Reinventing Grocery Shopping with Reinforcement Learning

Mandadi Geetha Pallavi

*Coumputer Science and Engineering Vellore Institute of Technology* India, Amaravathi [mandadipallavi994@gnail.com](mailto:mandadipallavi994@gnail.com)

Mallampati Bhavishya

*Computer Science and Engineering Vellore Institute of Technology* India, Amaravati

mallampatibhavishyachowdary@gmail. com

Gandla Sreeja

*Computer Science and Engineering Vellore Institute of Technology* India, Amaravathi [gandlasreeja5@gmail.com](mailto:gandlasreeja5@gmail.com)

*Abstract*— Grocery shopping stands as a quintessential task in everyday life, yet it often involves complex decision- making processes influenced by factors such as individual preferences, budget constraints, and time limitations. In response to the intricacies of this routine chore, the integration of artificial intelligence, particularly reinforcement learning (RL), has emerged as a promising avenue for optimization. RL empowers autonomous agents to navigate the grocery shopping landscape by learning optimal strategies through iterative interactions with the environment. By employing RL algorithms, these agents can adapt and refine their decision-making processes over time, aiming to maximize cumulative rewards such as cost savings, time efficiency, and customer satisfaction. Such personalized and dynamic approaches hold significant potential for revolutionizing traditional grocery shopping experiences, offering tailored recommendations, mitigating decision fatigue, and ultimately streamlining the overall shopping journey for consumers. In this context, the application of reinforcement learning (RL) techniques serves as a powerful tool to optimize the process of grocery shopping. By formulating the problem within a framework of states, actions, transition probabilities, and rewards, RL algorithms enable an autonomous agent to learn an optimal strategy for navigating through a network of shops to fulfill a given shopping list efficiently. In this project, through the implementation of value iteration, Q-learning, and representation policy iteration methods, the agent can learn and refine its decision-making policy, ultimately simplifying and enhancing the grocery shopping experience for consumers.

***Keywords— Gym, Grocery, Reinforcement Learning, numpy, pandas.***

1. Introduction

The landscape of grocery shopping has undergone a profound transformation in recent years, with the advent of online platforms revolutionizing the way consumers acquire essential goods. In this era of digital convenience, users are presented with a vast array of choices, making the process of selecting the right products a complex and often overwhelming task. To address this challenge and elevate the grocery shopping experience, this project proposes a groundbreaking approach that integrates Q-learning, a reinforcement learning algorithm, into the realm of sentiment-driven product recommendations.

Traditional grocery shopping has been characterized by routine visits to physical stores, where consumers makechoices based on personal preferences, brand loyalty, and real-time considerations. However, the emergence of online grocery platforms has introduced new dimensions to this process, enabling users to explore and select products from the comfort of their homes. Despite this convenience, the abundance of choices in the digital realm can lead to decision fatigue, prompting the need for intelligent systems that assist users in making informed and personalized selections.

The primary objective of this project is to reinvent the grocery shopping experience by seamlessly incorporating Q- learning, an iterative reinforcement learning algorithm, with sentiment-driven product recommendations. By leveraging user-generated reviews, sentiment analysis, and the adaptive nature of Q-learning, the system aims to optimize and refine product suggestions over time, providing users with tailored recommendations aligned with their preferences and sentiments.

The rationale behind integrating sentiment analysis into the recommendation system lies in the wealth of information embedded in user reviews. Each review encapsulates not only the user's opinion but also the sentiment associated with specific products, offering valuable insights into the preferences and experiences of fellow shoppers. Q-learning, with its iterative learning approach, allows the system to adapt and evolve continuously, learning from user interactions and feedback to improve the precision and relevance of recommendations.

This project stands at the intersection of artificial intelligence, reinforcement learning, and e-commerce, promising a dynamic solution to the challenges posed by the vastness of digital grocery platforms. The integration of Q- learning in the sentiment-driven recommendation system is poised to offer users a more personalized, efficient, and enjoyable grocery shopping journey. As we delve into the details of the project, we will explore the methodology, implementation, and expected outcomes of this innovative approach, with the ultimate goal of redefining the grocery shopping experience in the digital age.

1. Objective

The primary objectives of the project are:

1. Integrating Q-Learning with Sentiment Analysis:

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

- Develop a robust integration of Q-learning, a reinforcement learning algorithm, with sentiment analysis techniques to create a dynamic and adaptive recommendation system for grocery shopping.

1. Sentiment-Driven Product Recommendations:
   * Leverage sentiment analysis on user-generated reviews to extract valuable insights into the sentiments associated with various grocery products.
   * Utilize this sentiment data to drive personalized product recommendations, aligning with individual user preferences.
2. Iterative Learning Process:
   * Implement an iterative approach with Q-learning to continuously refine and optimize the recommendation model based on user interactions and feedback.
   * Ensure that the system adapts to changing user behaviors, preferences, and evolving sentiment trends over time.
3. Seamless Integration with E-Commerce Platforms:
   * Design and implement a user-friendly interface that seamlessly integrates with existing online grocery shopping platforms.
   * Facilitate easy navigation and exploration of sentiment- driven product recommendations, enhancing the overall user experience.
4. Efficiency and Accuracy in Recommendations:
   * Strive to minimize the learning curve for the recommendation system, providing users with accurate and relevant grocery product suggestions from the outset.
   * Focus on enhancing the precision of recommendations to improve user satisfaction and engagement with the grocery shopping platform.
5. Increased User Satisfaction and Engagement:
   * Aim to create a more personalized and enjoyable grocery shopping experience for users by offering recommendations that resonate with their sentiments and preferences.
   * Measure and evaluate user satisfaction and engagement metrics to assess the impact of the Q-learning-driven sentiment-based recommendation system.
6. Insights for Retailers:
   * Provide valuable insights for retailers into evolving sentiment trends and customer preferences within the grocery domain.
   * Enable retailers to make informed decisions regarding product assortment, marketing strategies, and inventory management based on the sentiment-driven data.
7. Optimization of Grocery Supply Chain:
   * Explore the potential impact of sentiment-driven recommendations on optimizing the grocery supply chain by influencing inventory levels, reducing waste, and enhancing sustainability.
8. Literature survey

