Sep. 17, 2021
The 20th Behavior Modeling in Transportation Networks
Lecture series #2-3

LSTM & RNN for day-to-day panel data

On the use of ML (particularly NN) for representing temporal dependencies in transport studies

Hiroshima University
Makoto Chikaraishi

INTRODUCTION

Applications of deep learning method in transport studies

	Accidents	Congestion	DB	M–AC	TD	TTP	Dist	Flow	Speed	Occ	Total
CNN	36	23	38	4	20	10	0	43	101	3	278
CNN-GRU	0	0	2	0	0	0	0	9	18	0	29
CNN-LSTM	3	2	2	0	9	3	0	11	18	0	48
CNN-RNN	0	0	0	0	0	0	0	0	10	0	10
LSTM- GRU	0	0	0	0	0	0	0	0	1	0	1
DBN	2	2	5	0	0	44	0	50	10	1	114
DNN	16	1	5	2	10	2	2	38	37	0	113
GRU	0	8	0	0	3	2	0	2	33	0	48
LSTM	10	16	12	4	69	11	0	77	90	0	289
RNN	1	9	12	1	1	4	0	12	20	0	60
SAE	4	1	0	0	14	20	0	151	32	0	222
TM	59	27	46	19	90	35	1	324	218	1	820
SNN	7	14	7	1	42	6	0	142	62	1	282
Total	138	103	129	31	258	137	3	859	650	6	2314

Abbreviations for area of application: DB, driver behaviour; M–AC, mode and activity choice; TD, travel demand; TTP, travel time prediction; Dist., travel distance; Occ, occupancy

Varghese, V. Chikaraishi, M., Urata, J. (2020) Deep Learning in Transport Studies: A Meta-Analysis on the Prediction Accuracy, Journal of Big Data Analytics in Transportation, Vol. 2, 199–220.

Tension between theory-driven methods (classical choice models) and data-driven methods (machine learning)

- **Winner of the 2018 Eric Pas Best Dissertation Award**
  - Timothy Brathwaite
    - The Holy Trinity: Blending Statistics, Machine Learning and Discrete Choice with Applications to Strategic Bicycle Planning

- **ICMC2019 keynote**
  - Joan Walker
    - Choice modelling in an age of machine learning

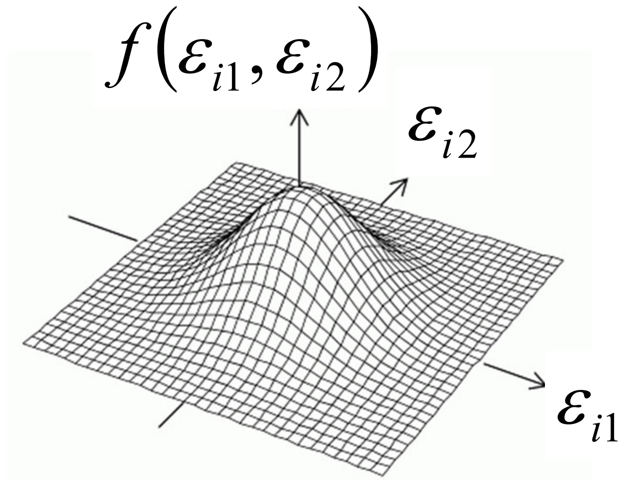
- **Honorable Mention of the 2019 Eric Pas Best Dissertation Award**
  - Shenhao Wang
    - Deep neural networks for choice analysis

And many others...

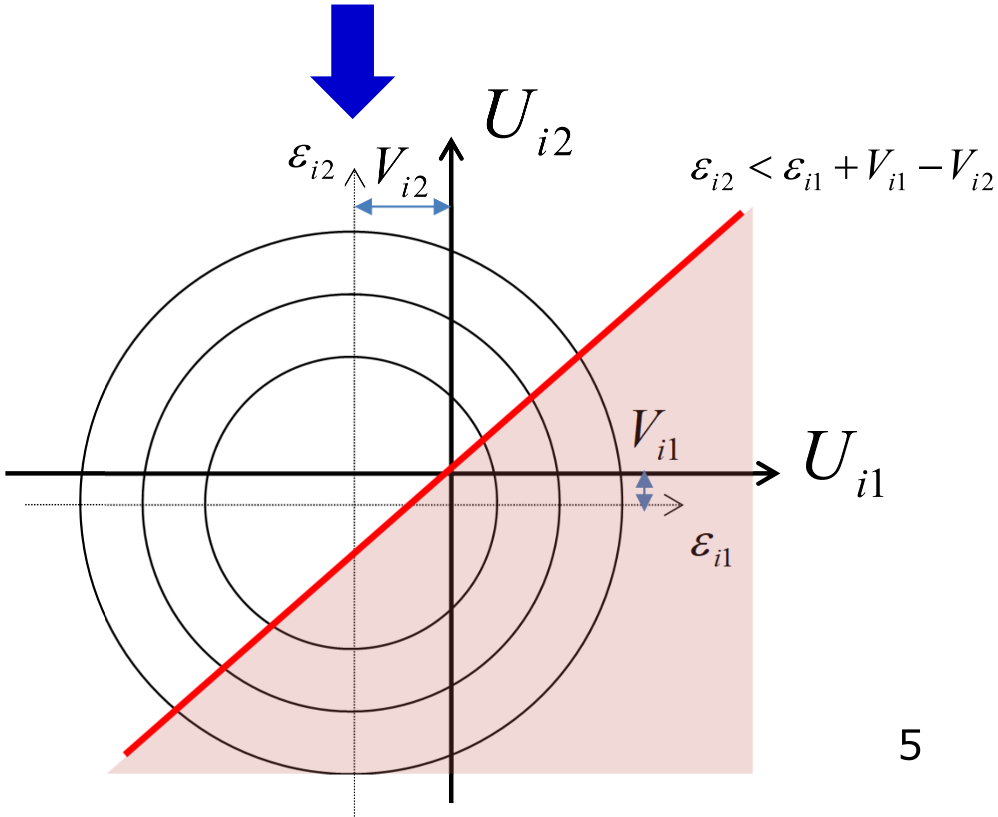
A conventional derivation of logit model

(for behavior modelers)

$$
\begin{aligned}
P_{i1} &= \Pr(U_{i1} > U_{i2}) \\
&= \Pr(V_{i1} + \varepsilon_{i1} > V_{i2} + \varepsilon_{i2}) \\
&= \Pr(\varepsilon_{i2} < \varepsilon_{i1} + V_{i1} - V_{i2}) \\
&= \int_{\varepsilon_{i1}=-\infty}^{\infty} \int_{\varepsilon_{i2}=-\infty}^{\varepsilon_{i1}+V_{i1}-V_{i2}} f(\varepsilon_{i1}, \varepsilon_{i2}) d\varepsilon_{i1} d\varepsilon_{i2}
\end{aligned}
$$



http://www.biwako.shiga-u.ac.jp/sensei/mnaka/ut/sozai/prob.html



A conventional derivation of logit model

(for behavior modelers)

Assuming ε_{i1} and ε_{i2} are independent,

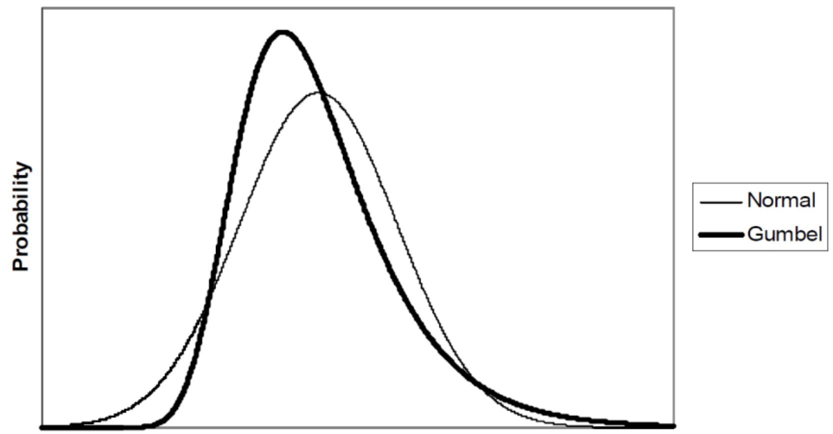


Figure. Probability Density Function for Gumbel and Normal Distributions

$$P_{i1} = \int_{\varepsilon_{i1}=-\infty}^{\infty} \int_{\varepsilon_{i2}=-\infty}^{\varepsilon_{i1}+V_{i1}-V_{i2}} f(\varepsilon_{i1}, \varepsilon_{i2}) d\varepsilon_{i1} d\varepsilon_{i2}$$

$$= \int_{\varepsilon_{i1}=-\infty}^{\infty} \left[\int_{\varepsilon_{i2}=-\infty}^{\varepsilon_{i1}+V_{i1}-V_{i2}} f(\varepsilon_{i2}) d\varepsilon_{i2} \right] f(\varepsilon_{i1}) d\varepsilon_{i1}$$

$$= \int_{\varepsilon_{i1}=-\infty}^{\infty} \left[\int_{\varepsilon_{i2}=-\infty}^{\varepsilon_{i1}+V_{i1}-V_{i2}} \underline{\exp(-\varepsilon_{i2}) \exp(-\exp(\varepsilon_{i2}))} d\varepsilon_{i2} \right] f(\varepsilon_{i1}) d\varepsilon_{i1}$$

↑ Assuming Gumbel distribution

$$= \int_{\varepsilon_{i1}=-\infty}^{\infty} [\exp(-\exp(\varepsilon_{i1} + V_{i1} - V_{i2}))] f(\varepsilon_{i1}) d\varepsilon_{i1}$$

$$= \int_{\varepsilon_{i1}=-\infty}^{\infty} [\exp(-\exp(\varepsilon_{i1} + V_{i1} - V_{i2}))] [\underline{\exp(-\varepsilon_{i1}) \exp(-\exp(\varepsilon_{i1}))}] d\varepsilon_{i1}$$

