

Capstone Project Report: Dynamic Parking Pricing

1. Introduction

This report details the development of a dynamic parking pricing system designed to optimize parking utilization and revenue generation for urban parking facilities. The project addresses the critical challenge of balancing parking demand with available supply, a common issue in metropolitan areas leading to congestion, inefficient space usage, and lost revenue opportunities. By leveraging real-time data and advanced pricing models, this system aims to provide a flexible and responsive pricing strategy that adapts to fluctuating demand, traffic conditions, and special events.

The traditional static pricing models often fail to account for the dynamic nature of urban parking, leading to either underutilization during off-peak hours or severe congestion during peak times. This project explores the implementation of several pricing models, ranging from a baseline linear model to more sophisticated demand-based and competitive pricing strategies. Furthermore, it integrates real-time data processing capabilities using Pathway and visualizes key metrics and pricing recommendations using Matplotlib, providing a comprehensive solution for modern parking management.

The primary objective of this capstone project is to demonstrate the feasibility and effectiveness of a data-driven approach to parking pricing. It encompasses data analysis, model development, real-time simulation integration, and the generation of insightful visualizations to support decision-making. The insights derived from this system can help urban planners and parking operators to implement more efficient and equitable parking policies, ultimately contributing to smarter city infrastructure.

2. Problem Statement Analysis

The core problem addressed by this project is the inefficiency of static parking pricing in dynamic urban environments. The provided problem statement highlights the need for a system that can adjust parking prices in real-time based on various factors to optimize utilization and revenue. Key aspects of the problem include:

- **Underutilization and Overcrowding:** Static pricing leads to empty parking spots during low-demand periods and overcrowded lots during high-demand periods, causing frustration for drivers and lost revenue for operators.
- **Lack of Responsiveness:** Current systems cannot react to sudden changes in demand due to events, traffic incidents, or time-of-day fluctuations.
- **Revenue Optimization:** There is a clear opportunity to increase revenue by implementing dynamic pricing strategies that capture higher value during peak demand.
- **Traffic Congestion:** Drivers circling for parking contribute to urban traffic congestion and increased emissions.

Data Analysis from `dataset.csv`

The `dataset.csv` file provides historical parking data, which is crucial for understanding parking patterns and developing predictive models. The dataset includes the following key features:

- **SystemCodeNumber** : Unique identifier for each parking facility.
- **Capacity** : Total parking capacity of the facility.
- **Occupancy** : Number of occupied parking spots at a given time.
- **LastUpdatedDate** and **LastUpdatedTime** : Timestamps for data collection, allowing for time-series analysis.
- **Latitude** and **Longitude** : Geographical coordinates of the parking facility, useful for spatial analysis and identifying nearby competitors.
- **QueueLength** : Number of vehicles waiting to park, indicating high demand and potential congestion.
- **TrafficConditionNearby** : Categorical data (e.g., 'Low', 'Medium', 'High') reflecting real-time traffic conditions around the parking facility.

- **IsSpecialDay** : Boolean indicator for special events or holidays that might influence parking demand.
- **VehicleType** : Type of vehicle (e.g., 'Car', 'Motorcycle'), which could influence pricing segments.

Initial analysis of the dataset reveals significant variations in occupancy and queue lengths across different parking facilities and times of day, confirming the dynamic nature of parking demand. The **TrafficConditionNearby** and **IsSpecialDay** features are particularly valuable for developing responsive pricing models, as they directly correlate with demand fluctuations. The presence of **Latitude** and **Longitude** also opens up possibilities for competitive pricing strategies based on proximity to other parking options.

This dataset forms the foundation for training and evaluating the proposed dynamic pricing models, enabling the system to learn from historical patterns and adapt to future conditions. The goal is to transform this raw data into actionable insights that drive optimal pricing decisions.