The modern era witnesses an ever-increasing demand for convenience, efficiency, and personalization in various aspects of daily life. One such area is grocery shopping, which traditionally involves navigating through multiple stores to find desired items at competitive prices. This process is not only time-consuming but also poses challenges for individuals with busy schedules or specific preferences. While online grocery platforms offer a degree of convenience, they often fall short in providing personalized recommendations tailored to individual preferences effectively. To address these challenges and enhance the grocery shopping experience, researchers and developers have explored various technologies and methodologies, one of which is the utilization of agent- based systems powered by reinforcement learning (RL) algorithms.

In their work, Kwang Hyoun Joo et al. **[ 1 ]**introduce an agent-based grocery shopping system designed to automate the process of grocery shopping while considering user preferences. Their system utilizes a network of agents to gather information from multiple grocery stores, compare it with user preferences, and adapt over time based on user feedback.

The concept of utilizing agent-based systems in e- commerce, as discussed by Kwang Hyoun Joo et al., resonates with the research conducted by T. Kinoshita. In their research, Kinoshita explores the potential of agent- based systems in organizing and coordinating various tasks within e-commerce environments. Their work highlights the role of organization agents in managing primitive agents and facilitating communication, which aligns with the organizational structure proposed in the agent-based grocery shopping system.

Furthermore, the approach of integrating user preferences into decision-making processes bears resemblance to the research conducted by N. Shiratori. In their work on personalized recommendation systems, Shiratori et al. emphasize the importance of considering user preferences and feedback to enhance the effectiveness of recommendation algorithms. This aligns with the goal of the agent-based grocery shopping system to adapt to user preferences over time by learning from previous shopping results.

By synthesizing insights from the works of Kwang Hyoun Joo, T. Kinoshita, and N. Shiratori, it becomes evident that the proposed agent-based grocery shopping system represents a convergence of ideas from the fields of agent- based systems, e-commerce organization, and personalized recommendation algorithms. The system aims to address the challenges of traditional grocery shopping by automating the process, considering user preferences, and facilitating efficient decision-making.

In their research, Hongying Du and Michael N. Huhns introduce **[ 2 ]** an innovative approach to social grocery shopping, where customers exchange information on item prices and quantities to identify the best deals and most convenient shopping plans. This strategy aims to empower consumers by enabling them to make more informed purchasing decisions and potentially achieve significant cost savings. Central to their approach is the utilization of agents

representing individual customers, which helps facilitate practical implementation and fosters trust among participants by potentially learning which agents can provide reliable information. By leveraging real-world shopping lists based on the U.S. Consumer Price Index, the authors ensure the realism and relevance of their findings.

The proposed system addresses a common challenge faced by consumers in traditional grocery shopping: the lack of easily accessible price comparison tools and the inconvenience of visiting multiple stores to find the best deals. By envisioning an online platform where customers can share price information and obtain pointers to stores offering the lowest total prices for their desired items, Du and Huhns aim to democratize the shopping experience and encourage fairer interactions between customers and stores. However, the success of such a system hinge on the willingness of customers to actively participate and contribute accurate price data, highlighting the importance of effective implementation and incentivization strategies.

A key aspect of the study involves assessing the robustness of the proposed multi-agent shopping system in the face of potential errors and manipulation, such as inaccuracies in reported prices or changes in store pricing strategies. Through simulations incorporating random and systematic errors, the authors evaluate the system's ability to withstand various challenges and maintain its effectiveness in optimizing cost savings for customers. This evaluation provides valuable insights into the feasibility and reliability of the proposed approach, helping to inform discussions on its practical implementation and potential benefits for consumers in real-world grocery shopping scenarios.

The key difference between this paper and the previous one lies in the approach to grocery shopping optimization. While both papers aim to enhance the shopping experience by leveraging agent-based systems, the focus here is on social interaction among customers to share price information and optimize savings. In contrast, the previous paper emphasizes automation and adaptation of agents to user preferences. Additionally, this paper highlights the potential savings and robustness of the proposed multi-agent shopping system in the presence of errors in reported prices, which is a unique aspect not explored in the previous work.

In their work, Kwang Hyoun Joo, T. Kinoshita, and N. Shiratori **[ 3 ]** introduce an agent-based grocery shopping system aimed at automating the shopping process by gathering information from multiple stores and comparing it with user preferences. Their system utilizes role agents, including user agents, information management agents, and store server agents, to facilitate cooperation and achieve user-defined goals.

The proposed system focuses on reducing user effort and saving time by purchasing the best groceries based on user preference. It covers functional requirements of a grocery shopping system and supports the five stages of consumer buying behavior model. However, it's worth noting that this paper does not explicitly mention the incorporation of reinforcement learning for adapting to user preferences over time, which could be a potential difference compared to other research works.

The third paper could build upon the findings of the first paper, incorporating feedback, addressing limitations, or presenting new developments in the agent-based grocery

shopping system. Overall, while the core concept of an agent-based grocery shopping system remains consistent across both papers, differences in scope, detail, publication venue, and temporal context could contribute to variations in their content and contributions.

In their work, L. Benedicenti, Xuguang Chen, Xiaoran Cao, and R. Paranjape **[ 4 ]** present an agent-based shopping system designed to assist supermarket shoppers both at home and during their shopping trips. The system utilizes lightweight agent implementation called TEEMA (TRLabs Execution Environment for Mobile Agents), which is built upon a microkernel concept providing basic services for agent communication, migration, and location. Additional services such as name services, storage services, security services, and database services can be added on top of TEEMA to enhance functionality.