↑ Assuming Gumbel distribution

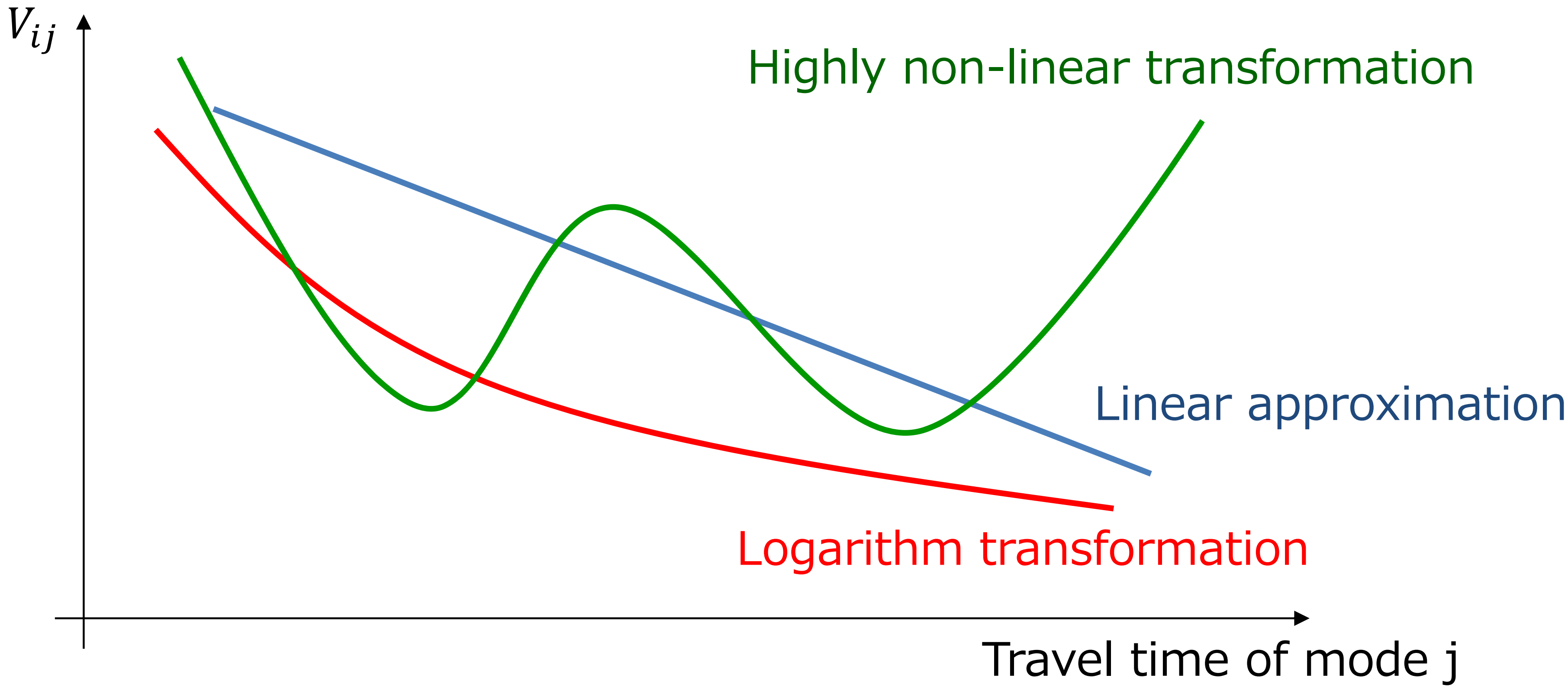
Problem setting

- Standard logit model: $P_{ij} = \dfrac{\exp(V_{ij})}{\sum_{j'=1}^{J} \exp(V_{ij'})}$
- The conventional form of V_{ij}:
  - Linear approximation (rooted to the Taylor's theorem)
  - Also known as a linear-in-parameter model
- Problem at hand:
  - Is there any better way to determine the functional form?
    - Obviously, taking into account the non-linearity of V_{ij} would improve the goodness-of-fit.
    - What is the cost of doing that?

Problem setting

- Can we understand the non-linear transformation of V_{ij} logically?

Example: contribution of travel time to mode/route choice model



Non-linearity through neural network (NN): It's about how to construct network architecture

A classical MNL

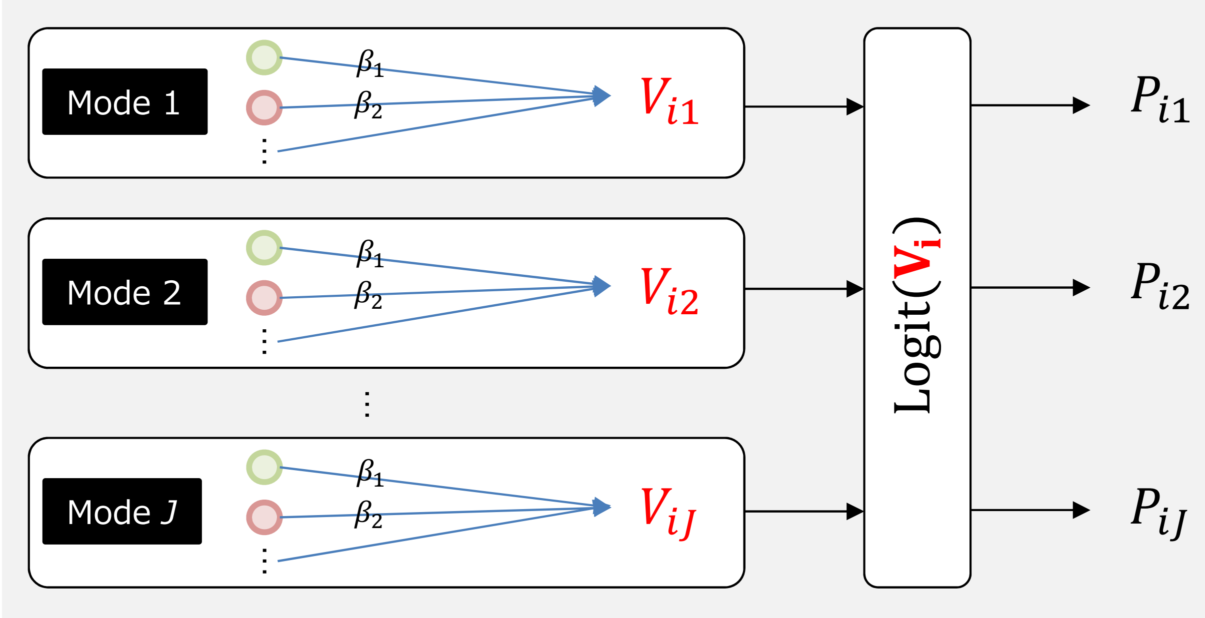
$$P_{ij} = \mathrm{Logit}(\mathbf{V_i}) = \frac{\exp(V_{ij})}{\sum_{j'=1}^{J} \exp(V_{ij'})}$$

$$V_{ij} = \beta_1 x_{ij1} + \beta_2 x_{ij2} + \cdots$$

Travel time of mode j

Travel cost of mode j

Network architecture



(Deep) NN

(Fully connected)

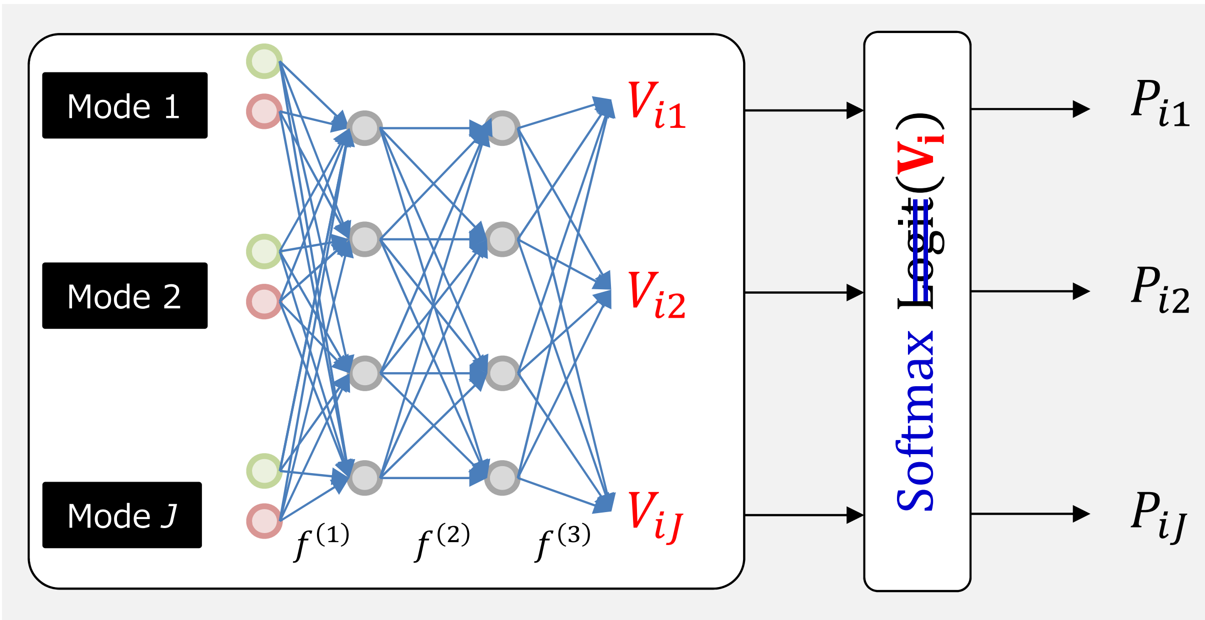
$$P_{ij} = \frac{\exp(V_{ij})}{\sum_{j'=1}^{J} \exp(V_{ij'})}$$

$$V_{ij} = f^{(3)}\left(f^{(2)}\left(f^{(1)}(\mathbf{x_i})\right)\right)$$

An example of f: Rectified linear unit (ReLU)

$$f(\boldsymbol{x}; \boldsymbol{W}, \boldsymbol{c}, \boldsymbol{w}, b) = \boldsymbol{w}^{\top} \max\{0, \boldsymbol{W}^{\top}\boldsymbol{x} + \boldsymbol{c}\} + b$$

