

Food Trucks Demand Analysis Model by Reinforcement Learning

強化学習によるキッチンカー
需要予測モデルの構築

Xueqing Bo · Tetsuya Higuchi · Koki Miyaji
Reo Kizaki · Ryota Koizumi · Kengo Satsuki
Sarina Tsutsumi · Tomoya Miyagawa · Kou Shiina

- 16 Tokyo Univ. of Science -

Food Truck

1. Background and Objective : 背景・目的
2. Explanatory Variables and Data : 説明変数・データ
3. Problem: 課題
4. Approach : 分析アプローチ
5. Results : 実行結果
6. Future prospects : 今後の展望

Background and Objective : 背景・目的

◆ Background

- About **30%** of businesspersons prefer to go out to eat lunch

- ビジネスパーソンの約 3 割が「外食ランチ派」

- No. 1 reason for not going out to lunch :

→ **"I don't have time to go out to eat"** (44.7%)

- 7割が外食ランチに行かない理由1位: 「外に食べに行く時間が取れない」

◆ Previous objective

- To find the optimal placement of kitchen cars using PP data!!

- PPデータを使って、キッチンカーの最適な配置を知りたい

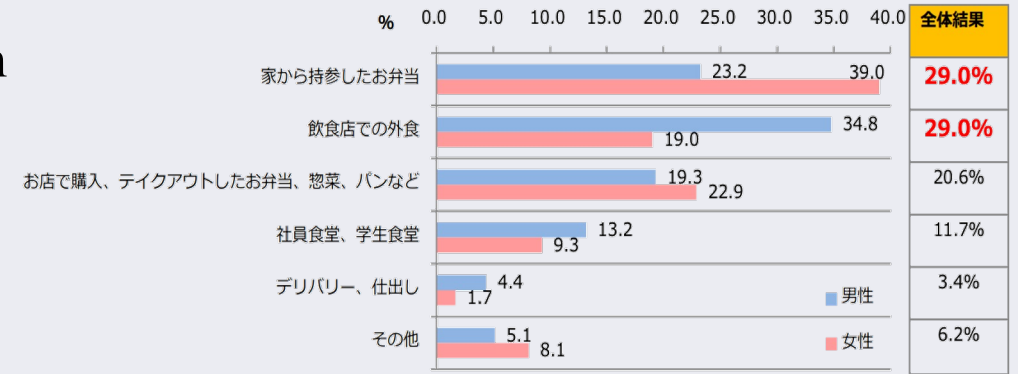


Figure 1. lunch on a workday

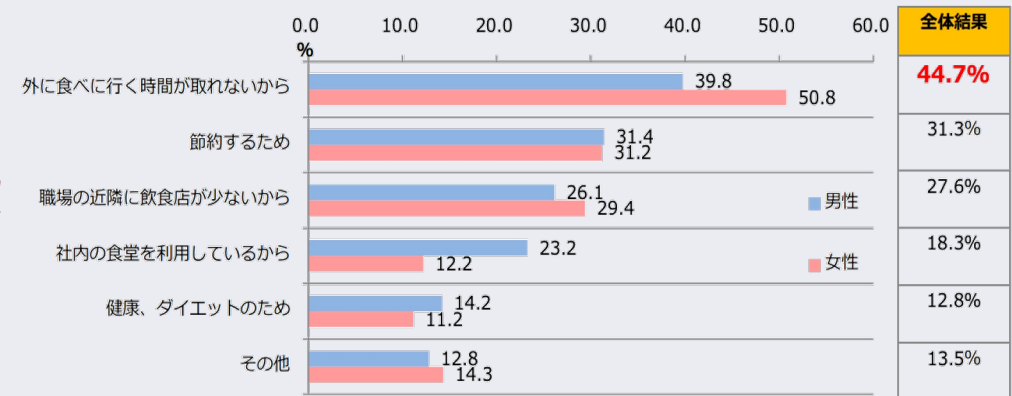
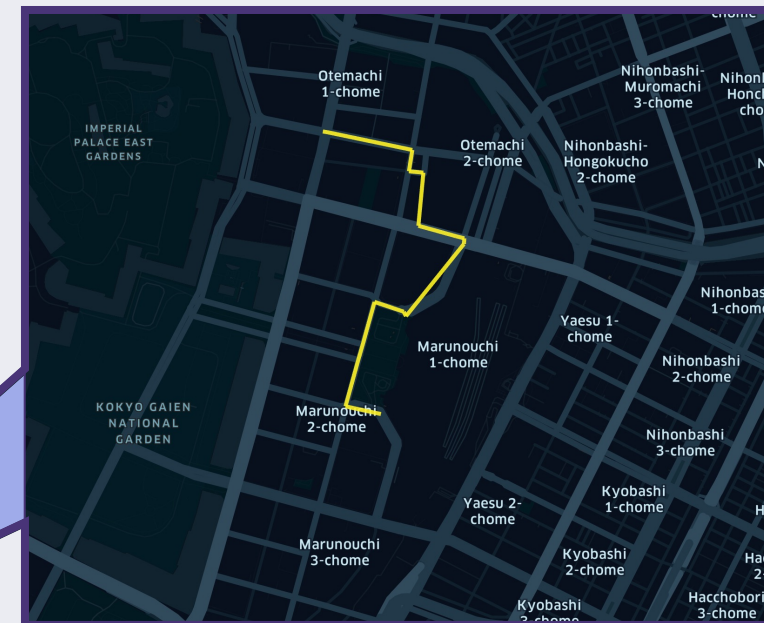
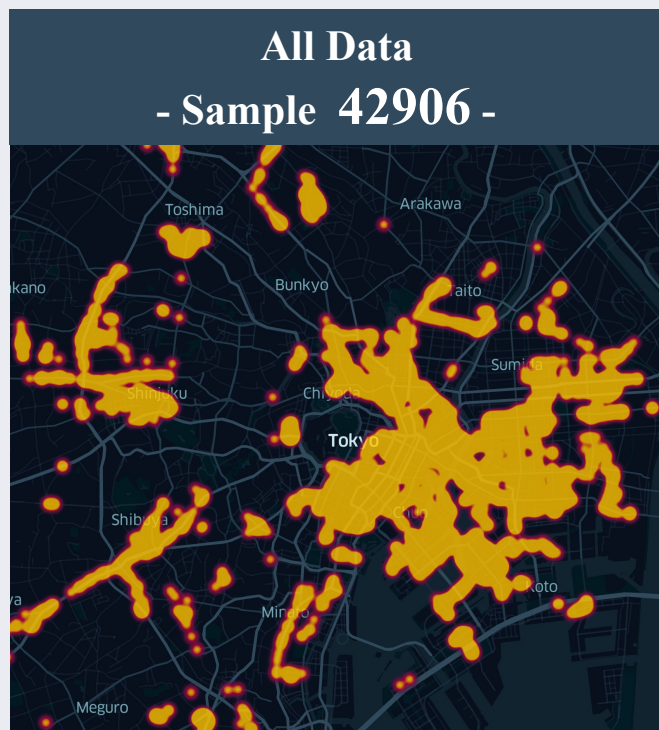