The proposed system facilitates supermarket shopping by enabling users to send agents with shopping lists to be selected supermarkets, where the agents retrieve limited price lists. The system also incorporates a residential gateway to protect user information during agent travel. Upon returning to the user, the system informs them of the search results, and if the user decides to visit a supermarket, an agent is sent there through the residential gateway to access complete price lists. The system architecture is distributed, with logical components located at the user's location, residential gateway, mobile terminal, and participating supermarkets.

The authors highlight the advantages of their agent-based approach, emphasizing its ability to automate inventory management, assist in supermarket selection based on price and availability, and provide real-time updates on special offers. Additionally, they discuss the integration strategy to make the system compatible with legacy database and server software. However, they also acknowledge the challenges and limitations, particularly regarding privacy protection, user authentication, and system robustness in the face of potential errors or manipulation. The comparison with other supermarket shopping systems underscores the unique strengths and weaknesses of the proposed agent-based approach in optimizing the shopping experience for consumers.

E-commerce has increased tremendously in recent decades because of improvements in information and telecommunications technology along with changes in societal lifestyles. As highlighted by Adrian Serrano- Hernandez et al. (2017) **[ 5 ]** , horizontal cooperation among companies can lead to significant benefits, such as reducing transportation costs, improving service quality, diminishing environmental impact, mitigating risk, and enhancing market share. More recently, e-grocery (groceries purchased online) including fresh vegetables and fruit, is gaining importance as the most-efficient delivery system in terms of cost and time. Javier Faulin and Rocio de la Torre have noted the logistics challenges associated with e-groceries, including food safety issues, differences in storage temperatures, and perishability over time (Fredriksson and Liljestrand 2015).

In this study, the authors evaluate the effect of cooperation- based policies on service quality among different supermarkets in Pamplona, Spain. Concerning the methodology, they deploy, firstly, a detailed survey in

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Pamplona to model e-grocery demand patterns. As highlighted by Luis Cadarso, consumers and sellers often have conflicting preferences regarding product perishability, with consumers preferring longer shelf lives and sellers benefiting from shipping shorter-lived items first to reduce food waste (Teller et al. 2018; Fikar 2018). Secondly, the authors develop an agent-based simulation model for generating scenarios in cooperative and non-cooperative settings, considering the real data obtained from the survey analysis. Thus, a Vehicle Routing Problem is dynamically generated and solved within the simulation framework using a biased-randomization algorithm. Finally, the results show significant reductions in lead times and better customer satisfaction when employing horizontal cooperation in e- grocery distribution.  **Table 1:** Contributions in E-Grocery, Horizontal Cooperation, and Agent-Based Simulation | | | | |  | | system |  | n Cao; R.  Paranja pe | agents with shopping lists to select supermarkets, retrieve price lists, and receive real-time updates on special offers, ensuring privacy and  compatibility with legacy software. |
| 5 | | Agent- based simulat ion improv es E- grocery deliveri es using horizon tal coopera  tion. | 14-18  Decembe r 2020 | Adrian Serrano  -  Hernan dez; Javier Faulin; Rocio de la Torre; Luis Cadars o | An agent-based simulation model to evaluate the effect of horizontal cooperation on lead times and customer satisfaction in e-grocery  distribution. |
| **S.N**  **o** | **Title** | **Date of**  **conferen ce** | **Author s** | **Proposed Method** |  |  |
| 1 | Agent- based grocery shoppin g system based on user's  prefere nce | 04-07  July 2000 | Kwang Hyoun Joo; T. Kinoshi ta; N. Shirato ri | An agent-based grocery shopping  system automates shopping by gathering information from  multiple stores, comparing it with user preferences, and adapting over time through feedback. |  |
|  | 6 | Agent- based simulat ion of consum er behavio r in  grocery shoppin g on a  regiona l level | August 2007 | Tilman A.  Schenk  ,  Günter Löffler, Jürgen Rauh | The proposed method is an agent-based micro model for simulating spatial choice in grocery shopping behavior based on individual population. |
| 2 | A  Multiag ent System Approa ch to Grocer y Shoppi ng | 2011 | Hongyi ng Du & Michae l N.  Huhns | The proposed method entails customers  exchanging information on item |  |
| prices and quantities to optimize savings and convenience,  facilitated by agents representing  customers. | | V | | | | |
| 3 | Design and implem entatio n of an agent- based grocery shoppin g system. | 25-11  Nov 2000 | Kwang Hyoun JOO  Tetsuo KINOS HITA  Norio SHIRA TORI | The proposed method involves an agent- based grocery shopping system that consists of three role agents: a user agent, an information management agent, and a store server agent, which cooperate to purchase groceries according to user  preferences. | |
| 4 | An agent- based shoppin  g | 02-05  May 2004 | L.Bene dicenti; Xuguan g Chen;  Xiaora | The method involves employing TEEMA an agent, to facilitate agent-based shopping,  enabling users to send | |

I. Preceding Work & Drawbacks

# Preceding Work:

The proposed development of an agent-based grocery shopping system addresses longstanding challenges inherent in traditional grocery shopping processes. Traditional methods often demand significant time and effort from consumers, requiring them to navigate through multiple stores to find desired items at competitive prices. Moreover, these methods may not sufficiently cater to individual preferences, leaving consumers with suboptimal choices. Recognizing these inefficiencies, the proposed system aims to revolutionize the grocery shopping experience by leveraging advanced technologies and algorithms to streamline the process.

Existing literature in e-commerce and consumer behavior underscores the importance of personalized recommendations in enhancing user satisfaction and engagement. Studies by researchers like N. Shiratori et al. have delved into the realm of personalized recommendation systems, which aim to tailor product suggestions to individual user preferences. However, traditional recommendation approaches often lack real-time

adaptability, failing to consider dynamic factors such as price fluctuations and item availability. By integrating reinforcement learning (RL) algorithms into the proposed system, it bridges this gap by enabling agents to learn and adapt their decision-making processes based on evolving conditions, thereby providing more personalized and relevant recommendations to users.