Network architecture

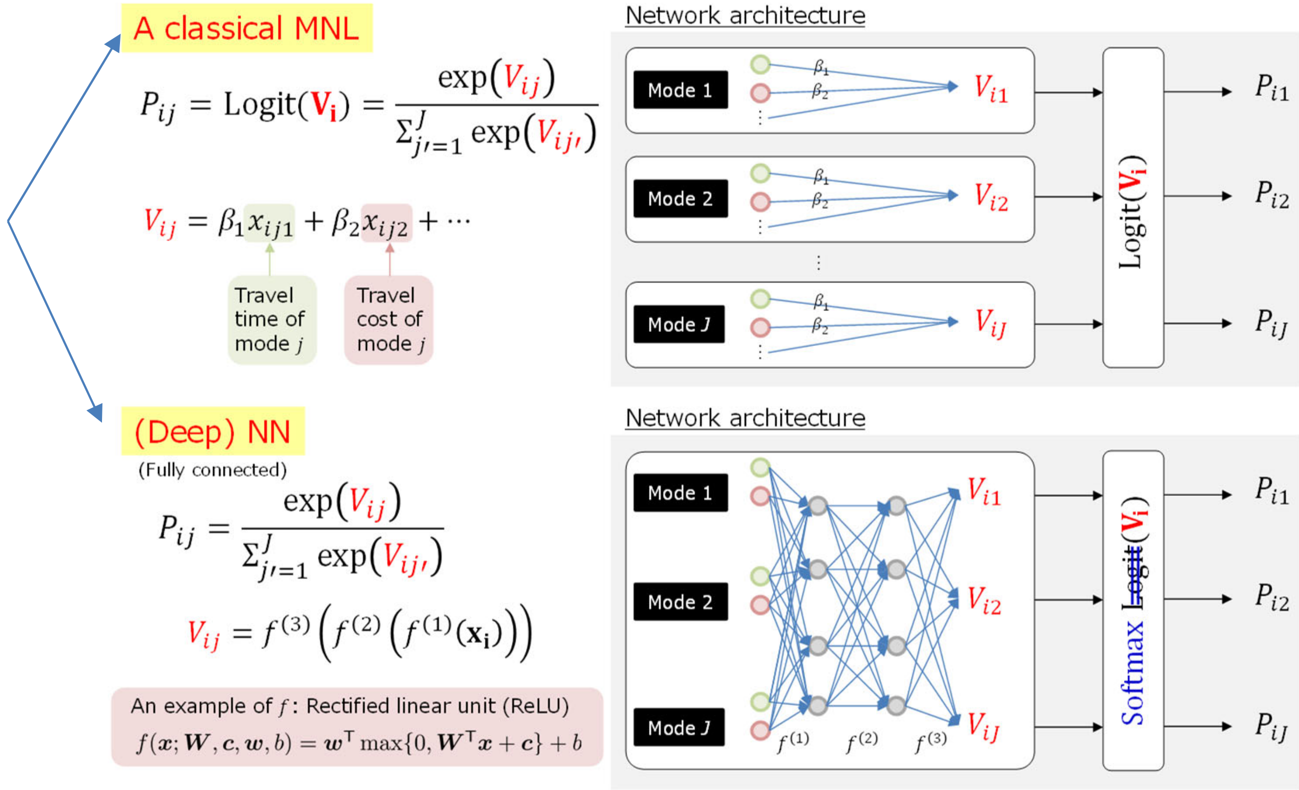


Universal approximation theorem

- Universal approximation theorem by Hornik et al. (1989), Cybenko (1989)
  - This theorem says that neural networks can approximate any function.
- This theorem also said that “shallow” network structure can approximate any function, while it is also known that more efficient learning can be achieved with “deep” network structure.
- Another important issue is the explainability of the fully connected NN.

Fully connected DNN often does not work well (and produce less explainable results)

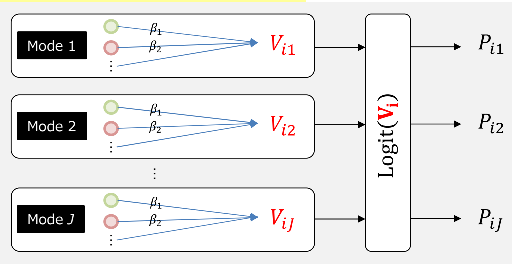
Seeking a better network structure in between.



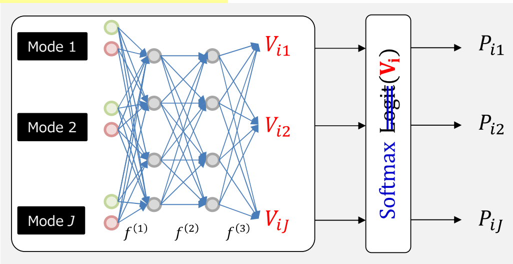
Efforts to keep both explainability and accuracy

- **Wang (2020)**
  - From fully connected deep neural network (F-DNN) to DNN with alternative-specific utility functions (ASU-DNN)

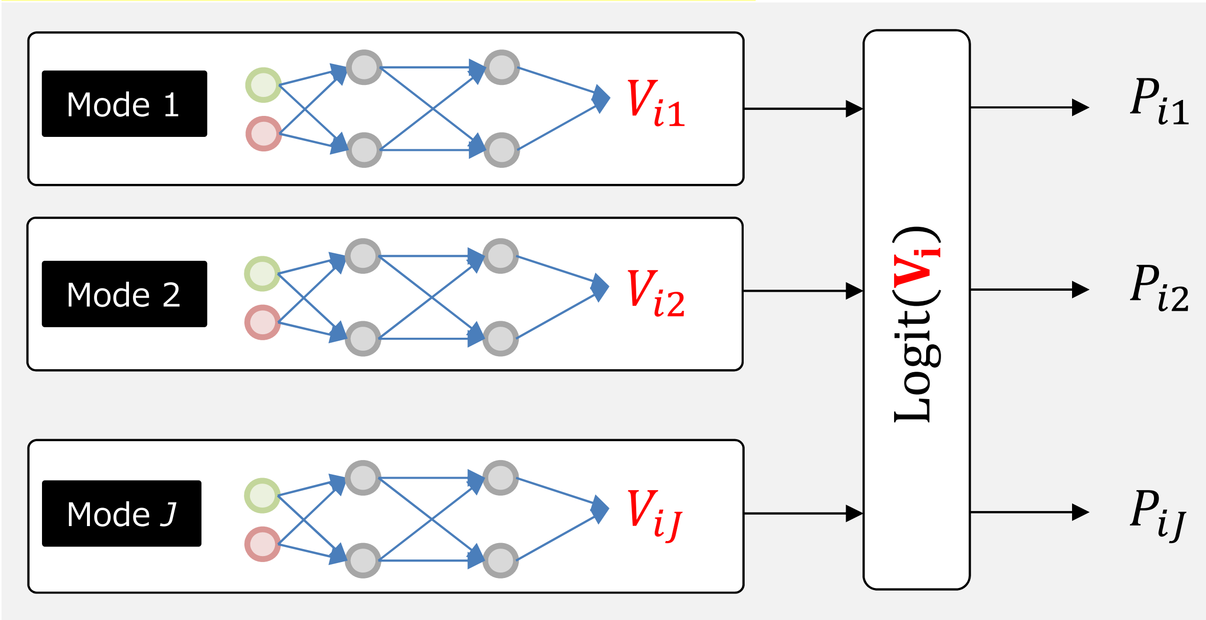
A classical MNL



(Deep) NN



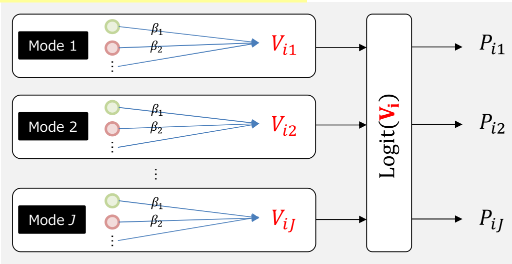
Proposed network architecture



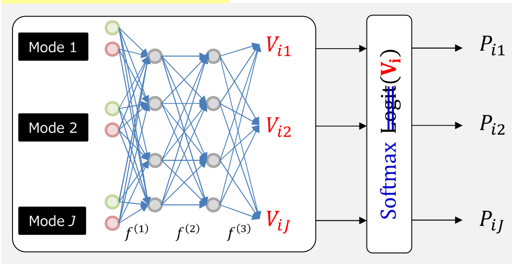
Efforts to keep both explainability and accuracy

- **Sifringer et al. (2020)**
  - Traditional linear-in-parameters are assumed for important policy variables, while DNN is used for the rest of variables (TB-ResNets proposed by Wang (2020) also follows a similar idea, but use a different method to implement it)

