Enhancing Sparse Data Performance in E-commerce Dynamic Pricing with Reinforcement Learning and Pre-Trained Learning

1st Yuchen Liu
Department of Computing
Xi'an Jiaotong-Liverpool University
Suzhou, China
Yuchen.Liu21@student.xjtlu.edu.cn

2nd Ka Lok Man
Department of Computing
Xi'an Jiaotong-Liverpool University
Suzhou, China
Ka.Man@xjtlu.edu.cn

3rd Gangmin Li
Faculty of Creative Arts, Technologies and Science
University of Bedfordshire
Luton, United Kingdom
Gangmin.Li@beds.ac.uk

4th Terry Payne
Department of Computer Science
University of Liverpool
Liverpool, United Kingdom
T.R.Payne@liverpool.ac.uk

5th Yong Yue
Department of Computing
Xi'an Jiaotong-Liverpool University
Suzhou, China
Yong.Yue@xjtlu.edu.cn

***Abstract*—This paper introduces a reinforcement learning-based framework designed to tackle dynamic pricing challenges in e-commerce. Prior research has predominantly concentrated on algorithm selection to enhance performance in dense data scenarios. However, many of these models fail to robustly address sparse data structures, such as low-traffic products, leading to the 'cold-start' problem [4]. Through numerical analysis, our framework offers innovative insights derived from the design of the reward function and integrates product clustering with pre-trained learning to mitigate this issue. As a result of this optimization, the performance of predictive models on sparse data is expected to see substantial improvement.**

***Index Terms*—Dynamic Pricing, Reinforcement Learning, Clustering, K-means, Sarsa, Markov decision process, Price elasticity of demand**

I. Introduction

The utilization of dynamic pricing strategies in online environments has been the subject of extensive research. This includes conventional strategies that overtly modify listed prices and innovative pricing strategies that subtly adjust prices behind the scenes. These include Bundle Pricing, Auctions, First Come-First Served, Price Discrimination, and Cashback mechanisms [5]. While these strategies may be governed by a dominant internal algorithm that drives changes, e-retailers aims to find optimal policy trajectories that can trigger price changes.

For e-retailers, profit maximization is a primary objective, often achieved by securing a sale price significantly higher than the cost price. However, price determination also necessitates consideration of competition among homogeneous products and customer satisfaction. Consequently, e-retailers establish fair prices to balance profitability and customer satisfaction, often called a 'collusion' price.

In practical terms, e-retailers encounter several hurdles when determining appropriate prices for newly introduced products, which typically lack substantial data accumulation:

1) Identifying a benchmark price to serve as the initial price.
2) Predicting the precise reactions of competitors and customers to the set price.
3) Establishing prices for all times of the year, recognizing that different periods may necessitate distinct pricing strategies.

Newly introduced products, which can be viewed as sparse data structures, may suffer from low profitability or extreme customer dissatisfaction if priced inappropriately. To address these pricing decisions for new products, we propose a method that integrates clustering, transfer learning, and reinforcement learning [3].

II. Fundamental Models

A. Clustering

In e-commerce, newly introduced products often grapple with a lack of historical transaction records and customer feedback. This absence of data can impede the process of price optimization, necessitating an effective strategy to circumvent this issue. One such approach is to minimize exploration by identifying clusters of similar products that can serve as benchmarks for reinforcement learning training.

Clustering is a powerful machine-learning technique that groups similar instances based on features or conditions[1]. K-means has been widely adopted among various clustering methods due to its simplicity and efficiency. The K-means algorithm partitions a given dataset into 'K' clusters, where each data point belongs to the set with the nearest mean. This

method is particularly effective when a clear distinction or separation exists between different data groups[2].

In the context of our study, we employ the K-means algorithm to group similar products based on their features. These clusters then serve as benchmarks for training our reinforcement learning model. This approach allows us to effectively address the challenges associated with sparse data structures and optimize pricing strategies for newly introduced products.

In this study, we focus on the 'Quilt Sets' category on Amazon.com, chosen for its homogeneity in product features. The methodologies applied here can be extended to other categories and platforms in future research. The data features considered include:

- Brand: The brand of the product.
- ASIN: The unique identifier for products on Amazon.
- Size: Variations include Twin, Queen, and King.
- Material: Available in 100% Cotton and 100% Microfiber.
- Colour: Available in Solid and Print.
- Quantity: Available in sets of 2, 3, or 4.
- Weight: Represents the shipping weight of the product.

