



CONTEXT-AWARE TAXI DISPATCHING AT CITY-SCALE USING DEEP REINFORCEMENT LEARNING

ZHIDAN LIU , JIANGZHOU LI AND KAISHUN WU

AGENDA

- Motivación
- Formalización del problema
- Diseño de COX
- Resultados obtenidos
- Análisis y conclusiones

MOTIVACIÓN: COORDINACIÓN DE RIDE-HAILING



Las plataformas de Ride-hailing (ej. Uber, Didi) han emergido en los últimos años como una opción atractiva de movilidad, moviendo millones de viajes diarios.

Sin embargo,

- Un número considerable de pedidos quedan sin atender por falta de vehículos.
- Existen numerosos taxis que pasan gran parte de su tiempo buscando pasajeros.

Desbalances oferta/demanda que deterioran la eficiencia y la experiencia de usuario.

MOTIVACIÓN: TAXI DISPATCHING PROBLEM

- Solución: Taxi Dispatching Problem.

Balancear las diferencias entre oferta y demanda mediante la relocalización proactiva de los taxis de manera de maximizar el cumplimiento de pedidos, tanto actuales como futuros.


Desafíos:

- Dimensión del problema.
- Dinamismo de la oferta y demanda.

FORMALIZACIÓN DEL PROBLEMA

Dada la localización y la disponibilidad de los taxis, y toda la información de las solicitudes de servicio (lugar de subida, lugar de bajada y tiempo de envío) el Taxi Dispatching Problem busca decidir en qué zonas y cuando estará localizado cada taxi disponible para tomar un viaje, de manera de maximizar el número total de viajes realizados y la calidad del servicio para el pasajero.

Usualmente se particiona tanto el tiempo como la distancia en sets discretos para simplificar el problema.



Time \ Zone	t_0	t_1	t_1
z_3	$[v_1]$	\emptyset	\emptyset
z_6	$[v_2]$	\emptyset	\emptyset
z_7	\emptyset	$[v_1, v_2, p_1]$	$[v_1, p_1]$
z_{10}	\emptyset	$[p_2]$	$[v_2, p_2]$

Ejemplo de motivación, extraída de Liu et al (2020)

MOTIVACIÓN: TAXI DISPATCHING PROBLEM

Dos enfoques principales:

- Modelación explícita de oferta/demanda basada en datos históricos.
- Enfoque libre de modelación, basado en Deep Reinforcement Learning.

Este último obtiene resultados muy superiores al primero, sin embargo, no son suficientemente eficientes debido a los siguientes factores.

Aspecto a mejorar DRL	Trabajos anteriores	Propuesta
Conformación de zonas consistentes con la conectividad vial	Dividen la ciudad en grillas	Clustering en gráfico de la red vial
Representación completa de estados de oferta/demanda	Consideran los taxis actualmente disponibles como la oferta	Considerar ambos, actuales y futuros, taxis disponibles como la oferta
	Codifican la demanda futura implícitamente en los estados del DRL	Se predice la demanda por zona de forma independiente
Coordinación de acciones	Se realizan acciones por taxi sin coordinación	Se modela un agente coordinador global del problema

DISEÑO DE COX

Tres módulos principales:

- Relevamiento de contexto (Context Acquisition)
- Modelo Deep Q-Network
- Simulador de ambiente (Environment Simulator)

RELEVAMIENTO DE CONTEXTO

- Conformación de zonas consistentes con la conectividad vial mediante **CARnet**.

Se inicializan los clústeres a partir de una solución simple (una grilla con k elementos), del cual se toma los nodos más cercanos a los respectivos centroides. Se clasifican los arcos a partir de los nodos de origen. Se computan pesos a cada clúster a partir de los arcos que contiene.