Figure 2. Why don't go out to take lunch on a workday?

There are few Tokyo PP data of "walking and meal"

- 歩行者×食事の首都圏PPデータが圧倒的に少ない



There is almost **No Data** near Tokyo Station in the business district

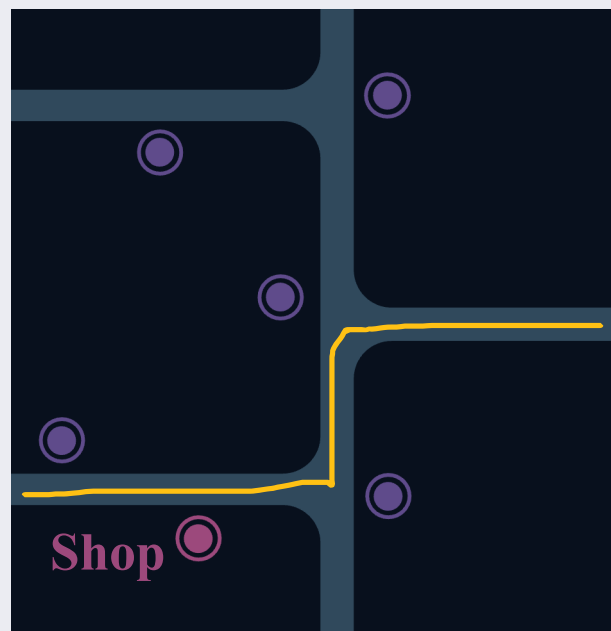
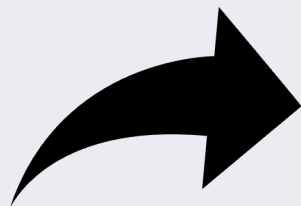
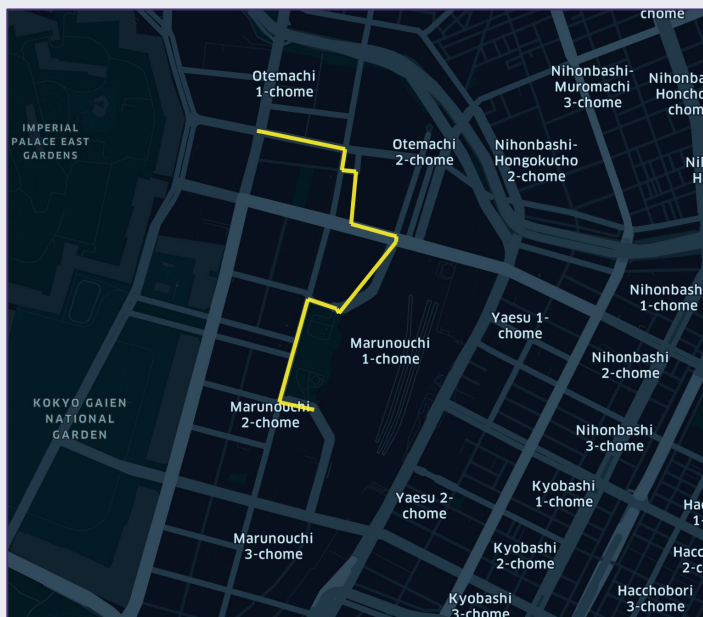
ビジネス街の東京駅付近のデータはほとんどなし

- Toyosu 2018~2019 PP Data -

◆Objective : 目的

Generate "detailed behaviors" that are difficult to observe using reinforcement learning.

