DoorDash Estimated Delivery Time Case Report

By DragonTotemRX

Background and Introduction

DoorDash is a technology company that connects consumers with their favorite local and national businesses in more than 7,000 cities across the United States, Canada, Australia, Japan and Germany. DoorDash enables local businesses to address consumers' expectations of ease and immediacy and thrive in today's convenience economy. By building the last-mile logistics infrastructure for local commerce, DoorDash is bringing communities closer, one doorstep at a time. DoorDash's mission is to grow and empower local economies.

As we can see from the DoorDash mobile app and web pages, DoorDash delivers food, convenience, grocery, alcohol, pets' food, and flowers, etc. DoorDash has become people's hero during the pandemic.

Business Problem and Key Findings

In this case, the business problem is to provide estimated delivery time in seconds as accurately as possible to improve customer satisfaction and build brand loyalty with customers to grow business.

To address this business problem, I created a predictive model using Catboost (Gradient boosting algorithm) which outperformed other models I tested. My catboost model has Mean Absolute Error (MAE) as 479.981 and Root mean squared error (RMSE) around 613.809 on the historical_data after data pre-processing and feature engineering. My predictions, on average, are 8 mins off the actual total delivery time.

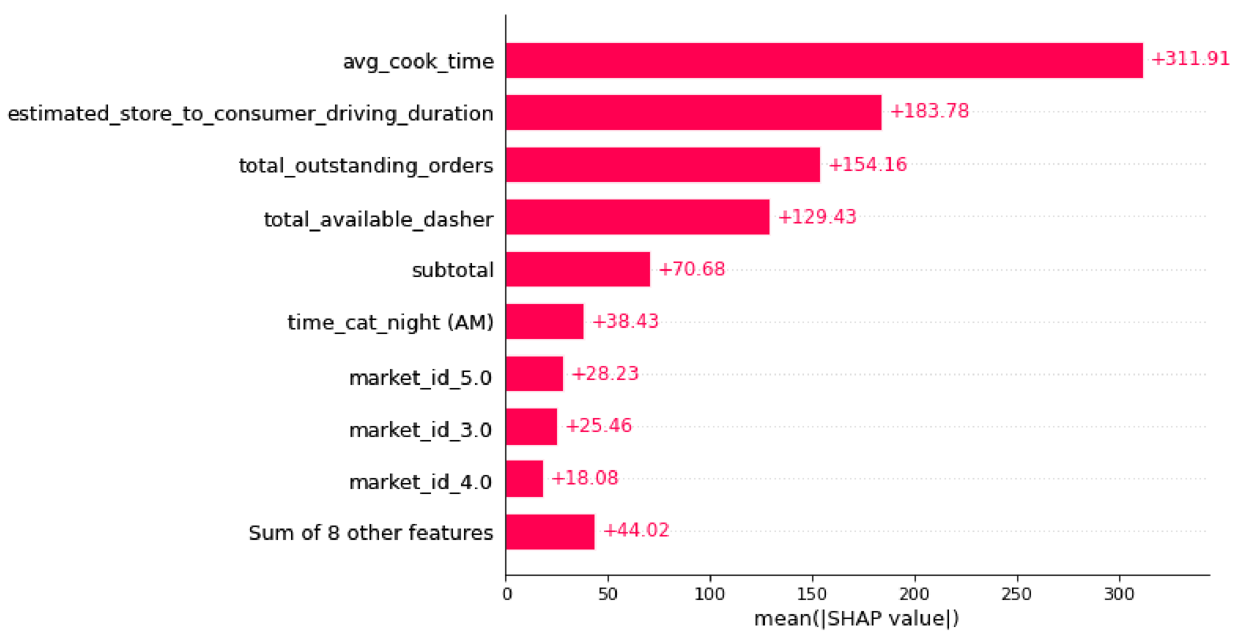
MAE is smaller than RMSE indicating that all prediction errors have different magnitudes. The reason I choose RMSE as the loss function is that RMSE has the benefit of penalizing large errors more. The errors are squared before they are averaged, the RMSE gives a relatively high weight to large errors. Therefore, RMSE is more appropriate in this DoorDash case, for example, estimated delivery time being off by 10 mins is more than twice as bad as being off by 5 mins. So I chose RMSE as the metric to measure accuracy of my model predictions.

Key Features And Potential Features To Add

As you can see in the bar chart below, the most important feature in my model is the "avg_cook_time" I created from historical data. It represents the average processing time from order created to driver pick up order for delivery. This store specific average processing time was averaged on different time periods (morning, afternoon, evening, night (AM/PM)). In this case, the model RMSE reduced to 613.809 from 672.047, and the model MAE reduced to

484.701 from 532.795. That is to say, adding “avg_cook_time” to the model, the model accuracy improved by about 48 seconds.

As I mentioned in the background section, DoorDash delivers food, convenience, grocery, alcohol, pets’ food, and flowers, etc. Different order types have different processing time before Dasher picks up the order for delivery. For example, a pre-order of a birthday cake from Walmart may take several days to deliver, while a cheese burger may just take 20 to 30 mins. Furthermore, even in the same fastfood restaurant, a family size combo takes more time to cook than a simple cheese burger. Thus, to have an informative order processing time feature, the key is to define the order types. Which category this order falls in? For example, is it food, convenience, grocery, alcohol, pets’ food, or flowers as shown in DoorDash mobile app? If it is food, is it fast food, healthy sandwich, salad, desserts, or Asian food? For example, Chinese cuisines usually cost more time to cook in comparison with cooking French fries. If it is fast food, is it just French fries or fried chicken. As I know, for Popeyes, their fresh cooked chicken tenders take at least 10 mins to process, while the fries take only 3 or 5 mins. Therefore, we can add order category features, food category features, food type features, pre-order type (how many days in advance). In summary, any feature added should help us better estimate order processing time. If we can better estimate order processing time, we can improve the satisfaction of both dashers and customers. For example, dashers may have more tips if they arrive on time and customers will be more loyal to DoorDash and even get a membership from DoorDash or become business partners with DoorDash. And if both dashers and customers’ satisfaction are improved, DoorDash business will grow faster and expand to more countries and eventually succeed in the mission of growing and empowering local economies.



Pre-launch Model Assessment

- Step 1: I would compare the RMSE using cross validation on my model with the historical data. And I did, the RMSE is stable.
- Step 2: Then I will test my model tolerance to noise and extreme scenarios by predicting the data_to_predict provided to see if the RMSE are still stable. Furthermore, add more noise and extreme values (reasonable on business sense) to create a new test data and apply my model to predict.
- Step 3: If the RMSE is still stable in step 2 , then I will compare my model with the previous production model on the same data (real business scale/ historical and new data) with the same size multiple times to make comparisons on average performances.
- Step 4: if my model is statistically better than the previous production model, I will communicate with my team and have meetings with leadership to set a business metric for A/B test to see if my model has higher business significance on improving customer satisfaction (for example, ratings, etc.) or business growing (e.g., higher retention in membership, better conversion rate, etc.)
- Step 5: If in general, my model is better than the previous production model via multiple A/B tests, I will test my model on various new data multiple times before launch. If before launch, there are any changes in DoorDash business model, I will re-train the model on new data and iterate through step 1 to 5 again before launch.