3. Project Design and Methodology

To address the complexities of dynamic parking pricing, this project adopts a multi-model approach, integrating various pricing strategies with real-time data processing and visualization capabilities. The overall architecture is designed for scalability, responsiveness, and data-driven decision-making.

3.1 Pricing Models

Three primary pricing models have been developed and evaluated:

3.1.1 Model 1: Baseline Linear Model

This model serves as a foundational approach, adjusting prices linearly based on current occupancy and capacity. It provides a simple yet effective mechanism for dynamic pricing, ensuring that prices increase as parking availability decreases. The formula for this model is:

$$\text{Price} = \text{Base Price} + \alpha * (\text{Occupancy} / \text{Capacity})$$

Where α is a sensitivity parameter that determines how aggressively the price changes with occupancy. This model is straightforward to implement and provides a basic level of demand-responsive pricing.

3.1.2 Model 2: Demand-Based Price Function

Building upon the baseline, Model 2 incorporates additional factors that influence parking demand, such as queue length, traffic conditions, special events, and vehicle type. This model aims to capture a more nuanced understanding of demand dynamics, allowing for more precise price adjustments. The pricing function considers:

- **Queue Length:** Longer queues indicate higher demand, leading to increased prices.
- **Traffic Conditions:** Heavy traffic nearby might deter drivers, potentially leading to lower prices or, conversely, higher prices if parking is scarce due to congestion.
- **Special Days:** Events or holidays typically lead to surges in demand, justifying higher prices.
- **Vehicle Type:** Different vehicle types might have varying willingness to pay, allowing for segmented pricing.

This model utilizes a more complex function that combines these factors, with adjustable weights (λ , α , β , γ , δ , ϵ) to fine-tune its responsiveness to each variable. The implementation allows for flexible adaptation to different urban contexts and policy objectives.

3.1.3 Model 3: Competitive Pricing Model (Optional)

This advanced model introduces a competitive element, where the pricing of a parking facility is influenced by the prices of nearby competing parking lots. The objective is to prevent significant price discrepancies that could drive customers to competitors or leave the facility underpriced. This model requires access to real-time pricing data from neighboring parking facilities and incorporates geographical proximity to determine the competitive landscape. The model considers:

- **Proximity to Competitors:** Parking lots closer to competitors will be more sensitive to their pricing.

- **Competitor Pricing:** If nearby competitors raise their prices, the current facility might also increase its prices, and vice-versa.
- **Occupancy and Queue Length of Competitors:** High demand at competitor lots could signal an opportunity to increase prices.

Implementing this model in a real-time streaming environment like Pathway would involve joining data streams from multiple parking facilities and continuously evaluating the competitive landscape. For this project, a simplified approach was taken where the model considers a dummy representation of competitor data, demonstrating the conceptual integration.

3.2 Real-Time Simulation with Pathway

Pathway is utilized for real-time data ingestion and processing, enabling the dynamic pricing system to react instantaneously to changes in parking conditions. The core functionalities of Pathway in this project include:

- **Data Ingestion:** Reading streaming data (simulated from `dataset.csv`) that represents real-time updates on occupancy, queue length, and other relevant metrics.
- **Schema Definition:** Defining a clear schema for the incoming data streams to ensure data integrity and facilitate processing.
- **Real-time Calculations:** Performing on-the-fly calculations for pricing models based on the latest ingested data. While direct `apply` operations on Pathway tables for complex Python functions can be challenging, the design allows for the integration of simpler, column-wise operations or the preparation of data for external model inference.
- **Output Streaming:** Directing the processed pricing recommendations to downstream applications or visualization tools.

The Pathway integration focuses on demonstrating the capability to handle continuous data flows and compute dynamic prices as new information becomes available. This is crucial for a responsive pricing system that can adapt to rapidly changing urban environments.

