International Journal of Advances in Engineering and Management (IJAEM)
Volume 6, Issue 11 Nov. 2024, pp: 475-479 www.ijaem.net ISSN: 2395-5252

Personalized Exercise Recommendation Using Reinforcement Learning

P.Sanjana, U.Sanjana, K.Santhoshini, G.Shivaja, Prof. Sabyasachi Chakraborty
Department of AIML, School of Engineering, Malla Reddy University

---

Date of Submission: 15-11-2024 Date of Acceptance: 25-11-2024

ABSTRACT: In today's fitness landscape, many individuals face challenges in developing workout plans that are tailored to their unique fitness levels, body types, and available equipment. Generic workout routines often fail to meet the specific needs of users, leading to suboptimal results and decreased motivation over time. This paper proposes developing a web-based system that utilizes reinforcement learning (RL) to deliver personalized exercise recommendations. By continuously learning from user feedback and progress, the system adapts workout plans dynamically to maximize effectiveness and sustainability. The RL model considers factors such as user fitness levels, personal preferences, body type, and available equipment to suggest optimal exercises and routines. This approach fosters an individualized fitness experience, improving adherence, results, and long- term user engagement. The proposed system could revolutionize how individuals approach fitness by offering customized solutions that evolve as the user progresses on their fitness journey.

I. INTRODUCTION

Achieving personal fitness goals requires more than just motivation; it demands a well-structured workout plan that aligns with an individual's unique fitness level, body type, and equipment availability. However, many fitness enthusiasts struggle to create personalized workout plans tailored to their specific needs. As a result, they often turn to generic fitness programs, which may not address their individual goals or physical capabilities. These generic plans, while easily accessible, frequently lack the customization necessary to maintain long-term effectiveness, leading to frustration, plateaus, and eventual disengagement.

In recent years, advancements in artificial intelligence (AI) and machine learning (ML) have introduced new possibilities for personalizing user experiences in various fields. One promising approach in the realm of fitness is reinforcement learning (RL), a branch of AI that enables systems to learn and adapt based on continuous feedback. This dynamic learning process can be leveraged to design workout programs that evolve with the user, providing tailored exercise recommendations that are both effective and sustainable. System offers personalized workout recommendations by tailoring plans to each user based on their input and feedback. It is adaptive, continuously learning from user behavior to improve and refine recommendations over time. Additionally, it is scalable, capable of expanding to include more exercises, equipment, and fitness levels. This paper presents a web-based system that uses RL to generate personalized workout plans. By considering user-specific factors such as fitness level, preferences, and equipment access, the system creates an adaptive and individualized fitness regimen. As the user interacts with the system and provides feedback, the RL model refines and optimizes future workout suggestions. This approach not only enhances the effectiveness of the workout plans but also encourages long-term adherence by keeping users engaged and motivated.

II. LITERATURE REVIEW

1. Research emphasizes that personalized approaches enhance user adherence and motivation compared to generic programs (Kelley et al., 2016). RL, as explored by Sutton and Barto (2018), enables adaptive learning through interactions, making it well-suited for dynamic environments that will give a good likely fitness.
2. Context-aware systems (Zhang et al., 2019) have shown that incorporating factors such as time and location can further refine exercise suggestions. Engagement strategies,

International Journal of Advances in Engineering and Management (IJAEM)
Volume 6, Issue 11 Nov. 2024, pp: 475-479 www.ijaem.net ISSN: 2395-5252

3. Including gamification and feedback mechanisms (Hamari et al., 2014), have been proven to motivate users effectively, a feature RL systems can leverage..
4. Integrating real-time health monitoring data from
5. wearables (Wang et al., 2020) allows for personalized adjustments based on users' physiological responses.
6. Existing frameworks, such as those developed by Chen et al. (2021), demonstrate improved adherence through user modeling and real-time feedback.
7. Alharbi et al. (2022) highlight the potential of multi-agent systems for collaborative workouts.
8. Despite these advancements, challenges such as data sparsity, scalability, and real-time adaptation remain, necessitating further research to enhance
9. the efficacy of RL in fitness recommendations. Overall, the ongoing exploration of RL in this domain promises innovative solutions for promoting healthier lifestyles.