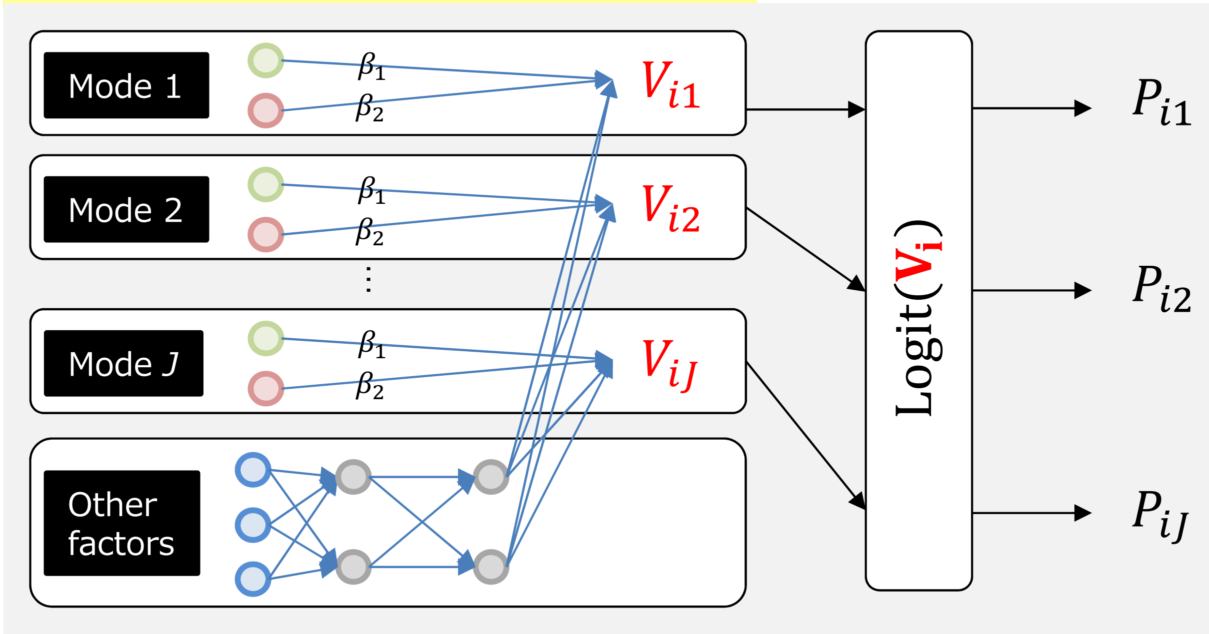
A classical MNL



(Deep) NN



Proposed network architecture



Chikaraishi, M., Garg, P., Varghese, V., Yoshizoe, K., Urata, J., Shiomi, Y. and Watanabe, R.: On the possibility of short-term traffic prediction during disaster with machine learning approaches: An exploratory analysis. *Transport Policy 98, 91-104, 2020*.

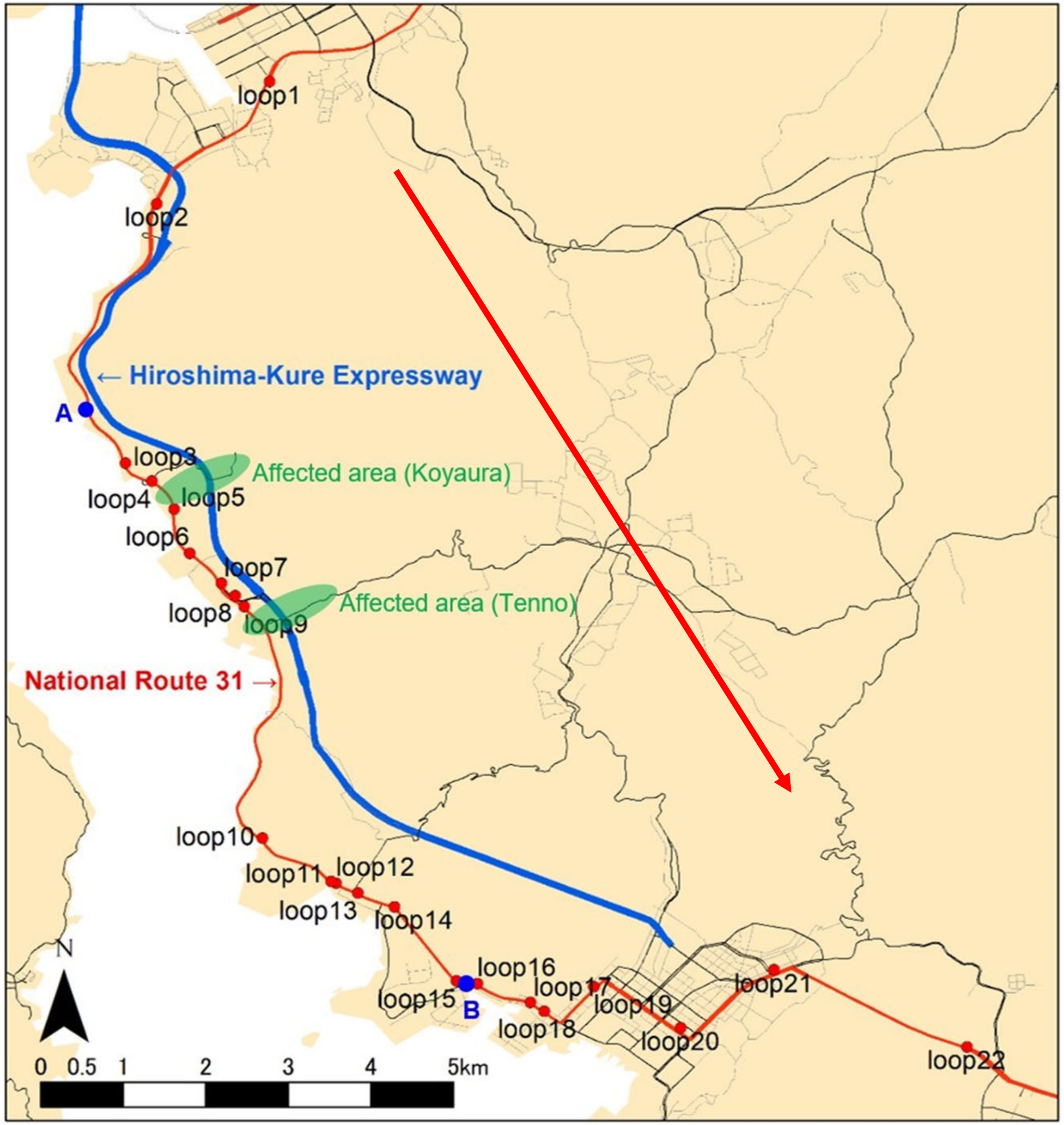
WHAT WOULD HAPPEN IF WE NAIVELY APPLY ML?

An example of less explainable results

1. Predict loop detector 9's traffic flow (Q) and time occupancy (K) using different ML methods.

2. Check the consistency of the results with theory

   Traffic flow theory said: traffic state should be dependent on traffic volume on the upstream and/or time occupancy on the downstream in congested situation (not the other way around).



An example of less explainable results

Prediction accuracy

XGB is the best in terms of prediction accuracy.
DNN also performs well.

Period	Time	5 min		10 min		20 min	
	Method	Flow	occupancy	Flow	occupancy	Flow	occupancy
		R^2(MAE)	R^2 (MAE)	R^2 (MAE)	R^2 (MAE)	R^2 (MAE)	R^2 (MAE)
A	ARIMA	0.86 (0.27)	0.87 (0.25)	0.81 (0.32)	0.81 (0.31)	0.74 (0.38)	0.72 (0.40)
	VAR	0.80 (0.30)	0.83 (0.27)	0.73 (0.35)	0.75 (0.33)	0.66 (0.39)	0.63 (0.38)
	RF	0.87 (0.22)	0.89 (0.21)	0.83 (0.27)	0.85 (0.25)	0.82 (0.27)	0.84 (0.26)
	SVM	0.85 (0.24)	0.85 (0.24)	0.83 (0.26)	0.83 (0.26)	0.81 (0.29)	0.80 (0.29)
	XGB	0.87 (0.23)	0.88 (0.22)	0.83 (0.27)	0.85 (0.25)	0.81 (0.29)	0.82 (0.28)
	FFNN	0.74 (0.34)	0.75 (0.35)	0.66 (0.40)	0.74 (0.35)	0.75 (0.36)	0.71 (0.39)
	DNN	0.83 (0.28)	0.85 (0.27)	0.80 (0.32)	0.79 (0.31)	0.74 (0.37)	0.77 (0.34)
B	ARIMA	0.77 (0.31)	0.91 (0.20)	0.68 (0.39)	0.84 (0.27)	0.60 (0.45)	0.75 (0.34)
	VAR	0.65 (0.45)	0.88 (0.25)	0.57 (0.50)	0.83 (0.30)	0.53 (0.51)	0.73 (0.34)
	RF	0.80 (0.31)	0.93 (0.18)	0.77 (0.36)	0.90 (0.21)	0.75 (0.38)	0.87 (0.26)
	SVM	0.78 (0.35)	0.88 (0.26)	0.76 (0.37)	0.85 (0.30)	0.75 (0.39)	0.79 (0.34)
	XGB	0.82 (0.29)	0.94 (0.17)	0.79 (0.33)	0.91 (0.21)	0.82 (0.29)	0.94 (0.17)
	FFNN	0.72 (0.42)	0.86 (0.28)	0.72 (0.41)	0.83 (0.32)	0.70 (0.42)	0.76 (0.37)
	DNN	0.79 (0.35)	0.89 (0.26)	0.76 (0.40)	0.84 (0.31)	0.69 (0.44)	0.74 (0.39)
C	ARIMA	0.76 (0.34)	0.90 (0.24)	0.70 (0.41)	0.84 (0.30)	0.65 (0.45)	0.76 (0.37)
	VAR	0.67 (0.40)	0.82 (0.27)	0.61 (0.45)	0.76 (0.31)	0.30 (0.45)	0.45 (0.36)
	RF	0.84 (0.25)	0.90 (0.19)	0.82 (0.29)	0.87 (0.23)	0.82 (0.30)	0.83 (0.26)
	SVM	0.80 (0.31)	0.87 (0.23)	0.77 (0.35)	0.84 (0.26)	0.74 (0.38)	0.81 (0.29)
	XGB	0.83 (0.27)	0.90 (0.19)	0.82 (0.29)	0.87 (0.23)	0.81 (0.31)	0.82 (0.27)
	FFNN	0.78 (0.32)	0.84 (0.25)	0.52 (0.51)	0.79 (0.30)	0.71 (0.38)	0.77 (0.32)
	DNN	0.79 (0.32)	0.86 (0.23)	0.71 (0.39)	0.80 (0.28)	0.70 (0.41)	0.70 (0.37)

Note: A represents the period before the disaster i.e. from July 1 to 5, 2018.
B represents the period immediately after the disaster i.e. July 12 to 18, 2018.
C represents the period after the disaster i.e. August 20-26, 2018.