3.3 Real-Time Visualizations with Matplotlib

Matplotlib is employed to create real-time visualizations of the dynamic pricing outputs, occupancy trends, and queue lengths. These visualizations provide immediate insights into the system's performance and the impact of the pricing models. Key visualizations include:

- **Price Comparison Plots:** Displaying the prices generated by Model 1, Model 2, and Model 3 over time for selected parking facilities, allowing for a direct comparison of their behavior.
- **Occupancy Trend Plots:** Illustrating the occupancy levels against the total capacity over time, helping to identify periods of high and low utilization.
- **Queue Length Trend Plots:** Showing the fluctuations in queue lengths, which serve as a strong indicator of unmet demand.

These visualizations are crucial for monitoring the system, identifying anomalies, and validating the effectiveness of the dynamic pricing strategies. They provide a clear and intuitive way for operators to understand the current state of parking facilities and the rationale behind the recommended prices.

3.4 Overall Project Architecture

The project architecture follows a modular design, ensuring flexibility and maintainability. It comprises the following key components:

1. **Data Ingestion Layer:** Responsible for collecting raw parking data. In this project, it's simulated via `dataset.csv` and processed by Pathway.
2. **Real-Time Processing Layer (Pathway):** Ingests streaming data, applies data transformations, and prepares data for pricing model inference.
3. **Pricing Model Layer:** Contains the implementations of Model 1, Model 2, and Model 3, which calculate dynamic prices based on input data.
4. **Visualization Layer (Matplotlib):** Generates real-time plots and dashboards to monitor system performance and pricing recommendations.
5. **Output Layer:** Delivers the calculated dynamic prices and insights to a CSV file for further analysis or integration with other systems.

This architecture ensures a clear separation of concerns, allowing for independent development and scaling of each component. The integration of Pathway for real-time processing and Matplotlib for visualization creates a robust and insightful dynamic parking pricing system.

4. Implementation Details

The dynamic parking pricing system is implemented using Python, leveraging several libraries for data manipulation, real-time processing, and visualization. The core components are organized into distinct scripts, each responsible for a specific part of the system.

4.1 `main.py`

This script serves as the orchestrator for the batch processing of the historical dataset. It performs the following key functions:

- **Data Loading:** Reads the `dataset.csv` file into a Pandas DataFrame.
- **Data Preprocessing:** Cleans and transforms the raw data, including combining date and time columns into a single `Timestamp` and handling categorical features.
- **Model Application:** Applies the implemented pricing models (Model 1, Model 2, and Model 3) to the historical data to calculate dynamic prices for each parking event.
- **Output Generation:** Saves the processed data, including the calculated prices, to a new CSV file (`processed_dataset.csv`), which can then be used for further analysis or visualization.

This script demonstrates the application of the pricing models to a static dataset, providing a comprehensive view of how prices would have fluctuated historically under different pricing strategies.

4.2 `pricing_models.py`

This module encapsulates the logic for the three dynamic pricing models. It defines functions for each model, allowing for modularity and reusability. The functions are

designed to take relevant parking parameters (e.g., base price, occupancy, capacity, queue length, traffic conditions) as input and return the calculated dynamic price.

- `model_1_pricing(base_price, occupancy, capacity, alpha)`: Implements the baseline linear pricing model.
- `model_2_pricing(base_price, occupancy, capacity, queue_length, traffic_condition, is_special_day, vehicle_type, lambda_val, alpha_val, beta_val, gamma_val, delta_val, epsilon_val)`: Implements the demand-based pricing model, incorporating various influencing factors.
- `model_3_pricing(current_price, occupancy, capacity, latitude, longitude, system_code_number, all_parking_data_at_timestep, base_price)`: Implements the competitive pricing model, considering competitor prices and proximity. (Note: The `all_parking_data_at_timestep` parameter is a simplified representation for demonstration purposes, as a full real-time competitive analysis would require complex data joins in a streaming environment).

By centralizing the pricing logic in this module, it becomes easier to modify, test, and extend the pricing strategies independently of the main data processing pipeline.