観測が難しい「マイクロ単位の行動」を強化学習を用いて生成



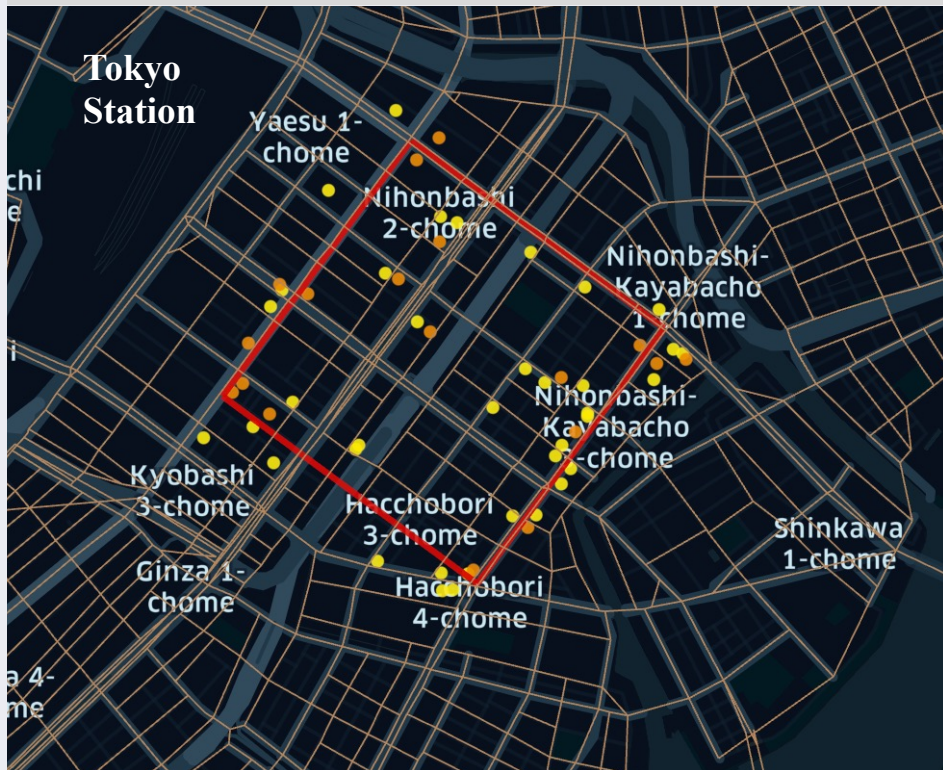
Restaurant selection results are not observable from PP, PT

PP,PTからは店舗選択実績は観測不可能

◆ Target Area : Target area: Nihonbashi (Nihonbashi, Hatchobori, Kayabacho, Kyobashi)

対象エリア“日本橋”(日本橋, 八丁堀, 茅場町, 京橋)

Target Area



1. Intersection on a Go board
2. Stores are not concentrated in buildings
3. Many genres of restaurants

1.碁盤の目, 2.店が建物に集約していない, 3.豊富な種類の飲食店

◆ Information of Restaurant : 飲食店情報

- Data : hot pepper API restaurants + chain restaurants

- ホットペッパーAPIの店情報 + チェーン店情報

- Sample numbers of restaurant : 108

- 飲食店サンプル数 : 108

- Category of food, Location of restaurant, Number of seats, Cost

- 食べ物カテゴリー, 店舗位置情報, 座席数, 費用



Compose a set of restaurant options : 飲食店の選択肢集合を構成



Figure 3 . The location of restaurants

1. Building Deep Learning Models “Grid World”

- 深層強化学習モデルの構築 “グリッドワールド”

2.Extracted from network data : ネットワークデータからの抽出

- Link Information リンク情報
- Building Information 建物情報

3. Setting up the reward function : 報酬関数の設定

- Constraint : 制約条件
- Waiting Time : 待ち時間
- Travel Distance : 移動距離
- Cost : 費用

1. Building Deep Learning Models “Grid World”

深層強化学習モデルの構築 “グリッドワールド”

Agent : Working people looking for lunch

エージェント: 昼食を探す社会人

Environment: Grid World

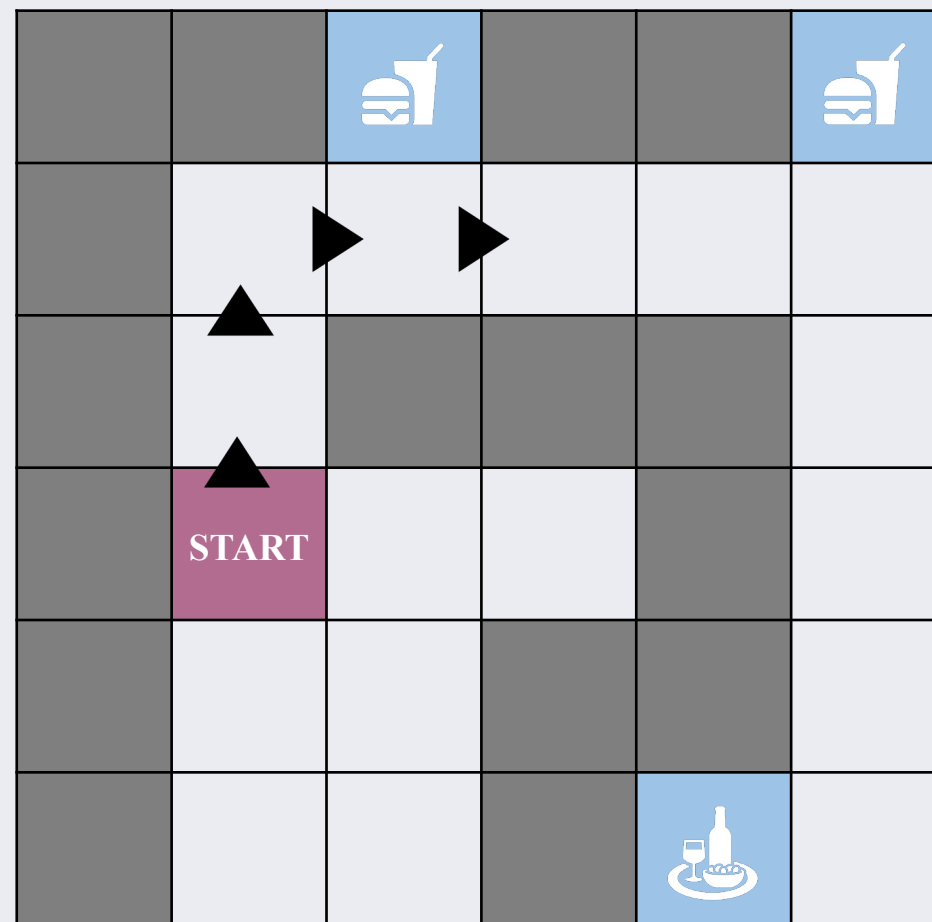
環境: グリッドワールド(10m (650m × 650m))

