Final Oral Examination: IoT Data Discovery and Learning

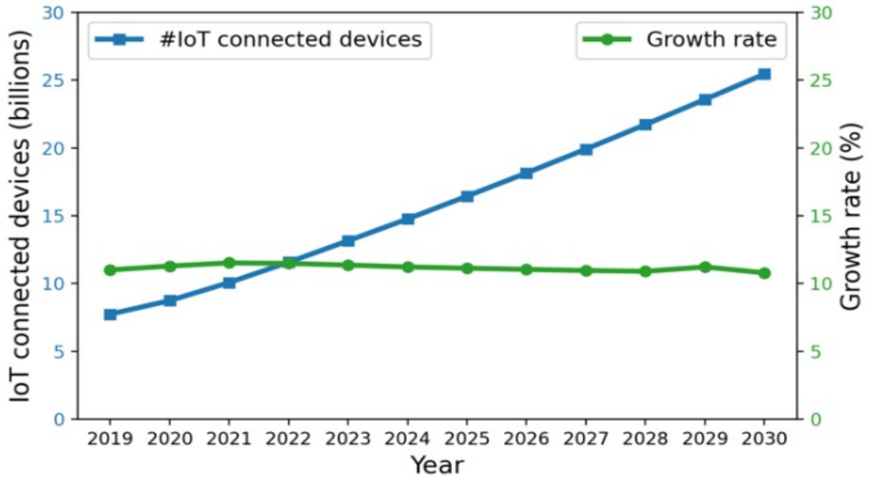
Ph.D. Candidate: Trung Hieu Tran
Department of Computer Science,
University of Texas at Dallas

Ph.D. Committee:
Dr. I-Ling Yen (Supervisor)
Dr. Farokh Bastani (Co-Supervisor)
Dr. Latifur Khan
Dr. Weili Wu
Dr. Kyeongjae Cho (Chair of the Examining Committee)

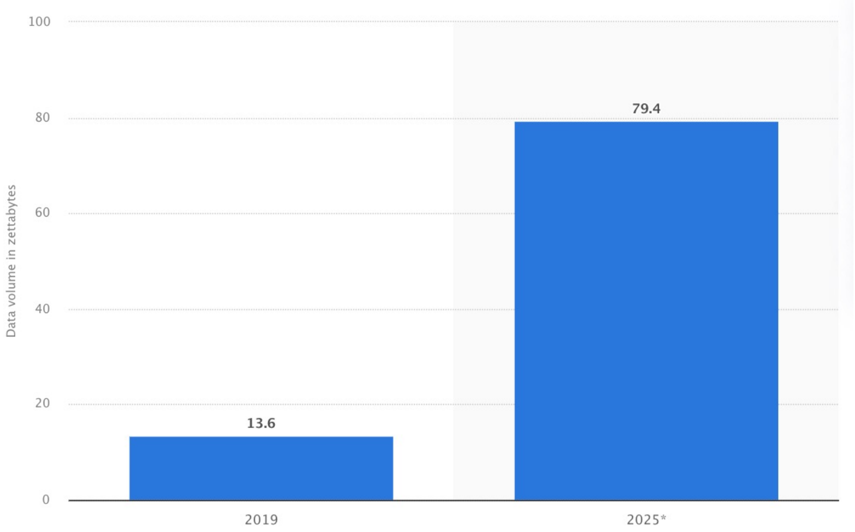
Presentation Outline

- Introduction – (Recap)
- Objective – (Recap)
- Our Approach
  - IoT Data Discovery – (Recap)
  - IoT Data Learning
    - Literature Review
    - Methodology
    - Experimental Studies
- Conclusion and Future Research

IoT Database Network (IoT-DBN)



Estimated Growth of the number of IoT devices worldwide [statista, 2021]

