Real-Time Taxi Demand Forecast Using Wireless Network Data

NTT DOCOMO, INC.
Service Development Group
Data Scientist
Shin Ishiguro
Background
Taxi drivers’ problem

- Listened from taxi drivers
  - Fresh drivers: don’t have enough knowledge
  - Experienced Drivers: After the delivery, They’d like to return to a familiar place from an unfamiliar place

Drivers can’t drive efficiently because they don’t have enough knowledge about the area demand

By using real-time population statistics data, can we estimate the future taxi demand?
Objective
Future taxi demand prediction

For improving the efficiency of taxi drivers,

Problem:
Is it possible to predict highly accurately with combination of Real-time population and taxi navigation data?

Approach:
Future demand forecast based on Deep Learning using Taxi service data and DoCoMo's "real-time population data"
Movie
About Real-time population
Real-time population

Population dynamics data using docomo’s mobile phone network mechanism. Periodically grasped mobile phones location in each area of each cellular base. About 60M subscribers' data. **Estimate the population in Japan all over the country 30 minutes before every 10 minutes with a 500m grid.**

**Such data is owned only docomo in Japan**

**We want operator partners’ in other countries!**

**Operation data**

**Data cannot identify individuals**

**Elimination of personal identity**

**Estimate real population by Considering docomos’ share rate in each area**

**Confidentialize small area**

**Estimated population**

<table>
<thead>
<tr>
<th>(40歳台男性)</th>
</tr>
</thead>
<tbody>
<tr>
<td>75人</td>
</tr>
<tr>
<td>90人</td>
</tr>
<tr>
<td>45人</td>
</tr>
</tbody>
</table>

**Real time population**

<table>
<thead>
<tr>
<th>(40歳台男性)</th>
</tr>
</thead>
<tbody>
<tr>
<td>75人</td>
</tr>
<tr>
<td>90人</td>
</tr>
<tr>
<td>45人</td>
</tr>
</tbody>
</table>
[Point A] Taxi boarding transition is synchronized with population transition

(As the population increases, the number of rides increases, and as the population decreases, the number of rides will decrease)

[Point B] Taxi boarding transition is synchronized with population transition 5 hours delay
• When and where population gathered more than usual,
• There is a possibility that the passenger acquisition rate is high and the demand can be found based on the population

Passenger Acquisition Rate

\[
\text{Passenger Acquisition Rate} = \frac{\text{Taxi num who get passengers}}{\text{Number of empty taxi}}
\]

* Per grid / per 30min
Approach
Demand forecast by Deep Learning

- Abstraction of the correlation between features by Deep Neural Network
  - Generate high dimensional features
  - Even when the correlation between data is not clear, high dimensional abstraction can be done

Depending on the characteristics of each area, the relationship between the population and the taxi that is synchronized but shifted could be expressed with DNN.
Proposed method: SdA based learning

- Enter time series / statistics data into Stacked denoising Autoencoders (SdA)
- Regression prediction in the final layer using abstracted feature quantities

Multiple kinds of time series data

- Time series feature quantity
  - (Taxi ride number, dynamic population, weather etc.)
  - Time axis (Information on past 6 hours of every 30 minutes)

+ Statistic features
  - (Same day of the week, average number of rides in the same time zone, etc.)

Input layer

Hidden layer

Output layer

Nodes are mutually Total binding

Arranged in one dimension

Stacked denoising Auto-Encoders

Abstraction of feature quantity

Prediction of correct answers

30 minutes later Demand forecast value

© 2017 NTT DOCOMO, INC. All Rights Reserved.
Creation of time series features

- **Target:** Taxi ride number 30 minutes later

- **Input:**
  - Population · rainfall · taxi data up to 6 hours ago at 30 minute intervals from target time ...
  - Population, rainfall, and taxi data one day ago ...
  - Population · rainfall · taxi data one week ago ...