Polisy: ϵ - greedy

方策: ϵ - greedy

Learning: Q learning × neural network

学習: Q学習 × ニューラルネットワーク



6*6 Grid World

1. Building Deep Learning Models “Grid World”

深層強化学習モデルの構築 “グリッドワールド”

Agent : Working people looking for lunch

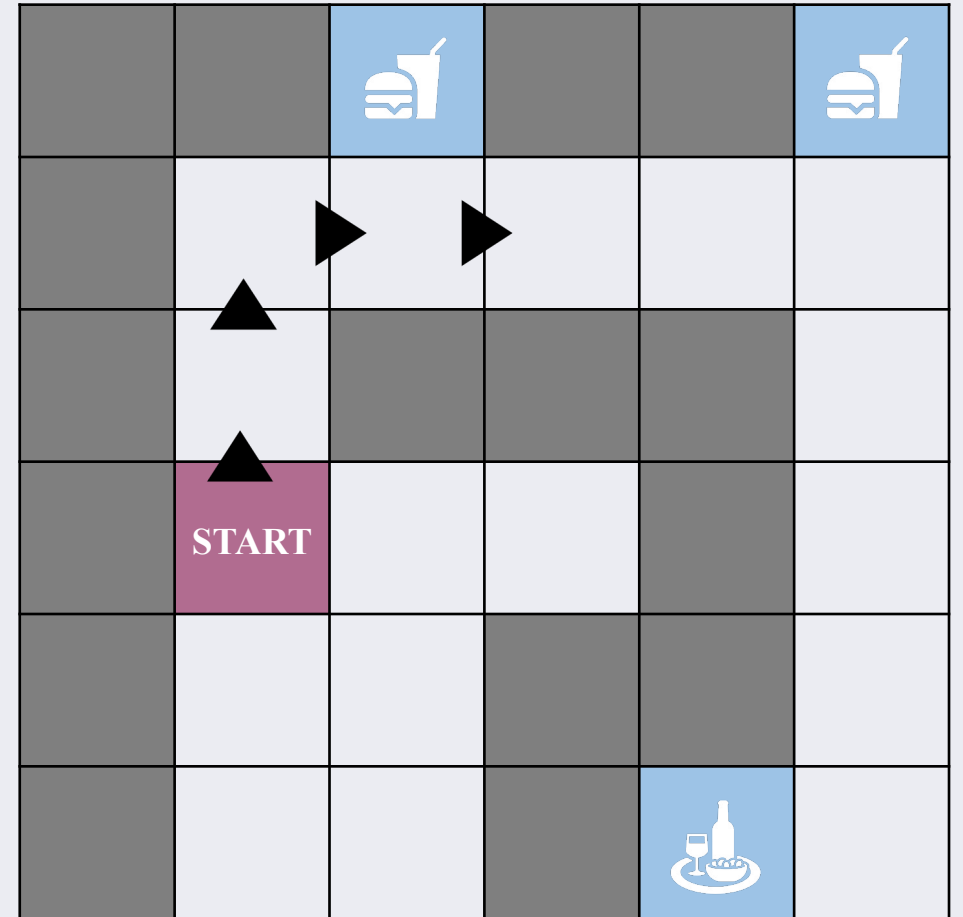
エージェント: 昼食を探す社会人

Simulation environment: 10meter mesh (650m x 650m)

シミュレーション環境: 10mメッシュ (650m × 650m)

Behavior: Searching and learning mesh for higher utility

行動: 高い効用を求めてメッシュを探索・学習



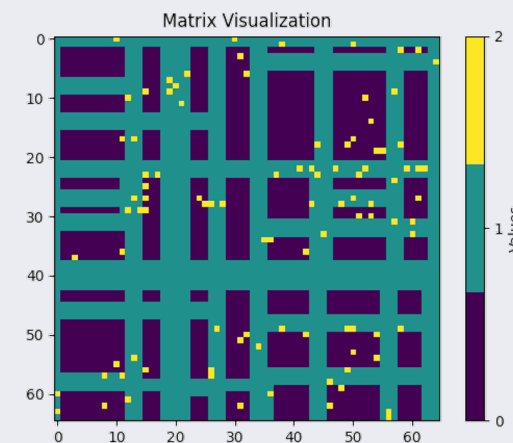
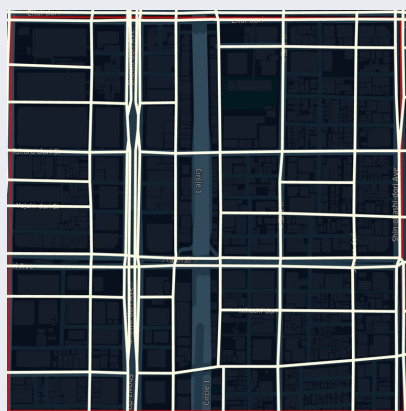
6*6 Grid World

