Johns Hopkins University

EN.560.650

Operation Research

Austin Sharing Bike Redistribution Project

Author:

Shaochong Xu

Yahan Li

Yinuo Fan

Weiting Yu

Congqi Lin

Introduction

As the proud owners of a shared bike company in Austin, we operate over 70 shared bike stations dispersed across the city with over 5000 bikes. We are not only making profits but also committed to promoting an environmentally friendly lifestyle while providing efficient and sustainable transportation options for the community. Bicycles tend to aggregate in certain geographical areas, leading to a scarcity of bicycles in other regions. Our approach addresses the issue by inferring the genuine transportation needs and using this data to better plan the system's usage. Through these calculations, we will determine when and where bicycles are needed. By employing machine learning, deep learning, and optimization models, we will accurately predict bike usage and optimize the distribution of shared bikes across various stations. This approach is designed to efficiently manage the allocation of resources, ensuring that bikes are strategically placed to meet demand while minimizing operational costs.

Literature

The literature on bike-sharing systems primarily focuses on optimizing operations and understanding user behavior to improve service efficiency. Notable studies in this field present various approaches to tackle key challenges in bike-sharing management.

O'Mahony and Shmoys (2015) delve into the data analysis and optimization for bike-sharing systems, particularly highlighting the issue of 'censoring' at stations that are either full or empty, thereby not representing actual demand accurately. They suggest using a matrix of observations and a level matrix to estimate the true demand, alongside considering soft outages and optimizing rebalancing operations.

Liu et al. (2015) address station site optimization, splitting the problem into two stages: predicting station demand and balance, and optimizing the station network. They utilize neural networks for prediction and seek to maximize demand fulfillment while minimizing imbalances, which provide valuable insights for our study.

Romero et al. (2012) present a simulation-optimization approach to design efficient bike-sharing systems by examining the interaction between private cars and public bicycles. They focus on user behavior to strategically locate public bicycle docking stations.

Jian et al. (2016) propose simulation optimization for large-scale bike-sharing systems, aiming to optimize bike and dock allocations to reduce customer dissatisfaction. They recommend advanced heuristic methods over traditional optimization techniques.

Sayarshad et al. (2012) introduce a multi-periodic optimization formulation for bike planning and utilization, aimed at balancing demand, fleet size, and relocation needs in bike-sharing systems.

Dell'Amico et al. (2014) studied the Bike-sharing Rebalancing Problem (BRP) to minimize costs using capacitated vehicles for redistribution. They develop mixed integer linear programming formulations and custom algorithms for solutions.

In the context of our study, while these methods offer comprehensive frameworks for addressing various aspects of bike-sharing systems, we face a limitation due to insufficient data to fully implement these approaches. Therefore, our focus is primarily on predicting the daily inflow and outflow of bikes at each station. This predictive data is then utilized to address the bike redistribution problem, a critical aspect of maintaining service efficiency in bike-sharing systems. Our approach, albeit simpler, aims to contribute to the broader understanding and optimization of bike-sharing operations.

Data Preprocessing

The sharing bike data was found on the official city of Austin data open data portal. This dataset contains trip data from the Austin MetroBike bicycle. The columns in this dataset are Trip ID: the MetroBike trip’s unique ID, Membership or Pass Type: the membership or pass type of the MetroBike, Bicycle ID: the MetroBike’s ID that was rented during the trip, Checkout Datetime: the date and time that the bike was checked out from the MetroBike kiosk, Trip Duration Minutes: the total number of minutes that the bicycle was checked out and Return Kiosk: the location of the Kiosk to return the bicycle. It records the sharing bike trips from 2013 to 2023. The data about the location and capacity of the sharing bike station was also found on the data portal.

To ensure our data analysis is not skewed by the extraordinary circumstances of the COVID-19 period, we've chosen to focus on data collected prior to the pandemic, specifically from June 1, 2016, to July 31, 2017. During this time frame, there were 59 bike-sharing stations in operation, and ideally, each station should have had 426 data entries. However, upon examining the records, we found inconsistencies in the data, with the most complete station record having only 394 entries, indicating missing data across the board.

