IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.53
Volume 4, Issue 6, April 2024

An Online Deep Reinforcement Learning Based Order Recommendation Framework for Rider-Centered Food Delivery System

Dr. M.A. Chaudhari[1], Pratiksha Balasaheb Tanpure[2], Prathmesh Hemant Mutke[3], Rutuja Ananda Bhalerao[4], Vrushali Namdev Avhad[5]
Department of Information Technology[1,2,3,4,5]
Amrutvahini College of Engineering, Sangamner, A. Nagar, Maharashtra

Abstract: *On-demand Food Delivery (OFD) has become a crucial component of intelligent transportation systems, revolutionizing logistics services in contemporary society. With transaction volumes soaring, a shift towards rider-centered assignment methods over traditional platform-centered approaches is evident among food delivery companies. Yet, challenges persist, including dynamic order arrivals, uncertain rider behaviors, and a plethora of false negative feedback, impeding effective decision-making by platforms during interactions with riders. To tackle these issues, we introduce an innovative online Deep Reinforcement Learning-based Order Recommendation (DRLOR) framework to navigate decisionmaking in the OFD setting. Formulated as a Markov Decision Process (MDP), our framework integrates three key networks: an actor-critic network for learning optimal order ranking policies, a rider behavior prediction network to forecast rider actions, and a feedback correlation network leveraging attention mechanisms to discern valid feedback amidst false reports, thereby constructing high-dimensional state representations of rider states. Rigorous offline and online experimentation on the Meituan delivery platform illustrates the efficacy of our DRLOR framework in significantly reducing interaction durations between riders and platforms, culminating in enhanced experiences for both riders and customers.*

Keywords: Online, Deep Reinforcement Learning, Order Recommendation, Rider-Centered, Food Delivery System

I. INTRODUCTION

1.1 Overview

In the dynamic landscape of modern food delivery services, the advent of On-demand Food Delivery (OFD) has transformed traditional logistics paradigms, with a notable shift towards rider-centered operations. This evolution reflects a strategic response to the burgeoning demand for efficient and timely delivery, necessitating innovative approaches to optimize rider-platform interactions. However, inherent challenges persist, including the unpredictable nature of order arrivals, the variability in rider behaviors, and the prevalence of false negative feedback, all of which undermine the efficacy of traditional platform-centric assignment models. To address these complexities, this paper proposes an advanced framework, termed the Online Deep Reinforcement Learning-Based Order Recommendation (DRLOR) system, specifically designed for Rider-Centered Food Delivery.Systems. Grounded in the principles of Deep Reinforcement Learning (DRL), the DRLOR framework leverages a Markov Decision Process (MDP) formulation to facilitate real-time decision-making in the face of dynamic operational conditions. Central to the DRLOR framework are three interconnected neural networks: an actor-critic network responsible for learning optimal order ranking policies, a rider behavior prediction network tasked with forecasting rider actions, and a feedback correlation network employing attention mechanisms to discern valid feedback amidst noise. These networks collectively enable the system to efficiently navigate the complexities of rider-platform interactions, enhancing overall operational effectiveness and customer satisfaction.

By conducting comprehensive offline and online experiments on the Meituan delivery platform, we demonstrate the tangible benefits of the proposed DRLOR framework. Specifically, our results showcase significant reductions in interaction durations between riders and the platform, leading to an improved experience for both riders and customers alike. In summary, this framework represents a novel and effective approach to address the intricacies of Rider-Centered Food Delivery Systems, offering promising avenues for enhancing efficiency and user satisfaction in the ever-evolving landscape of on-demand logistics services.

In today's rapidly advancing landscape of intelligent transportation systems, the integration of Online-to-Offline (O2O) businesses has become an essential facet of modern daily life. This paradigm shift enables individuals to access a diverse array of goods and services online, all from the convenience of their homes. Among the myriad of O2O services, On-demand Food Delivery (OFD) stands out as a pivotal player, offering consumers a vast selection of cuisines while relieving them from the burden of household chores simultaneously. In regions like China, where platforms like Meituan boast over 400 million active users and collaborate with more than 6 million merchants, the scale and potential of OFD services are undeniable, attracting significant attention from both researchers and investors. At the heart of the OFD ecosystem lies the platform's intricate orchestration of order collection, assignment to riders, and timely delivery to customers' doorsteps. This process involves a dynamic interplay of factors such as order locations, rider availability, and diverse order attributes, all of which collectively shape the operational landscape of the delivery.