An example of less explainable results

Particularly XGB **does NOT** really mimic the mechanisms of congestion occurrence.

What we have learned: The model which produces the best prediction accuracy is not always the best for practical use.

Table 2. Top 10 important features for DNN and XGB for LD-9

Method	Dependent variable	Period	Prediction horizon	Top 10 important features									
DNN	Q	July 1-5	5 minutes	Q9 (60)	Q9 (59)	Q9 (58)	Q5 (60)	Q4 (60)	Q3 (60)	Q2 (60)	Q2 (59)	K9 (60)	K9 (56)
			10 minutes	Q9 (60)	Q3 (60)	Q2 (60)	Q1 (60)	K22 (60)	K22 (59)	K22 (2)	K22 (1)	K21 (60)	K4 (1)
			20 minutes	Q22 (18)	Q21 (60)	Q3 (60)	Q2 (60)	K22 (60)	K21 (51)	K21 (1)	K15 (60)	K7 (1)	K2 (24)
		July 12-18	5 minutes	Q9 (60)	Q9 (6)	Q9 (59)	Q9 (56)	Q9 (36)	Q8 (60)	Q5 (60)	Q3 (60)	K12 (1)	K9 (60)
			10 minutes	Q9 (60)	Q9 (41)	Q9 (11)	Q5 (60)	Q3 (60)	K12 (1)	K4 (60)	K4 (59)	K2 (60)	K2 (59)
			20 minutes	Q17 (60)	Q17 (59)	Q17 (57)	Q17 (56)	Q9 (51)	Q9 (21)	Q5 (60)	K16 (1)	K10 (1)	K2 (60)
		August 20-26	5 minutes	Q9 (60)	Q9 (6)	Q9 (59)	Q9 (56)	Q9 (51)	Q9 (36)	Q9 (1)	Q7 (60)	Q4 (60)	Q3 (60)
			10 minutes	Q9 (60)	Q9 (56)	Q9 (41)	Q9 (36)	Q9 (11)	Q9 (1)	Q3 (60)	K17 (2)	K17 (1)	K3 (60)
			20 minutes	Q19 (57)	Q19 (2)	Q19 (1)	Q13 (60)	Q9 (6)	Q9 (51)	Q9 (46)	Q9 (21)	Q7 (60)	Q3 (60)
	K	July 1-5	5 minutes	Q9 (60)	Q9 (56)	Q4 (60)	Q2 (60)	K9 (60)	K9 (59)	K8 (60)	K8 (59)	K8 (58)	K3 (60)
			10 minutes	Q20 (43)	Q2 (60)	Q1 (60)	Q1 (59)	K22 (60)	K22 (59)	K22 (1)	K9 (60)	K8 (60)	K3 (60)
			20 minutes	Q21 (60)	Q21 (59)	Q21 (58)	Q20 (1)	Q2 (60)	K22 (60)	K22 (59)	K21 (2)	K21 (1)	K15 (60)
		July 12-18	5 minutes	Q9 (56)	Q5 (60)	Q4 (60)	K9 (60)	K9 (59)	K9 (58)	K9 (57)	K8 (60)	K8 (59)	K8 (58)
			10 minutes	Q9 (60)	Q7 (60)	Q5 (60)	K22 (60)	K9 (60)	K9 (59)	K9 (58)	K8 (60)	K8 (59)	K4 (60)
			20 minutes	Q9 (60)	Q9 (59)	Q5 (60)	Q4 (60)	K22 (43)	K9 (60)	K9 (59)	K9 (58)	K8 (60)	K2 (60)
		August 20-26	5 minutes	Q9 (6)	Q9 (56)	Q9 (36)	Q7 (60)	Q4 (60)	K9 (60)	K9 (59)	K9 (58)	K8 (60)	K8 (59)
			10 minutes	Q9 (41)	Q9 (11)	Q7 (60)	Q3 (60)	K9 (60)	K9 (59)	K9 (58)	K8 (60)	K8 (59)	K5 (60)
			20 minutes	Q9 (51)	Q9 (21)	Q7 (60)	Q5 (1)	Q3 (60)	K13 (1)	K10 (60)	K9 (60)	K8 (60)	K3 (60)
XGB	Q	July 1-5	5 minutes	Q9 (60)	Q3 (60)	Q3 (59)	Q2 (59)	Q2 (58)	Q2 (57)	K15 (59)	K15 (58)	K9 (60)	K5 (60)
			10 minutes	Q21 (60)	Q21 (51)	Q2 (60)	Q2 (58)	Q2 (57)	K15 (60)	K8 (57)	K8 (56)	K8 (55)	K1 (60)
			20 minutes	Q21 (60)	Q21 (56)	Q21 (51)	Q15 (56)	Q13 (60)	Q13 (55)	Q9 (60)	K15 (56)	K15 (53)	K4 (60)
		July 12-18	5 minutes	Q21 (60)	Q21 (56)	Q9 (60)	Q4 (60)	Q2 (56)	K9 (60)	K5 (60)	K4 (60)	K3 (60)	K2 (59)
			10 minutes	K9 (60)	K5 (60)	K5 (56)	K3 (60)	K3 (56)	K1 (60)	K1 (58)	K1 (56)	K1 (55)	K1 (53)
			20 minutes	Q21 (60)	Q21 (56)	Q9 (60)	Q4 (60)	Q2 (56)	K9 (60)	K5 (60)	K4 (60)	K3 (60)	K2 (59)
		August 20-26	5 minutes	Q21 (56)	Q9 (60)	Q8 (60)	Q7 (60)	Q4 (60)	K12 (60)	K7 (60)	K5 (60)	K3 (59)	K2 (57)
			10 minutes	Q8 (60)	Q7 (60)	Q3 (60)	K12 (60)	K11 (60)	K11 (55)	K7 (60)	K3 (60)	K2 (60)	K2 (57)
			20 minutes	Q2 (60)	Q2 (58)	K10 (58)	K9 (56)	K7 (60)	K3 (60)	K2 (58)	K1 (60)	K1 (59)	K1 (51)
	K	July 1-5	5 minutes	Q21 (60)	Q9 (60)	Q3 (60)	Q2 (58)	Q2 (57)	K6 (60)	K5 (60)	K4 (60)	K4 (58)	K2 (60)
			10 minutes	Q21 (60)	Q21 (57)	Q21 (46)	Q7 (57)	Q4 (59)	Q2 (60)	K10 (60)	K8 (57)	K1 (60)	K1 (59)
			20 minutes	Q21 (60)	Q21 (56)	Q21 (51)	Q13 (60)	Q13 (55)	Q2 (60)	Q7 (60)	K13 (56)	K13 (54)	K4 (60)
		July 12-18	5 minutes	Q13 (60)	Q4 (60)	Q3 (60)	K9 (60)	K8 (60)	K3 (60)	K3 (56)	K2 (58)	K1 (58)	K1 (57)
			10 minutes	Q4 (57)	K9 (60)	K8 (58)	K8 (57)	K8 (56)	K8 (55)	K3 (60)	K3 (54)	K1 (60)	K1 (55)
			20 minutes	Q13 (60)	Q4 (60)	Q3 (60)	K9 (60)	K8 (60)	K3 (60)	K3 (56)	K2 (58)	K1 (58)	K1 (57)
		August 20-26	5 minutes	Q21 (56)	Q1 (55)	K19 (60)	K9 (60)	K8 (60)	K5 (60)	K3 (59)	K3 (57)	K2 (41)	K1 (58)
			10 minutes	K9 (56)	K8 (60)	K8 (56)	K7 (60)	K5 (60)	K5 (59)	K4 (60)	K3 (60)	K2 (57)	K2 (47)
			20 minutes	Q2 (53)	K8 (60)	K8 (59)	K3 (60)	K2 (60)	K2 (59)	K2 (58)	K2 (57)	K2 (56)	K2 (55)

Notes: The bracket indicates time stamp, e.g., 1 means the 1st time stamp (i.e., data observed 60 min before the time prediction made) and 60 means the 60th time stamp (i.e., the newest data available at the time prediction made). The shaded feature means that the downstream traffic volume influences the upstream traffic states, which is difficult to explain from the perspective of traffic flow theory.

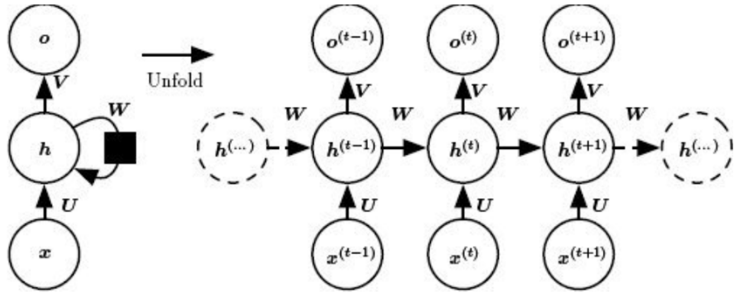
Goodfellow I, Bengio Y, Courville A. Deep learning, MIT Press; 2016.

