Case Study 4: Restaurant Visitor Prediction

Team 9
Tianhao Guo, Jiani Gao, Tingting Ma, Mo Cheng, Jinyan Lyu

1. Business Target

Modern catering makes it possible for people to have better food while spend less time in kitchen. When customers visiting a restaurant, they don't want to wait. Neither do restaurant owners. Because the longer they keep customers waiting, the more likely they will lose their business, while on the other hand it is also not wise to prepare more than enough food, which means a waste of cost. Thus, the ideal scenario would be, customers walks into a restaurant, and the restaurant is just ready to serve them with what they want, no more, no less. This win-win situation is hard to reach, but fortunately with the help of carefully collected data and machine learning methods we are able to create a product that helps customers and restaurants with information they need to be there.

To be more specific, our model predicts how many visitors a restaurant is expected to have in a given time.

The logic behind that is, number of customers directly affects the amount of food ingredients to prepare and staff working schedules. Ideally, a restaurant can make most efficient use of food ingredients and lower labor cost if customer visits can be precisely predicted. However in the reality, it is very difficult to make such prediction, due to many unexpected factors, like weather, local competitors. In this case study, we try to take all these factors into consideration and predict numbers of future visitors for local restaurants in Japan, based on historical data from a restaurant reservation website AirREGI and a restaurant rating website Hot Pepper Gourmet. And the well trained model can be used by both customers to decide which restaurant to attend, and restaurant owners to make decisions all the way from preparing a rush hour to expanding business.

2. Data Description

2.1 Data Overview

The original dataset which comes from Hot Pepper Gourmet (hpg) and AirREGI / Restaurant Board (air), has 7 subsets that contains reservation, visit, together with other information, is already split into training and testing sets. The training data covers the dates from 2016 until April 2017, and test set covers the last week of April and May of 2017.

The purpose of this section is to give support of our business target from the point of view of data, and a transaction to the math and machine learning session.

2.2 Data Exploration

For factors that may have effects in numbers of customers of a restaurant, we implemented several visualizations with Tableau from the given dataset for an intuitive view. This process helps us decide which feature to put into training session.

2.2.1 Hours

As presented in figure 1, line chart on left shows that visitors influx into restaurants around 6:00 pm of the day, while line chart on right suggests reservations start gradually around 11:00 am and peak in 5:00 pm. So, from the restaurant point of view, the two hours period from 4:00 pm to 6:00 pm is most critical to get prepared for the coming busy hours of the day.

If we look closer, there is a slight increase around 11:00 am in visit numbers due to lunch time, but the numbers of visitors between 10:00 am to 2:00 pm are smaller than reservation numbers of the given period, which means customers make reservation for dinner instead of lunch. This may give restaurants information about the potential customer traffic of the dinnertime, so that they can start the preparation early.