To address this, we implemented a data-cleaning strategy where stations with substantial data gaps (those with fewer than 330 records) were removed from our dataset. For the remaining stations, we used interpolation to fill in the smaller gaps in the data. This process resulted in a refined dataset comprising 31 stations. In our analysis and modeling, we operate under the assumption that these stations represent a closed system, allowing us to focus on a more consistent and reliable data set that excludes the anomaly of the COVID-19 impact.

Prediction Models

Machine Learning Models

By utilizing machine learning, we aim to achieve more accurate predictions in forecasting this need. We try to use four different machine learning models to predict this problem. Before we apply the model, we preprocess the data from the data we filtered before. Since the dataset contains the date column, we convert it to Python's datetime format, allowing us to extract and create distinct columns for the year, month, and day.

For training and testing our model, we filtered the dataset to a specific time frame. Testing data contained entries from July 2016 and 2017, with an emphasis on the days from July 2 to July 31. Training data consisted of entries from June 2016 and June 2017. The initial 'date' column was eliminated after filtering because the required temporal data had already been divided into several columns. The dataset was then split into features (X) and target variables (y), specifically focusing on 'in_bike' and 'out_bike' counts. Since bike usage is different during weekdays, weekends, and holidays, two binary variables are added to the data. "Is weekday" and " Is holiday" show Ture and False which represent the situation for that day. Four different prediction models will be used in this project, they are Random Forest, Decision Tree, Gradient Boosting, and K-Nearest

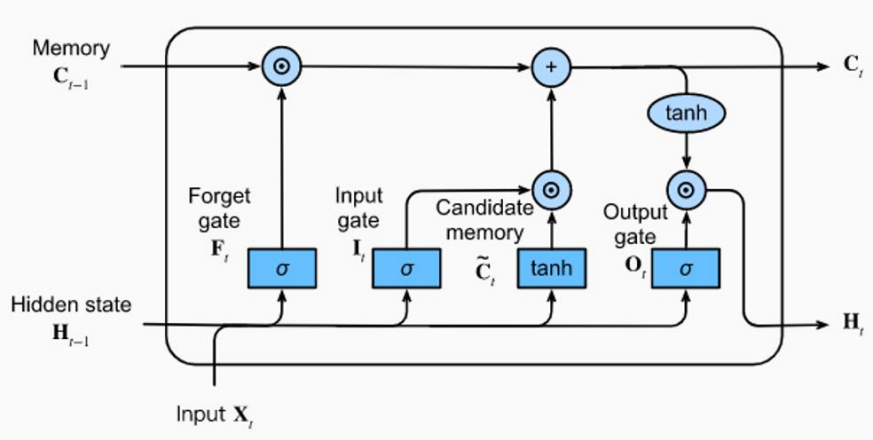
Neighbors. Each method will be used to predict incoming bike counts ('in_bike') and outgoing bike counts ('out_bike'). All models were trained using the data from June 2016 to June 2017.

After training, we get the result from the model for the amount of in and out bikes in this period. We want to know the result's accuracy of this model. The evaluation metrics used were Mean Squared Error (MSE), which was calculated for 'in bikes' and 'out bikes'. Using this method allows us to evaluate how well the model performs in estimating the net change in bike counts overall. The evaluation of various predictive models on bike usage data yielded the following results for Mean Squared Error (MSE): In the case of the Decision Tree model, MSE values were recorded as 53.30 for 'in bike' and 58.04 for 'out_bike'. For the Random Forest model, these figures were marginally different, with 'in bike' at 53.32 and 'out_bike' at 58.12. The Gradient Boosting model demonstrated improved performance, showing MSEs of 48.70 for 'in bike' and 52.27 for 'out_bike'. Lastly, the KNN model resulted in MSEs of 54.50 for 'in bike' and 59.34 for 'out_bike', indicating a varied range of effectiveness across these different modeling approaches.