Furthermore, research in RL, particularly the foundational work by Sutton and Barto, has demonstrated the efficacy of RL algorithms in optimizing decision-making processes in dynamic environments. By modeling the grocery shopping problem as a Markov Decision Process (MDP) and employing Q-learning, the proposed system empowers agents to make informed decisions regarding store selection and item purchases. This approach capitalizes on the inherent ability of RL algorithms to learn from experience and iteratively refine decision-making policies, ultimately optimizing the overall grocery shopping experience for consumers.

The integration of RL algorithms in the proposed system represents a significant departure from traditional e- commerce platforms, which often rely on static recommendation systems. By harnessing the power of RL, the system can adapt to changing user preferences, market conditions, and store offerings in real-time, thereby ensuring a more personalized and efficient shopping experience. Moreover, the proposed system aligns with broader trends in automation and optimization within the e-commerce landscape, reflecting a concerted effort to leverage cutting- edge technologies to address longstanding challenges and enhance consumer satisfaction.

In summary, the proposed agent-based grocery shopping system represents a convergence of insights from e- commerce, personalized recommendation systems, and reinforcement learning. By leveraging advanced algorithms and real-time data, the system seeks to revolutionize the grocery shopping experience, offering consumers unprecedented levels of convenience, personalization, and efficiency. Through iterative refinement and adaptation, the system holds the promise of reshaping the future of grocery shopping, making it more seamless, tailored, and enjoyable for consumers worldwide.

# Drawbacks:

While the proposed agent-based grocery shopping system holds promise for revolutionizing the grocery shopping experience, it also faces several potential drawbacks and challenges that warrant consideration:

**Complexity and Scalability:** Implementing an agent-based system can introduce complexity, particularly in terms of system architecture and scalability. As the number of users and transactions increases, managing a large network of agents and ensuring efficient communication between them may become challenging. This complexity could hinder the system's ability to scale effectively to accommodate a growing user base and handle increasing transaction volumes.

**Data Privacy and Security Concerns:** The system's reliance on gathering and analyzing user data to tailor recommendations raises significant privacy and security concerns. Collecting sensitive information about users' preferences, purchasing habits, and location data may pose risks if not adequately protected. Ensuring robust data

privacy measures and compliance with regulations such as GDPR (General Data Protection Regulation) is essential to maintain user trust and mitigate potential security breaches. **Algorithm Bias and Fairness**: The use of reinforcement learning algorithms introduces the risk of algorithmic bias, where the system's recommendations may reflect and perpetuate existing societal biases or preferences. Without careful design and oversight, the system may inadvertently favor certain demographics or product categories over others, leading to disparities in recommendation accuracy and fairness. Addressing algorithmic bias requires ongoing monitoring, evaluation, and mitigation strategies to ensure equitable treatment for all users.

**Limited User Control and Transparency:** While the system aims to automate and optimize the grocery shopping process, it may limit user control and transparency over decision-making. Users may feel disenfranchised if they perceive the system as making decisions on their behalf without sufficient input or explanation. Providing users with greater control over preferences, recommendations, and decision-making processes, as well as enhancing transparency in how the system operates, is essential to foster trust and user acceptance.

**Dependency on External Factors:** The effectiveness of the system is contingent on various external factors, including the availability and accuracy of data from grocery stores, fluctuations in market conditions, and the reliability of communication networks. Any disruptions or inaccuracies in these external factors could compromise the system's performance and user experience. Mitigating such dependencies may require redundancies, fallback mechanisms, and continuous monitoring to ensure robustness and resilience.

V.Hardware and software requirements

# Hardware Requirements:

**Processor:**

A multi-core processor with a clock speed of 2.5 GHz or higher is recommended to handle the computational demands of reinforcement learning algorithms, particularly during training phases.

# Memory (RAM):

A minimum of 16 GB RAM is advised to accommodate the memory-intensive nature of reinforcement learning tasks, especially when dealing with large datasets.

# Graphics Processing Unit (GPU):

An NVIDIA GPU with CUDA support is highly beneficial for accelerating the training of reinforcement learning models. A GPU with at least 4 GB of VRAM is recommended.

# Storage:

An SSD with a minimum of 512 GB storage capacity is recommended for faster data access, model storage, and retrieval.

# Internet Connectivity:

A stable and high-speed internet connection is essential for downloading datasets, model updates, and potential cloud- based training or deployment.

# Software Requirements:

**Operating System:**

Linux-based operating systems like Ubuntu 18.04 or higher are preferable for their compatibility with many machine learning libraries and frameworks.

# Python:

Python 3.x is the programming language of choice for implementing reinforcement learning algorithms. Essential libraries include NumPy, TensorFlow (or PyTorch), and OpenAI Gym.

# Reinforcement Learning Libraries:

OpenAI Gym: A toolkit for developing and comparing reinforcement learning algorithms.

TensorFlow or PyTorch: Deep learning frameworks for implementing neural network architectures used in reinforcement learning.

# Development Environment:

Jupyter Notebook or Visual Studio Code can serve as the integrated development environment (IDE) for coding, testing, and debugging reinforcement learning algorithms.

**Simulation Environment (Optional):**If the project involves simulating a grocery shopping environment, platforms like Unity ML-Agents or custom environments within OpenAI Gym can be used.

# Version Control:

Git and a version control platform like GitHub for tracking changes, collaborating with team members, and maintaining the project's version history.

VI.Modules

**Simulation Environment Module**: This module will contain functions/classes to simulate the environment, including shop availability, pricing, and distances between shops.

**Agent Module:** This module will contain classes/functions representing the agents, their decision- making processes, and interaction with the environment.