Additionally, we sampled data from the top 100 sellers and one newly introduced product in Nov. 2022. During feature engineering, we transformed the features into a one-hot encoding format and employed normalization to scale the range. We then applied the K-means clustering method, setting K=5, to divide the 98 samples into five groups. We examined the new product group (refer to Table I).

TABLE I
GROUPING OF NEWLY LAUNCHED PRODUCTS

Brand	ASIN	Price
Great Bay Home	B07PGQ3JYB	$54.99
Travan	B07TSHZKL1	$49.99
Mooreeke	B09VS9QSDD	$67.98
Lush Decor	B0006465C8	$51.09
Wongs Bedding	B07VBJ5P7Z	$39.99
Great Bay Home	B0778VZ7FP	$49.99
Wongs Bedding	B07WPHBNWS	$39.99
Levtex Home	B087LTGHHF	$89.99
DJV	B09721M4VC	$39.99
Belista	B09FNXM18F	$19.99
Woolrich	B01IROZDTW	$84.99
Smuge	B0995KVPR3	$16.99
Tigona	B0B5MWBRRQ	$44.99
Cozy Line Home Fashions	B07GSKRF89	$88.99
Janzaa	B08DK8JCYZ	$39.99
Uozzi Bedding	B07QXRHSNS	$39.99
Newlake	B078MH49YC	$73.89
Ycosy	BOBB5W21J6	$43.99
Lush Decor	B01MTMKFJ2	$42.67

B. Reinforcement Learning

In the realm of reinforcement learning, our study incorporates the Sarsa learning algorithm, which falls under the temporal difference methods within the reinforcement learning domain. The fundamental premise of Sarsa is to learn from the actual policy, as depicted by the following update rule[9]:

$$Q(s^t, a^t) \leftarrow Q(s^t, a^t) + \alpha[r^t + \gamma Q(s^{t+1}, a^{t+1}) - Q(s^t, a^t)] \tag{1}$$

Here, $\alpha \in [0, 1]$ denotes the learning rate, signifying the pace of learning, while $\gamma \in [0, 1]$ represents the discount factor, indicating the relative significance of the current reward versus future rewards.

In this study, we adopt the framework of a Markov Decision Process (MDP) to model dynamic pricing in e-commerce. This approach, widely recognized for its efficacy in decision-making problems [6], enables us to methodically delineate the key components of the reinforcement learning process: the state, action, and reward.

The state component encapsulates the system's current status, providing the necessary context for decision-making. The action component represents the decisions or moves made by the agent (in this case, the e-retailer) within the system. The reward component quantifies the outcome or result of an action, serving as a feedback mechanism that guides the learning process [7].

By structuring our problem as an MDP, we can leverage the powerful mathematical tools and algorithms associated with reinforcement learning. This approach facilitates the exploration of optimal pricing strategies in a systematic, data-driven manner, potentially leading to more effective and profitable decisions in e-commerce. The components are defined as follows:

1) **State Set:** We construct a state set S for a period of 30 days, represented as $S = s^1, s^2, \cdots, s^t, \cdots, s^{30}$. Each state s^t in this set corresponds to the system's state on day t.
2) **Action Set:** The agent's action corresponds to the product's retail price set. To simplify the process, we select a price from the same cluster for each day. Thus, the action set A for all days is defined as $A = a^1, a^2, \cdots, a^t, \cdots, a^{30}$, where each action a^t is within the price range of this cluster, i.e., $a^t \in \text{range}(\text{mincluster}, \text{maxcluster})$.
3) **Reward Set:** The reward set R for the 30-day period is defined as $R = r^1, r^2, \cdots, r^t, \cdots, r^{30}$, where each reward r^t corresponds to the reward received on day t.

To further refine our model, we incorporate the concept of price elasticity of demand from microeconomic theory. The formula for price elasticity is given by:

$$\xi = \frac{(q^t - q^0)/q^0}{(a^t - a^0)/a^0} \tag{2}$$

Here, ξ represents the product's price elasticity, assumed to be a fixed value. q^t and q^0 represent the sales volume at time t and the initial sales volume (based on the mean volume of this cluster), respectively. Similarly, a^t and a^0 represent the sales price at time t and the initial sales price (based on the mean price of this cluster), respectively. This formula allows us to model the relationship between price changes and changes in