III. PROBLEM STATEMENT

To develop a web-based platform that provides personalized workout recommendations by leveraging reinforcement learning. The system aims to Tailor workouts to the individual needs of users based on their fitness level, body type, preferences, and available equipment. Adapt dynamically by learning from user feedback and progress, ensuring that workout recommendations improve over time and remain effective. Ensure scalability, enabling the system to easily incorporate a broader range of exercises, equipment, and fitness levels, making it versatile for various applications. Offer solutions for personal trainers, fitness apps, and gyms to provide users with adaptive, data-driven exercise programs that maximize engagement and results.

IV. METHODOLOGY

1. User Input:

The user selects their fitness level, body part to train, and available equipment.
Initial State: User selects Intermediate, Abdominals, and Bands.

2. Exercise Recommendation:

The system suggests an exercise using the Q-Learning algorithm.generate corresponding outputs. Images depicting the same individual are labeled with the same unique identifier to ensure consistent recognition.
Recommended Exercise: The system suggests the "Front Plate Raise."

3. User Feedback:

The user provides feedback (like/dislike).
"User Feedback: Positive feedback (reward = 1)

4. Q-Table Update:

Based on the feedback, the Q-table (internal memory of the agent) is updated to improve future recommendations.
Q-Table Update: The system strengthens the association between the state and the recommended exercise for future recommendations.

Main Components:

i. Q-Learning Algorithm:

Handles exercise recommendations, and updates the Q- table based on user feedback.

ii. State Representation:

Combines Level, Body Part, and Equipment into a state.

iii. Handling User Feedback:

Rewards (1 for positive, 0 for neutral, -1 for negative) update the system's recommendations.

iv. Code Snippet: Showing a key part of the Q-learning update function.

Dataset Features: Columns:

1. **Level:** User's experience level (e.g., Beginner , Intermediate, Advanced)
2. **Body Part:** Targeted body part (e.g., Abdominal, Chest, Legs)
3. **Equipment:** Available equipment (e.g., Bands, Dumbbells)
4. **Title**: Exercise name
5. **Dataset Insights:** Variety of exercises across fitness levels and body parts. Diverse equipment-based exercise

V. DATA ANALYSIS

To perform training and testing with the given Q-learning algorithm and dataset, let's take the following steps:

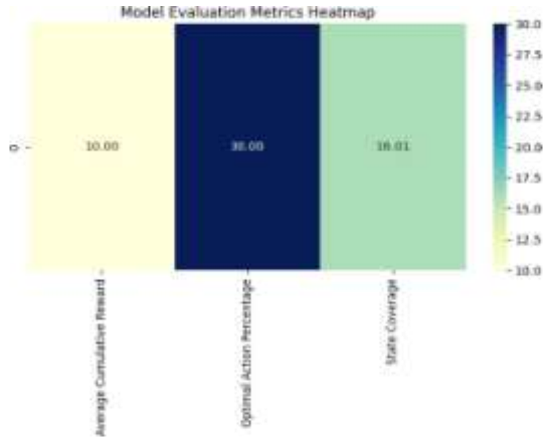
1. **Load and preprocess the dataset**.
2. **Set up the actions and states** based on the dataset.
3. **Simulate episodes** where the agent interacts with the environment by choosing actions based on states, updating the Q-table based on rewards, and moving to new states.

International Journal of Advances in Engineering and Management (IJAEM)
Volume 6, Issue 11 Nov. 2024, pp: 475-479 www.ijaem.net ISSN: 2395-5252

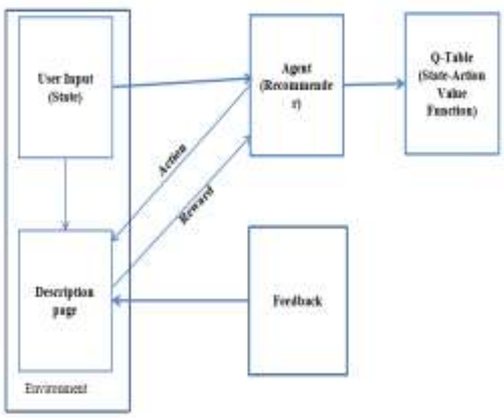
4. **Evaluate the performance** of the trained Q-learning agent.

MODEL EVALUATION MATRIX:

To visually represent the Q-learning model evaluation, we can create a matrix that shows the results for each evaluation metric as a heatmap. This can make it easy to see which metrics show strong or weak performance.



Model Selection And Architecture:



VI. MODEL DEVELOPMENT :TESTING AND VALIDATION

In deploying and validating a Q-learning-based exercise recommendation system, the objective is to ensure that the model provides accurate, reliable, and personalized recommendations in a production environment. Here's a structured overview of testing and validating the deployment phase, covering important aspects like testing methodologies, evaluation metrics, and model monitoring.

