香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY
THE DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
計算機科學及工程學系

Spatiotemporal Fuel Consumption Forecasting

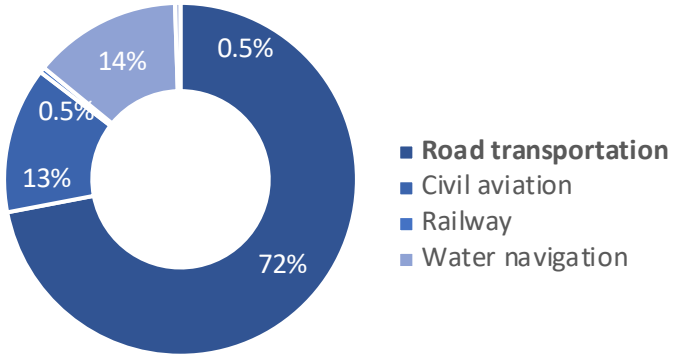
Supervised by: Professor Pan Hui

YAP Zhi Yun

Introduction to Fuel Consumption Forecasting

How can we curb transportation contamination with Intelligent Transport System (ITS)?

- Energy consumption of private cars in China to reach 117.38M tons of coals by 2030
- Road transportation forms 72% of CO2 emissions
- Improvement in driving behavior can improve up to 25% of the fuel economy



- Route recommendation algorithm focus on optimizing travelling time
- Users do not know about their carbon footprint during travelling

Source: European Environmental Agency 2016: https://www.eea.europa.eu/data-and-maps/indicators
IAEA 2020: https://inis.iaea.org/collection/NCLCollectionStore/_Public/36/045/36045273.pdf

Introduction to Fuel Consumption Forecasting

Over the years, eco-driving emerges as an effective strategy to reduce fuel cost

Recommended driving behavior

- ***Eco-driving*** emphasizes on sustainable driving behaviors to improve fuel economy
  - Choose a shorter travel path
  - Maintain a steady driving speed
  - Reduce engine idling time in traffic congestion

Algorithm design

- As the first step to recommend fuel-efficient route, the system needs to precisely predict regional fuel consumption level based on eco-driving factors:
  - Distance factor
  - Driving speed factor
  - Human mobility factor

Introduction to Fuel Consumption Forecasting

Formulating the fuel consumption forecasting problem:

- We propose fuel consumption forecasting as a new spatiotemporal prediction task in urban computing

- Given the static urban road network information, historical fuel consumption data as $X^{(t-T+1)}$, the model is to learn the function $f(\cdot)$ to predict regional fuel consumption level, $X^{(t+1)}$ for multiple future timesteps

$$\left[X^{(t-T+1)}, \dots, X^{(t)}\right] \xrightarrow{f(\cdot)} \left[X^{(t+1)}, X^{(t+2)}, \dots\right]$$

  - Incorporate multiple semantic relationship (distance, driving speed, and human mobility factor) of the road network in the model

Spatiotemporal Fuel Consumption Forecasting

Project scope and objective:

Position	Temporal Graph Neural Network for Spatiotemporal Fuel Consumption Forecasting		
Area of Focus	Dataset preparation	Research on SoTA models	DL framework design
Objectives	1 Develop **benchmark dataset** for fuel consumption forecasting task	2 Implement existing methodologies as **baselines** for comparison	3 Propose novel spatiotemporal prediction framework
Outcomes	Outperform state-of-the-art spatiotemporal models on two real world datasets		

Section 2: Design of SZ-FC and ST-MGAN

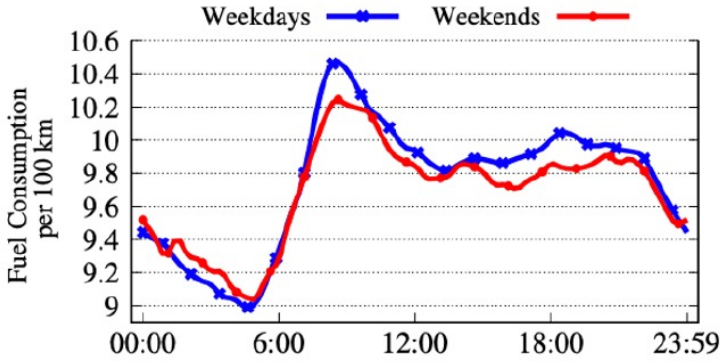
YAP Zhi Yun

Design of SZ-FC Dataset

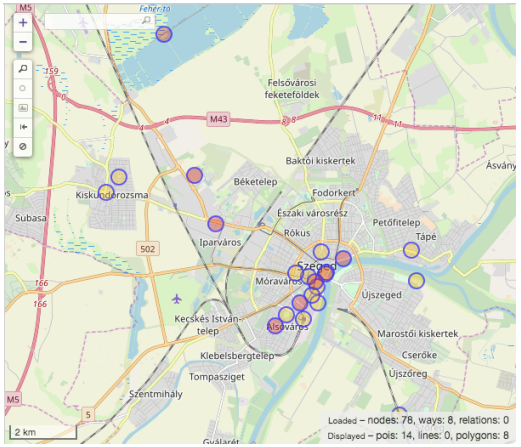
We follow the ETL process to prepare the Shenzhen Fuel Consumption (SZ-FC) dataset

Large Scale Private Car dataset

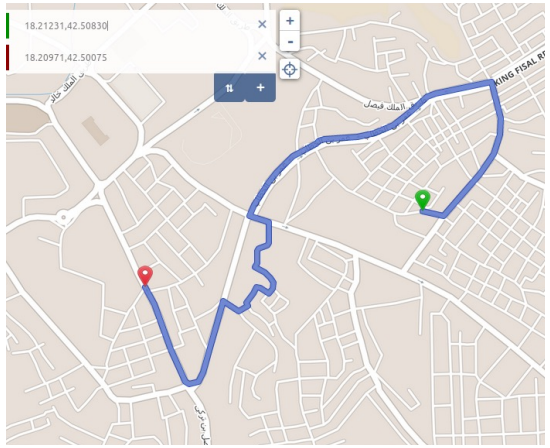
- Collected from 24th June to 31st July 2016
- 18M GPS trajectory data from 3,364 private cars
- Include highway, travel milage, road connectivity, and trip information



Open Street Map



Overpass API



OSRM API

Design of SZ-FC Dataset

We follow the ETL process to prepare the Shenzhen Fuel Consumption (SZ-FC) dataset

OpenStreetMap
The Free Wiki World Map

Large Scale Private Car dataset

Extract

- Extract surrounding POI data
- Extract pairwise road distance
- Extract trajectory data

Transform

- Remove outliers
- Remove erroneous data
- Infer fuel efficiency data*
- Infer missing values
- Extract top 500 active nodes