Deep Learning Model

Given the challenges we faced with the machine learning model in accurately predicting our time series data, we've shifted our focus toward exploring deep learning techniques. Considering the nature of our problem, which involves forecasting time series data, recurrent neural networks (RNNs) emerged as a promising avenue. Consequently, we've decided to develop a model using Long Short-Term Memory (LSTM) networks, a specialized type of RNN, to enhance our predictions for daily bike inflows and outflows at each station.

LSTM networks are particularly suited for time series forecasting due to their ability to remember long-term dependencies. Unlike traditional RNNs, which struggle with the vanishing gradient problem, LSTMs are designed to avoid this issue, making them more effective for learning from data where the context spans over longer sequences. This capability makes them an ideal choice for our task, where the prediction of bike usage patterns depends on understanding trends and patterns over time. With this LSTM model, we aim to achieve more accurate and reliable predictions for the next day's 'in bike' and 'out bike' numbers at each station.



LSTM models consist of three key gates: the forget gate, which determines what information to discard from the cell state; the input gate, which decides what new information to add; and the output gate, which controls the flow of information from the cell state to the hidden state. The hidden state carries information across time steps for predictions, while the cell state acts as the network's memory, maintaining information throughout the sequence's processing. This structure allows LSTMs to effectively learn from and remember information over long sequences, making them highly suitable for complex tasks involving time series data and more.

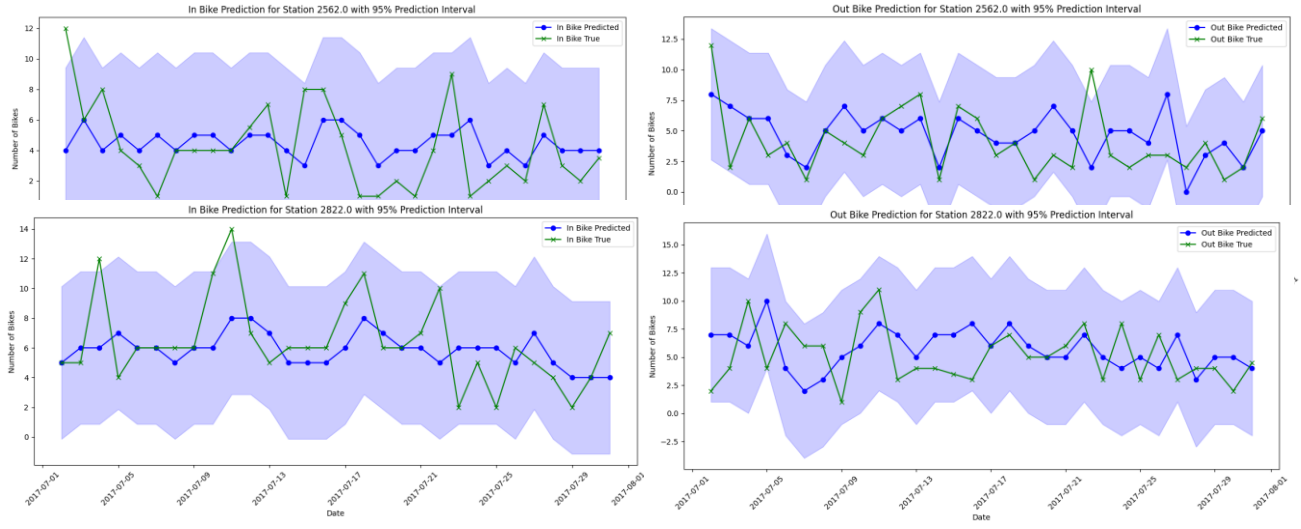
In our initial approach to the problem, we utilized a univariate time series model focusing solely on the 'in bike' or 'out bike' data as our primary input feature to predict the next day's values. However, this approach proved inadequate in capturing the complex patterns underlying our data, reflected in a mean square error exceeding 50 on our test set. Recognizing the need for additional context, we decided to incorporate weekday and holiday information into our model. Our preliminary data analysis revealed notable higher bike usage on Mondays and Tuesdays, indicating these as potentially significant predictive features. We implemented one-hot encoding to integrate this weekday and holiday information into the model effectively. Additionally, we augmented our data with a smoothed target variable, calculated using a moving average over a three-day window, to enhance the model's learning capability by presenting a more nuanced view of the trends and reducing the noises.