**Reward Module:** Contains functions/classes to calculate rewards for different actions taken by the agent.

**Transition Probability Module:** Contains functions/classes to calculate transition probabilities based on actions and outcomes.

**Value Iteration Module:** Implements the value iteration algorithm to find the optimal policy.

**Q-Learning Module:** Implements the Q-learning algorithm to learn the optimal policy through trial and error.

**Representation Policy Iteration Module:** Implements the representation policy iteration algorithm for policy approximation.

1. Project flow

# Data Setup:

In this part, the code initializes various parameters required for the problem. This includes the prices of items (MRPs), the number of shops, the number of items, etc. MRPs are stored in a dictionary where each item is associated with its price.

# State Space Definition:

The state space is a set of all possible states the agent can be in. In this case, each state consists of two components: the shop number and the buying status of items. The buying status of items is represented as a binary vector indicating whether each item is bought or not.

# Transition Probabilities and Rewards:

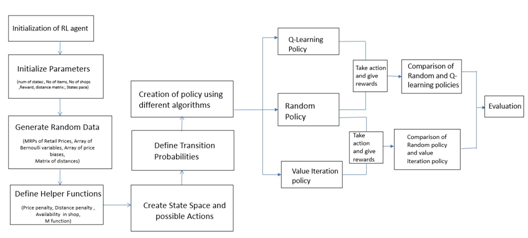
Transition probabilities determine the likelihood of transitioning from one state to another when an action is taken. Rewards represent the immediate benefit (or cost) associated with taking an action from a particular state to another state.

The transition probabilities and rewards for each possible action (selecting a shop) from a given state to another state. It considers factors such as availability of items in shops, prices of items, and distances between shops.

# Q-learning Algorithm:

Q-learning is a model-free reinforcement learning algorithm for learning optimal policies. It learns a Q-value for each state-action pair, representing the expected future reward of taking that action in that state.

The Q-learning function iterates over episodes, where each episode consists of multiple steps. In each step, the agent selects an action based on an epsilon-greedy policy (balancing exploration and exploitation), updates Q-values based on the observed reward and transitions, and collects rewards.



Fig(1). Complete Process of the Grocery Shopping

# Testing:

After learning the optimal policy using Q-learning, the code tests this policy against a random policy. It does so by simulating shopping scenarios and comparing the rewards obtained using both policies.

# Visualization:

The rewards obtained over episodes during Q-learning and rewards obtained from testing the learned policy against the random policy. This visualization helps in understanding the learning progress and performance of the learned policy.

The best action (shop) for each state, indicating the learned optimal policy. It also displays plots of rewards obtained during learning and testing, providing insights into the learning process and policy performance.

# Functions:

These are utility functions used in the Q-learning algorithm and reward calculation process. They calculate penalties based on item prices, shop distances, item availability probabilities, and parameters for transition probabilities.

This executes the Q-learning algorithm for a specified number of episodes and steps. It updates Q-values based on observed rewards and transitions, iteratively improving the learned policy.

This section tests the learned policy against a random policy for multiple scenarios and collects rewards. It helps evaluate the performance of the learned policy in comparison to a baseline (random) policy.

Overall, aim of this project comprehensive Q-learning- based solution to optimize shopping decisions across multiple shops, considering factors such as item availability, prices, and shop distances.

VII. Proposed model

To address the above challenges in grocery shopping faced by the people, this project has proposed a solution. In this we created an environment representing of various shops available for grocery shopping. Each shop will have its own inventory of items along with their availability (whether the item is in stock or not) and pricing patterns (the price of each item). Here we also define the distances between each pair of shops in the environment. This could be based on real-world distances or simplified distances for simulation purposes. And then the state of the agent will include the current shop it is in and the buying status, which indicates whether each item on the shopping list has been bought or not. The agent can take actions to move to different shops based on the current state. The available actions will be a list of shops that the agent can visit from the current shop. The agent will make decisions based on a policy, which determines the action to take given a state. This policy can be learned through reinforcement learning algorithms like Q-learning or policy iteration methods.

If the agent finds the required item in a shop, it receives a positive reward. The reward could be a fixed value or based on the price of the item. If the item is not found in the shop, the agent might receive a penalty to incentivize finding the item elsewhere. The agent might incur a penalty for traveling between shops to encourage minimizing travel distance. The agent might receive a penalty based on the prices of the items bought, encouraging it to find cheaper options.

The transition model defines the probabilities of moving from one state to another based on the action taken by the agent and the outcome. The probability of finding an item in a shop could be based on historical data or modeled using a distribution (e.g., Bernoulli distribution). The probability of moving from one shop to another could depend on the distance between the shops and possibly other factors like traffic conditions. The transition probabilities might also consider the prices of items in different shops, favoring shops with lower prices if available and shows the best available option to the user so that they can save time and won’t get confused in selecting the required products from different vendors.

By incorporating these elements into the simulated environment and agent design, you can create a realistic and effective system for optimizing grocery shopping. The rewards and transition probabilities play a crucial role in guiding the agent's decision-making process, while the environment provides a realistic setting for testing and evaluating different strategies.