2. Extracted from network data : ネットワークデータからの抽出

-Link Information : リンク情報

- Building Information : 建物情報

- Grid world : グリッドワールド



- Converts network and point data overlaid image to grayscale
ネットワークとポイントデータを重ねた画像をグレースケールに変換

Green = Road
Yellow = Restaurant
Black = Wall

Generate **grid world** from images of **Restaurants, Walls, and Links**

飲食店, 壁面, リンクの画像からグリッドワールドを生成

3. Setting up the reward function : 報酬関数の設定

$$V = \text{Waiting time} + \text{time}_{\text{walk}} + \text{Cost} + \text{Eating time}$$

- **Waiting Time** 待ち時間 (queueing theory: 待ち行列理論)

$$\text{Waiting time} = \frac{\rho}{1 - \rho} \times \frac{1}{\mu} \quad \rho = \frac{\lambda}{\mu} * \frac{1}{\text{cap}}$$

- $\rho = \lambda/\mu$ = Average utilization rate : 平均利用率
- μ = Average service rate : 平均サービス率
- λ = Average arrival rate : 平均到着率
- cap = capacity : 席数

- **Travel Distance** 累積移動距離

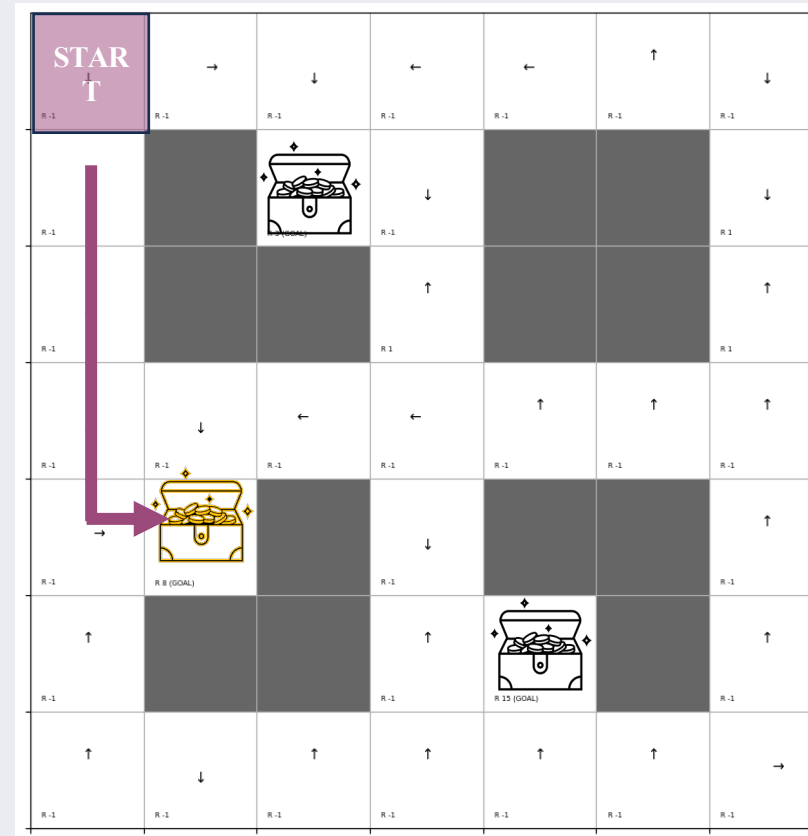
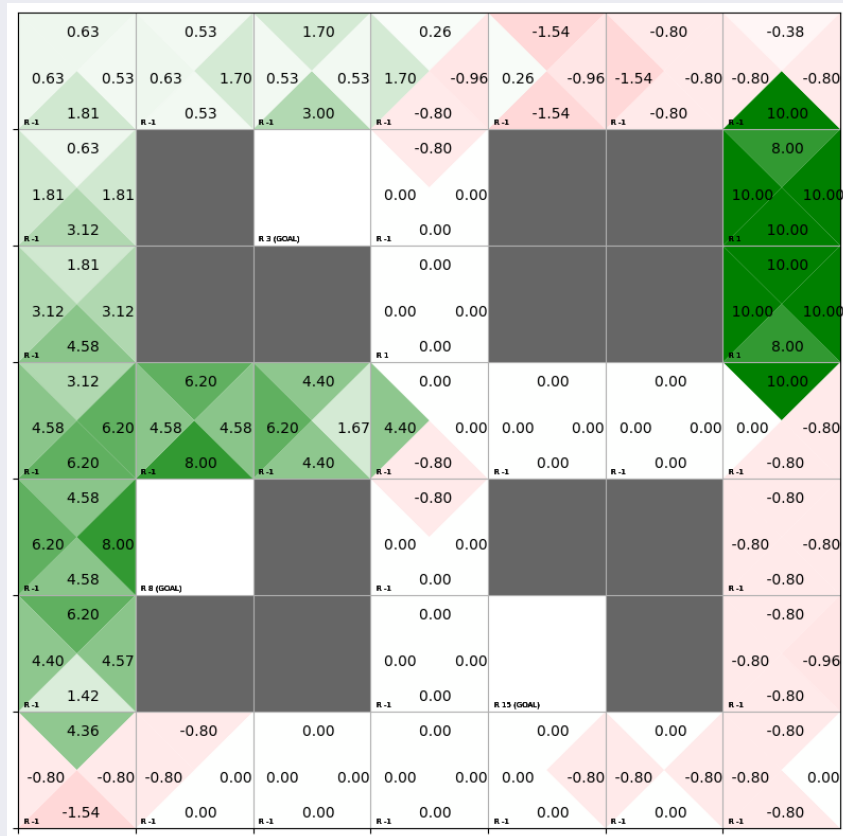
$$\text{Travel distance} = n_{\text{agent}} \times \text{gridsize} , \quad \text{time}_{\text{walk}} = \frac{\text{Travel distance}}{v_{\text{walk}}}$$