In our LSTM model, we utilized data from June 1, 2016, to July 1, 2017, as the training set and data from July 2, 2017, to July 31, 2017, as the testing set. It's important to note that having a larger dataset doesn't always equate to improved model performance. During the process of fine-tuning the model, we conducted a thorough grid search to determine the optimal combination of

hyperparameters, including the ideal size of the training set. We experimented with training set sizes based on the most recent 60, 90, 120, and 150 days. These subsets were then divided into training and validation sets to monitor and minimize loss, thus preventing overfitting or underfitting.

Our comprehensive grid search revealed that the best-performing LSTM model configuration comprised a single layer, a learning rate of 0.0001, and a sequence length of 30 days, meaning that the model used data from the past 30 days to predict the following day's values. Additionally, we settled on a 90-day range for the dataset and a hidden layer size of 64. This particular model configuration achieved a mean squared error (MSE) of 20.2 on the validation set, encompassing all stations, marking it as the most effective model based on our search criteria.

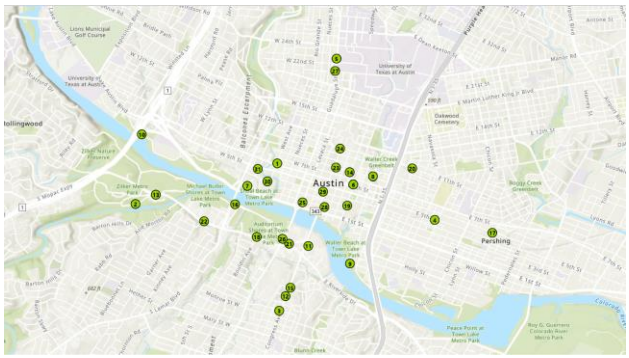
The Mean Squared Error (MSE) of our LSTM model could potentially be reduced further with a more robust dataset. In our test set analysis, we observed that most stations exhibited a relatively low MSE, typically under 10. However, a handful of stations showed anomalously high MSE values, exceeding 80. Upon investigating these outliers, we discovered that these 'problematic' stations shared a common characteristic: they were heavily 'interpolated' stations with over 50 instances of missing dates in their data. This suggests that the high MSEs were largely influenced by the gaps in data, which were filled via interpolation. Despite these few exceptions, the overall performance of our model was satisfactory, indicating its effectiveness in handling most of the dataset's complexities.

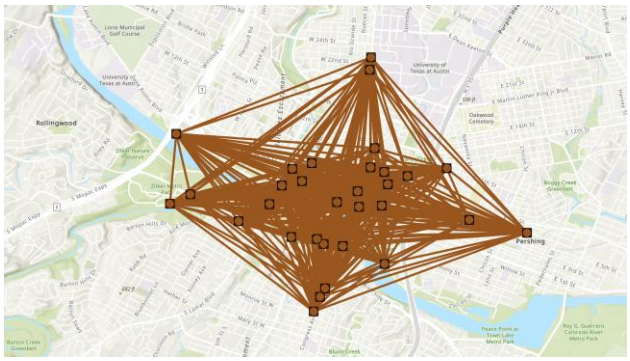


Based on the figures presented, it's evident that our model demonstrates strong performance. This is particularly highlighted by the fact that almost all the actual ('true') values fall within a relatively narrow 95% prediction interval, which is approximately 6.5. The ability of the model to contain the true values within such a precise range is indicative of its accuracy and reliability in making predictions. A narrow prediction interval, especially one that encapsulates the majority of true data points, suggests that the model's predictions are not only consistent but also closely aligned with the actual observed values, affirming the model's effectiveness in forecasting. Hence, we are confident in using our predicted flow as the input for our next step.