1.2 Motivation

The motivation behind exploring novel order recommendation methods in On-demand Food Delivery (OFD) systems stems from the recognition that traditional approaches often prioritize platform metrics over rider preferences and constraints. By overlooking the needs of riders, such methods risk compromising efficiency and profitability. Addressing this gap is crucial as it directly impacts operational effectiveness and overall user satisfaction, underscoring the necessity for tailored solutions that harmonize platform objectives with the wellbeing of delivery personnel.

1.3 Problem Definition and Objectives

- To reduce number of declined orders and improve overall delivery experience.
- To Promoting a healthy work life balance for riders, preventing burnout. .
- To create a rider centered food delivery system aims to enhance the experience for delivery riders while ensuring efficiency and customer satisfaction.
- Enhanced Rider Centered Experience
- Real time Recommendation.
- Order recommendations to individual user preferences, enhancing customer satisfaction

II. LITERATURE SURVEY

“An Imitation Learning Enhanced Iterated Matching Algorithm for On Demand FoodDelivery [1]: This study looks into a problem faced by a real food delivery platform. This problem is tricky because things are always changing, it's really big and complicated, there's not much time to make decisions, and people really care about getting good results. The researchers figured out that they can break down this problem into smaller pieces that stay the same for a short period of time. This makes it easier to solve. To make sure they can solve it quickly but still get good results, they suggest creating some quick tricks to solve the problem and using machine learning to help make decisions by looking at past data. They tested this idea using a computer program that learns from past mistakes and improves over time.

“Short Term Demand Prediction for On Demand Food Delivery with Attention Based Convolutional LSTM [2]: This paper talks about a smart computer system called "At ConvLSTM" that helps predict how much food people will order for delivery in a city, but just for a short time. It uses a special type of technology called deep learning, which works like our brains, to look at patterns in how many orders are made and where they're coming from. This helps the system figure out how much food will be needed in different parts of the city in the near future. It looks at things like where and when people want food. It also pays more attention to some parts of the city when making predictions. They tested this system with real data and found that it's pretty good at guessing how much food people will order for delivery in

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.53
Volume 4, Issue 6, April 2024

the short term. This special attention thing they added also helps make the predictions more accurate. So, this system is helpful for figuring out how much food to have ready for delivery in a city.

"Deep Reinforcement Learning based Recomm. with Explicit User Item Interactions Modeling [3]: This paper introduces a new way to make recommendations called DRR. Instead of just suggesting things based on what's popular or similar, DRR treats recommending things like a game where you have to make decisions over time. It looks at both short-term and long-term benefits to give better suggestions. In DRR, there's a part that figures out what state things are in, like how many times you've clicked on something or how often you buy a certain type of product. Then, it uses this information to create different ways to make recommendations that understand how people interact with different items

III. REQUIREMENT AND ANALYSIS

Software Requirements:

- Python
- Flask
- Microsoft Excel

Hardware Requirements:

- RAM 8 GB or Higher
- HDD 100 GB(Processor – Intel Core I4 or Higher.

This module works in an efficient way by following a particular set of instructions in proper manner are as follows:

Problem Definition:

the problem statement and objectives of the framework. Identify the key challenges, such as optimizing order recommendations, minimizing delivery times, and maximizing rider efficiency while ensuring a rider-centered experience.

Data Collection :

Gather relevant data from various sources. This may include user profiles, order history, restaurant information, rider availability, delivery locations, and feedback data.

Preprocessing and Feature Engineering:

Features Prepare and preprocess the data. This involves cleaning, normalizing, and transforming the data into a format suitable for deep reinforcement learning. Extract relevant features that represent the state of the system, including spatial and temporal information.

System Architecture Design:

Design the architecture of the online framework, which includes: The policy network (actor) for order recommendations. The value network (critic) for evaluating actions and states. Integration with data sources and feedback mechanisms. Incorporation of convolutional neural networks (CNNs) for spatial-temporal featureextraction.

Deep Reinforcement Learning Implementation:

Implement the deep reinforcement learning algorithm, such as Actor-Critic or Deep Q- Network (DQN), to make sequential decisions. Train the policy network to recommend orders and the value network to estimate expected returns based on historical data.

Training and Simulation :

Train the deep reinforcement learning model on a simulated environment that emulates the real-world food delivery system. Use historical data and reinforcement learning techniques tooptimize the model's policy.

Feedback Mechanism:

Implement mechanisms for collecting user and rider feedback. This feedback loop is crucial for system improvement an Quality Control.

Rider Assignment Optimization:

Develop algorithms for optimizing rider assignments, considering factors like rider availability, delivery routes, and minimizing wait times

Testing and Evaluation:

Conduct thorough testing and evaluation of the framework using real-world data. Measure key performance metrics, such as order success rates, delivery times, and user satisfaction.