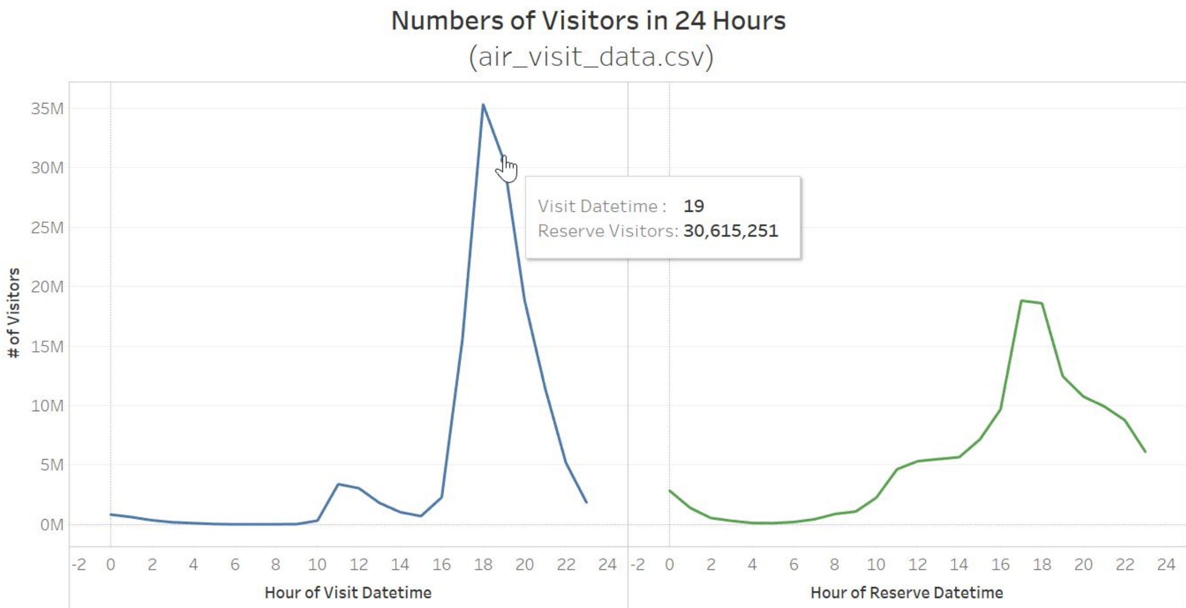

Figure 1: Total numbers of visitors of all resturants in 24 hours from AirREGI dataset.

2.2.2 Workday

Same as hours, which day of the week also influence customer's willingness to eat in restaurants. The bar chart in figure 2 shows numbers of visitors of restaurant ID: air_00a91d42b08b08d9 during a week. As we can see, Friday has the most visitors while not many in Saturday, which may due to different open hours. And we can safely predict this restaurant close in Sunday as there's almost no visitors in that day of the week.

However, rather than being an effect itself, several potential factors may make the difference of visitors in a week a corresponding result. Take the same restaurant as an example, serving

Italian/French food and locate in Tokyo-to Chiyoda-ku Kudanminami, one of the most crowded commercial district in Japan, it is possible that most of the customers are office workers and tourist, whom are unlikely to visit on off days. A ramen restaurant in residential area may have a totally different story (which means, data). So, when it comes to training session, the correlation between factors is one major concern.

Figure 2: Number of visitors in a week of a resturant (ID: air_00a91d42b08b08d9) from 11/July/2016 to 22/April/2017.

2.2.3 Locations

Among all factors a restaurant owner need to take care, location is probably the most important one. An excellent location suggests large customer traffic, which is the necessary condition of great benefit. But the down side and challenge is it is little chance other investors ignore the business opportunity, and that means fierce competition.

Figure 3 shows the distribution of restaurants in Tokyo and Japan. Having the highest population density among world metros, numbers of restaurants in Tokyo also overwhelmingly larger than other cities in Japan. As tooltips of the two charts point out, Sendai as one of the biggest city in northeast Japan has 17 restaurants in record, while the number of a single district in Tokyo is three times of that. Customer diversion should definitely be taken into consideration in prediction of customer numbers.