A. Testing Methodologies:

- **Offline Testing:** To evaluate the model on historical or simulated data before moving to a live environment. This phase helps ensure the model performs well on past data and avoids negative experiences in production.

A/B Testing: To compare the Q-learning recommender model's effectiveness against a baseline (such as a rule-based or random recommendation system) with a group of real users.

- **User Testing and Feedback**: To gather real-time feedback on the quality and relevance of the recommendations from actual users.

B. Validation Matrix:

- **Cumulative Reward**: Measures the sum of rewards obtained over time, indicating the model's overall performance in achieving positive user interactions.
- **Precision K**: Calculateing the proportion of the relevant recommendations within the top K results.
- **Average Session Duration and Engagement**: Measures the amount of time users spend on recommended exercises, serving as a proxy for engagement.
- **Action Convergence Rate**: Calculates the percentage of instances where the model consistently recommends the same or similar exercises for specific states.
- **User Satisfaction Score**: A direct measurement of user satisfaction collected via surveys or feedback forms.

C. Real Time Monitoring:

- **Real-time Monitoring:** To continuously track the model's performance metrics, ensuring it performs well in a live environment and meets predefined KPIs.
- **Drift Detection:** To monitor for distributional shifts or changes in user behavior that may affect recommendation accuracy.
- **Dynamic Parameter Tuning:** To dynamically adjust the model's exploration rate, learning rate, or discount factor based on real-time user interactions.
- **User Feedback Loop:** To incorporate real-time user feedback into the model learning and imprlearning and improvement.

International Journal of Advances in Engineering and Management (IJAEM)
Volume 6, Issue 11 Nov. 2024, pp: 475-479 www.ijaem.net ISSN: 2395-5252

D. User Feedback loop:

- **Scheduled Retraining:** Periodically retrain the model on new data to account for shifts in user behavior and preferences.
- **Feedback-based Adjustments:** Continuously refine the reward function and model parameters based on observed performance and user feedback.
- **Feature Expansion:** Add new features, such as seasonal preferences or fitness goals, to make recommendations even more personalized and relevant.

VII. MODEL IMPLEMENTATION

This code sets up a Flask-based web GUI for a Q-learning-based exercise recommender system. The application allows users to select their preferences (e.g., level, body part, equipment) and receive exercise recommendations. The system also incorporates a feedback mechanism that allows users to give feedback on the recommended exercises, which in turn updates the Q-learning model.

Here's an overview of each part and improvements that can enhance usability:

1. Data and Model Loading:

- The app loads the dataset (megaGymDataset.csv) and initializes unique states and actions.
- It then creates an instance of QLearningExerciseRecommender, the Q-learning model for exercise recommendations.

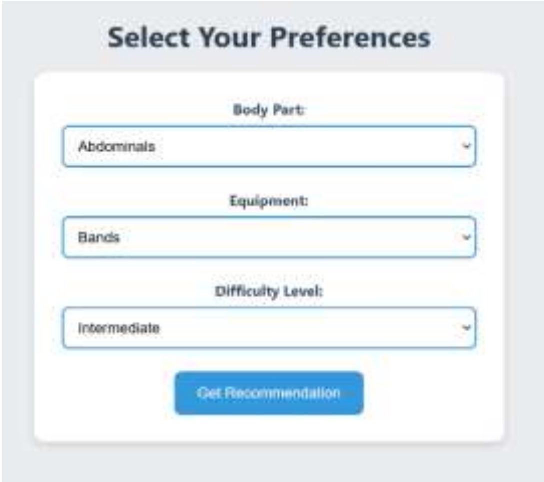
Endpoints:

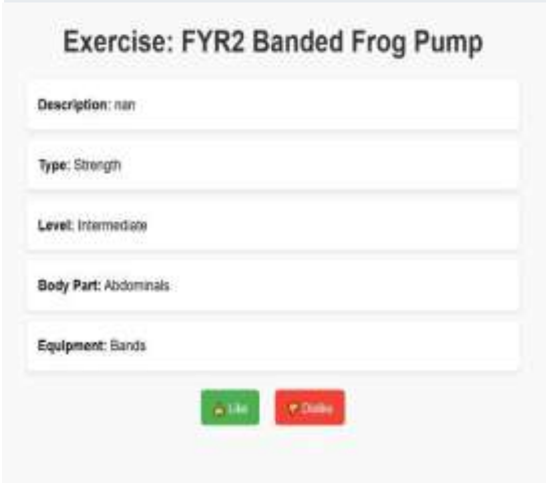
- **Home Page (/)**: Displays the home page, where users can choose options.
- **Recommendation Page (/recommendation)**: Users input their desired level, body part, and equipment, which creates a state. The system uses this state to recommend an exercise via the Q-learning model and redirects to the exercise description.
- **Description Page (/description/<exercise>)**: Shows details of the recommended exercise. Users can provide feedback by pressing "like" or "dislike" buttons, which update the Q-learning model's Q-table.

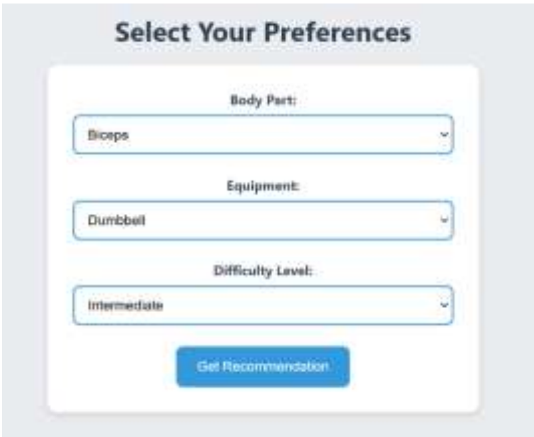
Feedback Mechanism:

- When a user rates a recommendation, the model uses this feedback to update the Q-table, enabling better recommendations over time.

VIII. RESULTS







IX. CONCLUSION

Reinforcement learning is a powerful tool for creating personalized fitness recommendations. The system adapts to user preferences over time, improving the quality of recommendations.

Final Thoughts: This system is a promising step towards smarter, personalized fitness technology.

X. FUTURE WORK

A limited dataset might impact the diversity of recommendations. User feedback is essential for the system's learning curve. Add more user-specific features like goals, workout history, and health conditions. Integrate with wearable devices for real-time feedback on exercise effectiveness.

REFERENCES

[1]. **Sutton, R. S., & Barto, A. G. (2018).** Reinforcement Learning: An Introduction (2nd ed.). MIT Press. This book is the definitive guide to reinforcement learning (RL), providing a thorough understanding of Q-learning and other RL algorithms. It includes examples and techniques relevant to learning from user feedback and building adaptive recommendation systems.

[2]. **Zhao, X., Liu, X., & Tang, J. (2022).** "Deep Reinforcement Learning for Online Personalized Recommendation: Methods, Applications, and Challenges." Artificial Intelligence Review, 55(1), 1–26. This paper focuses on reinforcement learning methods in the recommendation space, specifically how they apply to real-time and personalized recommendations. It provides valuable insights into RL techniques, which are essential for Q-learning-based exercise recommendation systems.

[3]. **Shani, G., & Gunawardana, A. (2011).** "Evaluating recommendation systems." In Ricci, F., et al. (Eds.), Recommender Systems Handbook, 257-297. This chapter covers evaluation metrics and methodologies for recommender systems, which is crucial for testing and validating a Q-learning-based exercise recommendation system. It discusses metrics like precision, recall, and engagement, which are applicable to personalized recommendations.

[4]. **Maugeri, G., & Barchitta, M. (2019).** "A systematic review of physical activity interventions in primary care." International Journal of Environmental Research and Public Health, 16(16), 3026. This systematic review covers various exercise interventions and provides insights into the types of exercises and factors (like user fitness level and equipment availability) that are crucial for personalization in exercise recommendations.

[5]. **He, Z., Li, X., Chen, X., Huang, T., Sheng, Q., & Zhao, Y. (2020).** "Modeling user fitness data for personalized activity recommendation." IEEE Access, 8, 108411-108421.This article focuses on using fitness data to recommend personalized activities. It explores data-driven approaches and features relevant for adapting recommendations to users' preferences and activity history.

[6]. **Kang, C. R., & Park, J. Y. (2021).** "A personalized exercise recommendation system using fitness records in a mobile application." Healthcare Informatics Research, 27(1), 57-66. This research presents a personalized exercise recommendation system using mobile app data, examining how features like user history, exercise difficulty, and preference improve recommendation accuracy and user satisfaction.