Continuous Learning and Improvement:
Enable the system to continue learning and adapting to changing user preferences, environmental factors, and feedback, thus improving the quality of recommendations and rider assignments over time.

IV. SYSTEM DESIGN

4.1 System Architecture

The methodology of the Online Deep Reinforcement LearningBased Order Recommendation Framework for a RiderCentered Food Delivery System is meticulously designed to optimize order recommendations and rider assignments through a systematic approach. Beginning with problem formulation, the framework identifies key objectives such as minimizing delivery times and maximizing rider efficiency while ensuring a rider-centered experience. Data collection follows, gathering pertinent information from various sources including user profiles, order history, and rider availability. Subsequently, preprocessing and feature engineering transform this data into a format suitable for deep reinforcement learning, extracting relevant features to represent the system's state effectively.

Actor-Critic Algorithm:

1. Initialization: Start by initializing the actor's policy network and the critic's value network with random weights.
2. Interaction with the Environment: The actor selects an action based on the current state using its policy network, executes the action in the environment, and observes the outcome.
3. Update the Critic: The critic evaluates the value of the action taken using its value network and computes the advantage, which is the difference between the observed reward and the critic's estimate of the expected reward.
4. Update the Actor: The actor's policy network is updated to maximize the expected return, using the advantage obtained from the critic.
5. Repeat: Steps 2-4 are repeated for multiple iterations, with the actor and critic networks continuing to interact with the environment, gather experiences, and refine their policies and value estimates.
6. Convergence: The actor-critic algorithm continues to update the actor and critic networks until it converges to a good policy that maximizes the expected return.

Rider Behavior Prediction Newtwork:

1. In online reinforcement learning for platforms, like those used by ridesharing apps, the system needs to learn from immediate feedback from users.
2. The app suggests a list of options to users, and their response determines how good or bad the suggestions were.
3. This feedback is crucial for teaching the system to improve. But in real life, user behavior can be unpredictable, making it hard to test changes offline. So, we use a special kind of neural network called RBP network to mimic user behavior.
4. This helps us accurately predict how users will react to suggestions, providing reliable feedback for the learning system.

Deep Reinforcement Learning (DRL) Algorithm:

1. Initialization: Set up the neural network architectures for the policy network (actor) and the value network (critic).
2. Environment Interaction: Interact with the environment, observe the current state, and generate a probability distribution over possible actions using the policy network.
3. Action Selection: Sample an action from the probability distribution generated by the policy network.
4. Execution and Feedback: Execute the recommended action in the environment, receive feedback, and calculate a reward based on the feedback received.
5. Value Estimation: Estimate the expected return or value of the current state using the value network.

6. Policy and Value Update: Update the policy network (actor) to maximize the expected cumulative reward and update the value network (critic) to better estimate the expected return of states.
7. Continue Iteration: Repeat steps 2-6 for multiple iterations, allowing the system to learn and improve its policy over time.
8. Convergence and Evaluation: Run the DRL algorithm until the policy converges to a satisfactory level of performance, leading to improved user experiences and system efficiency.

4.2 Working of the Proposed System

System architecture design integrates policy and value networks for order recommendations and evaluation, respectively, while incorporating feedback mechanisms and convolutional neural networks for spatial-temporal feature extraction. The deep reinforcement learning implementation trains these networks on historical data, optimizing the model's policy through training and simulation in a simulated environment mirroring the real-world food delivery system. Feedback mechanisms facilitate continuous learning and improvement, with algorithms developed to optimize rider assignments and ensure efficient delivery routes. Rigorous testing and evaluation, along with comparison with baseline models, validate the framework's effectiveness, leading to model fine-tuning and integration into the food delivery system's online platform. Continuous learning and maintenance ensure ongoing system optimization, supported by thorough documentation and reporting for ongoing analysis and improvement efforts. This structured methodology leverages advanced machine learning techniques and data-driven decision-making to enhance the rider-centered food delivery experience systematically.