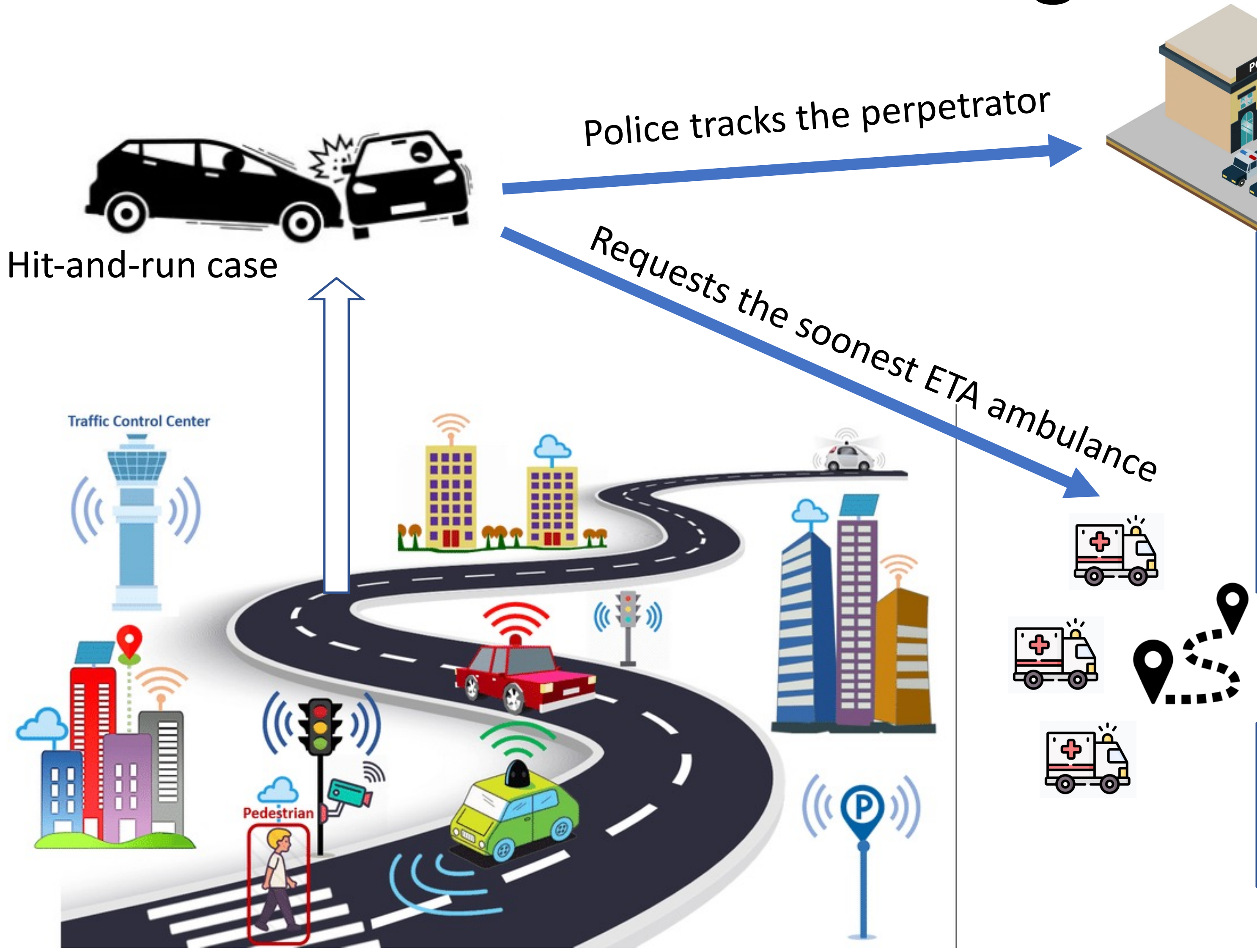


Data volume of IoT connections worldwide in 2019 and 2025 (in zettabytes*) [statista,2022]

* 1 Zettabytes = 10^{12} Gigabytes

- Growing ubiquity of IoT devices (~ 16 Billion devices in 2025)
- Creating a torrent of IoT data

