NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE
School of Computer Science and Engineering
College of Engineering

Autonomous Vehicle Routing in Smart Cities

Fast Heuristic Algorithms for the Travelling Salesman Problem using Graph Convolutional Neural Networks

Student: Chaitanya K. Joshi, Supervisor: Dr Xavier Bresson

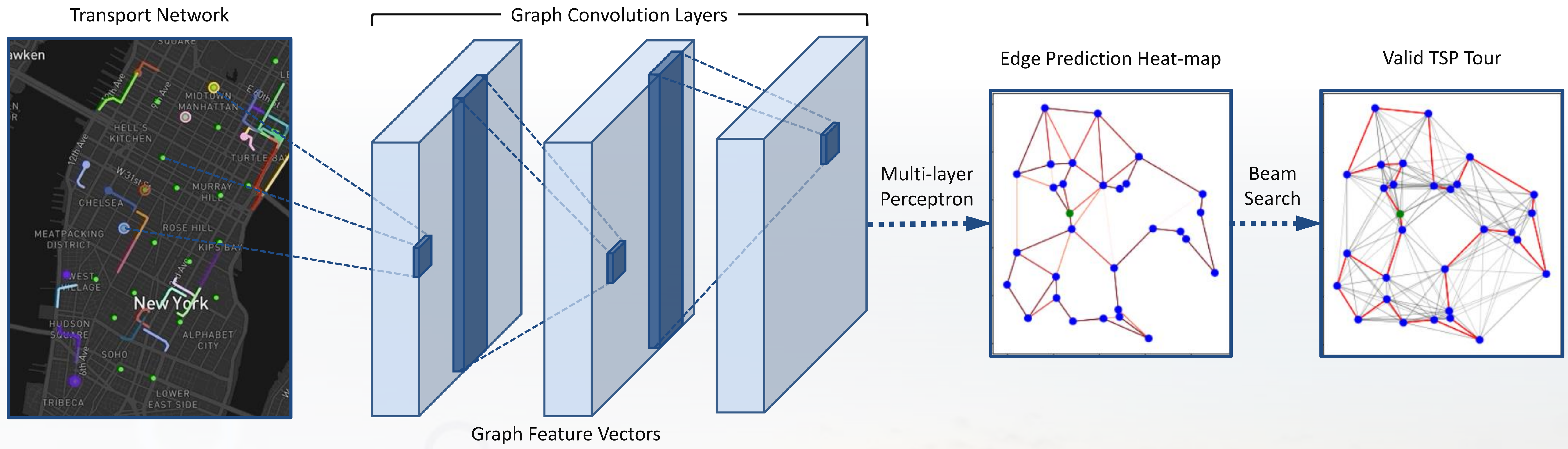
Motivations

- Routing a fleet of self-driving vehicles around a transport network: **many cars, many stops, dynamic fleet size.**
- Exact/optimal algorithms: slow, not parallelized and **intractable for large scales** (NP-hard problem).
- Heuristic algorithms: **hand-crafted**, might not be optimal.
- Solution: Can we use Artificial Intelligence to ***automatically learn*** better heuristics which are **highly parallelizable** (using GPUs) and **faster** (polynomial time complexity)?

The Travelling Salesman Problem

"Given a list of cities and the distances between each pair of cities, what is the shortest possible route that visits each city and returns to the origin city?"

- Most famous NP-hard problem today (introduced in 1832).
- Backbone of modern industries such as transportation, supply chain, genetics, and scheduling.
- **Vehicle routing == TSP on a transport network graph.**



Graph ConvNet Model

Procedure

1. Convert transport network into **2D Euclidean graph** where each stop is a node and each connecting road is an edge.
2. The **Graph Convolutional Neural Network** extracts compositional feature vectors for all nodes and edges in the graph via multiple layers of graph convolution.
3. The **Multi-layer Perceptron** uses the final edge feature vectors to compute an edge adjacency matrix of probabilities for each edge occurring on the TSP tour.
4. The adjacency matrix or *Heat-map* is converted to a valid TSP tour using **Beam Search** (*i.e.* limited-width BFS).

Advantages

- **Time complexity**: $O(n^2)$ where n is the number of nodes.
- **Batched computation**: Graph ConvNet, MLP and Beam Search are all parallelizable on GPUs.
- **No hand-crafting**: learning from solved TSP instances.

State-of-the-Art Results

- State-of-the-art performance among deep learning approaches:

Method & Description	Avg. Predicted Tour Length		
	20 Nodes	50 Nodes	100 Nodes
Concorde Solver Best exact solver, uses handcrafted heuristics	3.84	5.70	7.76
Graph Attention Network Previous best deep learning solver, uses RL	3.84	5.73	7.94
Graph ConvNet Our solver, uses SL and Beam Search	3.84	5.70	7.87

- Cost saving for a fleet of 100 vehicles, needing 1,000,000 computations of TSP for routing every day:

(Taking average rate of GPU server from Google Cloud to be US$ 0.65 per hour.)

Method	Avg. Solution Time	Total Cost
Concorde Solver	250 ms	US$ 45.15
Graph Attention Network	360 ms	US$ 65.00
Graph ConvNet	60 ms	US$ 10.84