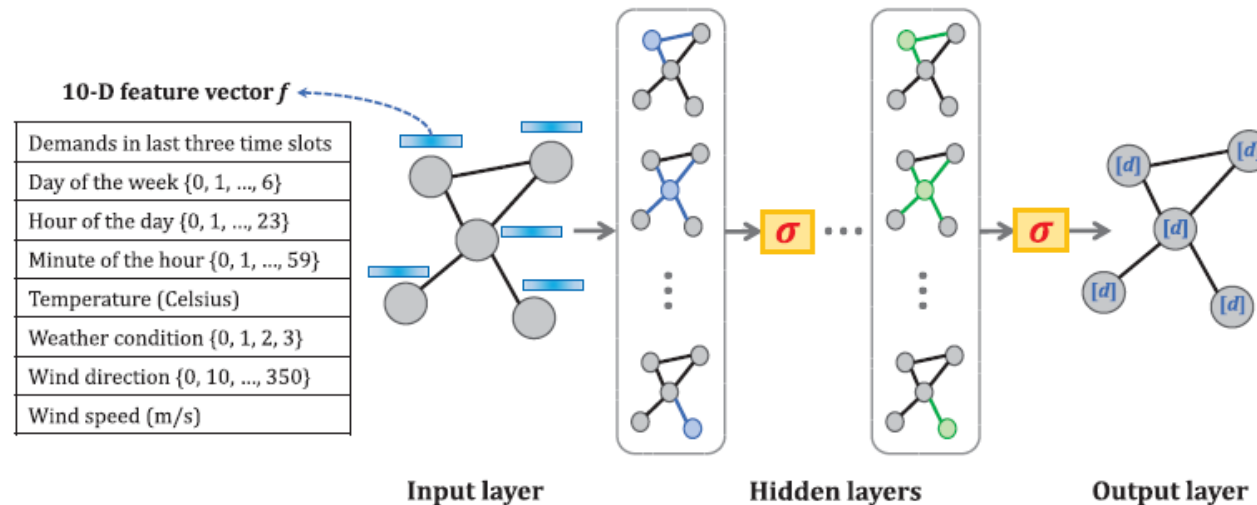
Se toma el clúster con menos arcos y se le agrega el nodo (y arcos correspondientes) con menor costo nodo-clúster. Dicho costo se define como la suma de: el costo de viaje desde el centroide al nodo y el menor costo de viaje a los vértices del clúster.

Se repite hasta tener todos los nodos clasificados

RELEVAMIENTO DE CONTEXTO

- Predictor de demanda.

Se utiliza un Graph Convolutional Network (GCN). Consiste en aplicar kernels a un gráfico para aprender patrones considerando la vecindad dada por la red.



MODELO DEEP Q - NETWORK

Se considera el centro de despacho como el agente. Al principio de cada período, el agente utiliza el modelo DQN para generar una acción para cada taxi disponible, de forma secuencial, basado en el contexto. Se considera:

- Estado contextual: información de oferta/demanda a nivel zonal de los k vecinos más cercanos.
- Acciones: enviar un vehículo a alguno de los k – vecinos más cercanos o que permanezca en la zona.
- Recompensa: basada en el impacto respecto a la tasa oferta/demanda tanto en la zona objetivo como en la actual. Suponiendo que i es la zona actual y g la objetivo, se define de la siguiente manera:

$$r_t = \begin{cases} 5 & 0 \leq \omega_{zi} \leq 1 \text{ \& } i == g, \\ -5 & 0 \leq \omega_{zi} \leq 1 \text{ \& } i \neq g, \\ \frac{1}{\omega_{zg}} & \omega_{zi} > 1 \text{ \& } 0 \leq \omega_{zg} \leq 1, \\ 0 & \omega_{zi} > 1 \text{ \& } \omega_{zg} > 1 \text{ \& } i == g, \\ -\omega_{zg} & \omega_{zi} > 1 \text{ \& } \omega_{zg} > 1 \text{ \& } i \neq g. \end{cases}$$

MODELO DEEP Q - NETWORK

Utilizan una arquitectura basada en V. Mnih (2015) para aproximar Q .

- Entrenamiento.

Se enfrenta al agente a una situación simulada, se ejecuta una decisión y se actualiza Q de la siguiente manera:

$$Q^*(s, a) \leftarrow Q(s, a) + \alpha [r + \gamma \max_{a^*} Q(s, a^*) - Q(s, a)]$$

Se adoptan dos técnicas para evitar inestabilidades en el entrenamiento:

- Target Network. Se “congela” una copia de la estimación de la función de valor en una red *target* para tomarla como referencia estable por un determinado número de pasos de entrenamiento.
- Prioritized Experience Replay. Registra experiencias previas del agente en forma de tuplas, para luego utilizar muestras de experiencia previa en la actualización de mini batches o directamente de la función de Q – learning.

