medical practitioners and experts, it was discovered that the exercise recommendations were more generic compared to the dietary recommendations. An expert system was designed to give out exercise recommendations to patients based on their medical factors such as retinopathy level , body mass index, and cholesterol levels. This expert system would give out a set of exercises , from which the recommendation system would choose an exercise through reinforcement learning techniques to present to the patient.

The ”retina-care” application was designed considering the necessary UI /UX aspects, while considering the target user group which are diabetic retinopathy patients. Actions such as getting user reinforcement for recommendations were tailored in a manner that the user experience of the app was kept positive. Furthermore, the application was developed adhering to proper software engineering guidelines, making sure of the security aspect, ethical aspect and protecting user privacy. The retina-care web application was developed in a similar manner, ensuring that the medical practitioners and experts has a platform to manage their patient data, and add expert data. As mentioned before, the mobile application was developed through the react-native framework , while the web application was developed through the react framework.

The recommender system was implemented through a flask server , and version controlling for the server, web application and mobile application was done through GitHub. In terms of deployment, the recommendation server was deployed through a CI/CD pipeline, with GitHub actions and using AWS cloud services in a manner that the recommender service would always be accessible with WEB API calls from either web client or mobile client.

The below diagram depicts an architectural overview of the recommendation system

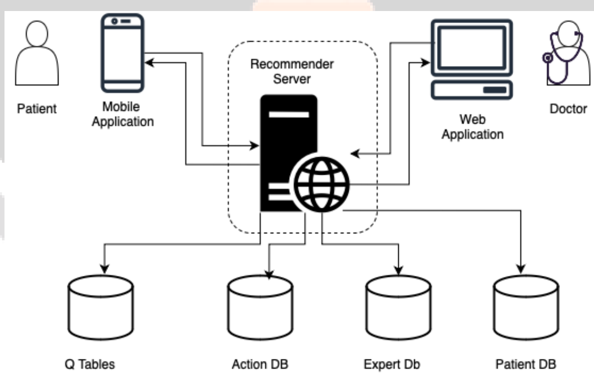

Fig-4: System Architecture Diagram

4. RESULTS AND DISCUSSIONS

When Selecting a solution for the cold start Problem , the following factors were considered :

Usage of the epsilon greedy method in the proposed system was found to be problematic because it uses “Explore and Exploit” method, and random actions are chosen in the exploration stage. After discussion with medical practitioners, it was agreed upon that this method was not ethical to be used on a medical system since giving out wrong health recommendations to users can be harmful to the patients’ health. Hence, the Epsilon Greedy method was not used.

When implementing the Initialized Q tables solution, a prototype Q table was initialized which contained only 4 states and 9 actions (fig. 1). Discussions with experts for this prototype Q table took a considerable amount of time since each meal plan had to be carefully considered with each condition of a state, and Q values were to be assigned comparatively. This deemed the solution impractical, given the actual Q table size of the proposed system, and the considering the issues that may arise when scaling the application and dynamically adding more meal recommendations (actions)

Compared to the other solutions, Integrating with an expert system was more successful, So the system used this solution for the cold start problem. Furthermore, when considering the reinforcement learning algorithm, Q learning was chosen over SARSA, since the epsilon based exploration in SARSA was not deemed ethical in the context of giving out health recommendations.

Evaluating the recommender system is a crucial step to be done before deploying a reinforcement learning based recommender system, Specially in a medical context such as this , where real diabetic retinopathy patients would be using the system. Usually in reinforcement learning , including Q learning the accuracy is measured by the metric "Cumulative reward". Cumulative reward is measured by the reward metric given by the user in each iteration and measuring the average reward value.

Prior to deployment, a set of diabetic retinopathy patients were selected with different conditions , where the patients had a variety based on conditions such as age, retinopathy level and body mass index. These patients were then given the recommender system and the cumulative reward graph was recorded on each patient, such as depicted in fig. 5.

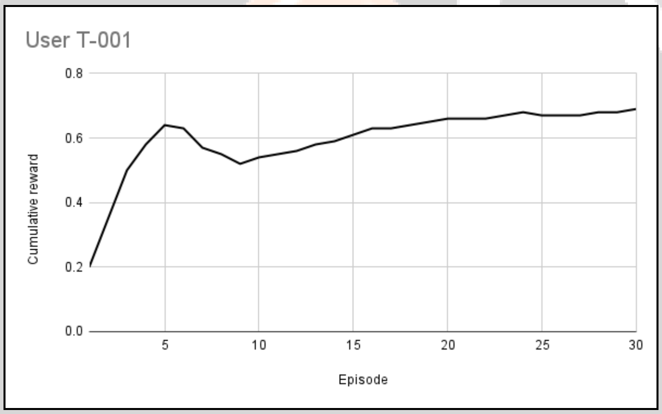

Fig-5: User Cumulative Reward Graph

The graph shown below (fig. 6) depicts the average reward given by users in the first 30 episodes using the recommender system. As seen in the graph , the reward level of users was found to be comparatively low throughout the first 10 iterations , and had higher reward values afterwards.