Motivating Example



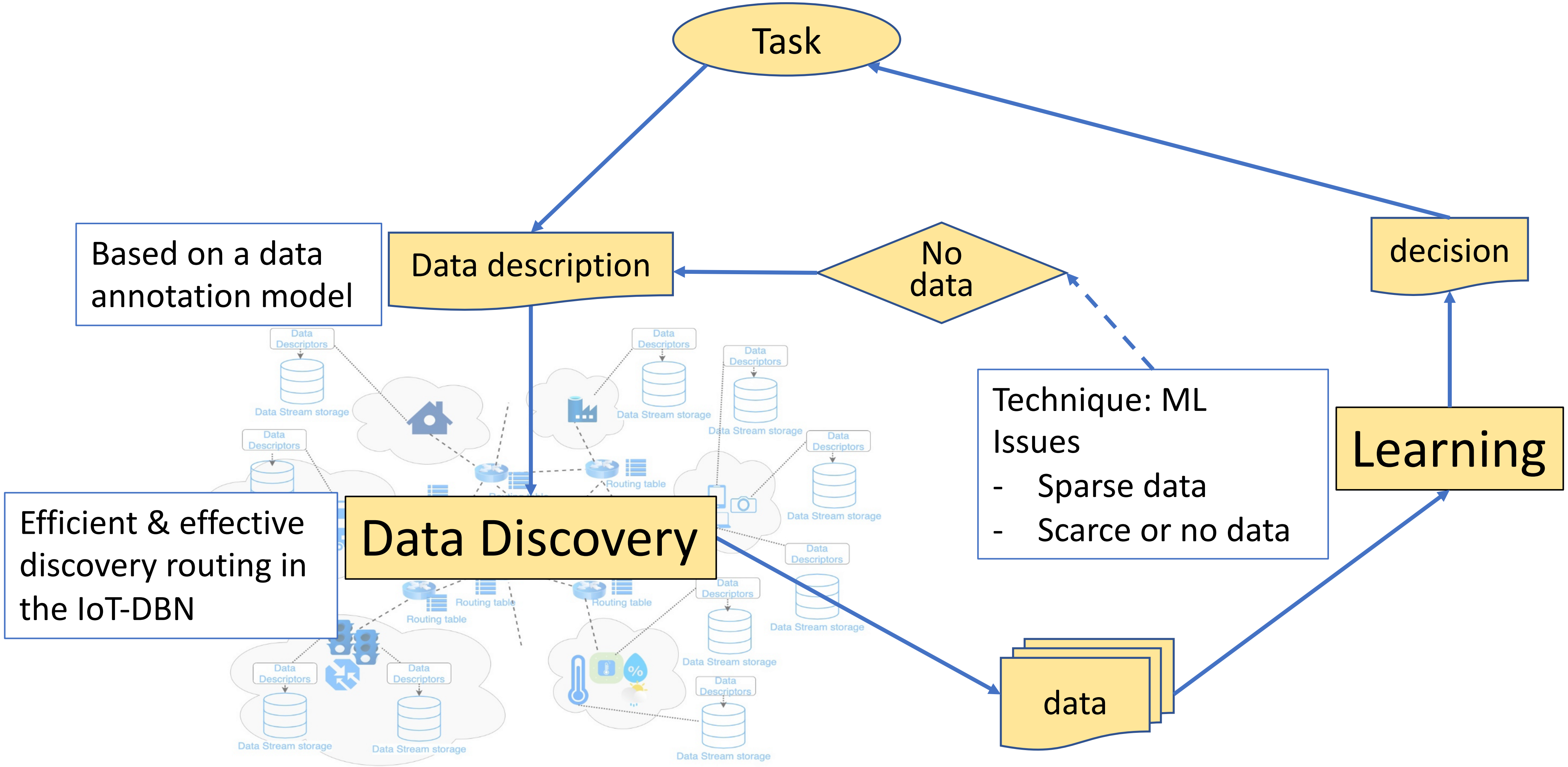
- Can use Google Map, Apple Map, etc.
- But estimators are mostly for regular cars, not for ambulances
- IoT sensors near the incident may capture the data that are useful for identifying and tracking the perpetrator
  - E.g., smart car sensors, roadside cameras

- Need to discover the relevant data
- Need learning methods to do ETA for ambulances