- **Cost** 費用

$$\text{Cost} = 800\text{yen}/60\text{min}$$

- n_{agent} = Agent Movement Count
エージェントの移動回数
- gridsize = Grid length
設定グリッドの長さ

• Constraint 制約条件: 60min (12:30~13:00)



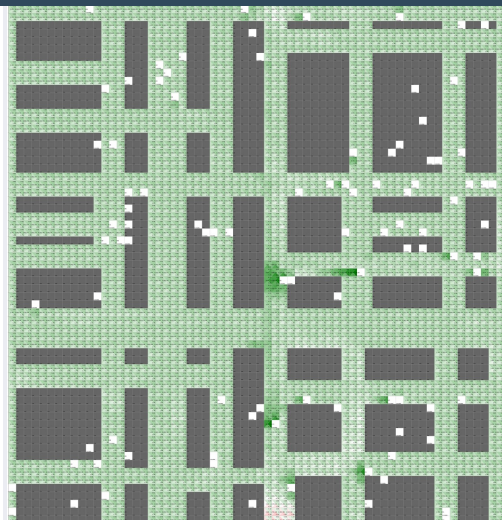
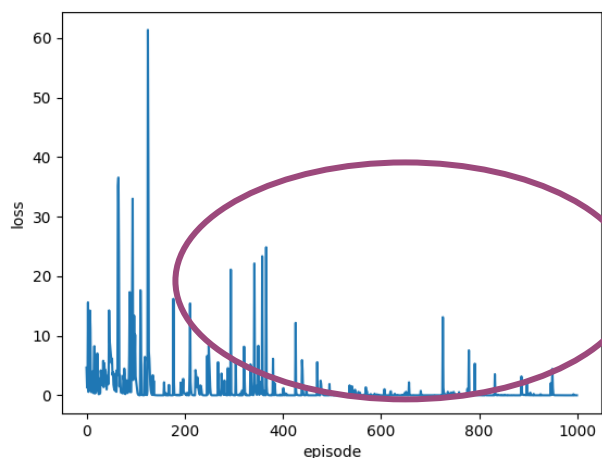
Low **Good Behavior** **High**

 低 行動の良さ 高

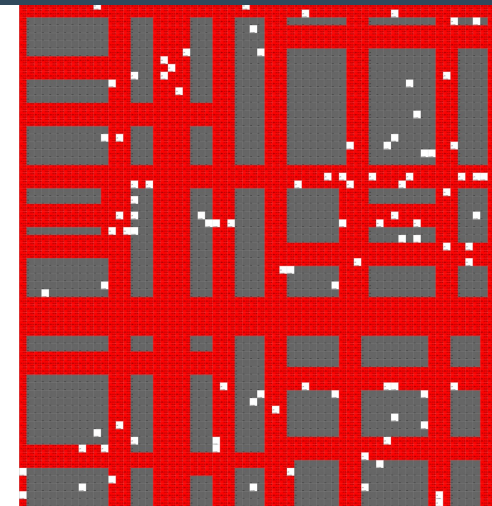
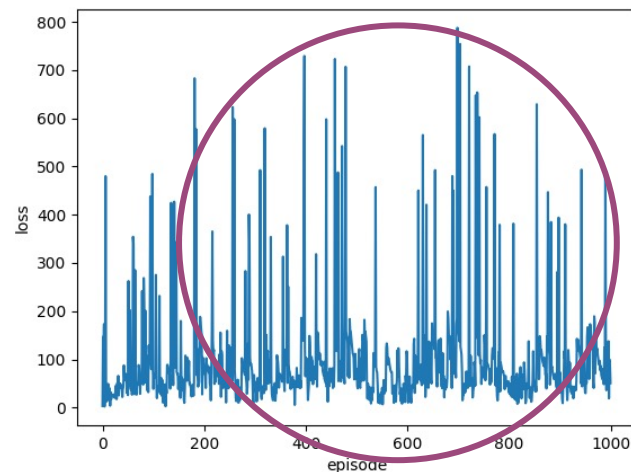
Route optimization

 ルートの最適化

Flow of people = 20 (35s)



Flow of people = 30 (411s)



- **Learning is not advanced as there are many people**
人手が多いほど学習が進んでいない
- **The Q function is generally reduced when the number of people increases**
人手が多くなった時にQ関数が全体的に低下している



Expressing hesitation in choosing a restaurant by waiting in line
店選びの迷いを待ち行列で表現できた