**one-hop neighborhood inference policy*
**using adjusted fuel efficiency measure*

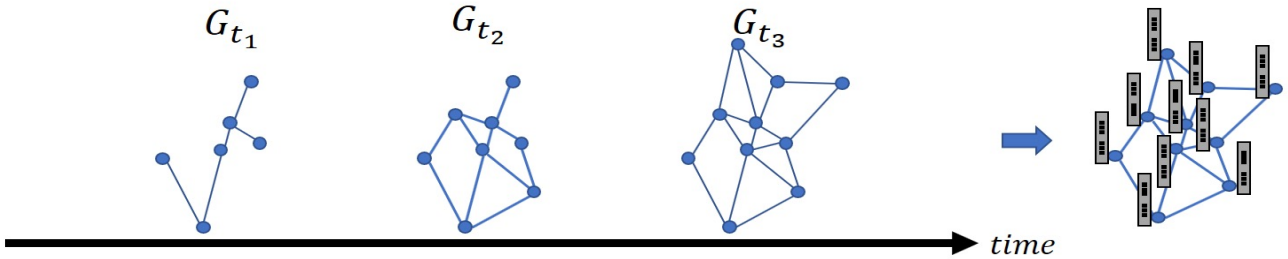
Load

- Load data to *Tensorflow* for model training and evaluation

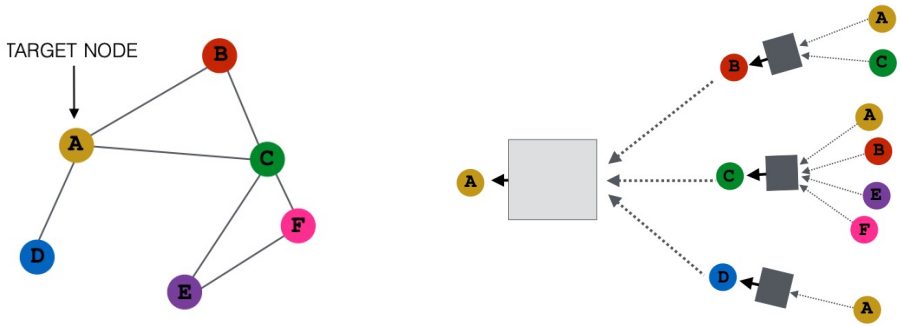
Shenzhen Fuel Consumption (SZ-FC) dataset	• aggregated using 15-min interval • 500 road nodes • 3,576 timesteps

Introduction to Temporal Graph Neural Network

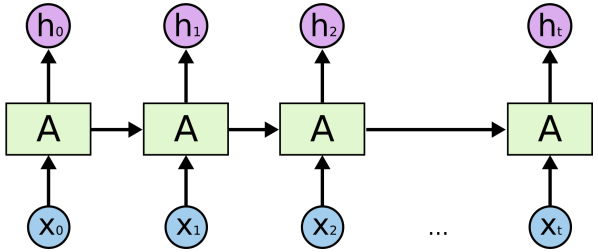
Temporal Graph Neural Network (T-GNN) combines sequential model and GNN to make prediction



We represent each road as a node, and spatial connectivity as vertices in the graph. GNN aggregates and propagates node representations using ***neighborhood aggregation scheme***



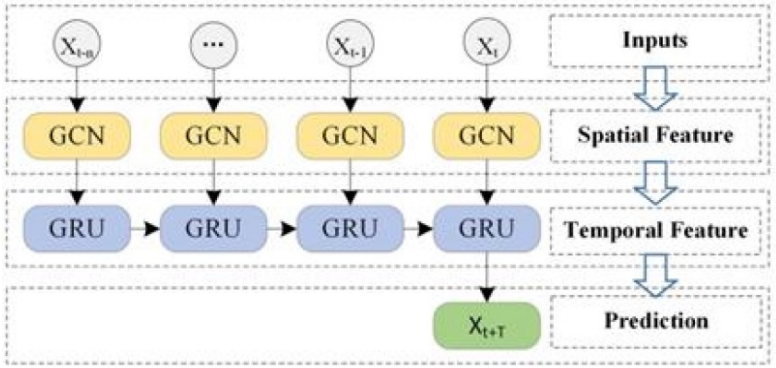
Sequential model like RNN encodes the **temporal dependency** of graph signals

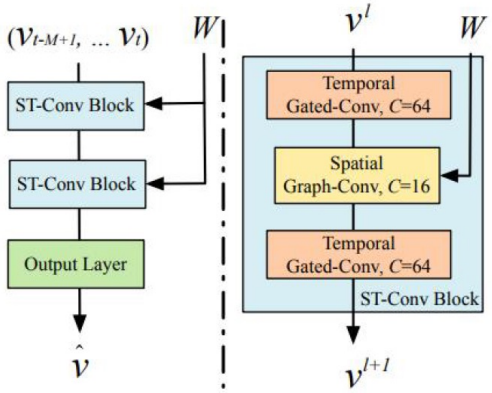


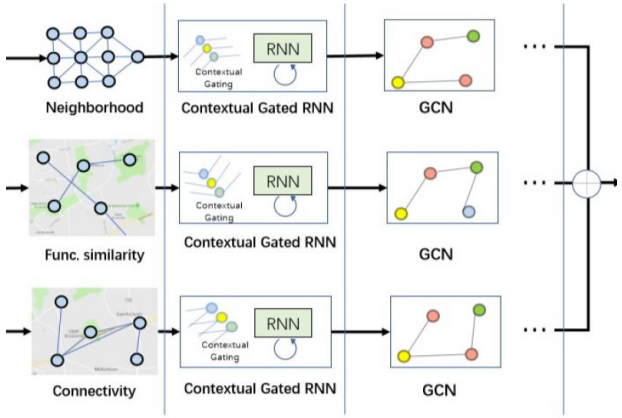
Related Work on Spatiotemporal Forecasting

We reviewed variants of Graph Neural Network for learning spatial dependency

TGCN, STGCN, and STMGCN apply ***Graph Convolution Network***







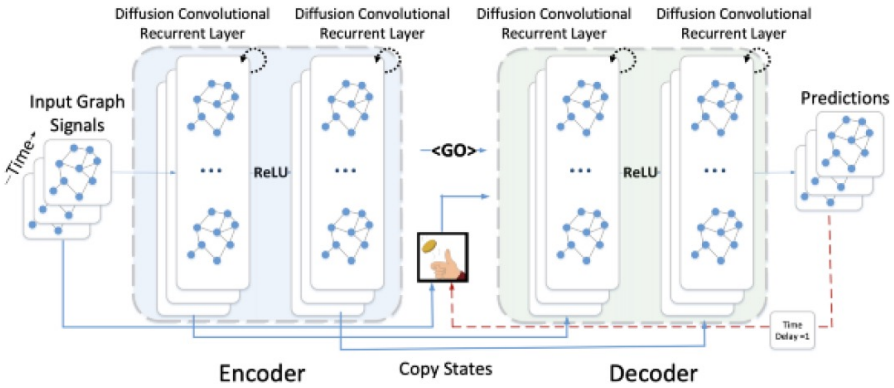
Use of mean aggregator

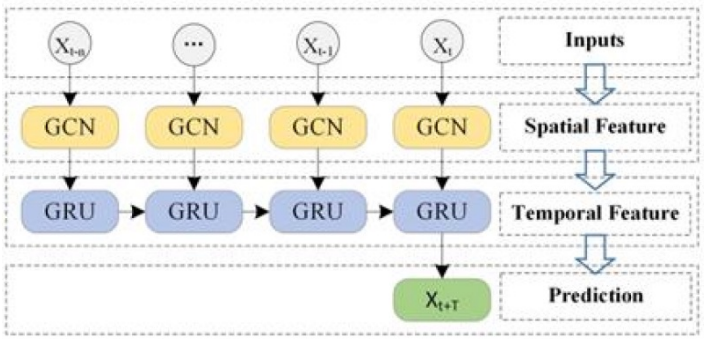
- Assume equal strength of connectivity in the network graph and neglect the representative power of edge features
- Lead to over-smoothing problem when the number of layers increase