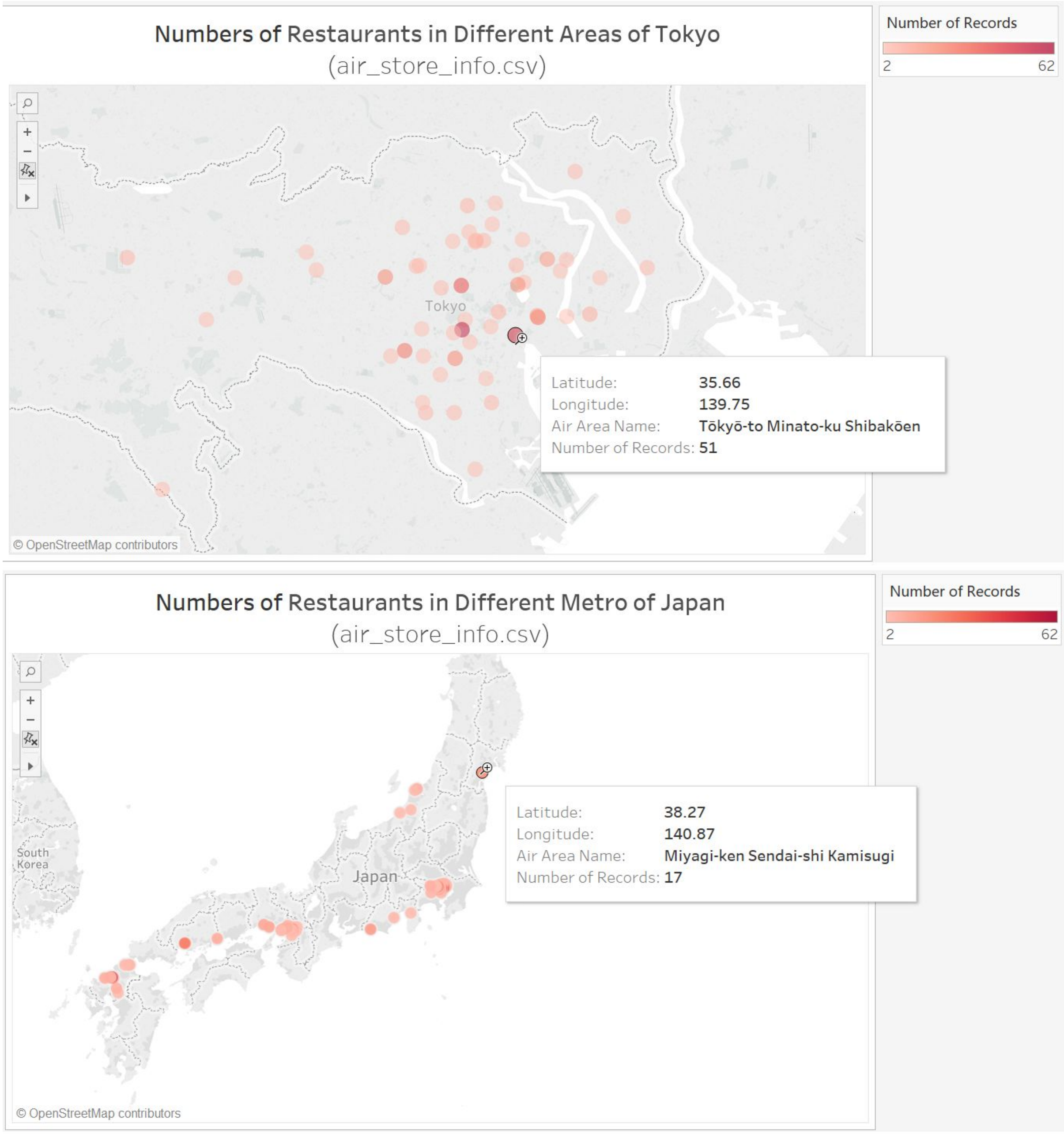

Figure 3: Distribution of restaurants in Tokyo and Japan.

2.2.4 Genres

Other restaurants in the same location are the potential opponents in business battle for sure, but among them who serve same types of food are those restaurant owners need to pay special attention to, because it directly affects the customer traffic. In figure 4, we can see the numbers of different genres of restaurants in Fukuoka-ken Fukuoka-shi Daimyo, a mature business district, are distributed pretty evenly, which makes a lot of scene because every restaurant owner has a share. For training session, genre is a feature to count.

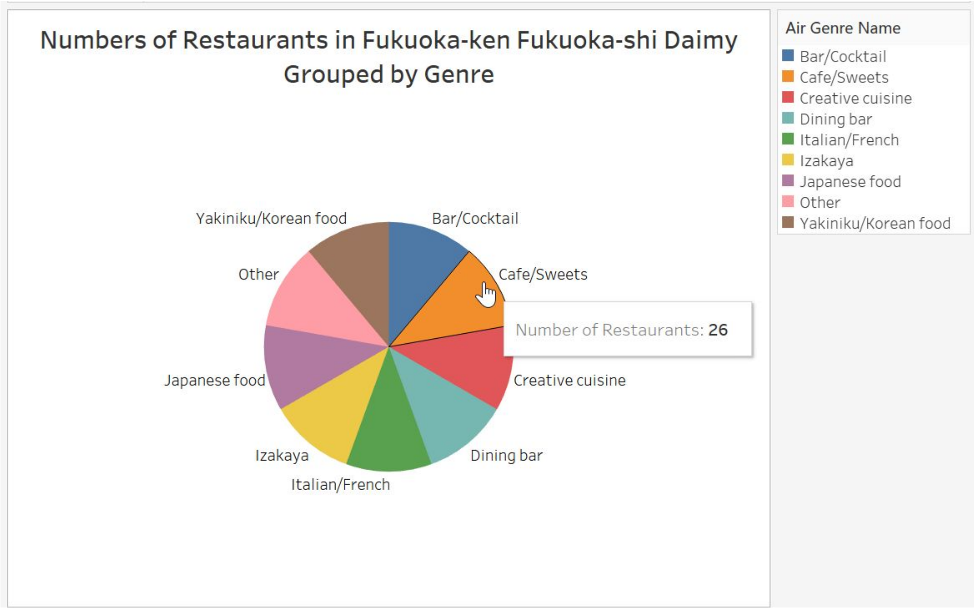

Figure 4: numbers of different genres of restaurants in Fukuoka-ken Fukuoka-shi Daimyo.