1. Reinforcement Learning Model : 強化学習の改良

- Complexity of reward functions : 報酬関数の複雑化
 - Restaurant information etc... : 店舗情報・個人情報・実際の混雑率の導入
- Space-time prism : 時空間制約(来た方向に戻らない)
- Multi-agent : マルチエージェント化

2. Comparison with the Destination Choice model : 目的地選択モデルとの比較

3. Model application : モデルの空間移転性の確認

- ゼロから作るdeep learning4
- https://www.jstage.jst.go.jp/article/srs1970/33/3/33_3_269/_pdf

Number of samples : サンプル数

- Sample numbers is insufficient サンプル数が足りない
- The choice set of restaurant selection is huge 店舗選択の選択肢集合は膨大
→ 推定状の問題

Determining selection performance : 選択実績の判断

Impossible to determine restaurant selection from PP data : PPデータから店舗の選択を判断不可能

→ Inference from route data : 経路データから推測

→ Final point is a node such as intersection : 最終地点は交差点等のノード

V = Waiting time at the restaurant + Walking time to the restaurant + Eating time + Food expense



Queueing theory



Calculate the time from the place of departure to the restaurant using coordinate information



Category of food



Generalization cost

V = 店での待ち時間 + 店までの徒歩時間 + 飲食時間 + 食費



待ち行列理論



座標情報を用いて
出発地から店までの
所要時間を算出



食事ジャンル



一般化費用

◆ Waiting Time : 待ち時間

Waiting time at the restaurant : Waiting time occurs after full occupancy

- 待ち時間 : 待ち時間 T_w は満席後に発生

(1) Determination of waiting time : 待ち時間発生有無の判定

- **Full or Empty** : 満席 or 空席

Full : Capacity < Total number of customers - customer leaving a restaurant

- 満席 : 店の収容人数 < 合計来店人数 - 退店人数

(2) Queueing theory : 待ち行列理論

$$\text{Waiting time (} T_w \text{)} : \text{待ち時間 } T_w = \frac{\rho}{1-\rho} \times \frac{1}{\mu}$$

$\rho = \frac{\lambda(t)}{\mu}$: level of congestion : 混雑度

$\lambda(t)$: t時刻での来店人数/時間

μ : ジャンル/時間 (時間当たり何人捌けるか)