NETWORK ARCHITECTURE FOR REPRESENTING TEMPORAL DEPENDENCIES

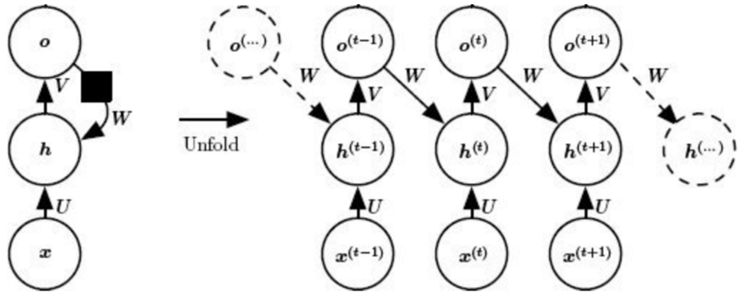
Recurrent neural network

- Recurrent neural network
  - A neural network that is specialized for processing a sequence of values (e.g., time series data).
  - Parameter sharing
    - A recurrent neural network typically shares the same parameters across time steps.
    - This is needed to generalize and make it possible to predict future.
    - An example:
      - Recurrent structure:
        - » Tomorrow will come after today.
      - Non-recurrent structure:
        - » Sep. 18, 2021 will come after Sep. 17, 2021.
  - There are a wide variety of recurrent neural networks (next slide).

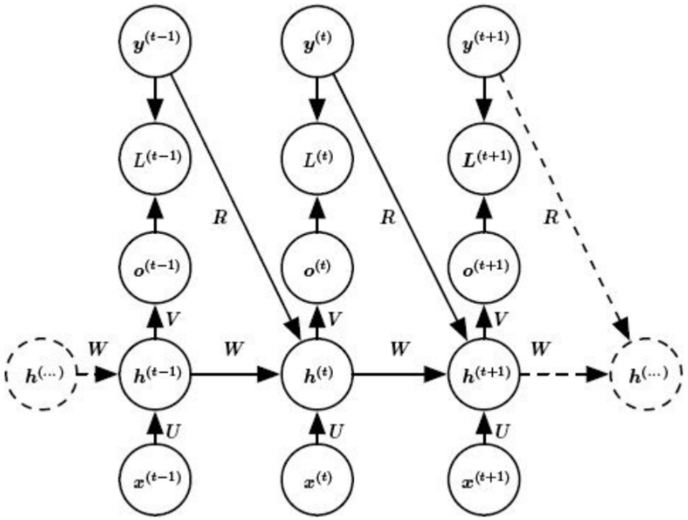
Examples of RNN structures



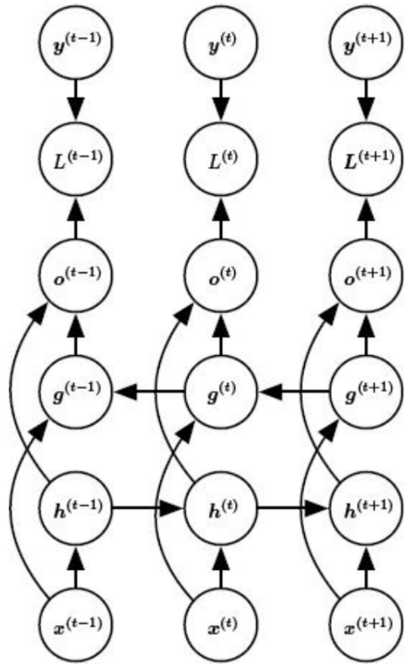
Recurrent networks that produce an output at each time step and have recurrent connections between hidden nodes.



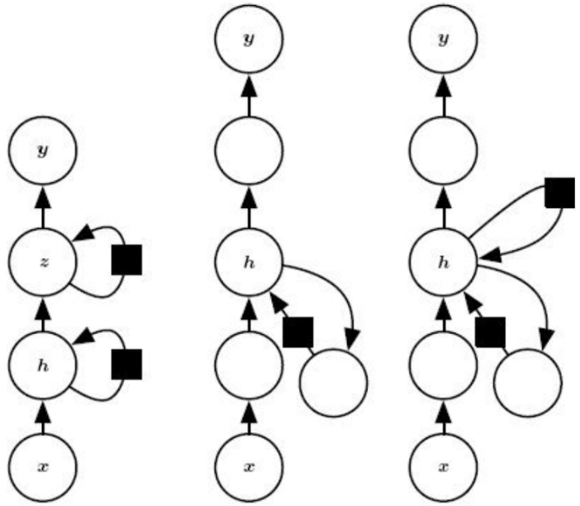
Recurrent networks that produce an output at each time step and have recurrent connections only from the output at the next step to the hidden units at the next time step.



Adding connection from the output at time t to the hidden unit at time t+1



Bidirectional recurrent networks



Make network deeper

Various structures exist (similar with time series models with lagged variables)

Hierarchical structure of network: use the concept of "cell"

Having a cell (a set of nodes with a particular network structure), instead of simply having a node.

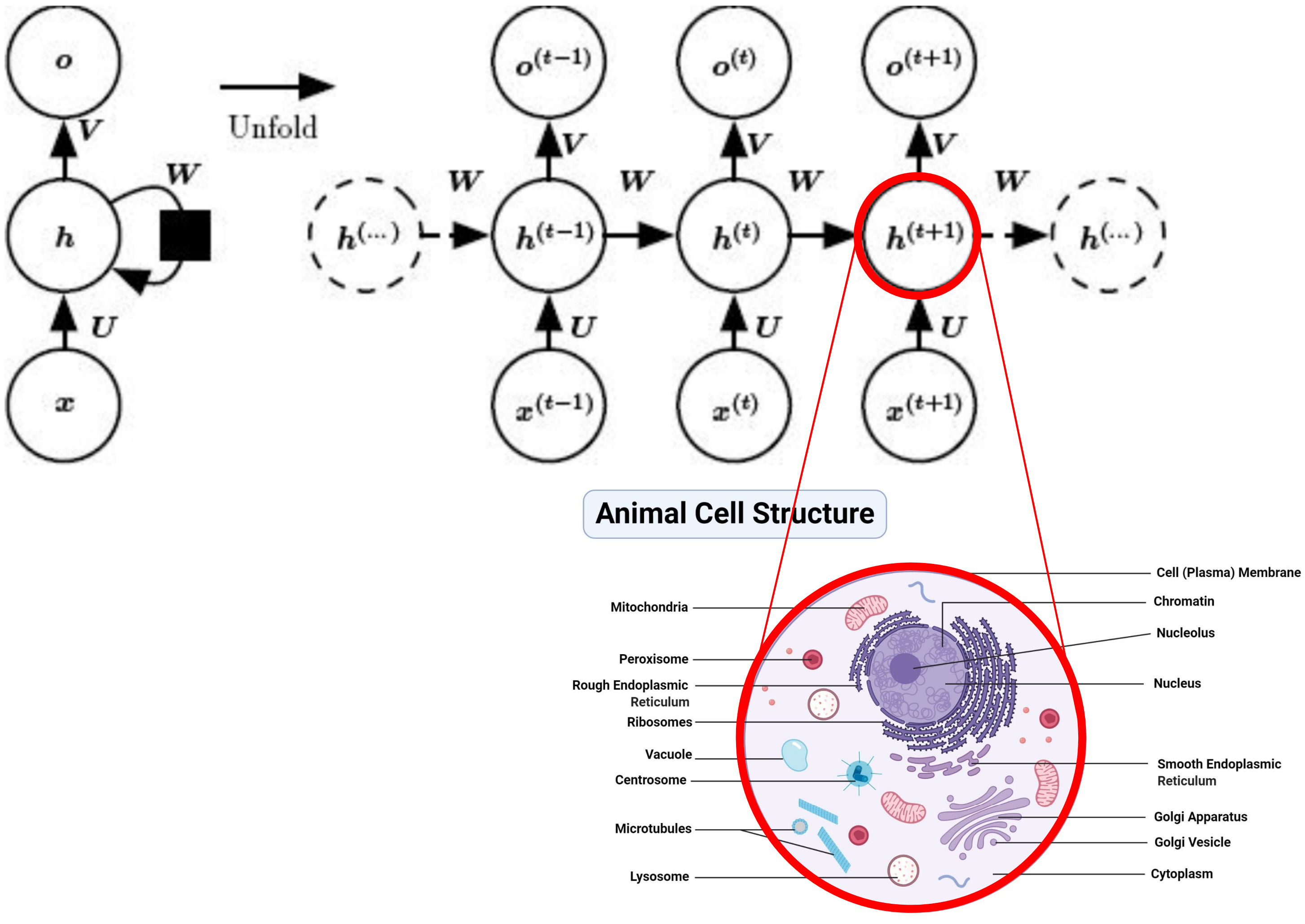


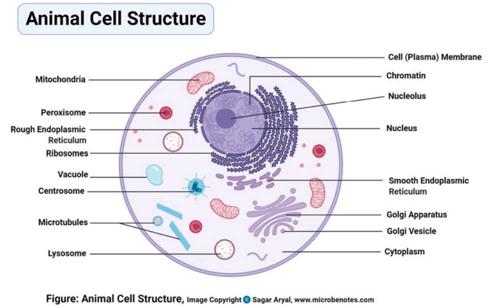
Figure: Animal Cell Structure, Image Copyright © Sagar Aryal, www.microbenotes.com