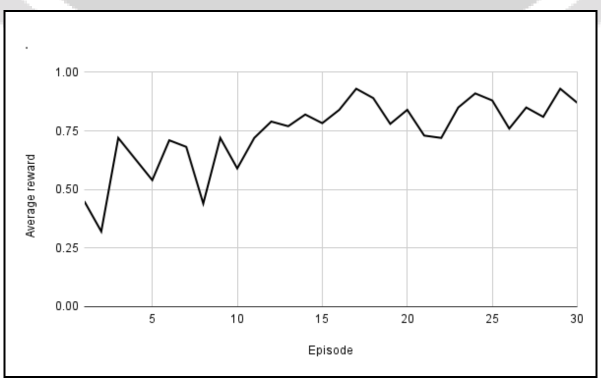

Fig-6: Average Reward per Episode

The average reward value of the first 10 iterations tend to have a lesser value because the system uses an expert system to train the reinforcement agent, hence the recommendations are not yet personalized. Furthermore, there are instances where the reward values drop over time even when user receives recommendations through the reinforcement agent. The reason for this is that patients tend to prefer a diversity when choosing their meals. This in turn helps the recommender system adjust the Q values and give out a variety of recommendations.

The framework of the Recommendation System implemented in the "Retina-care" application is not just limited to this specific use case. This can be applied effectively in any scenario where there is a requirement for a recommendation system where explore and exploit cannot be used, and a need to integrate an expert system is present, Especially in the medical domain. For instance, consider an activity recommendation system for autism patients , the activities would differ from many factors such as level of autism and patient age, and given that there is a variety of autism management activities , this framework can be used to recommend activities that the patient would prefer , while also ensuring that the said activities are appropriate to the condition of the patient. Furthermore, medical professional would be able to add new activities to the activity pool, even after deployment of the system. In complex scenarios such as that, our recommendation system framework can be of much value.

5. CONCLUSION

The paper dives into the development process of the recommendation system of the Retina-care application, which is an application designed for Diabetic Retinopathy patients. The recommendation system which is a part of retinopathy management, provides user with exercise and dietary recommendations, while making sure that these recommendations adhere to both user preferences and medical restrictions. Medical practitioners add new recommendations specified for patient states through the web portal. This specific requirement needs an approach that can manage all these challenges, while ensuring the user frequently engages with the application.

One of the key features of the developed recommender system is that it successfully solves the cold start problem which is a crucial challenge in the medical recommendation context. This was implemented by a combination of integrating an expert system and usage of the epsilon greedy method. The system gives out recommendations and receives feedback required for the reinforcement agent while maintaining a positive user experience of the application. To evaluate the performance of the recommender system, the cumulative reward index was measured in a sample set of diabetic retinopathy patients, and the results indicated a high cumulative reward as the users continued to use the system.

Moving forward, there is a promising path for future research involving Q learning based recommender systems in the medical domain, where there are medical constraints on recommendations, leaving no room for error in recommendations given. Integrating expert systems to overcome the cold start problem and dynamically managing actions can open new solutions towards problems that were previously classified hard to overcome.

6. REFERENCES

[1]. S. V. Bartholomeusz, H. M. S. Chathuranga and D. Ganegoda, "Stress Relieving and Emotion Alleviating Activities Recommendation System Based on Deep Reinforcement Learning - A Novel Method Providing Slate Recommendations Which Utilizes SlateQ with Multi-Objective Optimization," 2023 5th International Conference on Advancements in Computing (ICAC), Colombo, Sri Lanka, 2023, pp. 680-685

[2]. Wei Chu and Seung-Taek Park. 2009. Personalized recommendation on dynamic content using predictive bilinear models. In Proceedings of the 18th international conference on World wide web (WWW '09). Association for Computing Machinery, New York, NY, USA, 691–700.

[3]. Dash, S., Acharya, B. R., Mittal, M., Abraham, A., & Kelemen, A. (Eds.). (2020). Deep Learning Techniques for Biomedical and Health Informatics. Studies in Big Data.

[4].Edirisinghe, Kelum. (2020). Reinforcement Learning For Personalized Recommendations. 10.13140/RG.2.2.32213.45286.

[5].García-Fiñana M, Hughes DM, Cheyne CP, et al. Personalized risk-based screening for diabetic retinopathy: A multivariate approach versus the use of stratification rules. Diabetes Obes Metab. 2019; 21: 560–568..

[6].Sang Ho Oh, Jongyoul Park, Su Jin Lee, Seungyeon Kang, Jeonghoon Mo, Reinforcement learning-based expanded personalized diabetes treatment recommendation using South Korean electronic health records, Expert Systems with Applications, Volume 206,2022, 117932,ISSN 0957- 4174

[7].Baha Ihnaini, M. A. Khan, Tahir Abbas Khan, Sagheer Abbas, Mohammad Sh. Daoud, Munir Ahmad, Muhammad Adnan Khan, "A Smart Healthcare Recommendation System for Multidisciplinary Diabetes Patients with Data Fusion Based on Deep Ensemble Learning " , Computational Intelligence and Neuroscience, vol. 2021, Article ID 4243700, 11 pages, 2021.
[8]. Mulani, J., Heda, S., Tumdi, K., Patel, J., Chhinkaniwala, H., Patel, J. (2020). Deep Reinforcement Learning Based Personalized Health Recommendations. In: Dash, S., Acharya, B., Mittal, M., Abraham, A., Kelemen, A. (eds) Deep Learning Techniques for Biomedical and Health Informatics. Studies in Big Data, vol 68. Springer, Cham
[9].Stelios Giannikis, Flavius Frasincar, David Boekestijn,Reinforcement learning for addressing the cold-user problem in recommender systems, Knowledge-Based Systems,Volume 294,2024,111752,ISSN 0950-7051
[10]. Kaelbling, L. P., Littman, M. L., & Moore, A. W. (1996). Reinforcement learning: A survey. Journal of Artificial Intelligence Research, 4, 237-285
[11]. B. Jang, M. Kim, G. Harerimana and J. W. Kim, "Q-Learning Algorithms: A Comprehensive Classification and Applications," in IEEE Access, vol. 7, pp. 133653-133667, 2019