4.3 `pathway_integration.py`

This script demonstrates the integration with Pathway for real-time data ingestion and simulation. While the complex `apply` operations for the pricing models were challenging to implement directly within Pathway's streaming context, this script showcases Pathway's capability to handle continuous data streams and prepare them for analysis.

- **Data Streaming:** Utilizes `pw.demo.replay_csv` to simulate a real-time data stream from a subset of the historical `dataset.csv`.
- **Schema Definition:** Defines a `ParkingSchema` to ensure that incoming data conforms to the expected structure.
- **Timestamp Processing:** Combines date and time columns and converts them into a Pathway-compatible timestamp format.
- **Output:** Writes the processed stream data to a CSV file (`pathway_processed_output.csv`), demonstrating the flow of data through the

Pathway pipeline.

This script highlights the potential of Pathway for building real-time data pipelines for dynamic pricing systems, where rapid ingestion and transformation of data are critical.

4.4 visualizations.py

This script is responsible for generating insightful visualizations using Matplotlib, based on the processed data from `main.py`. These plots provide a clear understanding of the pricing model behaviors and parking trends.

- **Price Comparison Plots:** Generates line plots comparing the prices calculated by Model 1, Model 2, and Model 3 over time for selected parking facilities. (e.g., `price_comparison_*.png`)
- **Occupancy Trend Plots:** Visualizes the occupancy levels against the capacity over time for selected parking facilities, showing utilization patterns. (e.g., `occupancy_trend_*.png`)
- **Queue Length Trend Plots:** Displays the fluctuations in queue lengths over time, indicating periods of high demand. (e.g., `queue_length_trend_*.png`)

These visualizations are essential for monitoring the system's performance, validating the pricing strategies, and communicating insights to stakeholders.

5. Results and Discussion

The implementation of the dynamic parking pricing system yielded several key insights into the behavior of different pricing models and their potential impact on parking management. The visualizations generated from the `processed_dataset.csv` provide a clear picture of these dynamics.

5.1 Price Comparison Analysis

The price comparison plots (e.g., `price_comparison_*.png`) illustrate how each model adjusts prices over time. Model 1, the baseline linear model, shows a direct correlation between price and occupancy, with prices increasing as parking lots fill up. Model 2, the demand-based model, exhibits more volatile price fluctuations, reacting

to changes in queue length, traffic conditions, and special events. This responsiveness is crucial for capturing peak demand value and deterring congestion.

Model 3, the competitive pricing model, demonstrates how external factors (simulated competitor prices) can influence pricing decisions. While simplified in this implementation, it highlights the importance of considering the broader market when setting prices. The interplay between these models suggests that a hybrid approach, combining the stability of Model 1 with the responsiveness of Model 2 and the market awareness of Model 3, could yield optimal results.

5.2 Occupancy and Queue Length Trends

The occupancy trend plots (e.g., `occupancy_trend_*.png`) reveal the utilization patterns of parking facilities. Periods of high occupancy often coincide with increased prices from Model 2, indicating its effectiveness in responding to demand. Conversely, lower occupancy periods might see reduced prices, encouraging utilization.

The queue length trend plots (e.g., `queue_length_trend_*.png`) are critical indicators of unmet demand. Spikes in queue length are directly addressed by Model 2, which increases prices to manage demand and potentially reduce waiting times. Analyzing these trends helps in understanding the effectiveness of the dynamic pricing system in balancing supply and demand.

5.3 Pathway Integration Insights

The Pathway integration, while not fully implementing the complex pricing logic within the streaming environment due to current limitations, successfully demonstrated real-time data ingestion and transformation. The `pathway_processed_output.csv` file confirms that data can be continuously processed and prepared for downstream applications. This capability is fundamental for a truly dynamic system that needs to react to live data feeds. Future work would involve optimizing the pricing models for native Pathway execution or integrating a high-performance inference engine for real-time model predictions.