Data obtained for a certain grid

<table>
<thead>
<tr>
<th>Time period</th>
<th>A</th>
<th>B</th>
<th>...</th>
<th>Rain amount</th>
<th>Drop off counts</th>
<th>Get on counts</th>
</tr>
</thead>
<tbody>
<tr>
<td>2015-08-02 10:10</td>
<td>3197</td>
<td>2912</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>2015-08-02 10:40</td>
<td>2589</td>
<td>2343</td>
<td>0</td>
<td>1</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>2015-08-02 11:10</td>
<td>2308</td>
<td>2080</td>
<td>0</td>
<td>6</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>2015-08-02 11:40</td>
<td>2182</td>
<td>1842</td>
<td>4.1</td>
<td>2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2015-08-02 12:10</td>
<td>2102</td>
<td>1690</td>
<td>3.5</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>2015-08-02 15:10</td>
<td>1698</td>
<td>392</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>2015-08-02 15:40</td>
<td>1750</td>
<td>320</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>2015-08-02 16:10</td>
<td>1834</td>
<td>234</td>
<td>0</td>
<td>3</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

Explanatory variable (input data)

In the past 6 hours data

Target variable (What to forecast)
Creation of statistic features

For each taxi / population, various statistics are designed as feature quantities in order to capture long-term and short-term trends

- Statisticize the average value, maximum value, minimum value, total of the same day of week and the same time zone
- Use statistics for one year and statistics one year ago

Examples of statistics

- Use the same month's statistics one year ago
  Acquire seasonality

- Use statistics for 1 year
  Acquire average demand

Prediction target date
2016/09/01

2015/09/01 ~ 2016/8/31 statistic features

2015/09 statistic features
Two kinds of predictive model design

• In the proposed method, all meshes are predicted with a single model
  – By adding a large amount of data at multiple points
    It can utilize "data of other places similar to each other"

Create a single predictive model as a whole
(Proposed method)
• Generalized performance is high because it uses data of all places
• Robust against noise

Create different forecast models for each location
• Effective in a unique place such as a place where there are extremely a lot of people
• it is sensitive to noise

Place 1’s model
Place 2’s model
Place 3’s model
Experiment
Based on the past data, estimate the accuracy of taxi boarding number every 500 minutes for every 500 m grid every 30 minutes and evaluate accuracy.
Evaluation experiment: Precondition

1. Taxi data: Operation data in Tokyo
2. Population data: Real-time demographic data
3. Weather data: National synthetic radar data

- Learning data period: April 1, 2016 - August 31, 2016
- Evaluation data period: September 1, 2016 - September 14, 2016
- Target area: Tokyo 23 wards + Musashino city + Mitaka city

<table>
<thead>
<tr>
<th>Data type</th>
<th>Car num</th>
<th>Resolution</th>
<th>Acquisition frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Taxi data</td>
<td>4425 units</td>
<td>GPS accuracy</td>
<td>Per 5～10 sec</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Resolution</td>
<td>500 m</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Acquisition frequency</td>
<td>Per 10 min</td>
</tr>
<tr>
<td>Population data</td>
<td></td>
<td>Resolution</td>
<td>250 m</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Acquisition frequency</td>
<td>Per 10 min</td>
</tr>
<tr>
<td>Weather data</td>
<td></td>
<td>Resolution</td>
<td>250 m</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Acquisition frequency</td>
<td>Per 10 min</td>
</tr>
</tbody>
</table>

- Convert to Last 30 min / 500 m Boarding or dropoff number count data
- Conversion to Last 30 minutes / Total for every 500 m
Evaluation experiment: Setting of neural network

- Hyper parameter search by random search is performed using Python's Hyperopt
  - For each experiment, 1000 model creation

- The activation function is ReLU
- Using Batch Normalization
- The optimization function is Adam

<table>
<thead>
<tr>
<th>Hyper parameters</th>
<th>Search space</th>
</tr>
</thead>
<tbody>
<tr>
<td>Denoising Autoencoder’s Noise coefficient</td>
<td>$0 \leq q_D \leq 0.2$</td>
</tr>
<tr>
<td>Sparse Autoencoder’s Regularization factor</td>
<td>$0 \leq \rho \leq 0.02$</td>
</tr>
<tr>
<td>Dropout ratio</td>
<td>$0.3 \leq \varepsilon \leq 0.7$</td>
</tr>
<tr>
<td>Batch size</td>
<td>$50 \leq BS \leq 200$</td>
</tr>
<tr>
<td>Unit counts</td>
<td>$10 \leq N \leq 1000$</td>
</tr>
</tbody>
</table>
Evaluate prediction accuracy by the 4-layer network
Perform learning by changing data