3. The Math Part

In order to predict the visitors of restaurants, we need to build a model using the multivariate time series data. Here, we choose Long Short-Term Memory (LSTM) recurrent neural network, which is well-suited to classify, process and predict time series given time lags of unknown size and duration between important events.

Recurrent neural network is extremely powerful for handling the sequential data. But, one of the most challenging quest for deep learning trying to generate sequences is to create long-term structures. Long-term structure is natural for people, however, it is complicated for machine to have this ability comparing short memories.

Long Short Term Memory networks are a special kind of RNN which is capable of learning long-term dependencies. LSTMs are explicitly designed to avoid the long-term dependency problem. All recurrent neural networks have the form of a chain of repeating modules of neural network. In standard RNNs, this repeating module will have a very simple structure, such as a single tanh layer.

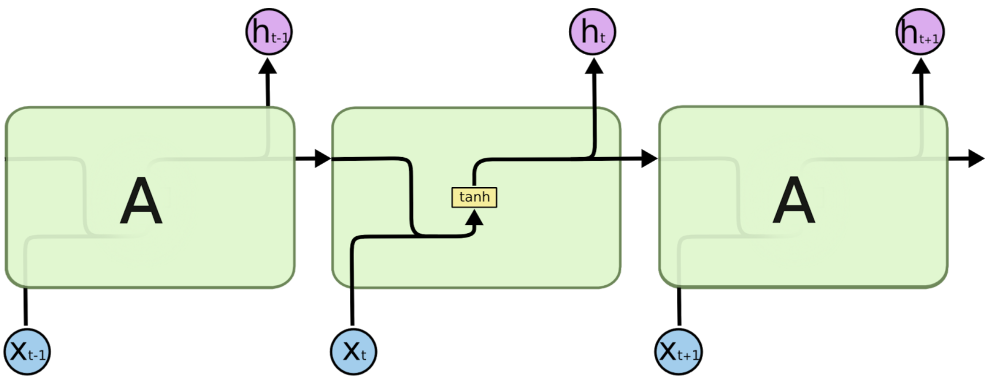

Figure 5: repeating module in standard RNN.

An LSTM network contains a (memory) cell. An LSTM cell "remembers" a value for either long or short time periods. A "standard" LSTM block contains three gates that control or regulate information flow: an input gate, an output gate and a forget gate. These gates compute an activation often using the logistic function. These gates can be thought as conventional artificial neurons. Thus each of the gates has its own parameters. Their output is multiplied with the output of the cell or the input to the LSTM to partially allow or deny information to flow into or out of the memory. More specifically, the input gate controls the extent to which a new value flows into the memory, the forget gate controls the extent to which a value remains in memory and the output gate controls the extent to which the value in memory is used to compute the output activation of the LSTM block.

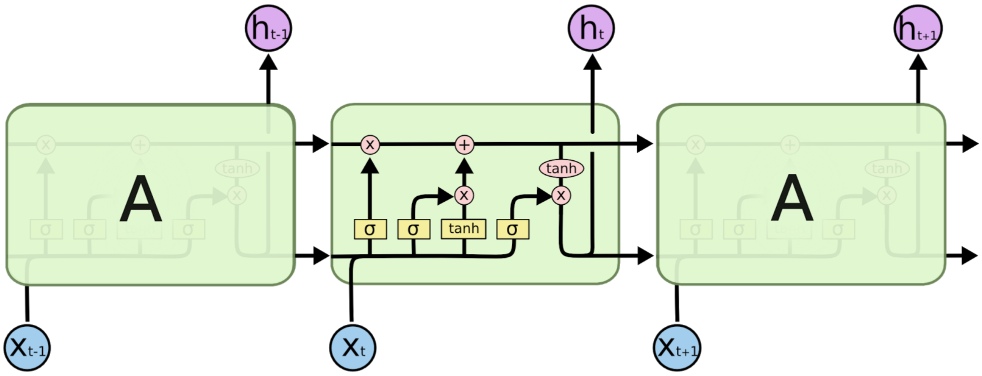

Figure 6. repeating module in LSTM.