Distance Matrix

Arcgis was used in this project to find the distance between each station. The longitude and latitude of each station were plotted on the map shown in the following picture. Each green cycle represents a sharing bike station and there are 31 stations in Austin. To measure the distance between each station, all the stations are connected. Since the MetroBike company will use trucks to redistribute the bikes, the trucking distance between each point will be considered. It will avoid the truck-restricted roads and use preferred truck routes.





Optimization Model

Incorporating forecasted data on incoming and outgoing vehicle information for each site into our optimization model stands as a pivotal measure for enhancing the efficiency of our shared bike system. Employing Gurobi optimization technology, our objective is to determine an optimal solution, considering factors such as fuel and time costs associated with extended transportation distances. We aim to maximize the reduction of bike travel distance while strategically minimizing operational costs. Within our model, the decision variable signifies the number of bikes to be moved, allowing for the strategic allocation of resources based on forecasted demand.

To ensure the robustness of our system, we have established constraints. These constraints elucidate the predicted changes in vehicle demand, ensuring that bike movements remain within the station's capacity and above zero. We define *Xi* as the initial number of bikes at station *i*, *Ci* as the maximum capacity of station *i*, *Dij* as the distance from station *i* to station *j*; and *Mij* as the number of bikes moved from station *i* to station *j*, *Ni* as the netflow between *Ini* (incoming bike to station *i*) and *Outi* (outcoming bike from station *i*), it can be solved by the integer program below:

$$Min \sum D_{ij} \times M_{ij}$$

$$s.t. \sum_j \left(X_i + N_i - M_{ij} + M_{ji}\right) \geq 0 \qquad for\ all\ i$$

$$N_i = In_i - Out_i$$

$$\sum_j \left(X_i + N_i - M_{ji} + M_{ij}\right) \leq C_i \qquad for\ all\ i$$

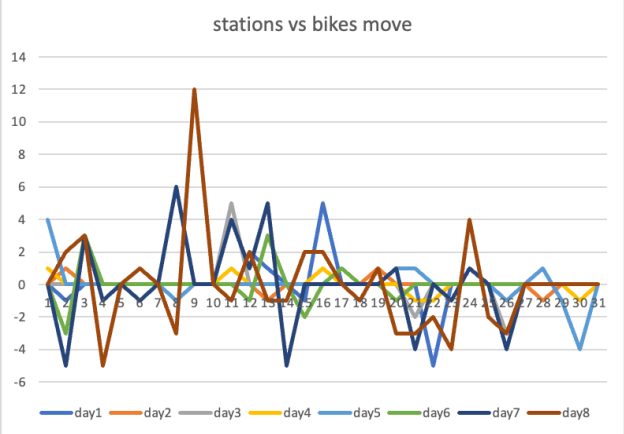
$$M_{ij} \geq 0$$

We determined the strategy for the number of bikes to be moved for a station based on the predicted incoming and outgoing quantities for the day. We then evaluated the accuracy of the strategy by comparing it with the actual incoming and outgoing quantities for the day. If, after considering the movement of bikes, the number of bikes at the station remains within the capacity limits, we consider the strategy to be correct.

Results

We tested the integer programming approaches on real-world cases in Austin. We implemented the IP in Gurobi and carried out a number of experiments. The prediction results from the LSTM model, which exhibited the lowest Mean Squared Error, were set as our input for the number of bikes in and out. Starting our prediction on July 2nd, 2017, we gathered the initial number of bikes in all 31 stations on that day, considering them as the original, and also inputted them into the optimization model. Gurobi identified three feasible solutions, with the optimal one recommending the transfer of 1 bike from Station 2 to Station 13, 1 bike from Station 11 to Station 15, 2 bikes from Station 15 to Station 12, and 5 bikes from Station 22 to Station 16. To validate this solution, we applied these transfer values to the real net flow. The result showed that 29 out of 31 stations still met the constraints. We updated the bike counts for the successful stations as the original for the next day, while for the stations that did not meet the constraints, we either

retained the original counts or set them to capacity values to continue the optimization for the subsequent day. Through our day-by-day optimization process and validation, we discovered that no optimal solution was found on July 8th, 2017, when we assigned capacity values as the new initial to those stations that had been unsuccessful in meeting constraints. During this period, adjusting with the original counts yielded an accuracy of 83.87%, whereas adjusting with capacity values resulted in a slightly lower accuracy of 80.65%. This indicates that the strategy, which considers the need for bike movements, is effective in a large majority of cases.



