Estimating New York Travel Times by Leveraging Geographical Separations

Sungjun Choi, Amrit Singhal, Niles Christensen

Introduction

The goal was to predict travel time of a taxi ride given its time and date, and which of New York City's 263 taxi zones it starts and ends in.

Key Insights

- **Different traffic patterns** in various boroughs of NY
- **Different levels of connectedness** between pairs of boroughs, e.g., Manhattan and Bronx are very connected, as are Queens and Brooklyn
- **Harder to move between certain sets of boroughs** due to separation by water, can only be travelled between at a limited number of places

Idea of Super-Boroughs

Key to our approach are the **sets of well-connected boroughs**, which we call "super-boroughs"

- Some connections are clear: Brooklyn and Queens are not significantly separated by geography, and Manhattan and the Bronx are connected by so many bridges that transit between them isn't very affected
- Our architecture can be simplified greatly when all trips could be done by **crossing only 1 bridge**
- Decided upon following super-boroughs:
  1. Manhattan / Bronx / Newark
  2. Queens / Brooklyn
  3. Staten Island
- Added **Newark airport** to Bronx/Manhattan super-borough, with an artificial "bridge" to Staten Island

Dealing with Size of Data

- Set up an **SQL database**
- **Structured queries** to avoid large joins
- Wrote multiple queries to access data even under resource limitations

Feature Engineering

1. Preprocessed PULoc, DOLoc, and PUDt as features
2. Found **GPS coordinates** of a bounding box for each zone, and used the mean as the location of the zone
3. Matched each taxi zone to the **borough** it belonged
4. Added **day-of-the-week** feature, as it can be an important factor that controls traffic patterns
5. Experimented with **different representations of features**, i.e. as one-hot or numeric values

Approach

1. Train **simple submodels** to predict travel times starting and ending within each super-borough
2. Train a model to **predict the most likely bridge** used for trips between super-boroughs
3. Train a **larger model to combine predictions** for time to reach bridge, time to cross bridge, and time to reach destination from bridge

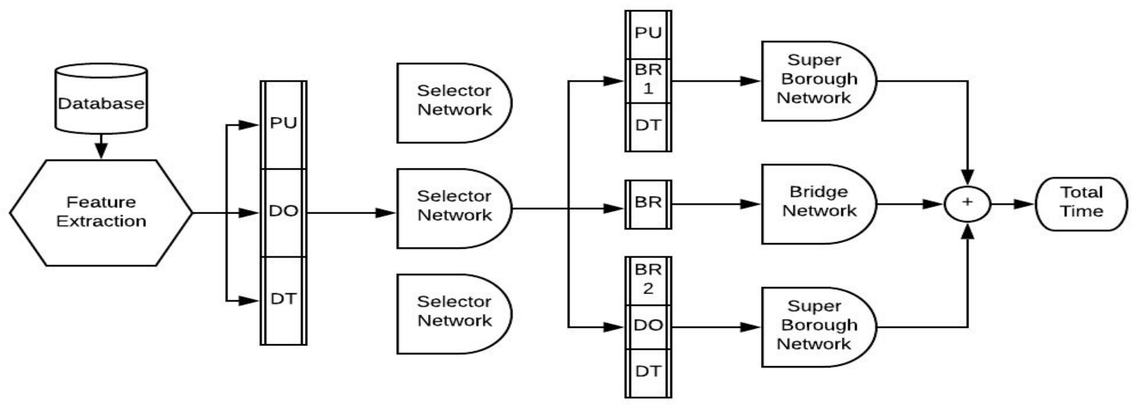
Modelling Techniques Used

1. **XGBoost Regression Trees (GBRT)**
   Gradient-boosted regression trees for each borough
   - Better exploit abundant **ordinal/nominal features**
   - Gradient boosting to do boosted regression
   - XGBoost used to speed up training
   - Can only be combined based on heuristics
   - Hard to fully utilize **cross-borough datapoints**

$$-g_m(\mathbf{x}_i) = -\left[\frac{\partial L(y_i, F(\mathbf{x}_i))}{\partial F(\mathbf{x}_i)}\right]_{F(\mathbf{x})=F_{m-1}(\mathbf{x})} \quad \mathbf{a}_m = \arg\min_{\mathbf{a},\beta} \sum_{i=1}^{N} [-g_m(\mathbf{x}_i) - \beta h(\mathbf{x}_i;\mathbf{a})]^2. \quad [1]$$

$$F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + \rho_m h(\mathbf{x};\mathbf{a}_m).$$

2. **Neural Networks**
   - Developed an **ensemble network**, combining networks for predicting the correct bridge, predicting the travel time in each borough and in the bridge, and backpropagating the final MSE loss of the total travel time throughout the network
   - Each **super-borough net is trained separately** and fine-tuned furthur while training the ensemble



Results

- With GBRTs, individual super-borough models give smaller RMSE than a single model for the whole city
- The neural net method is implemented and ongoing

Conclusion

The results we have so far suggest that leveraging the underlying geographical structure can yield improvements.

Image Courtesy: NYC Open Data (https://data.cityofnewyork.us/Transportation/NYC-Taxi-Zones/d3c5-ddgc)

[1] Friedman, J. (2001) Greedy function approximation: A gradient boosting machine. *The Annals of Statistics* **29**(5):1189-1232