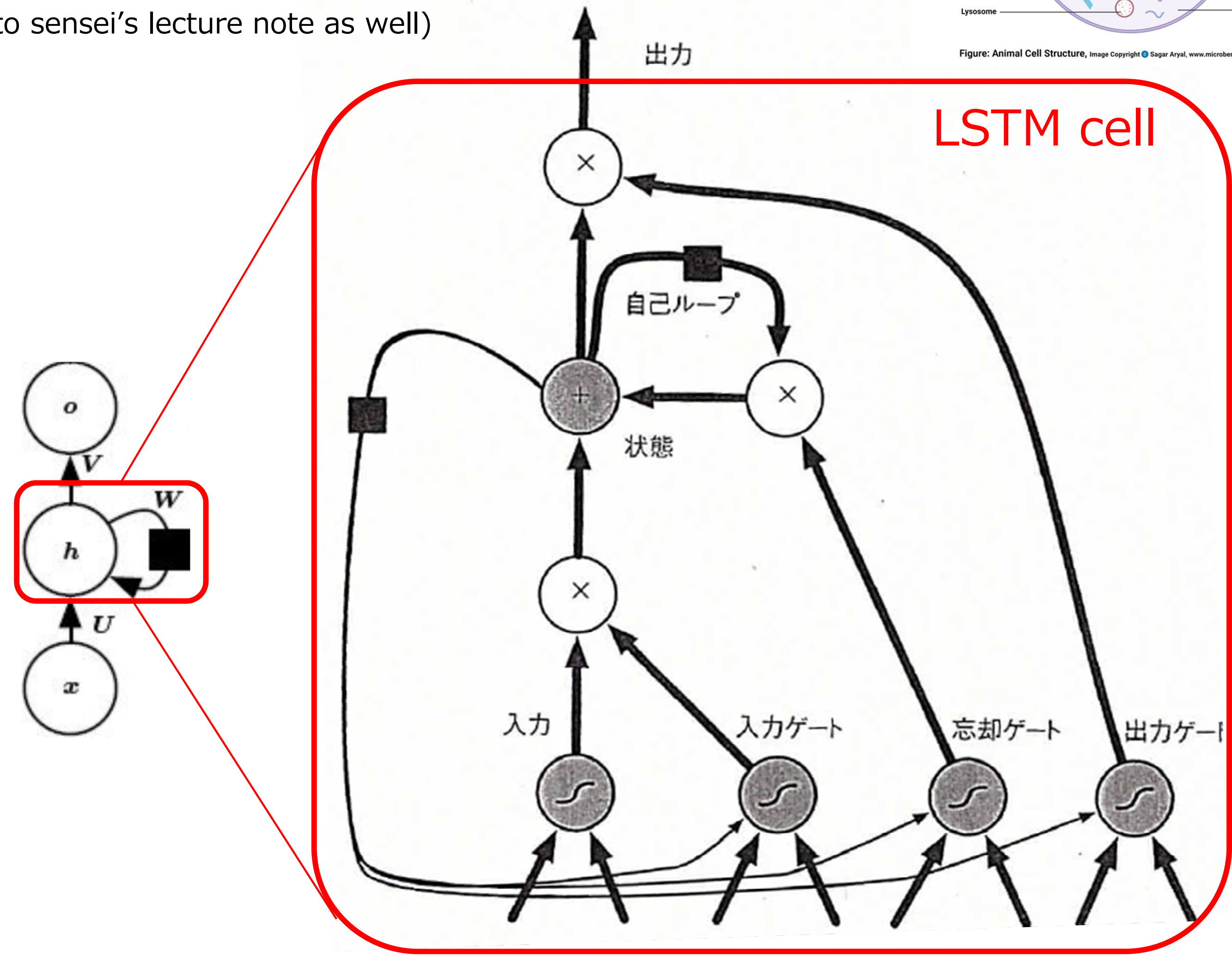
LSTM cell (Long short-term memory)

Hochreiter and Schmidhuber (1997)

The weight on the self-loop <u>conditioned on the context</u> (rather than fixed), resulting in the dynamic control of the time scale and forgetting behavior of different units.

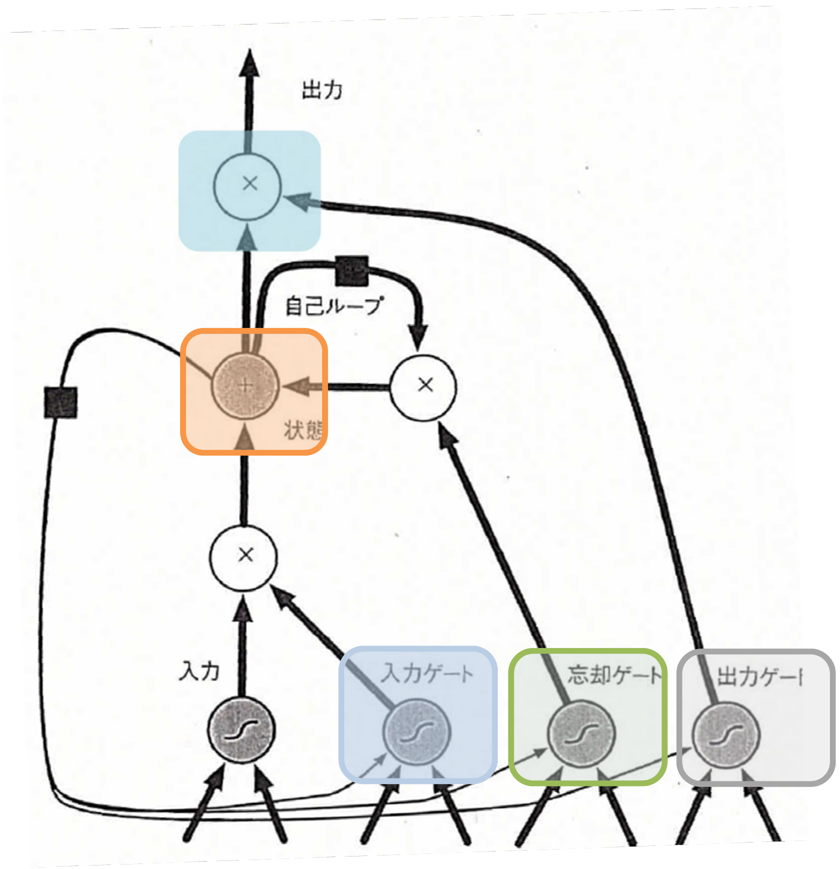
(Conceptually, it is similar with the introduction of moderators.
Please refer to Yamamoto sensei’s lecture note as well)





LSTM cell (Long short-term memory)

Hochreiter and Schmidhuber (1997)



Forget gate

$$f_i^{(t)} = \sigma\left(b_i^f + \sum_j U_{i,j}^f x_i^{(t)} + \sum_j W_{i,j}^f h_j^{(t-1)}\right)$$

Input gate

$$g_i^{(t)} = \sigma\left(b_i^g + \sum_j U_{i,j}^g x_i^{(t)} + \sum_j W_{i,j}^g h_j^{(t-1)}\right)$$

State

$$s_i^{(t)} = f_i^{(t)} s_i^{(t-1)} + g_i^{(t)} \sigma\left(b_i + \sum_j U_{i,j} x_j^{(t)} + \sum_j W_{i,j} h_j^{(t-1)}\right)$$

Output

$$h_i^{(t)} = \tanh\left(s_i^{(t)}\right) q_i^{(t)}$$

Output gate

$$q_i^{(t)} = \sigma\left(b_i^o + \sum_j U_{i,j}^o x_i^{(t)} + \sum_j W_{i,j}^o h_j^{(t-1)}\right)$$

Other variants...

- ✓ GRU: Gated recurrent unit (Cho et al., 2014)
  - ✓ Simpler than LSTM, but the performance is similar to that of LSTM for some applications.

Remaining concerns in applications to transport issues

- Temporal dependencies are not independent from spatial dependencies.
  - LSTM is designed for handling temporal dependencies, not spatial dependencies.

Cui, Z., Henrickson, K., Ke, R.and Wang, Y.: Traffic Graph Convolutional Recurrent Neural Network: A Deep Learning Framework for Network-Scale Traffic Learning and Forecasting. *IEEE Transactions on Intelligent Transportation Systems 21, 4883-4894, 2020.*

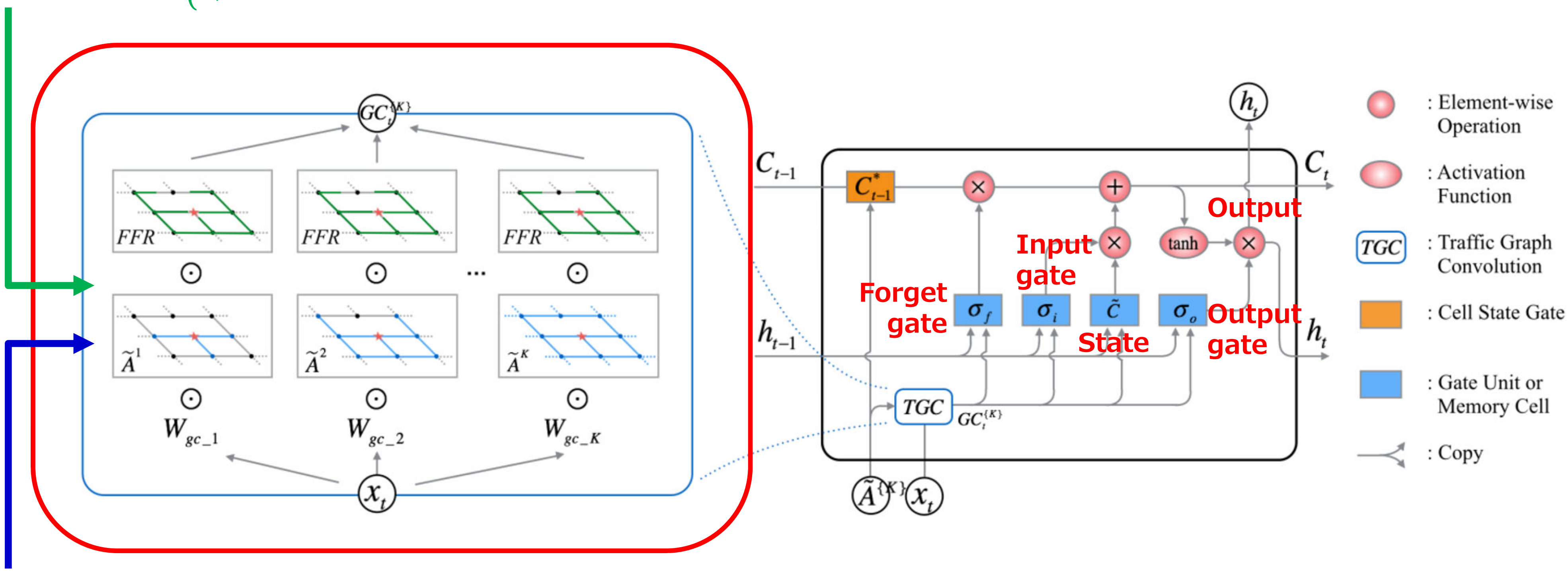
TGC-LSTM

(TRAFFIC GRAPH CONVOLUTIONAL RNN)