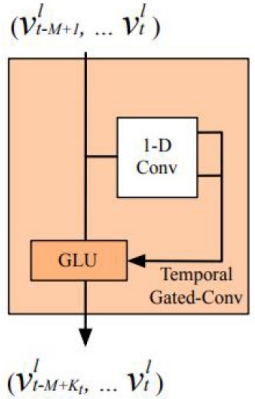
Related Work on Spatiotemporal Forecasting

We reviewed variants of Sequential Model in T-GNN for learning temporal dependency

DCRNN, TGCN, STGCN, and STMGCN extend the power of RNN







RNN-based structure

- Generally, more costly than convolution-based algorithm due to sequence-to-sequence learning nature
- Can suffer from gradient vanishing problem in long-term dependency

Proposed Framework

For timely prediction of regional fuel consumption data, our proposed T-GNN model should be able to:

1. How to alleviate, if not avoid **over-smoothing problems** in the conventional graph convolution network approach?

2. How to incorporate **polymorphic spatial correlations** (distance, driving speed, and human mobility factor) of a road network for fuel consumption forecasting?

3. How to learn **long-term dependency** better and with higher computational efficiency while making multi-step predictions

Design of ST-MGAN

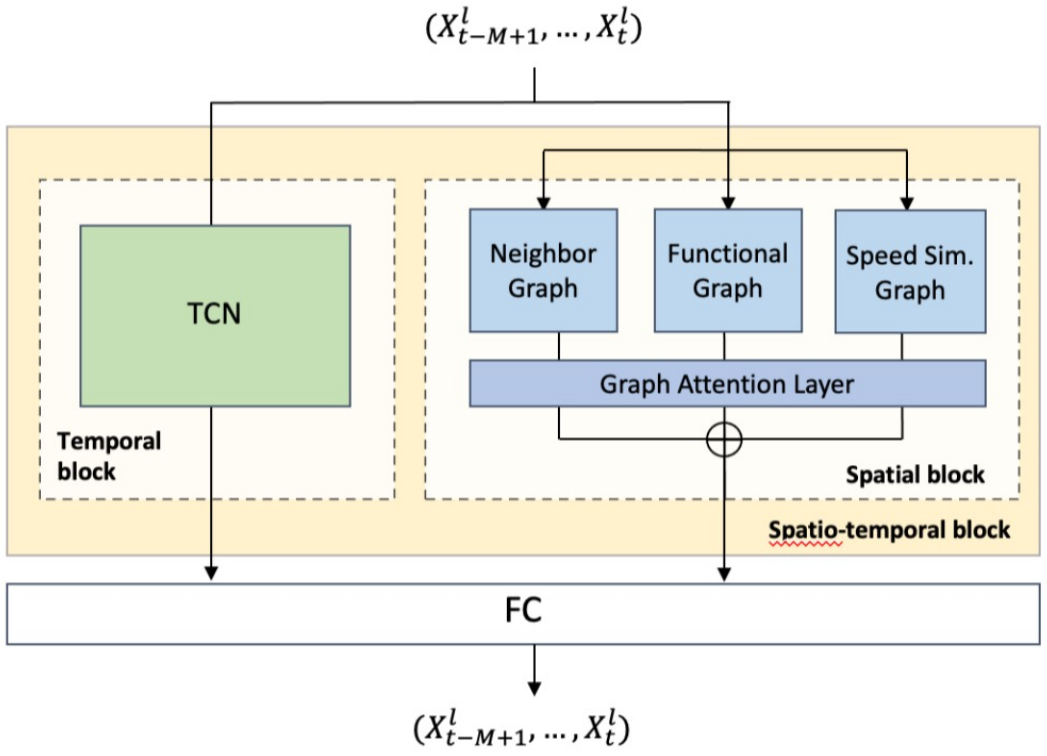
How to incorporate polymorphic relationships and long-term dependency?

- For learning spatial dependency:
  - Introduce **graph attention mechanism** that leverages self-masked attention layer
  - Approach of constructing **multi-graphs** is more scalable than multi-relational graph in CompGCN (Shikhar, 2020)

- For learning temporal dependency:
  - Convolution-based **temporal convolutional network** trains much faster than recurrent neural network-based sequential model

Design of ST-MGAN

Spatiotemporal Multi-graph Attention Network (ST-MGAN)



- Learns spatial and temporal dependency in parallel in temporal and temporal block
- Encodes polymorphic relationship in three pre-computed graphs (***multi-graph strategy***)
- Combines ***graph attention*** representation using multilayer perceptron (MLP)
- Concatenates spatial and temporal embedding from ***temporal convolutional network*** for prediction

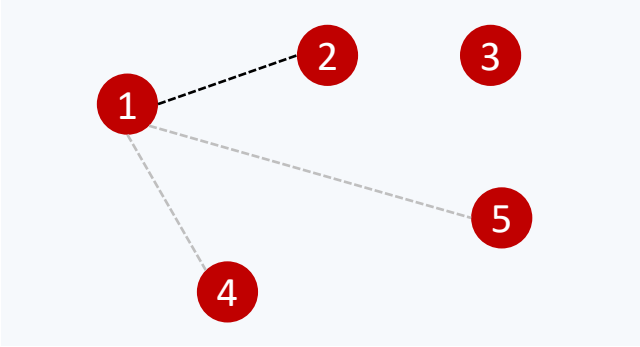
Design of ST-MGAN

Recall that the model is required to model distance, driving speed, and human mobility factors

Spatial correlation

Region 1 and 2 are **geographically connected** on the map, and would inhibit similar fuel consumption trend

Graph representation



Node 1 and 2 are connected by **spatial proximity**

Neighborhood Graph

Thresholded Gaussian kernel captures inverse relationship of distance and spatial connectivity

$$A_{N,i,j} = \begin{cases} exp\left(-\frac{d_{ij}^2}{\sigma^2}\right), & if\ i \neq j \\ 0 & otherwise \end{cases}$$

where $A_{N,i,j}$ is the edge weight, d_{ij} is the pairwise great circle distance

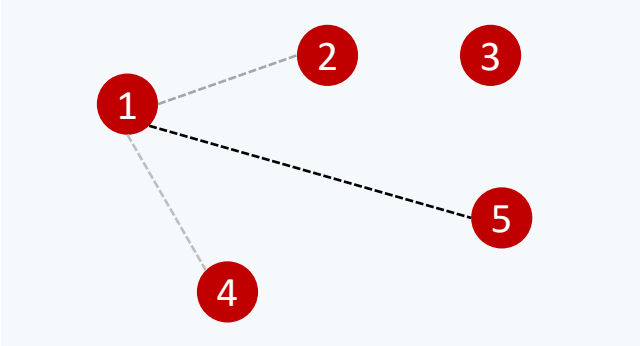
Design of ST-MGAN

Recall that the model is required to model distance, driving speed, and human mobility factors