Función de pérdida:

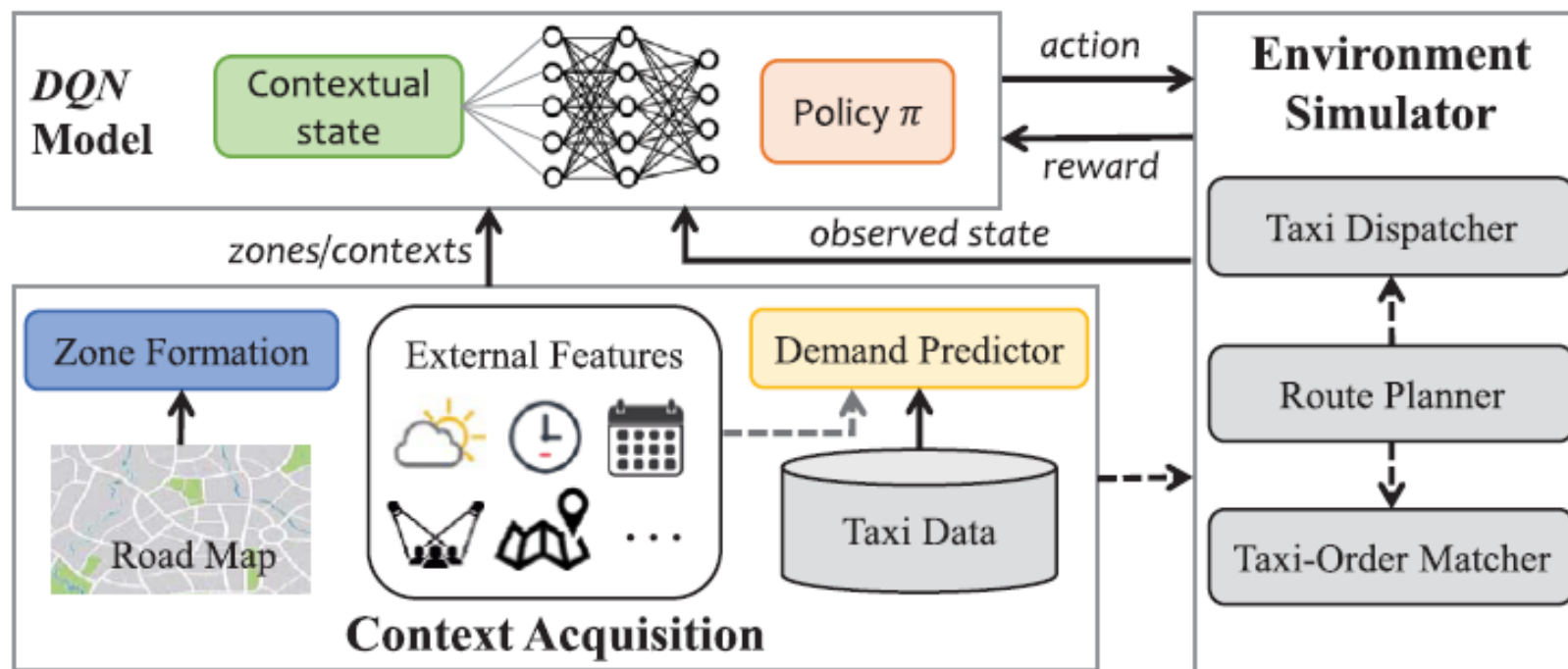
$$\mathcal{L}(\theta) = \mathbb{E} [((r + \gamma \max_{a^*} Q(s, a^*; \theta')) - Q(s, a; \theta))^2]$$

SIMULADOR

Se diseña e implementa un entorno de simulación que emula todo el procedimiento de una plataforma de ride-hailing a la hora de manejar vehículos y procesar pedidos. Sus componentes principales son las siguientes.