Algorithm 1 Sarsa: An on-policy TD control algorithm

Initialize $Q(s,a), \forall s \in S, a \in A(s)$ **arbitrarily**

for each episode **do**
  Initialize s
  Choose a from s using policy derived from Q (i.e., ϵ-greedy)
  for each step of episode **do**
    Take action a, observe r, s^{t+1}
    Choose a^{t+1} from s^{t+1} using policy derived from Q (i.e., ϵ-greedy)
    $Q(s^t,a^t) \leftarrow Q(s^t,a^t) + \alpha[r^t + \gamma Q(s^{t+1},a^{t+1}) - Q(s^t,a^t)]$
    $s \leftarrow s^{t+1}, a \leftarrow a^{t+1}$

sales volume, providing a more nuanced understanding of the dynamics at play. From this formula, we derive the following:

$$q^t = \frac{(a^t - a^0) \times q^0 \times \xi}{a^0} + q^0 \tag{3}$$

Then, the reward at time t can be expressed as follows, where c represents the fixed cost:

$$r^t = (a^t - c) \times q^t = (a^t - c) \times (\frac{(a^t - a^0) \times q^0 \times \xi}{a^0} + q^0) \tag{4}$$

The pseudocode for the Sarsa learning algorithm is presented in Algorithm 1.

C. Transfer Learning

Transfer learning has emerged as a powerful technique in machine learning and artificial intelligence, providing a strategic solution to the pervasive problem of insufficient training data. In such contexts, transfer learning serves as a bridge, connecting disparate but related domains and enabling knowledge transfer. This is especially evident in reinforcement learning, where transfer learning has been employed to expedite the learning process and enhance performance in tasks characterized by sparse rewards [8]. For instance, a model trained to navigate one type of video game can transfer its learned knowledge to a different but related match, reducing the required training. This exemplifies the power of transfer learning in creating efficient learning pathways between related domains, even in the face of data limitations.

III. Computational Modeling

A. Hyperparameters Setting

Within our computational simulation, we conduct extensive manipulations of the price elasticity variable (ξ) across two distinct values, 0.5 and 0.8, while operating within the encompassing framework of sales volume clusters. The primary objective of this experimentation is to meticulously evaluate the adaptability of our methodology across a broad spectrum of environments. By undertaking these adjustments, we aim to obtain a comprehensive understanding of the impact that variations in elasticity have on the outcomes.

To facilitate our analysis, we meticulously configure the decision-making algorithm employed in our system, namely Sarsa, with specific parameter values. The learning rate (α) is meticulously set at 0.1, the discount factor (γ) at 0.9, and the exploration rate (ϵ) at 0.1. Additionally, within the designated product cluster, we meticulously establish the average price at \$53 while setting the cost (c) at \$30. These specific values enable us to carefully examine and interpret the results obtained through our computational simulation.

B. Outcome and Analysis

The figures provided alongside this research article offer valuable insights into the reinforcement learning process. Notably, the x-axis represents the number of episodes, denoting the progression of the learning process. Conversely, the y-axis reflects the cumulative Q value, which indicates the overall reward accumulated over 30 days. The attainment of convergence in the cumulative Q value implies that the system has successfully explored and identified a near-optimal pricing strategy.

Upon meticulous analysis of the obtained results, a prominent and evident pattern emerges, highlighting substantial variations in performance across different elasticity conditions. This observation underscores the profound impact of diverse elasticity of demand values and initial pricing strategies on the outcomes of the learning process. A comprehensive depiction and thorough analysis of this influential aspect can be found in Figures 1 and 2.

Regarding the reward component, it is crucial to note that initializing the pricing strategy at the mean value of the cluster prices (\$53) yields significantly superior outcomes compared to alternative approaches for price learning. This observation accentuates the criticality of establishing a well-calibrated starting point within the reinforcement learning process, particularly within the dynamic pricing domain.

Regarding the stability aspect, Figures 1 and 2 provide valuable insights. Our transfer learning method demonstrates more substantial stability than other stochastic pricing strategies. It consistently explores within a reasonably confined space, resulting in returns that exhibit minimal fluctuations. Additionally, the convergence speed is relatively rapid, further solidifying the efficacy and stability of our approach.

C. Future Work

In future research endeavours, a promising avenue for exploration entails the seamless integration of the aforementioned technique with other state-of-the-art deep reinforcement learning methods. This integration can significantly bolster the robustness and efficacy of the algorithm under investigation. Notably, prominent examples of these methods include DQN, DDPG, SAC, and PPO, each of which has demonstrated its ability to enhance the algorithm's performance.