Spatial correlation

Region 1 and 5 share the same social functions, **human mobility**, and traffic pattern throughout the day

Graph representation



Node 1 and 5 are connected by **functional similarity**, which is characterized by its surrounding point-of-interest

Func. Similarity Graph

Jaccard distance captures the POI similarity of two road nodes

$$A_{F,i,j} = sim(P_{vi}, P_{vj}) \in [0, \infty)$$

$$J(v_i, v_j) = \frac{|v_i \cap v_j|}{|v_i \cup v_j|}$$

where $A_{F,i,j}$ is the edge weight, v_i is the POI vector of node i

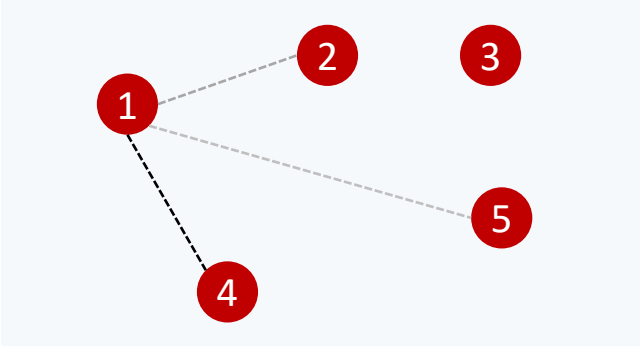
Design of ST-MGAN

Recall that the model is required to model distance, driving speed, and human mobility factors

Spatial correlation

Region 1 and 4 have similar driving speed and fuel consumption pattern

Graph representation



Node 1 and 4 are connected by **speed similarity**, which captures the traffic congestion trend of the road

Speed Similarity Graph

Dynamic time warping captures the similarity of two temporal speed sequences

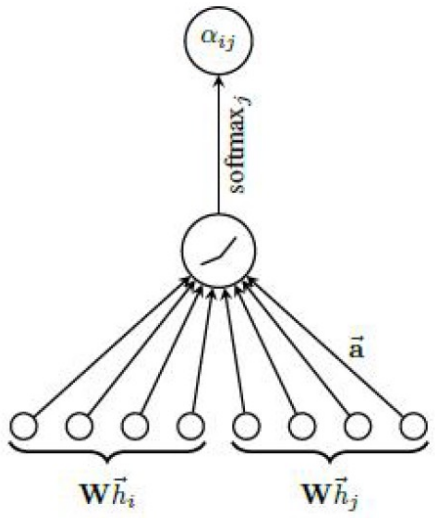
$$A_{S,i,j} = DTW(S_{vi}, S_{vj}) \in [0, \infty)$$

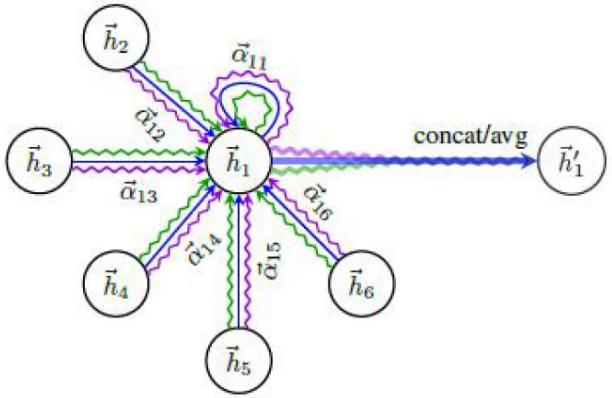
where $A_{S,i,j}$ is the edge weight, S_{vi} is the speed sequence of node i

Design of ST-MGAN

With the three precomputed semantic graphs, graph attention mechanism of message passing within the network:

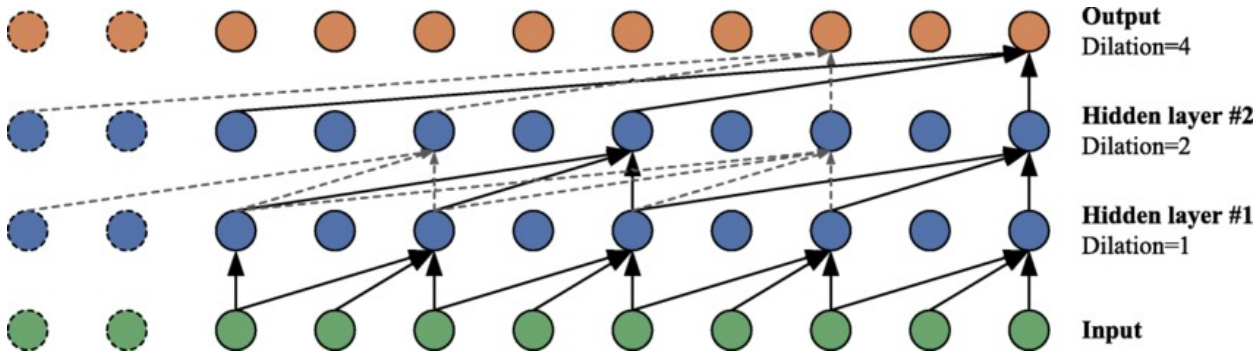
- Unlike GCN, graph attention calculates and assigns nodes in one-hop neighborhood with **attention coefficient**
- During message passing, features of the neighboring nodes are scaled using the attention coefficient
- This allows the model to learn the strength of connectivity between neighboring nodes through trainable parameters





Design of ST-MGAN

ST-MGAN captures the temporal dependencies using temporal convolutional network (TCN)



- Harness power of TCN with high parallelism and stable gradients
- ***Dilated convolution*** to skip certain nodes during convolution operation, which increases the receptive field while maintaining high computational efficiency
- ***Causal convolution*** to mask future timesteps and prevent leakage of information

Section 3: Experiments and Results

YAP Zhi Yun

Experimental Setup

We use a sliding window approach to generate training, validation, and testing set

We evaluate the performance of the proposed ST-MGAN model against baselines on two real-world dataset

- SZ-FC dataset
- PeMS-M dataset

Generate training sequence with 1-hour input, and prediction interval of 15-, 30-, and 45-minute intervals

Shenzhen Fuel Consumption (SZ-FC) dataset

- 1-month trajectory data in Shenzhen
- aggregated using 15-min interval
- 500 road nodes
- 3,576 timesteps
- Train:val:test ratio of 60:15:25

Catrans Performance Measurement System (PeMS-M) dataset

- Collected from May to Oct 2012 weekdays
- 6-month traffic speed data in California
- 228 trac sensors
- 12,672 timesteps
- Train:val:test ratio of 60:20:20

Experimental Setup

Experiments are conducted to evaluate performance of ST-MGAN against 5 statistical and DL baselines:

Baselines
• Historical average (HA) • Autoregressive integrated moving average (ARIMA) • Long short-term memory (LSTM) • Spatiotemporal Graph Convolution Network (STGCN) • Spatiotemporal Multi-graph Convolution Network (STMGCN)

Evaluation Metrics
• Mean absolute error (MAE) • Root-mean-squared error (RMSE) • Mean absolute percentage error (MAPE)

Experiment on SZ-FC and PeMS-M Dataset

We conducted experiments on ST-MGAN and baselines using SZ-FC dataset:

Model	SZ-FC (15/30/45 min)		
	MAE	RMSE	MAPE (%)
HA	2.74/3.12/3.09	4.71/4.81/4.84	7.02/7.27/7.33
ARIMA	4.30/4.22/4.68/	6.30/6.47/6.59	7.33/7.41/7.46
LSTM	2.94/3.58/4.95	5.05/5.23/5.43	6.21/6.12/6.44
STGCN	2.25/3.03/3.57	3.04/4.70/4.77	5.36/7.33/8.69
STMGCN	**1.55**/2.47/3.43	2.75/**3.43**/3.99	**3.75**/**4.67**/5.84
ST-MGAN	1.58/**2.46**/**3.26**	**2.24**/3.68/**3.86**	4.70/5.21/**5.23**

- ST-MGAN outperforms the baselines in over 50% prediction intervals in SZ-FC dataset

Model	PeMS-M (15/30/45 min)		
	MAE	RMSE	MAPE (%)
HA	4.01	10.61	7.20
ARIMA	5.55/5.86/6.27	9.00/9.13/9.38	12.92/13.94/15.20
LSTM	3.57/3.94/4.16	6.20/7.03/7.51	8.60/9.55/10.10
STGCN	**2.26**/3.93/3.79	**4.07**/5.77/7.03	**5.24**/**7.39**/**9.12**
STMGCN	2.70/3.97/4.10	4.23/5.96/7.24	6.21/8.58/9.53
ST-MGAN	2.60/**3.87**/3.95	4.22/**5.76**/**7.01**	5.61/7.45/9.18

- ST-MGAN outperforms in over 50% prediction intervals in SZ-FC dataset
- STGCN is still the best performing model for PeMS-M dataset.

Section 4: Conclusion

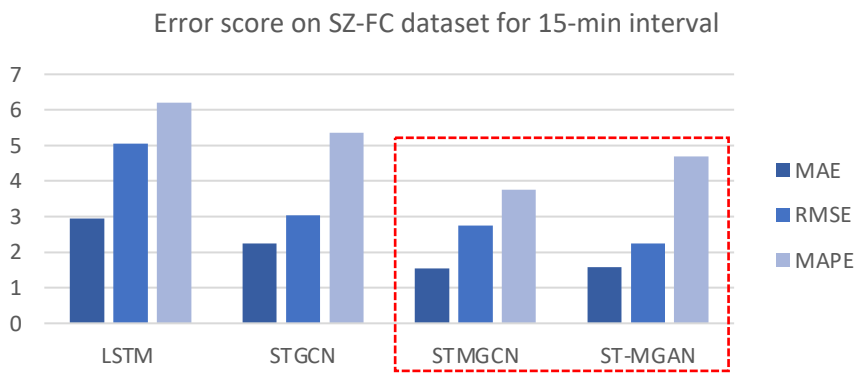
Analysis on ST-MGAN methodology

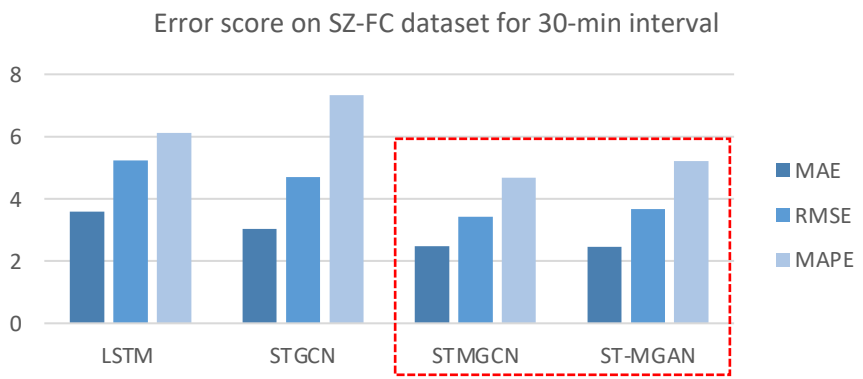
We want to answer the following question:

1. How effective is the *multi-graph strategy* in modelling complex spatial dependency?
2. How effective is the *graph attention mechanism* in modelling complex spatial dependency?
3. How effective is the *temporal convolutional network* in capturing long-term dependency?
4. How is the *computational efficiency* of ST-MGAN?

Analysis on ST-MGAN methodology

How effective is the multi-graph strategy in modelling complex spatial dependency?





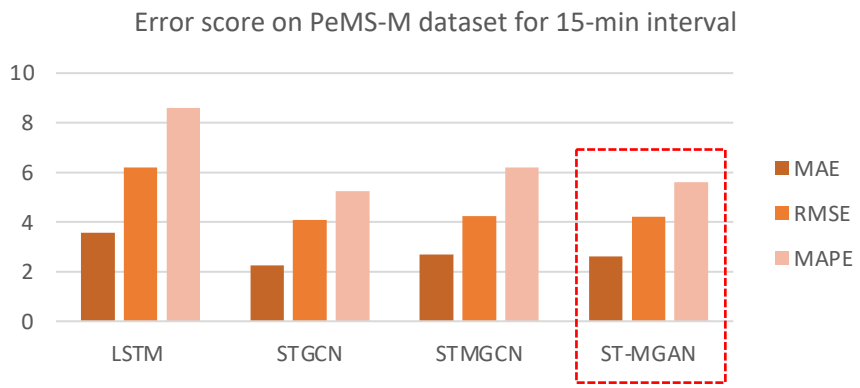
To evaluate the predictive power of ***multi-graph strategy***,

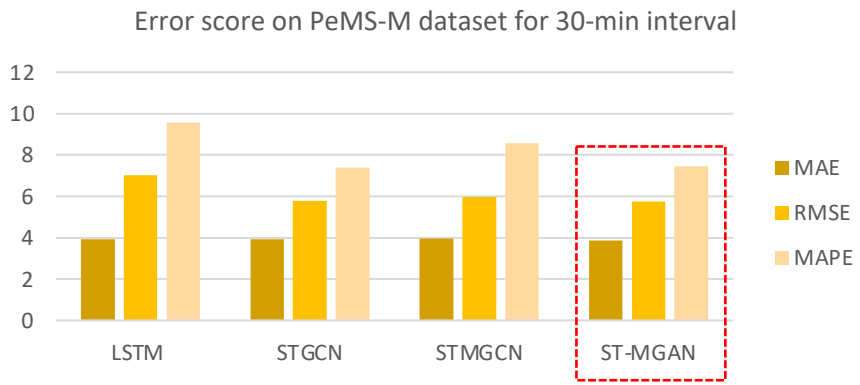
- We investigate performance of multi-graph frameworks (STMGCN, ST-MGAN) on SZ-FC dataset
- STMGCN and ST-MGAN outperform LSTM and STGCN by approximately 2% error score in 15-, 30-, and 45-minute prediction intervals

Multi-graph strategy improves spatiotemporal framework in modelling polymorphic relationships

Analysis on ST-MGAN methodology

How effective is the graph attention mechanism in modelling complex spatial dependency?