7.1 Tentative proposed Model

To address the above challenges in grocery shopping faced by the people, this project has proposed a solution. In this we created an environment representing of various shops available for grocery shopping. Each shop will have its own inventory of items along with their availability (whether the item is in stock or not) and pricing patterns (the price of each item). Here we also define the distances between each pair of shops in the environment. This could be based on real-world distances or simplified distances for simulation purposes. And then the state of the agent will include the current shop it is in and the buying status, which indicates whether each item on the shopping list has been bought or not. The agent can take actions to move to different shops based on the current state. The available actions will be a list of shops that the agent can visit from the current shop. The agent will make decisions based on a policy, which determines the action to take given a state. This policy can be learned through reinforcement learning algorithms like Q-learning or policy iteration methods. If the agent finds the required item in a shop, it receives a positive reward. The reward could be a fixed value or based on the price of the item. If the item is not found in the shop, the agent might receive a penalty to incentivize finding the item elsewhere. The agent might incur a penalty for traveling between shops to encourage minimizing travel distance. The agent might receive a penalty based on the prices of the items bought, encouraging it to find cheaper options. The transition model defines the probabilities of moving from one state to another based on the action taken by the agent and the outcome. The probability of finding an item in a shop could be based on historical data or modeled using a distribution (e.g., Bernoulli distribution). The probability of moving from one shop to another could depend on the distance between the shops and possibly other factors like traffic conditions. The transition probabilities might also consider the prices of items in different shops, favoring shops with lower prices if available and shows the best available option to the user so that they can save time and won’t get confused in selecting the required products from different vendors. By incorporating these elements into the simulated environment and agent design, you can create a realistic and effective system for optimizing grocery shopping. The rewards and transition probabilities play a crucial role in guiding the agent's decision-making process, while the environment provides a realistic setting for testing and evaluating different strategies.

**List of Abbreviations:**

Bernoulli distribution:

The Bernoulli distribution is a discrete probability distribution that models a single binary outcome, typically denoted as success or failure, where success occurs with a probability *p* and failure with probability 1−*p*.

Model-Function(M-Function):

The M-function maps states and actions to the probabilities of success and failure, aiding the agent in decision-making by quantifying the likelihood of receiving rewards for different actions.

Q-learning:

Q-learning is a model-free reinforcement learning algorithm that iteratively updates action-values to find the optimal policy for maximizing cumulative rewards in a Markov decision process.

• Shops= [S1, S2 ,….., Sn]

• buying\_status=[I1,I2,…..Im] where Ii be;longs to {Y,N} depicting whether the item i has been bought or not

• States={current\_shop,current\_buying\_status}

• State Space=No of shops x 2^(no: Of items to buy)=n\*(2^m) • Actions=[ S1, S2 ,….., Sn]

Item Availability Model

Function availability\_in\_shop

We will use a Bernoulli Distribution for each item in each shop.Either the item is available or it’s not.(X=0 or 1) This function returns the probability P(X=next\_status) using the distribution for that particular item and shop.

M function The M function used ensures that if the action Si chosen ,then the probability of ending up in shop Sj is greater than that of ending up in shop Sk if and only if Sj is closer to Si than Sk. The function is described as:

Price of Item Model:

Function:price\_penalty

We will use a Gaussian Distribution for price of each item in each shop.The MRP,the price fluctuates slightly around the MRP.Based on the price a penalty (cost) is added.

Distance Travelled Model

Function:distance\_penalty

We will have fixed distances between the shops.To account for real world conditions like traffic congestions,we use a Poisson Distribution- parameter passed is distance travelled.

Transition Probabilities and Rewards(for 1 item to buy):

1.If item has been bought(current\_status=Y)then,session terminates

2. If chosen to try same shop again (action=current\_shop,next\_shop=current\_shop) P[current\_shop][current\_status][action][next\_shop][next\_status]=

availability\_in\_shop(next\_shop,next\_status)

•a.ItemFound:

R[current\_shop][current\_status][action][next\_shop][next\_status]=reward\_buying+distance\_penalty(distance(current\_shop,next\_shop))+price\_penalty(Price[next\_shop])

•b.ItemNotFound:

R[current\_shop][current\_status][action][next\_shop][next\_status]=distance\_penalty(distance(current\_shop,next\_shop))

2. If chosen to go to another shop

• If you end up in the chosen shop (next\_shop=action) P[current\_shop][current\_status][action][next\_shop]

[next\_status]=0.9\*availability\_in\_shop(next\_shop,

next\_status) - i.ItemFound:R[current\_shop][current\_status][action]

[next\_shop][next\_status]=reward\_buying+

distance\_penalty(distance(current\_shop,next\_shop))+ price\_penalty(Price[next\_shop])

- ii.ItemNotFound:

R[current\_shop][current\_status][action][next\_shop][next \_status]=distance\_penalty(distance(current\_shop,next\_shop))

•b. If you end up in some shop not chosen(next\_shop!=action) P[current\_shop][current\_status][action][next\_shop][next\_status]=0.1\*availability\_in\_shop(next\_shop,next\_status)\*M(next\_shop,current\_shop) -

i.ItemFound:

R[current\_shop][current\_status][action][next\_stop][next\_status]=reward\_buying+distance\_penalty(distance(current\_shop,action)+distance(action,next\_shop))+price\_penalty(Price[next\_shop])

- ii.ItemNotFound:

R[current\_shop][current\_status][action][next\_shop]

[next\_status]=distance\_penalty(distance(current\_shop,action)+distance(action,next\_shop))

**7.2 Reinforcement Learning Methods**

**Q-learning:** Q-learning is a model-free RL algorithm that learns the optimal actionselection policy for a given environment without requiring a model of the environment's dynamics. - In the context of grocery shopping optimization, Q-learning is used by the autonomous agent to learn the Q-values, which represent the expected cumulative rewards for taking a particular action in a given state. -

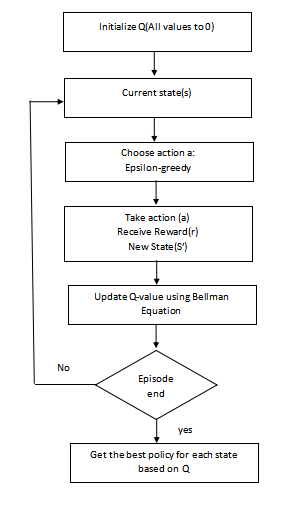
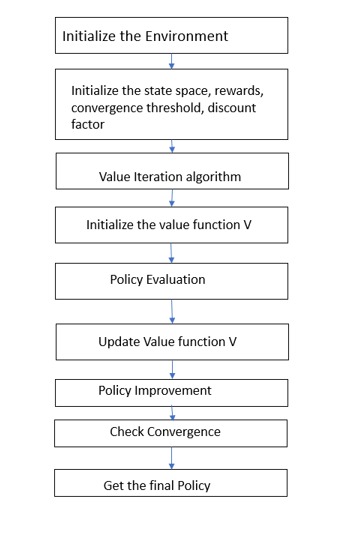