Through a week-long strategy analysis, we observed that certain stations, such as station 5, 8, and 31, did not require any bike movements, indicating that the capacity settings for these stations fully met the demand. However, station 9 experienced a significant increase in demand on July 9, possibly due to increased nearby activities. Based on this analysis, we can adjust the station capacities more reasonably and analyze the bike movement requirements in conjunction with nearby events.

Sensitivity Analysis

In our sensitivity analysis, we tried to heighten the demand by 20% and 50% to evaluate the model's response to increased pressure on the system. Remarkably, the model continued to generate optimal solutions, even though with a higher frequency of bike transfers, leading to increased transfer costs. Taking the first day as an example, a 20% demand increase resulted in 23 bike transfers, while a 50% increase led to 37 transfers. In this case, we could not only transfer bikes between stations but also transfer from the warehouse just like the approach used for stations that failed to meet constraints. Additionally, we explored the model's resilience by modifying

capacities and even closing one station, finding that it maintained efficacy and adaptability across diverse scenarios.

Conclusion

In conclusion, our exploration into predicting bike inflow and outflow for each station using machine learning and deep learning techniques has yielded valuable insights. Notably, the Gradient Boosting Model within the realm of machine learning has exhibited superior accuracy, evidenced by a lower Mean Squared Error (MSE) compared to other models. The LSTM model, despite encountering outliers, demonstrated commendable performance in handling the intricacies present in the dataset, underscoring its resilience to complex data scenarios. Leveraging our predictive data and employing Gurobi, we formulated an Integer Programming (IP) model, offering a novel perspective on addressing bike rebalancing challenges in bike-share systems. Our study acknowledges that vehicle damage or warehouse transportation factors were not taken into account, thus it may influence the conclusions. With more detailed data and the inclusion of fiscal constraints, the applicability of the IP model can be improved. Continuous efforts are being made to improve the predictive model by incorporating other factors such as weather and humidity, aiming to further improve the accuracy of the predicted model. This continuous improvement represents a dynamic approach that can improve the overall effectiveness of bike rebalancing strategies in bike-sharing systems. For our shareholders, the shared bike company and the government, the utilization of bike stations could be maximized with the minimum rebalancing cost using our model.

References

O'Mahony, E., & Shmoys, D. (2015, February). Data analysis and optimization for (citi) bike sharing. In Proceedings of the AAAI conference on artificial intelligence (Vol. 29, No. 1).

Liu, J., Li, Q., Qu, M., Chen, W., Yang, J., Xiong, H., ... & Fu, Y. (2015, November). Station site optimization in bike sharing systems. In 2015 IEEE International Conference on Data Mining (pp. 883-888). IEEE.

Romero, J. P., Ibeas, A., Moura, J. L., Benavente, J., & Alonso, B. (2012). A simulation-optimization approach to design efficient systems of bike-sharing. Procedia-Social and Behavioral Sciences, 54, 646-655.

Jian, N., Freund, D., Wiberg, H. M., & Henderson, S. G. (2016, December). Simulation optimization for a large-scale bike-sharing system. In 2016 Winter Simulation Conference (WSC) (pp. 602-613). IEEE.

Sayarshad, H., Tavassoli, S., & Zhao, F. (2012). A multi-periodic optimization formulation for bike planning and bike utilization. Applied Mathematical Modelling, 36(10), 4944-4951.

Dell'Amico, M., Hadjicostantinou, E., Iori, M., & Novellani, S. (2014). The bike sharing rebalancing problem: Mathematical formulations and benchmark instances. Omega, 45, 7-19.