Walking time to restaurant

歩行速度

$$v_{walk} = 80 \text{ m/min} \times 0.67 \\ \cong 55 \text{ m/min}$$

累積移動距離

$$x_{sum} = n_{agent} \times gridsize$$

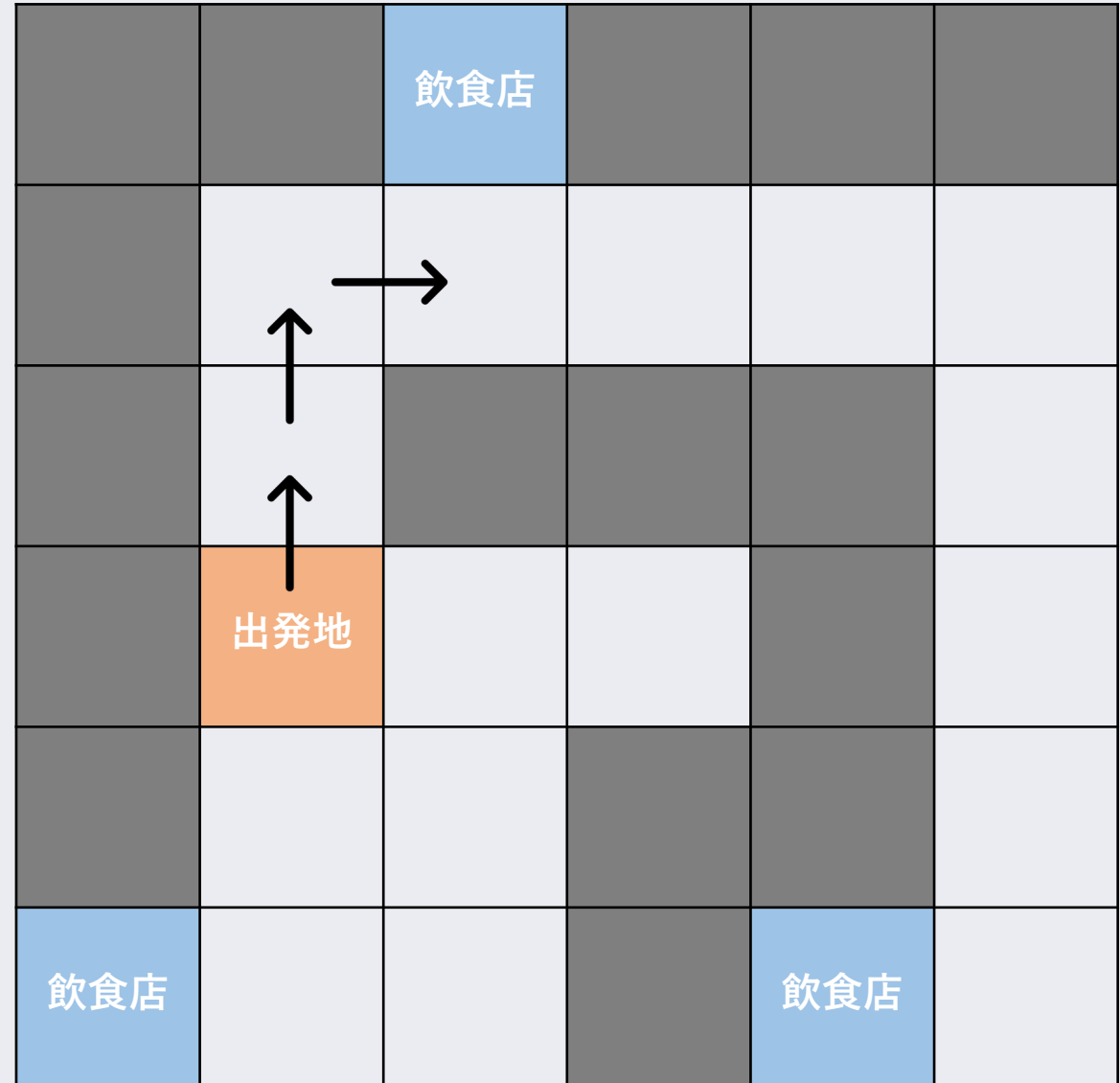
n_{agent} エージェントの移動回数

$gridsize$ 設定グリッドの長さ

徒歩時間

$$t_{walk} = \frac{x_{sum}}{v_{walk}}$$

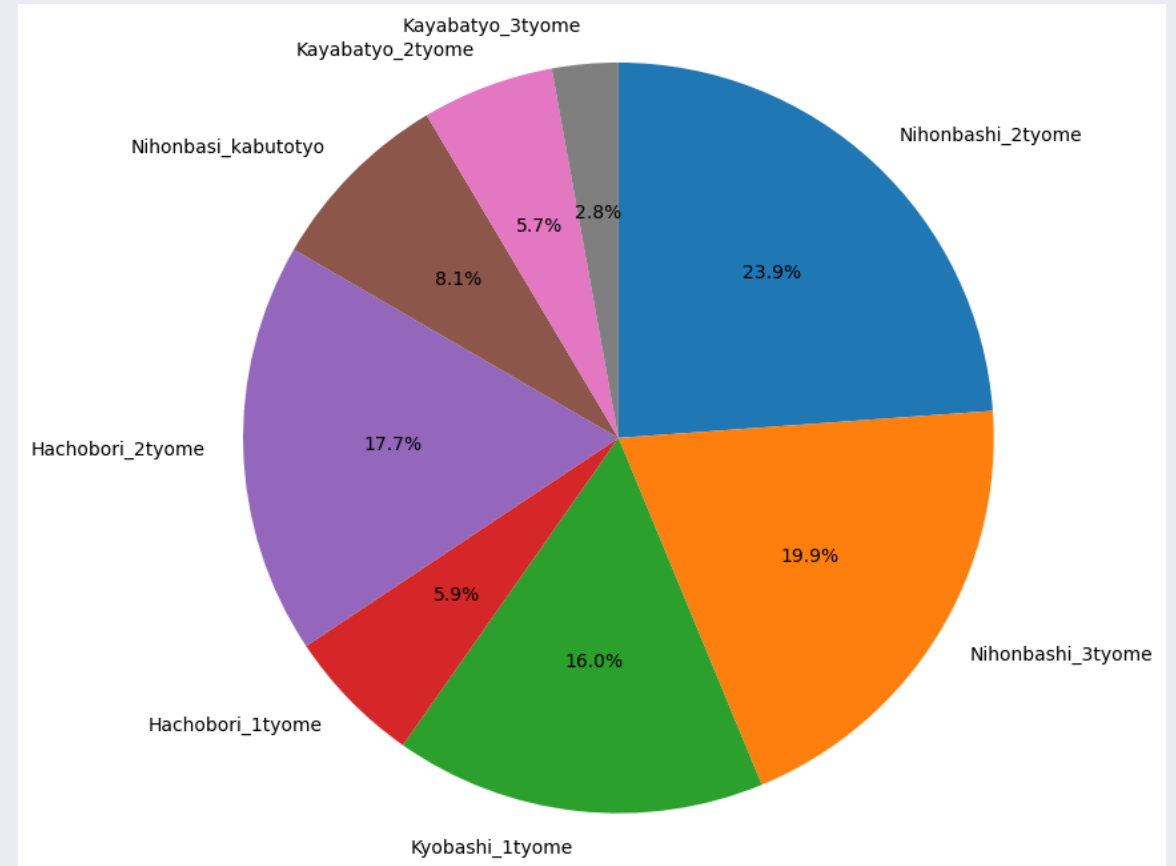
→片道10分, 530mほどの移動を想定



- 時間帯別来店人数

昼間人口 = 町丁目で6~9に分類

仮定: エリア内全店舗で一緒の来店人数



Category of food

| Category (16) ジャンル | Eating time 食べる時間 |
|-----------------------------------|----------------------|
| Soba · Udon そば · うどん | 20min |
| Ramen ラーメン | 20min |
| Hamburger バンバーガー | 20min |
| Beef bowl 牛丼 | 20min |
| Chinese food 中華料理 | 30min |
| Curry カレー | 30min |
| Coffee カフェ | 30min |
| Japanized western food 洋食 | 30min |
| Family restaurant ファミレス | 30min |
| Izakaya 居酒屋 | 30min |
| Japanese food 和食 | 30min |
| International food 各国料理 | 30min |
| Dining bar ダイニングバー・バル | 30min |
| Korean food 韓国料理 | 40min |
| Italian · France food イタリア · フレンチ | 60min |
| Yakiniku · organ meat 焼肉 · ホルモン | 60min |