

Shelf Space Allocation

The shelf space allocation problem has been studied in the operations research literature for many decades. Some classical work approaches the problem by proposing a dynamic programming algorithm to allocate limited shelf space among a finite set of products. In this case, the objective function is composed of revenue, costs and a set of constraints (?). Later work proposed a simulated annealing optimization approach that accounts for two primary decision variables: product assortment and allocated space for each product (?). This optimization technique accounts for many different environment variables such as item profitability, brand elasticities, and supply chain features. More recently, frequent pattern mining algorithms have been proposed to allocate product shelf space. For instance Brijs et al. (?) propose the PROFSET algorithm, which is an association rule algorithm that mines customer basket sets to identify profitable product pairings. This algorithm is an extension of frequent item set algorithms that also accounts for product value. Extensions of this idea have also been proposed. Aloysius and Binu propose a PrefixSpan algorithm for shelf allocation that first identifies complementary categories from historical purchase data before identifying product mix strategies within categories (?).

These existing studies differ from our work in the following ways. First, they all focus on micro-regions (shelves) within the retail environment. The spatial effects these models capture are markedly different from the macro-level ones tackled in the current work. Second, these studies focus on the number of each product on a shelf. They try to maximize profitability given the fixed shelf volume. This optimization problem is fundamentally different from allocating products across the entire store. For these reasons, none of these methods can be directly applied to our problem.

Deep Reinforcement Learning for Spatial Resource Allocation

Recent breakthroughs in reinforcement learning (?) (?) (?) have spurred interest in RL as an optimization approach in complex and dynamic environments. In particular, recent studies have proposed RL algorithms as a mechanism for spatiotemporal resource allocation.

Order dispatching. Significant attention has been paid to the order dispatching problem in ride sharing systems. Briefly, order dispatching refers to the problem of efficiently matching riders and drivers in an urban environment. The RL agent must learn the complex spatial dynamics to learn a policy to solve the dispatching problem. For example, Lin et al. (?) tackle the dispatch problem by proposing a contextual multi-agent reinforcement learning framework that coordinates strategies among a large number of agents to improve driver allocation in physical space. Additionally, Li et al. (?) also approach the order dispatching problem with multi-agent reinforcement learning (MARL). Their method relies on the mean field approximation to capture the dynamic, spatially distributed fluctuations in supply and demand. They empirically show that MARL can reduce supply-demand gaps in peak hours.

Traffic signal control Increasing traffic congestion is a key concern in many urban areas. Recent efforts to optimize traffic control systems via reinforcement learning has shown encouraging results. These systems seek to adjust traffic lights to real-time fluctuations in traffic volume and road demand. Wei et al. (?) propose IntelliLight, which is a phase-gated deep neural network that approximates state-action values. More recently (?) proposes a graph attentional network to facilitate cooperation between many traffic signals.

Spatial demand for electronic tolls Chen et al. (?) propose a dynamic electronic toll collection system that adjusts to traffic patterns and spatial demand for roads in real time. Their proposed algorithm, PG- β , is an extension of policy gradient methods and decreases traffic volume and travel time.

While these reinforcement learning methods deal with the large-scale optimization of spatial resource, they cannot be directly applied to the product allocation problem because they all rely on domain-specific simulators. We propose our model in an effort to extend these state-of-the-art optimization techniques to our problem.

Conclusion

In this paper, we studied the automation of product placement in retail settings. The problem is motivated by the fact that well placed products can maximize impulse buys and minimize search costs for consumers. Solving this allocation problem is difficult because location-based, historical data is limited in most retail settings. Consequently, the number of possible allocation strategies is massive compared to the number of strategies typically explored in historical data. Additionally, it is generally costly to experiment and explore new policies because of the economic costs of sub optimal strategies, and operational cost of deploying a new allocation strategy. Therefore, we propose a probabilistic environment model called that is designed to mirror the real world, and allow for automated search, simulation and exploration of new product allocation strategies. We train the proposed model on real data collected from two large retail environments. We show that the proposed model can make accurate predictions on test data. Additionally, we do a preliminary study into various optimization methods using the proposed model as a simulator. We discover that Deep Q -learning techniques can learn a more effective policy than baselines. On average, DQN offers an improvement of 24.5% over Tabu search in terms of cumulative test reward.

Dicta nihil beatae, porro recusandae fuga sit obcaecati adipisci distinctio quasi quaerat, repellat sed cumque consequatur. Corrupti cupiditate rem nobis dignissimos recusandae, incidunt a earum pariatur quia ex, soluta dolor illo excepturi dolorum dicta. Illum sit ex recusandae illo commodi voluptates deserunt, aperiam non minima maiores commodi quam nisi porro? Natus molestiae totam soluta culpa cum, aperiam voluptatibus asperiores in illum, laudantium magni illum omnis fugiat mollitia corrupti veniam fuga numquam beatae, labore commodi rerum reiciendis maiores nobis quisquam incidunt et blanditiis sint, porro officiis quaerat velit repellat adipisci aliquam voluptate? Eum ipsum culpa

facilis optio fugit adipisci illo dolorem odio nemo dicta, sus-
cipit aliquid sed repellat quos sint accusamus ab minima do-
lorum debitis?