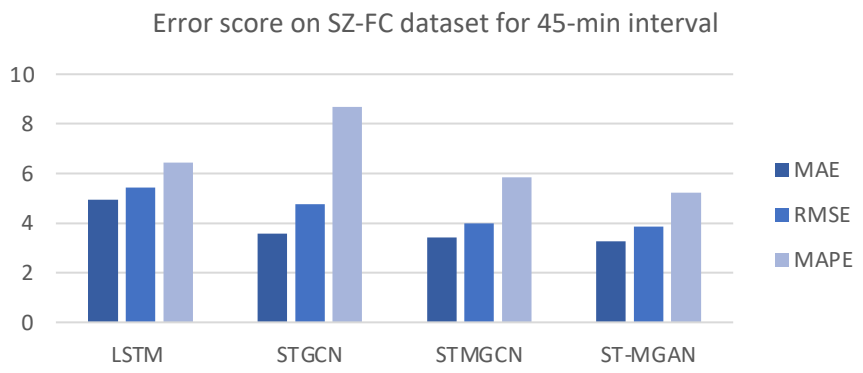
To evaluate the predictive power of ***graph attention mechanism***,

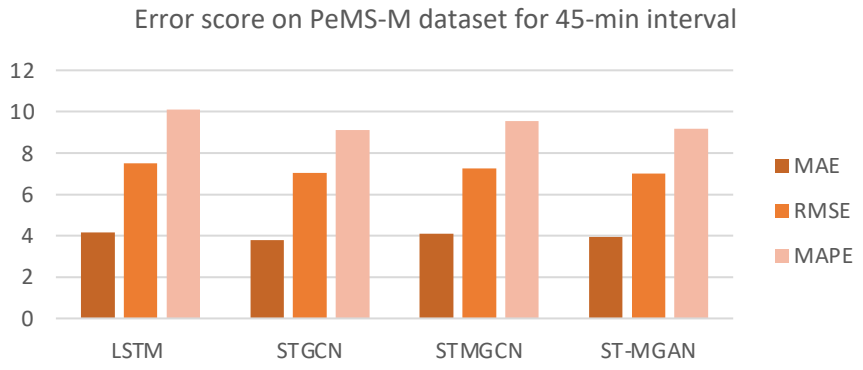
- In PeMS-M dataset, multi-graph strategy is not employed in ST-MGAN and STMGCN
- ST-MGAN outperforms graph-based spatiotemporal models without attention mechanism in over 40% prediction intervals

Employing ***attention mechanism*** in single graph slightly improves spatial dependency modelling

Analysis on ST-MGAN methodology

How effective is the temporal convolutional network (TCN) in capturing long-term dependency?





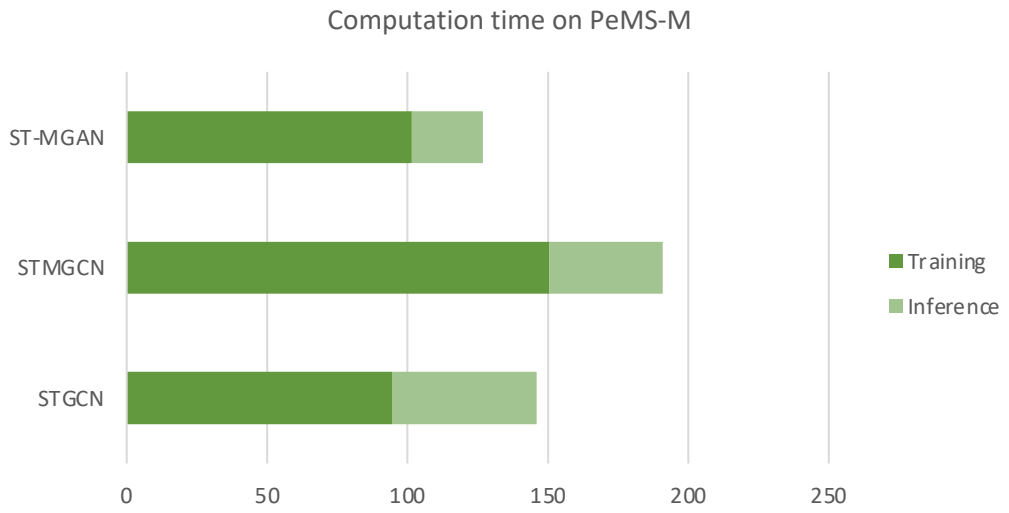
To evaluate the predictive power of ***TCN***,

- We investigate performance of ST-MGAN on SZ-FC and PeMS-M datasets for prediction interval of 45-min
- ST-MGAN outperform all models by approximately 1% in percentage error

TCN enables ST-MGAN to better capture long-term dependency in fuel consumption and traffic speed trend

Computation Time Analysis of ST-MGAN

How is the computational efficiency of ST-MGAN?



We report the time taken (sec) for each model to train and infer a 9-step prediction on PeMS-M dataset

- Overall, ST-MGAN has the most stable computation efficiency
- With convolutional-based sequential model, ST-MGAN runs much faster in model inference stage
- Longer training time of ST-MGAN could be attributed to the computation of attention coefficient

Future Work

- Enrich SZ-FC dataset with more trajectory data on inactive nodes.
  - Data collection period of PeMS-M dataset as 6 months

- Review choice of multi-graphs to be constructed
  - Explore other measure that can better encode the spatial relationship
  - Explore other spatial relationship that is more representative to the fuel consumption forecasting task

- Train and evaluate ST-MGAN on other benchmark dataset of spatiotemporal forecasting task.

Conclusion

Review of project scope and objective:

Position	Temporal Graph Neural Network for Spatiotemporal Fuel Consumption Forecasting		
Area of Focus	Dataset preparation	Research on SoTA models	DL framework design
Objectives	1 Developed **benchmark SZ-FC dataset** for fuel consumption forecasting	2 Implemented 5 existing methodologies as **baselines** for comparison	3 Proposed and implemented ST-MGAN as a novel DL framework
Outcomes	(i) ST-MGAN outperforms baselines in over 50% prediction intervals in SZ-FC dataset. (ii) ST-MGAN demonstrates good computation efficiency, model generalization and transferability.		

Acknowledgement

Special thanks to:

- **Mr. Ahmad Alhilal,**
  - who provided advice on processing and transforming the Large-Scale Private Car dataset, helped on setting up the computing resources, and guided me to define the project scope.
- **Dr. Tristan Braud**,
  - who advised on building the data pipeline, setting up the experiments, and formulating the research problem.
- **Mr. Mingyang Zhang**,
  - who guided me on the T-GNN architecture design and provided feedback on proposed methodology.

Thank You!