Fig (2). Implementation using Q\_Learning

The agent explores the grocery shopping environment by interacting with it, making decisions (actions) based on its current state and receiving feedback (rewards) based on the outcomes of its actions. - During each interaction, the agent updates its Q-values using the Q-learning update rule, which incorporates the observed reward and the maximum Q-value of the next state. - Over time, through repeated interactions and updates, the agent learns the optimal Q-values, enabling it to make informed decisions about which actions to take in different states to maximize cumulative rewards, such as cost savings, time efficiency, and customer satisfaction.

**Value Iteration: -**

****

**Fig (3). Implementation of Value Iteration**

Value iteration is a model-based RL algorithm that iteratively computes the optimal value function for each state in the environment, representing the expected cumulative rewards achievable from that state onwards. - In the grocery shopping optimization project, value iteration is used to compute the optimal value function by iteratively updating the values of each state based on the Bellman equation, which expresses the relationship between the value of a state and the values of its neighboring states. - The agent initializes the value function arbitrarily and then iteratively updates it until convergence, with each iteration bringing the values closer to their optimal values. - Once the optimal value function is computed, the agent can derive the optimal policy by selecting the action with the highest expected value in each state, enabling it to make decisions that maximize cumulative rewards. - Value iteration provides a principled approach to finding the optimal policy in environments with known dynamics, making it particularly useful for grocery shopping optimization when the structure of the shopping environment is well-defined.

VIII .RESULT:

The graph depicts the relationship between the rewards obtained in each episode and the number of episodes, showcasing how progression of rewards over episodes, providing insight into the learning performance of the Q-learning algorithm. The upward trend indicates the algorithm's ability to gradually maximize rewards as it gains more experience in the environment.fig.(2)

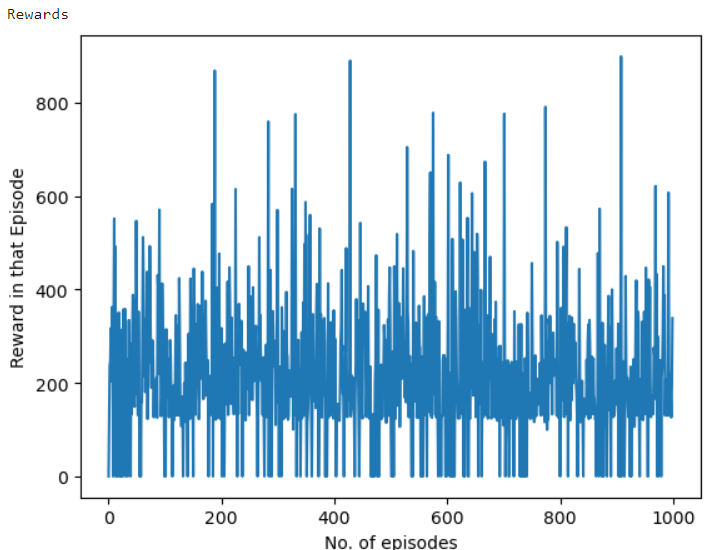


Figure.2- Rewards on each Iteration

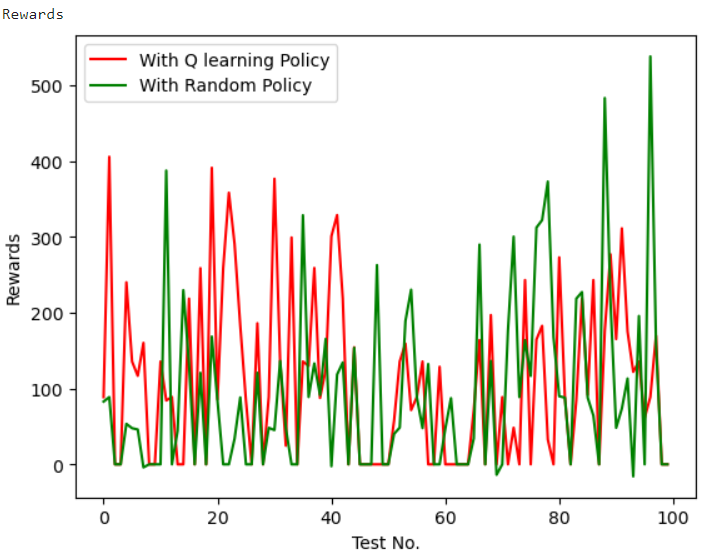


Figure.3 – Comaprision of Q-Learning with Random Algorithm for rewards Vs Test No

This graph, runs 100 tests comparing the rewards obtained using the Q-learning policy against those obtained using a random policy. For each test, it selects a random starting state, executes actions according to the respective policies, and accumulates rewards until the episode terminates or a maximum step limit is reached. The rewards from each test are plotted, with the x-axis representing the test number and the y-axis representing the rewards achieved, allowing for a comparison between the two policies' performance