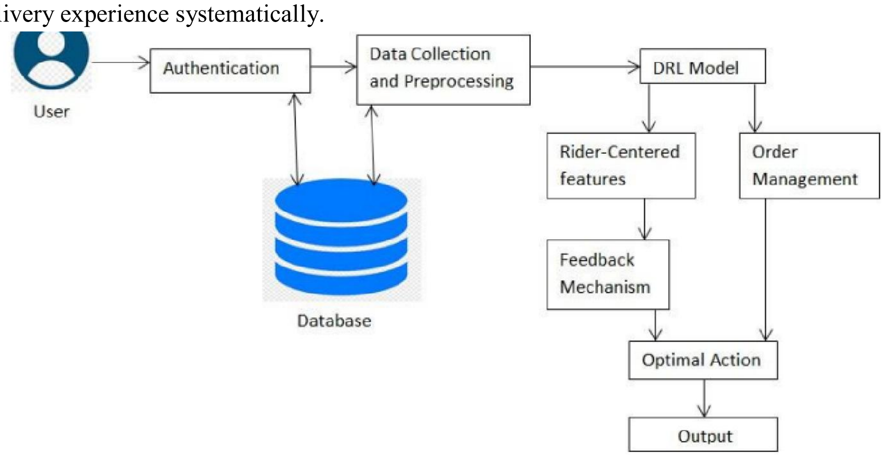

Fig. 4.1 System Architecture

4.3 Result

The Online Deep Reinforcement Learning-Based Order Recommendation Framework for a Rider-Centered Food Delivery System delivers tailored and efficient order recommendations, optimizing the food delivery experience. Leveraging sophisticated technologies, it provides personalized suggestions for customers while streamlining rider assignments. Through advanced algorithms and data processing, the system ensures timely and accurate order deliveries, enhancing user satisfaction. By continuously learning from user interactions and feedback, it evolves to meet evolving preferences, ultimately providing a seamless and rewarding food delivery experience for both customers and delivery riders.