Appendix

Title
Proposed vs. Previous Work
Multi-relational Graph in CompGCN
Dynamic Time Warping Algorithm
Graph Attention Score Computation
One-hop neighborhood inference scheme
ST-MGAN - Neighborhood graph
Data Processing – Speed
Data Processing - Sparsity
OSM Overpass API
POI tags and distribution

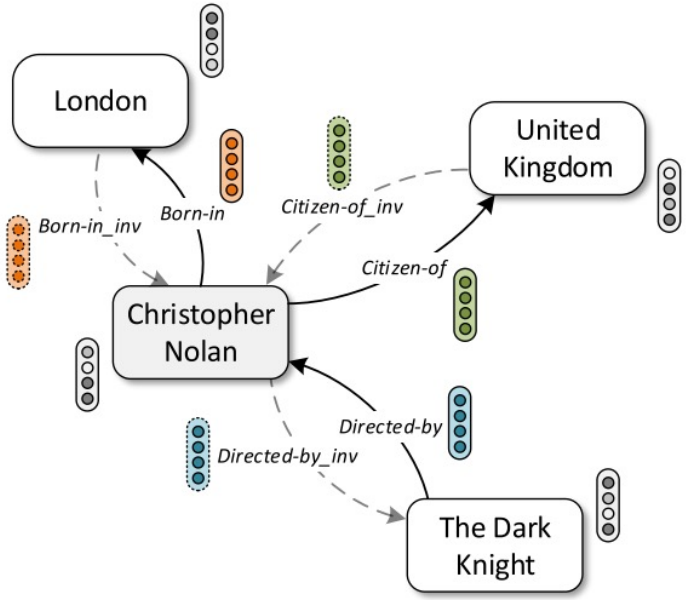
Appendix: Proposed vs. Previous work

Previous work	Proposed methodology
Graph convolution network used by most spatiotemporal frameworks - TGCN, STGCN, STMGCN	Introduce graph attention layer to TGNN framework - Alleviate potential over-smoothing problem with multi-graphs approach
RNN-based sequential model architecture - Long training and inference time - Potential gradient vanishing problem	Convolution-based sequential model architecture - Better parallelism; faster training speed - Stable gradient; no gradient vanishing problem - Flexible receptive field with adjustable dilated rate
Multi-relational graph to encode multiple relations - Does not scale with number of relations	Harness power of multi-graph strategy previously proposed in STMGN - Scale with number of relations

Appendix: Multi-relational Graph in CompGCN

- Multi-relational graph is first introduced in knowledge graph application to encodes multiple directed relationships within the network

- Number of trainable parameters scale with the number of relations
  - Leading to a scability issue when applying to the fuel consumption forecasting task



Relational Graph with Embeddings

Appendix: Dynamic Time Warping algorithm

Algorithm 2 Dynamic Time Warping algorithm

Input: series $u = \{u_1, u_2, ..., u_{T_u}\}$ and series $v = \{v_1, v_2, ..., v_{T_v}\}$

Output: DTW-distance $D(u, v) = k(w) \cdot g(T_u, T_v)$

Initialize:

$g(0, 0) = 0$

$g(1, 1) = d(u_1, v_1) \cdot w_D$

$g(i, 0) = \infty \quad for \ 1 \ \leq \ i \ \leq \ T_u$

$g(0, j) = \infty \quad for \ 1 \ \leq \ j \ \leq \ T_v$

for $1 \leq i \leq T_u and 1 \leq j \leq T_v$ **do**

$$g(i, j) = min \begin{cases} \mathrm{g(i,j\text{-}1)} \ + \ \mathrm{d}(\mathrm{u}_i, v_j) \cdot w_V \\ \mathrm{g(i\text{-}1,j\text{-}1)} \ + \ \mathrm{d}(\mathrm{u}_i, y_j) \cdot w_D \\ \mathrm{g(i\text{-}1,j)} \ + \ \mathrm{d}(\mathrm{u}_i, v_j) \cdot w_H \end{cases}$$

end for

Appendix: Graph Attention Score Computation

Calculating attention coefficient

$$z_i^{(l)} = \mathbf{W}^{(l)} h_i^{(l)}$$

$$e_{ij}^{(l)} = LeakyReLU(\overrightarrow{a}^{(l)^T}(z_i^{(l)} || z_j^{(l)}))$$

$$\alpha_{ij}^{(l)} = softmax_j(e_{ij}^{(l)}) = \frac{exp(e_{i,j}^{(l)})}{\sum exp_{k \in N(i)}(e_{ik}^{(l)})}$$

$$h_i^{(l)} = \sigma(\sum_{j \in N(i)} \alpha_{ij}^{(l)} z_j^{(l)})$$

Appendix: One-hop neighborhood inference scheme

Fill missing values of fuel consumption data using context-aware information

Algorithm 1 One-hop Neighborhood Inference

Input: $[x_{t,i},\ x_{t+1,i+1}, ...]$
Output: $[y_{t,i},\ y_{t+1,i+1}, ...]$

```
for all x_{t,i} do
    connectedNodeList = n_i
    connectedNodeList += [node(s) connected to n_i]
    while period t < 0 do
        for all n_j do
            y_{t,i} += x_{t,j}
        end for
        period -= 1
    end while
    y_{t,i} = y_{t,i} / length(connectedNodeList)
end for
```

Appendix: ST-MGAN - Neighborhood graph

Inverse relationship between pairwise road distance and thresholded Gaussian kernel measure

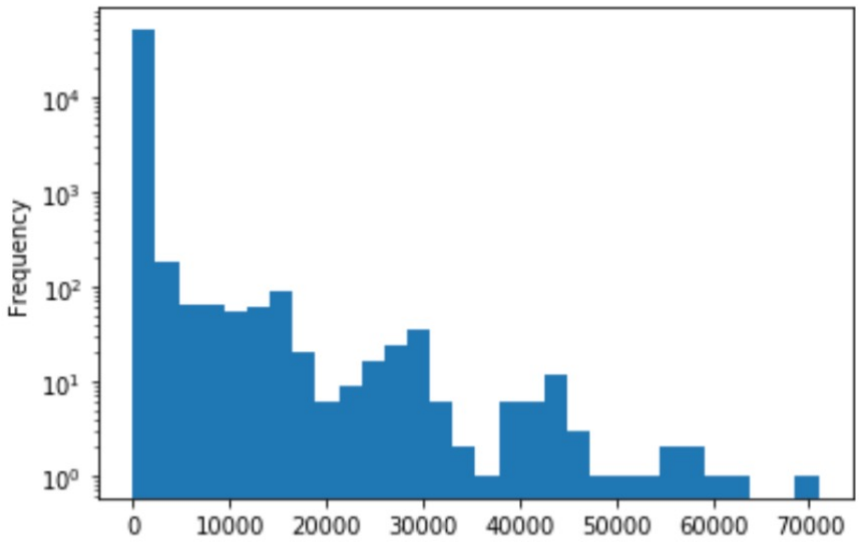


Figure B.10: Pairwise road node distance in SZ-FC dataset

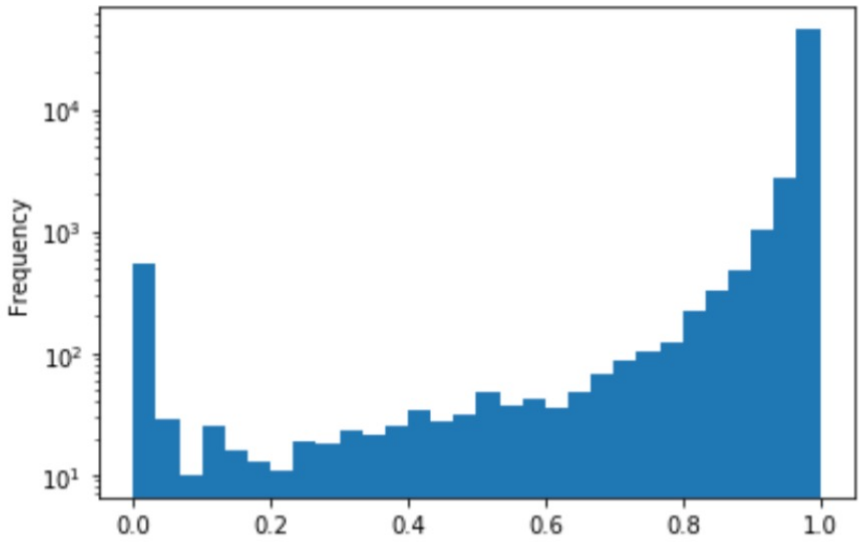


Figure B.11: Thresholded Gaussian kernel measure captures the inverse relationship between spatial proximity and pairwise road node distance