5.4 Challenges and Limitations

- **Model Complexity in Streaming:** Directly translating complex Python functions (like those in `pricing_models.py`) into Pathway's streaming `apply` operations

proved challenging. This highlights a common hurdle in integrating sophisticated analytical models with real-time streaming platforms.

- **Data Granularity:** The effectiveness of dynamic pricing heavily relies on the granularity and real-time availability of data. The current dataset, while comprehensive, is historical. A live data feed would be essential for a production-ready system.
- **Competitive Data:** The competitive pricing model (Model 3) was simplified due to the lack of real-time competitor pricing data. A robust implementation would require access to and integration of such external data sources.
- **User Acceptance:** The success of dynamic pricing also depends on user acceptance and clear communication of pricing strategies to the public.

Overall, the results demonstrate the significant potential of dynamic parking pricing to optimize urban parking resources. The system provides a robust framework for data-driven decision-making, offering valuable insights into parking demand and pricing strategies.

6. Conclusion

This capstone project successfully developed and demonstrated a dynamic parking pricing system capable of adapting to real-time changes in parking demand and traffic conditions. By implementing and evaluating three distinct pricing models—a baseline linear model, a demand-based function, and a conceptual competitive pricing model—the project showcased the potential for optimizing parking utilization and revenue generation in urban environments. The integration of Pathway for simulated real-time data ingestion and Matplotlib for comprehensive visualizations underscored the feasibility of building a data-driven solution for modern parking management.

The project highlighted that dynamic pricing is not merely about increasing revenue but also about improving urban mobility by reducing congestion, optimizing space allocation, and enhancing the overall parking experience. The insights gained from analyzing historical data and simulating dynamic pricing scenarios provide a strong foundation for urban planners and parking operators to make informed decisions that benefit both the public and the economy.

While challenges exist, particularly in the direct integration of complex analytical models within streaming environments and the acquisition of real-time competitive

data, the modular architecture developed in this project provides a flexible framework for future enhancements. The ability to quickly adapt pricing strategies based on live data is paramount in today's rapidly evolving urban landscapes, and this project serves as a significant step towards achieving that goal.

7. Future Work

Several avenues for future development and research can further enhance the dynamic parking pricing system:

- **Advanced Pathway Integration:** Explore more sophisticated methods for integrating complex pricing models directly within Pathway's streaming environment. This could involve using Pathway's native capabilities for stateful computations or integrating with external model serving platforms for real-time inference.
- **Machine Learning Models:** Incorporate advanced machine learning techniques for demand forecasting and price optimization. This could include models that learn from historical data to predict future occupancy and queue lengths with higher accuracy, leading to more proactive pricing adjustments.
- **Real-time Data Sources:** Integrate with actual real-time data feeds from parking sensors, traffic management systems, and third-party data providers (e.g., for special events, weather, or competitor pricing). This would transform the simulated environment into a live, operational system.
- **User Interface Development:** Develop a user-friendly web interface (e.g., using Bokeh or Panel for interactive dashboards) that allows parking operators to monitor the system, adjust parameters, and visualize real-time pricing recommendations and performance metrics.
- **A/B Testing and Optimization:** Implement mechanisms for A/B testing different pricing strategies in a live environment to continuously optimize pricing algorithms based on real-world performance data.
- **Predictive Maintenance:** Extend the system to include predictive maintenance for parking infrastructure, using data from sensors to anticipate and address potential issues before they impact operations.
- **Integration with Smart City Platforms:** Explore integration with broader smart city initiatives, allowing the dynamic parking pricing system to contribute to and

benefit from a larger ecosystem of urban data and services.

- **Ethical and Social Considerations:** Conduct further research into the ethical implications of dynamic pricing, including fairness, accessibility, and potential impacts on different socioeconomic groups. Develop strategies to mitigate any negative consequences and ensure equitable access to parking.

By pursuing these future work directions, the dynamic parking pricing system can evolve into a more robust, intelligent, and impactful solution for urban parking management.