About 1st to 14th September 2016, every 10 minutes
We predicted the sum of the number of times of riding in the Tokyo metropolitan area 500 meters for 30 minutes,
Evaluate error from correct answer

RMSE

\[
\text{RMSE} = \left( \frac{1}{n} \sum_{i=1}^{n} (t_i - \hat{t}_i)^2 \right)^{\frac{1}{2}}
\]

Each 500 m mesh / 30 minutes, stratified by the number of total boarding

Boarding counts
Evaluation results for each input feature quantity

Compared to only taxi data
Improved accuracy by adding population and rainfall as a whole

Rainfall is working well,
Learning period: Since rain hardly fell during the evaluation period against rainy seasons from April to August,
Was the information "did not rain" work?

RMSE

<table>
<thead>
<tr>
<th>Boarding counts</th>
<th>Only taxi</th>
<th>Taxi/Population</th>
<th>Taxi/weather</th>
<th>Taxi/population/weather</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 1 2 3 4 5-</td>
<td>10-20-30-</td>
<td>40-50-60-70-80-90-</td>
<td>0 1 2 3 4 5-</td>
<td>10-20-30-</td>
</tr>
</tbody>
</table>
Evaluation results for each input feature quantity

In the place / time zone of the boarding record 50 to 90, When population is added, great accuracy improvement can be expected
In the place / time zone of the boarding record 4 to 40, using population data, accuracy improved but slightly reversed when adding rainfall data.

Possible Factors:
- The influence of the population greatly affects where the demand is high?
- The model with population has not been converged yet?

Evaluation results for each input feature quantity
Prediction example: Around Hamamatsucho

- Deep learning models accurate events
Prediction example: Around Kamata station

- Working relatively well
- Sometimes it is impossible to follow the extreme peak

Boarding counts

- target
- DL

weekends  weekends
- It is comparatively well predictable
- We can consider the difference for each day of the week
Although the overall trend is caught, prediction looks downward at the peak.
Demonstration experiment
Online estimation uses docomo's cloud

- Update once every 10 minutes, predict future ride demand for 30 minutes

**docomo’s real-time trip demand prediction**

- Multivariable Auto-Regression
- Deep Learning

Inferred from docomo’s mobile network
Population Statistics*
Taxi Data (location, status, etc.)
Weather and etc.

Prediction Result
Online Forecast Result Evaluation in Trial

Prediction accuracy: 92.9%

- High prediction accuracy is achieved especially in high demand area and low demand area
- Tackle further improvement in accuracy in the future

Areas by demand

- **Correct**
  The predicted value is within the actual value ± 20% or within ± 1 unit

- **Quasi-correct**
  The predicted value is within the actual value ± 50%
Result of sales of field demonstration

Increase sales of participants in field trials for the entire period of demonstration experiment

Average sales improvement is 1,409 JPY per driver per day (1,409 JPY ≈ 13.16 USD)

Sales result per driver per day
Sales results for December and March based on November (pre-experiment)

- **Field trial participants (26 people)**
- **All drivers (10,640 people)**

<table>
<thead>
<tr>
<th>Month</th>
<th>Field Trial Participants</th>
<th>All Drivers</th>
</tr>
</thead>
<tbody>
<tr>
<td>December</td>
<td>2,223 JPY</td>
<td></td>
</tr>
<tr>
<td>January</td>
<td>988 JPY</td>
<td></td>
</tr>
<tr>
<td>February</td>
<td>905 JPY</td>
<td></td>
</tr>
<tr>
<td>March</td>
<td>1,521 JPY</td>
<td></td>
</tr>
</tbody>
</table>
Commercial version released on February 15th, 2018 in Tokyo and Nagoya, Japan

For 1,350 taxis

For 1,150 taxis
Summary
Summary

- Taxi future demand prediction using SdA
- Confirm that prediction accuracy improves in high demand place / time zone by using population data
- Verification test realizes accuracy of 92.9%
- Confirmed sales improvement of 1,409 yen average per day per person

Future works

- Adding data: Using event information
- Realization of deeper learning algorithm with higher precision
  - Full utilization of DGX-1!

Thank you