- Generación de viajes basada en datos reales, de la plataforma Didi en la ciudad de Chengdu, China. Se incorporaron además datos externos de meteorología.
- Módulo de planificación de rutas. Se utiliza Dijkstra para hallar la ruta mínima y con ello, estimar los costos y tiempos de viaje asociados a cada ruta.
- Condición de rechazo de viaje. Si un pedido no puede ser atendido por un taxi disponible en un cierto tiempo, se rechaza el viaje.
- Taxi Dispatcher. Módulo de ejecución de ordenes establecidas por el agente y de evaluación de recompensa para cada zona

RESUMEN ARQUITECTURA



Esquema de arquitectura COX, extraída de Liu et al (2020)

EVALUACIÓN

Se compara COX con seis enfoques representativos al problema de taxi dispatching:

- Sin relocalización de unidades.
- Relocalización aleatoria.
- Greedy.: cada taxi se mueve hacia la zona cercana con menor tasa oferta/demanda.
- cDQN
- MOVI
- Adaptative cooperative rebalance: enfoque greedy que contempla la oferta/demanda futura.

Métricas de comparación:

- Tasa de rechazo.
- Tasa de relocalización.
- Costo promedio de coordinación.
- Tiempo de espera promedio.
- Valor total de servicio

EVALUACIÓN

PERFORMANCE COMPARISONS OF DIFFERENT APPROACHES ON THE FIVE METRICS, WHERE THE UNITS FOR BOTH COORDINATION COST AND WAITING TIME ARE *Minutes*, N/A REPRESENTS *Not Available*, AND *cDQN** INDICATES THE RESULTS OF *cDQN* ON THE SMALLER GRIDS

# of taxis	Approach	Reject rate	Average repositions	Coordination cost	Waiting time	Normalized TOV
6000	<i>Simulation</i>	40.94%	N/A	N/A	1.10	100.00%
	<i>Random</i>	31.28%	8.65	6.79	0.96	116.16%
	<i>Greedy</i>	31.98%	18.72	6.77	1.83	115.64%
	<i>cDQN</i>	43.24%	10.17	6.29	1.11	96.10%
	<i>cDQN*</i>	28.41%	14.20	4.03	1.23	122.01%
	<i>MOVI</i>	28.77%	8.76	6.30	0.98	120.46%
	<i>ACR</i>	28.06%	8.51	7.03	0.94	121.81%
	<i>COX</i>	24.32%	7.04	5.92	0.68	127.71%
9000	<i>Simulation</i>	30.62%	N/A	N/A	0.93	100.00%
	<i>Random</i>	19.36%	11.23	6.81	0.68	116.04%
	<i>Greedy</i>	21.91%	24.54	6.72	0.77	112.94%
	<i>cDQN</i>	37.18%	10.62	6.98	0.99	91.03%
	<i>cDQN*</i>	19.32%	16.77	4.00	1.03	116.20%
	<i>MOVI</i>	16.40%	11.91	6.20	0.74	120.40%
	<i>ACR</i>	14.64%	9.64	6.98	0.62	122.98%
	<i>COX</i>	11.48%	9.60	5.60	0.46	126.95%
12000	<i>Simulation</i>	24.71%	N/A	N/A	0.82	100.00%
	<i>Random</i>	13.87%	13.03	6.85	0.52	114.21%
	<i>Greedy</i>	16.13%	28.08	6.62	0.64	111.67%
	<i>cDQN</i>	34.04%	9.43	6.84	0.91	88.03%
	<i>cDQN*</i>	12.24%	16.15	4.04	0.85	116.31%
	<i>MOVI</i>	10.28%	11.81	6.15	0.60	119.09%
	<i>ACR</i>	8.67%	9.46	6.88	0.44	121.10%
	<i>COX</i>	7.15%	8.31	5.63	0.35	122.77%

EVALUACIÓN

Se evalúan los distintos componentes del diseño. Se tienen los siguientes resultados.

- *Eficiencia computacional: ejecutable en tiempo real.*
- CARnet resulta significativamente beneficiosa a la hora de solucionar el Taxi Dispatching Problem.
- El enfoque propuesto con GCN resulta superior en la estimación de demanda respecto a enfoques tradicionales y MOVI.
- La incorporación de información futura de la oferta impacta significativamente en la tasa de rechazo.

CONCLUSIONES Y ANÁLISIS

- Se mejoran los resultados obtenidos en el TDP respecto a los enfoques previamente reportados en la literatura.
- Se obtiene un modelo ejecutable a tiempo real.
- El modelo propuesto tiene flexibilidad para incorporar más características a la descripción de entorno y abarcar situaciones inusuales.
- El enfoque de agente centralizado, sin incorporar incentivos a los conductores, no parece realista.



¡MUCHAS
GRACIAS!

MGUTIERREZ5@UC.CL