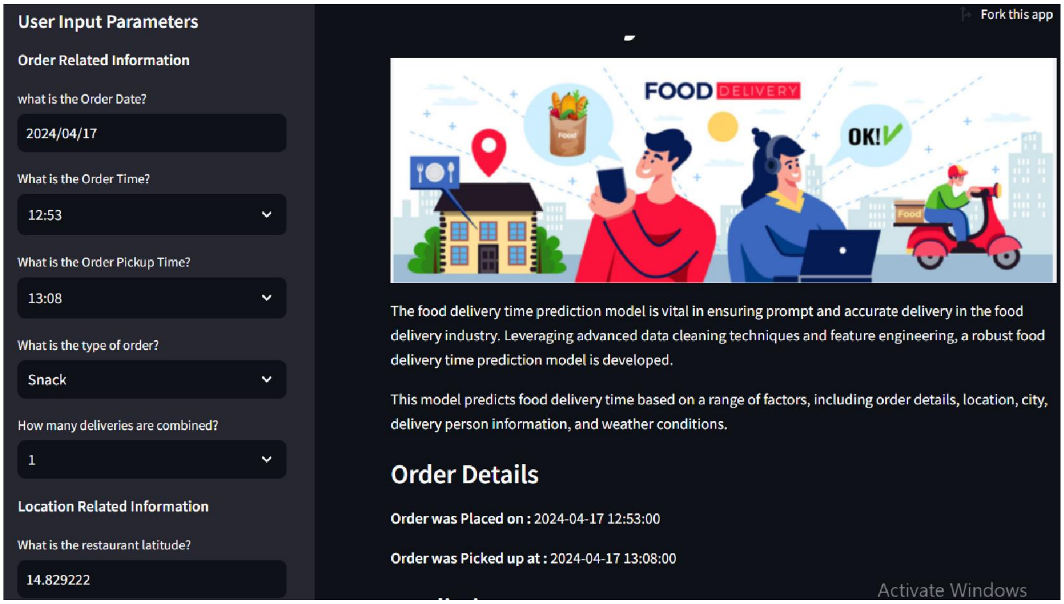

Fig. 4.2 Output

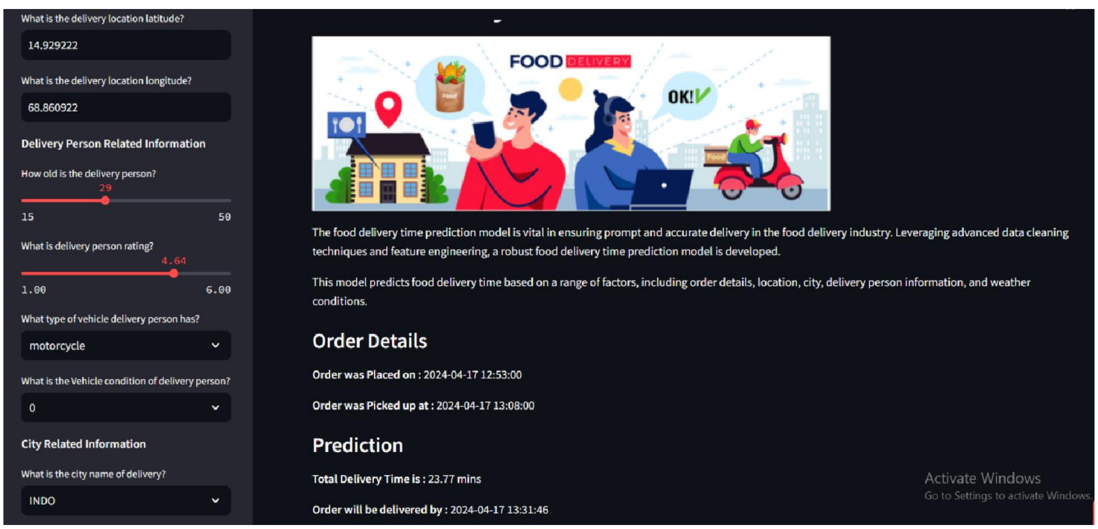

Fig. 4.3 Output

V. CONCLUSION

The decision-making problem within the rider-centered food delivery system, formulated as a Markov Decision Process (MDP), necessitates a sophisticated approach to address challenges such as dynamism and uncertain rider behaviors. To tackle these complexities and effectively utilize feedback information, we propose the Deep Reinforcement Learningbased Order Recommendation (DRLOR) framework. DRLOR leverages online deep reinforcement learning, offering several advantages over traditional recommendation models. Firstly, the Actor-Critic (AC) network facilitates continuous policy updates based on real-time rider feedback, resulting in shorter interaction sessions and faster service provision for customers. Secondly, the Feedback Correlation (FC) network employs an attention mechanism to identify and mitigate false negative feedback, while also capturing the intricate relationships between different feedback data. This approach aids in constructing robust state embedding's that accurately describe the diverse states of riders, thereby enhancing the overall performance and efficiency of the recommendation system

VI. ACKNOWLEDGMENT

We extend our sincere appreciation to Dr. M.A. Chaudhari from the Department of Information Technology Amrutvahini College of Engineering, Sangamner, A.Nagar, Maharashtra, for his invaluable guidance and support in the development of the Online Deep Reinforcement Learning-Based Order Recommendation Framework for Rider-Centered Food Delivery System. We also acknowledge the collaborative efforts of our team members, whose dedication and contributions have been instrumental in the success of this project. Additionally, we express our gratitude to the institution for providing the necessary resources, and to colleagues, friends, and family membersfor their unwavering encouragement and support throughout this endeavor

REFERENCES

[1]. J. Chen et al., "An imitation learning-enhanced iterated matching algorithm for ondemand food delivery," IEEE Trans. Intell. Transp. Syst., vol. 23, no. 10, pp. 18603– 18619, Oct. 2022

[2]. Xinlian Yu, "Short Term Demand Prediction for On Demand Food Delivery with Attention Based Convolutional LSTM," 20

[3]. T. Xue and Z. Jieru, "Deep Reinforcement Learning based Recommendation with Explicit User Item Interactions Modeling," 2019

[4]. Feng Liu , "Crowdsourced delivery: A review of platforms and academic literature," Omega, vol. 98, Jan. 2021, Art. no. 102139

[5]. X. Wang, L. Wang, S. Wang, Y. Yu, J. Chen, and J. Zheng, "Solving online food delivery problem via an effective hybrid algorithm with intelligent batching strategy," in Proc. Int. Conf. Intell. Comput., 2021, pp. 340–354.

[6]. A. Seghezzi, M. Winkenbach, and R. Mangiaracina, "Ondemand food delivery: A systematic literature review," Int. J. Logistics Manage., vol. 32, no. 4, pp. 1334–1355, Oct. 2021.

[7]. S. Liu, L. He, and Z.-J. Max Shen, "On-time last-mile delivery: Order assignment with travel-time predictors," Manage. Sci., vol. 67, no. 7, pp. 4095–4119, Jul. 2021.

[8]. S. Paul, S. Rathee, J. Matthew, and K. M. Adusumilli, "An optimization framework for ondemand meal delivery system," in Proc. IEEE Int. Conf. Ind. Eng. Eng. Manage. (IEEM), Dec. 2020, pp. 822–826.

[9]. H. Huang, C. Hu, J. Zhu, M. Wu, and R. Malekian, "Stochastic task scheduling in UAVbased intelligent ondemand meal delivery system," IEEE Trans. Intell. Transp. Syst., vol. 23, no. 8, pp. 13040–13054, Aug. 2021.