4. The Hacking Part

First step is to process the raw data, since we have two sets of data stored in several files. We load the data to pandas DataFrames and aggregate the number of visitors by store id and date. We have the store id relationship of these two datasets and we will merge them to a single dataframe. Also, for each store, we compute the min, mean, median, max visitor numbers. We noticed that data in some fields such as genes are string type, so we need to convert them to numerical type. We used LabelEncoder in scikit-learn to encode them to numerical type and then make all values to float32 type. Now the data looks like this.

	air_store_id	visit_date	visitors	dow	year	month	min_visitors	mean_visitors	median_visitors	max_visitors	count_observations	air_genre_name
0	air_ba937bf13d40fb24	2016-01-13	25	2	2016	1	7.0	23.843750	25.0	57.0	64.0	4.0
1	air_ba937bf13d40fb24	2016-01-14	32	3	2016	1	2.0	20.292308	21.0	54.0	65.0	4.0
2	air_ba937bf13d40fb24	2016-01-15	29	4	2016	1	4.0	34.738462	35.0	61.0	65.0	4.0
3	air_ba937bf13d40fb24	2016-01-16	22	5	2016	1	6.0	27.651515	27.0	53.0	66.0	4.0
4	air_ba937bf13d40fb24	2016-01-18	6	0	2016	1	2.0	13.754386	12.0	34.0	57.0	4.0

Next, we need to convert the time series to a supervised learning problem. A time series is a sequence of numbers that are ordered by a time index. This can be thought of as a list or column of ordered values. A supervised learning problem is comprised of input patterns (X) and output patterns (y), such that an algorithm can learn how to predict the output patterns from the input patterns.

	var1(t-1)	var2(t-1)	var3(t-1)	var4(t-1)	var5(t-1)	var6(t-1)	var7(t-1)	var8(t-1)	var9(t-1)	var10(t-1)	var11(t-1)	var12(t-1)	var13(t-1)	var14(t-1)	var15(t-1)	var16(t-1)
1	0.666667	0.0	0.0	0.022556	0.076686	0.071895	0.023918	0.985714	0.214286	0.038835	0.768135	0.904147	0.0	0.0	0.0	0.0
2	0.666667	0.0	0.0	0.030075	0.132432	0.124183	0.055809	0.971429	0.214286	0.757282	0.815045	0.968543	0.0	0.0	0.0	0.0
3	0.666667	0.0	0.0	0.082707	0.288725	0.274510	0.095672	1.000000	0.571429	0.650485	0.815973	0.968186	0.0	0.0	0.0	0.0
4	0.666667	0.0	0.0	0.015038	0.035856	0.026144	0.015945	0.857143	0.142857	0.902913	0.794438	0.940490	0.0	0.0	0.0	0.0
5	0.666667	0.0	0.0	0.120301	0.228969	0.209150	0.082005	1.000000	0.500000	0.300971	0.793805	0.938517	0.0	0.0	0.0	0.0

The key function to help transform time series data into a supervised learning problem is the Pandas shift() function. Given a DataFrame, the shift() function can be used to create copies of columns that are pushed forward or pulled back. This is the behavior required to create columns of lag observations as well as columns of forecast observations for a time series dataset in a supervised learning format. We defined our own series_to_supervised() function to shift columns of variables. We used MinMaxScaler from scikit-learn to normalize all features to range(0,1). Then we shifted all columns. The data looks like this after dropping unnecessary columns.

Next, we need to define and fit the model. 75% data are splitted as training data and 25% are test data. Input data are reshaped into a 3D format [samples, timesteps, features] for LSTM. We defined the LSTM model with 64 neurons in first hidden layer and 1 neuron in output layer for predicting number of visitors. We used mean_squared_error loss function and adam optimizer. After 10 training epochs with batch size 100, we get the model and we plot the loss of training and test. We can see the trend looks similar, but the loss of training is smaller than test which means underfitting. We used this model to predict the test data and computed the root-mean-square error (RMSE). RMSE is a frequently used measure of the differences between values predicted by a model and the values actually observed. Here we get the value 0.797.

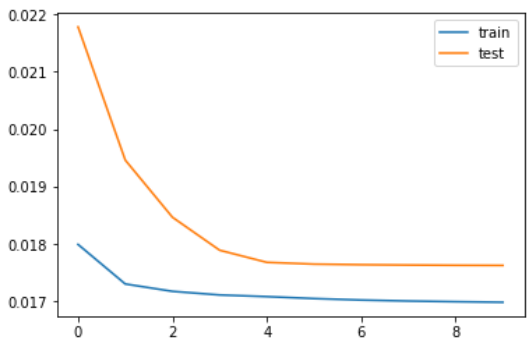

Figure 7:Loss of training and test in the model

5. Conclusion

By analyzing the result of training model and predicted test data, we can see the difference between prediction and actual data is small for enough to business product. Thus, we believe that our model can help restaurants to reasonably allocate time and prepare food according to the predicted customer traffic we provide.