Furthermore, to ensure the validity and generalizability of our conclusions, we aim to collect datasets from diverse categories. This comprehensive dataset collection process allows

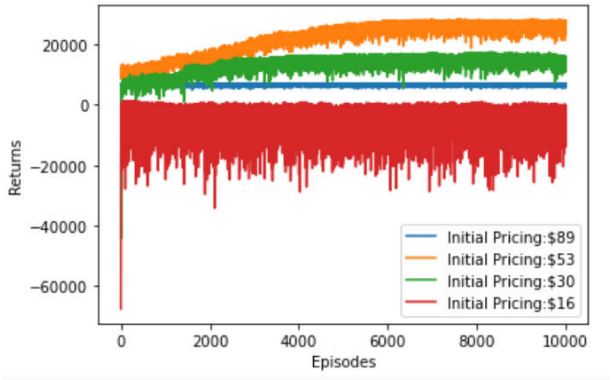

Fig. 1. Sales Reward by $\xi = 0.5$

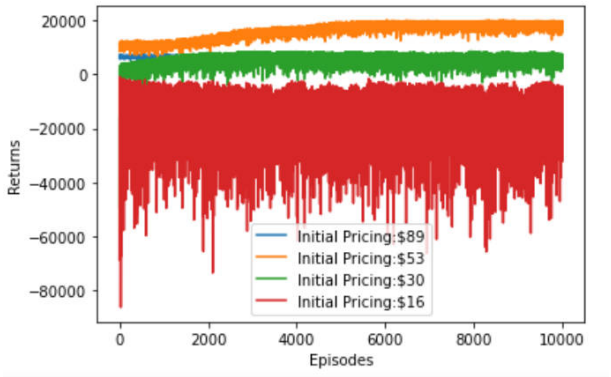

Fig. 2. Sales Reward by $\xi = 0.8$

us to rigorously test and validate the efficacy of our findings across various domains.

By synergistically combining these cutting-edge approaches, researchers can harness the cumulative benefits and leverage the advancements in deep reinforcement learning. This integration and collaboration propel the algorithm's performance and expand its capabilities, resulting in more robust and effective outcomes.

IV. Conclusions

In this study, we address the issue of sparse data structures in dynamic e-commerce pricing by employing k-means clustering, transfer learning and reinforcement learning. Our findings demonstrate that transferring a price point similar to a new product from an existing cluster can significantly expedite the system's learning process. This approach effectively reduces the number of exploratory steps required in practice, enabling the system to establish an optimal pricing action more swiftly.

This methodology offers a novel solution to the challenge of sparse data in dynamic pricing, particularly for new products. By leveraging the similarities within a cluster, we can provide a more informed starting point for the reinforcement learning process. This not only accelerates the learning phase but also enhances the overall efficiency of the pricing strategy. The implications of this approach extend beyond e-commerce, potentially benefiting any industry or sector that relies on dynamic pricing in the face of sparse data.

Acknowledgment

This work is partially supported by the XJTLU AI University Research Centre and Jiangsu Province Engineering Research Centre of Data Science and Cognitive Computation at XJTLU. Also, it is partially funded by the Suzhou Municipal Key Laboratory for Intelligent Virtual Engineering (SZS2022004) as well as funding: XJTLU-REF-21-01-002 and XJTLU Key Program Special Fund (KSF-A-17).

References

[1] Anil K Jain. "Data clustering: 50 years beyond K-means". In: *Pattern recognition letters* 31.8 (2010), pp. 651–666.

[2] Tapas Kanungo et al. "An efficient k-means clustering algorithm: Analysis and implementation". In: *IEEE transactions on pattern analysis and machine intelligence* 24.7 (2002), pp. 881–892.

[3] Deqian Kong et al. "Dynamic pricing of demand response based on elasticity transfer and reinforcement learning". In: *2019 22nd International Conference on Electrical Machines and Systems (ICEMS)*. IEEE. 2019, pp. 1–5.

[4] Jiaxi Liu et al. "Dynamic pricing on e-commerce platform with deep reinforcement learning". In: (2018).

[5] Y. Liu et al. "Dynamic Pricing Strategies on the Internet". In: in Proceedings of International Conference on Digital Contents: AICo (AI, IoT and Contents) Technology. 2022.

[6] Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

[7] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[8] Matthew E Taylor and Peter Stone. "Transfer learning for reinforcement learning domains: A survey." In: *Journal of Machine Learning Research* 10.7 (2009).

[9] Weinan Zhang, Jian Shen, and Yong Yu. *Hands-on reinforcement learning*. The People's Posts and Telecommunications Press, 2022.