Network architecture employed

An free-flow reachable matrix

$$FFR_{i,j} = \begin{cases} 1, & if\ S_{i,j}^{FF} m\Delta t - Dist_{i,j} \geq 0 \\ 0, & otherwise \end{cases}, \forall\, v_i, v_j$$

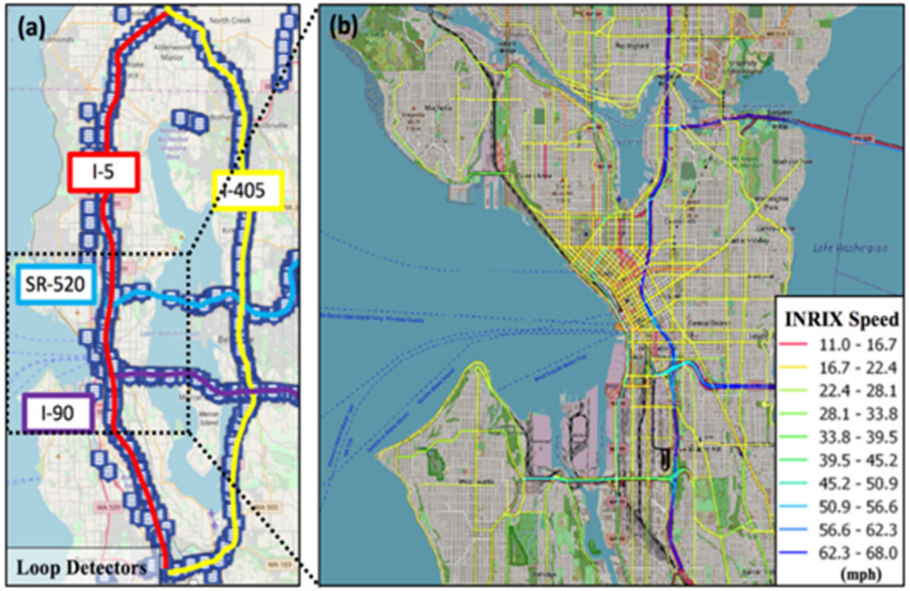


An adjacency matrix $A \in \mathbb{R}^{N \times N}$, in which each element $A_{i,j}$=1 if there is an edge connecting node i and node j and $A_{i,j}$=0 otherwise.

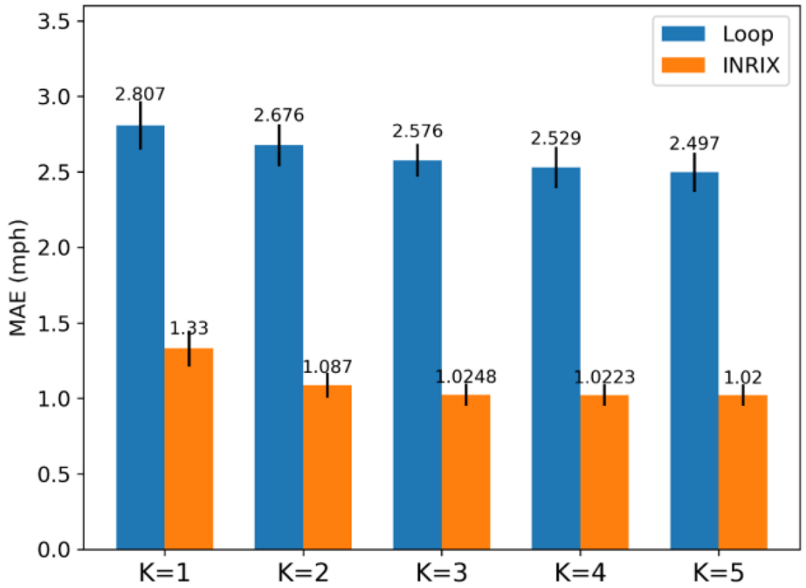
$$\tilde{A}_{i,j}^{k} = \min\left((A+I)_{i,j}^{k}, 1\right)$$

(called a k –hop neighborhood matrix)

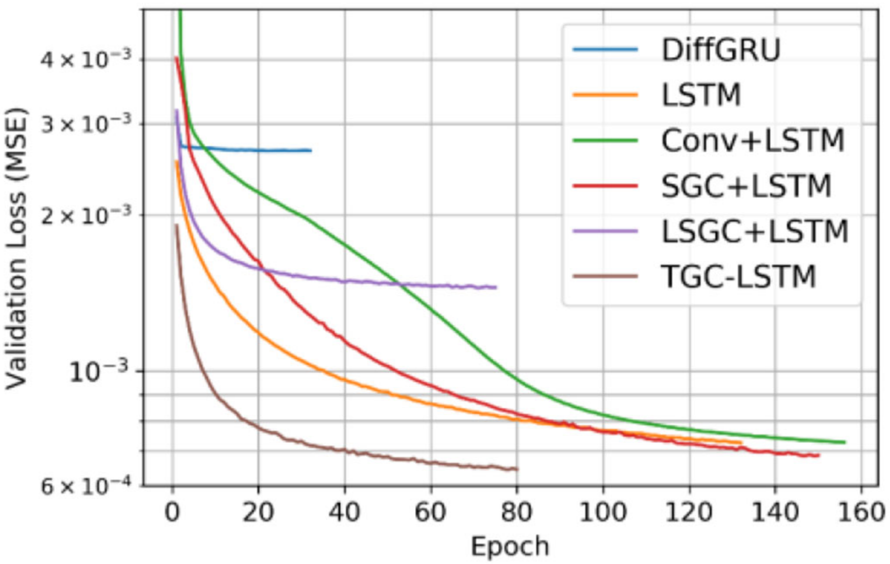
Empirical results



a) LOOP dataset covering the freeway network in Seattle area; (b) INRIX dataset covering the downtown Seattle area, where traffic segments are plotted with colors.



Histogram of performance comparison for the influence of orders (hops) of graph convolution in the TGC LSTM on INRIX and LOOP datasets.



Validation loss versus training epoch (batch size = 40 and early stopping patience = 10 epochs)

Limitations

1. Undirected graph is used, not directed one.
   - Limited applications of directed graph convolution.
   - Recently, some researchers have proposed approximation methods such as Tong et al. (2020)

2. Travel time and congestions are not endogenously modeled.
   - choice [C] = f(travel time [TT]), but also TT = g(C) = g(f(TT))

Conclusions

- Take-away messages:
  - Shifting from "choosing theory-driven OR data-driven" to "integrating theory-driven AND data-driven".
  - Temporal dimension can be well modeled using LSTM etc., while temporal dimension and spatial dimension cannot be simply separated in transport studies.
  - Some researchers have been actively working on the development of the methods for handling both temporal and spatial dependencies in a consistent way with theories in the transportation field. Yet, still a number of challenges (e.g., endogenous representation of travel time with directed graph) remain.

References

- Brathwaite, T.: The Holy Trinity: Blending Statistics, Machine Learning and Discrete Choice, with Applications to Strategic Bicycle Planning, PhD Dissertation at UC Berkeley, 2018, https://escholarship.org/uc/item/1pk9p2ct
- Chikaraishi, M., Garg, P., Varghese, V., Yoshizoe, K., Urata, J., Shiomi, Y. and Watanabe, R.: On the possibility of short-term traffic prediction during disaster with machine learning approaches: An exploratory analysis. Transport Policy 98, 91-104, 2020.
- Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H.and Bengio, Y.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078, 2014.
- Cui, Z., Henrickson, K., Ke, R.and Wang, Y.: Traffic Graph Convolutional Recurrent Neural Network: A Deep Learning Framework for Network-Scale Traffic Learning and Forecasting. IEEE Transactions on Intelligent Transportation Systems 21, 4883-4894, 2020.
- Goodfellow I, Bengio Y, Courville A. Deep learning, MIT Press; 2016.
- Hochreiter, S.and Schmidhuber, J.: Long short-term memory. Neural computation 9, 1735-1780, 1997.
- Sifringer, B., Lurkin, V.and Alahi, A.: Enhancing discrete choice models with representation learning. Transportation Research Part B 140, 236-261, 2020.
- Wang, S.: Deep neural networks for choice analysis, PhD Dissertation at MIT,2020 https://dspace.mit.edu/handle/1721.1/129894
- Tong, Z., Liang, Y., Sun, C., Rosenblum, D.S.and Lim, A.: Directed graph convolutional network. arXiv preprint arXiv:2004.13970, 2020. https://arxiv.org/pdf/2004.13970.pdf