Appendix: Data Processing - Speed

Remove erroneous data in speed distribution

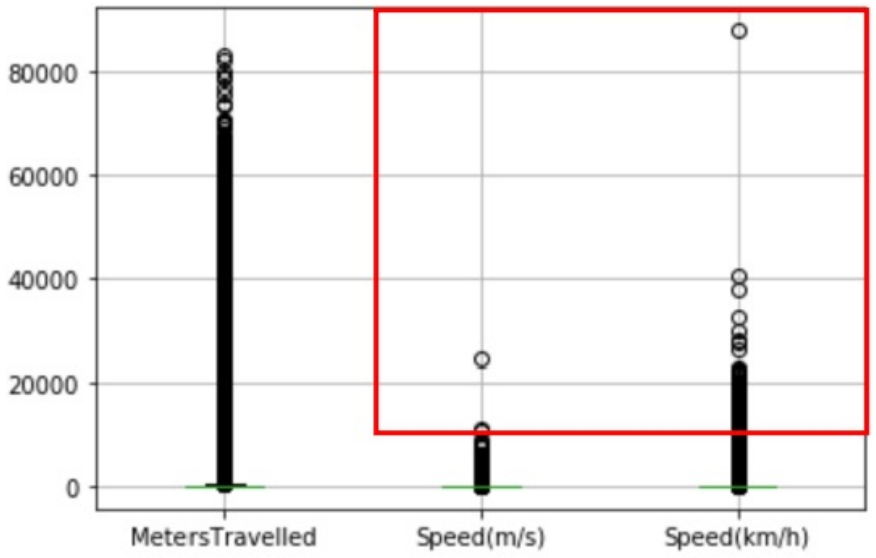


Figure B.9: Trajectory data with instantaneous speed of over 200km/h is considered as erroneous data to be filtered out

Appendix: Data Sparsity

Ground truth fuel consumption data suffers from severe data sparsity issue

Sparsity of Fuel Consumption Matrix

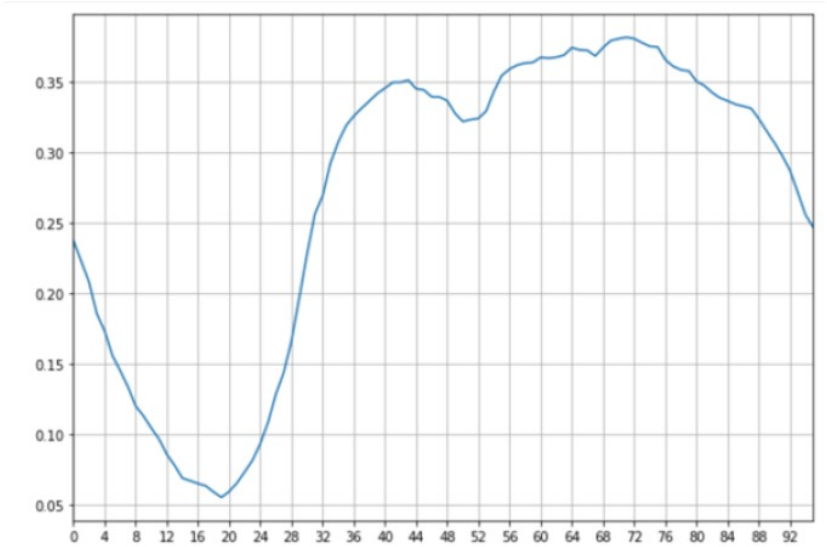


Figure B.6: Derived fuel consumption data only covers 15% of the nodes amongst 37,174 nodes in the road network

Appendix: OSM Overpass API

Extract surrounding POI data points from OSM Overpass API

```python
# DIST = 200
url = "http://overpass-api.de/api/interpreter"
def get_nearby_poi(edge_id, lat1, lon1, lat2, lon2):
  if lat1 > lat2:
    lat1, lat2, lon1, lon2 = lat2, lat1, lon2, lon1
  try:
      query = f"""
                  [out:json][timeout:100];
                  (node["tourism"]({DIST}, {row.StartLat}, {row.StartLng}, {row.EndLat}, {row.EndLng}););
                  (node["leisure"]({DIST}, {row.StartLat}, {row.StartLng}, {row.EndLat}, {row.EndLng}););
                  (node["shop"]({DIST}, {row.StartLat}, {row.StartLng}, {row.EndLat}, {row.EndLng}););
                  (node["amenity"]({DIST}, {row.StartLat}, {row.StartLng}, {row.EndLat}, {row.EndLng}););
                  out center;
                  """
      response = requests.get(url, params={'data': query})
      assert(response.status_code==200)
      data = response.json()
  except Exception as e:
      print('Error', edge_id, e.args, response.status_code)
```


```json
"features": [
  {
    "type": "Feature",
    "properties": {
      "@id": "node/277045518",
      "amenity": "fuel"
    },
    "geometry": {
      "type": "Point",
      "coordinates": [
        114.0431003,
        22.6009762
      ]
    },
    "id": "node/277045518"
  },
  {
    "type": "Feature",
    "properties": {
      "@id": "node/277501742",
      "amenity": "school",
      "created_by": "Potlatch 0.9c",
      "name": "Tsinghua Experimental School",
      "name:zh": "桃源居中澳实验学校"
    },
```

Appendix: POI tags and distribution

Distribution of POI tags from the Overpass API and Remapped POI categories

POI tag distribution from Overpass API

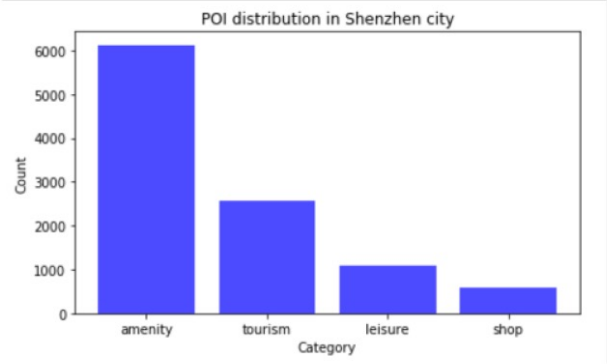


Figure B.3: Skewed distribution of 4 major POI tags provided by Overpass PI.

POI categories

Accommodation	Amenity	Art	Beauty
Clothing	Education	Electronic	Entertainment
Finance	F&B	Food	Healthcare
Household	Leisure	Outdoor	Shopping
Shop(other)	Transportation	Tourism	

Table B.1: 196 POI categories from OSM database are remapped to 19 new POI tags