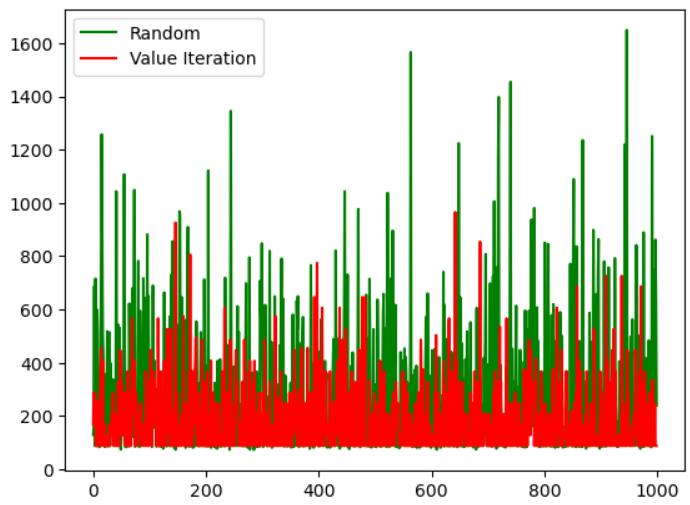


Figure.4- Value Iteration Vs Random Algorithm

This graph iterates 1000 times, each time performing value iteration to derive an optimal policy. It then compares the rewards accumulated using a random policy versus the rewards obtained with the optimal policy derived from value iteration, totaling the rewards for each iteration and plotting them for visual comparison. Any encountered exceptions during execution are handled and displayed.

IX. Conclusion

In conclusion, the application of reinforcement learning (RL) techniques offers a promising avenue for optimizing the grocery shopping experience. Through iterative interactions with the environment, RL-powered agents can learn and refine optimal strategies tailored to individual preferences, budget constraints, and time limitations. By maximizing cumulative rewards such as cost savings, time efficiency, and customer satisfaction, these agents streamline the shopping journey for consumers. The implementation of value iteration, Q-learning, and representation policy iteration methods enables continuous improvement in decision-making policies, leading to enhanced shopping experiences

X. future scope

Moving forward, the evolution of AI-enabled grocery shopping optimization presents several exciting opportunities for further exploration and development:

1. Enhanced Personalization: Continuously refining RL algorithms to provide even more personalized recommendations, considering factors like dietary preferences, health goals, and cultural backgrounds, to offer a truly tailored shopping experience.

2. Multi-Agent Systems: Advancing research in multi-agent systems to model complex interactions among shoppers, retailers, and other stakeholders, thereby facilitating collaborative optimization strategies and improving overall system efficiency.

3. Real-Time Adaptation: Developing adaptive RL algorithms capable of dynamically adjusting shopping strategies in response to real-time changes in factors such as product availability, pricing fluctuations, and store layouts.

4. Sustainability Integration: Incorporating sustainability metrics into RL frameworks to promote environmentally conscious shopping behaviors, including reducing food waste, selecting eco-friendly products, and optimizing transportation routes to minimize carbon footprint.

5. Integration with E-commerce: Expanding RL-based optimization techniques to online grocery platforms, enabling seamless integration between physical and digital shopping experiences, and leveraging data-driven insights to enhance customer satisfaction and operational efficiency. By embracing these future directions and continuing to innovate in the field of AIenabled grocery shopping, we can strive towards creating more efficient, convenient, and sustainable shopping experiences for consumers worldwide.

XI. References

1. Joo, Kwang Hyoun, Tetsuo Kinoshita, and Norio Shiratori. "Agent-based grocery shopping system based on user's preference." Proceedings Seventh International Conference on Parallel and Distributed Systems: Workshops. IEEE, 2000.
2. Du, Hongying, and Michael N. Huhns. "A Multiagent system approach to grocery shopping." Advances on Practical Applications of Agents and Multiagent Systems: 9th International Conference on Practical Applications of Agents and Multiagent Systems. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011.
3. Joo, Kwang Hyoun, Tetsuo Kinoshita, and Norio Shiratori. "Design and implementation of an agentbased grocery shopping system." IEICE TRANSACTIONS on Information and Systems 83.11 (2000): 1940-1951.

[4]L. Benedicenti, Xuguang Chen, Xiaoran Cao and R. Paranjape, "An agent-based shopping system," Canadian Conference on Electrical and Computer Engineering 2004 (IEEE Cat. No.04CH37513), Niagara Falls, ON, Canada, 2004, pp.703-705Vol.2, doi:

10.1109/CCECE.2004.1345210.

1. Serrano-Hernandez, Adrian, et al. "Agent-based simulation improves E-grocery deliveries using horizontal cooperation." 2020 Winter Simulation Conference (WSC). IEEE, 2020.
2. Schenk, Tilman A., Günter Löffler, and Jürgen Rauh. "Agent-based simulation of consumer behavior in grocery shopping on a regional level." Journal of Business research

60.8 (2007): 894-903.

1. Yu-San Chen, Chang-Franw Lee, Proceedings of the 2009 International Conference on Artificial Intelligence (ICAI'09), “Agent-Based Simulation of Consumer Purchasing Behaviour in a Virtual Environment”.
2. Giuseppe Mangioni, Rosario Sinatra, Vincenzo Nicosia, Vito Latora, Journal of Artificial Societies and Social Simulation, “A Multi-Agent System for Modelling Consumer Behaviour in Supermarkets”.
3. Yuri Kaniovski, Martin Summer, Journal of Artificial Societies and Social Simulation, “Agentbased Modeling of Store Choice Dynamics”.
4. Kozma Zsolt, Máté Csorba, 7th International Conference on Applied Informatics,2015, “Agentbased Simulation of Consumer Behaviour in the Retail Sector”.
5. Michele Piccione, Annual Review of Economics, 2016, “Agent-Based Modeling in Consumer Economics”.
6. Pavle Boskovic, Srdjan Boskovic, Aleksandar Ivanovic, 2018 17th International Symposium INFOTEH-JAHORINA (INFOTEH), “Agent-Based Model of Consumer Decision Making Process in Online Grocery Shopping”.

[13]”Jungsoo Park, Hyunju Park, 2014 International Conference on Control, Automation and Information Sciences (ICCAIS), “Agent-Based Simulation for Understanding Consumers' Shopping Behavior in Virtual Supermarket.”