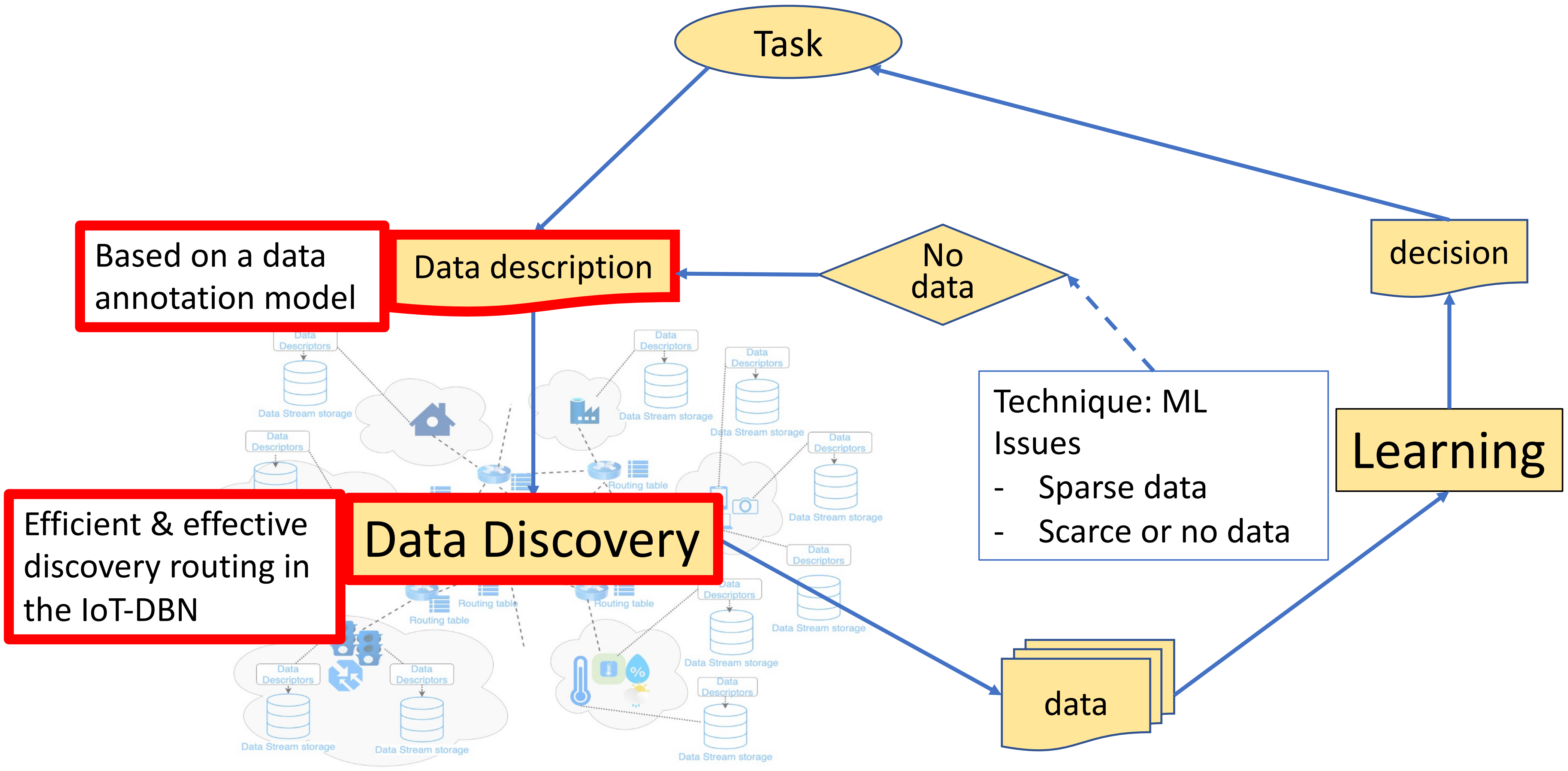
Objective

- **Make use** of the huge amount of data (from IoT sensors) to help improve the daily social operations
- Address existing issues:
  - How to discover the useful data for the current situation?
  - How to learn from the discovered data to make problem solving decisions?
  - How to address to data-sparsity in learning?

Overview of Our Approach

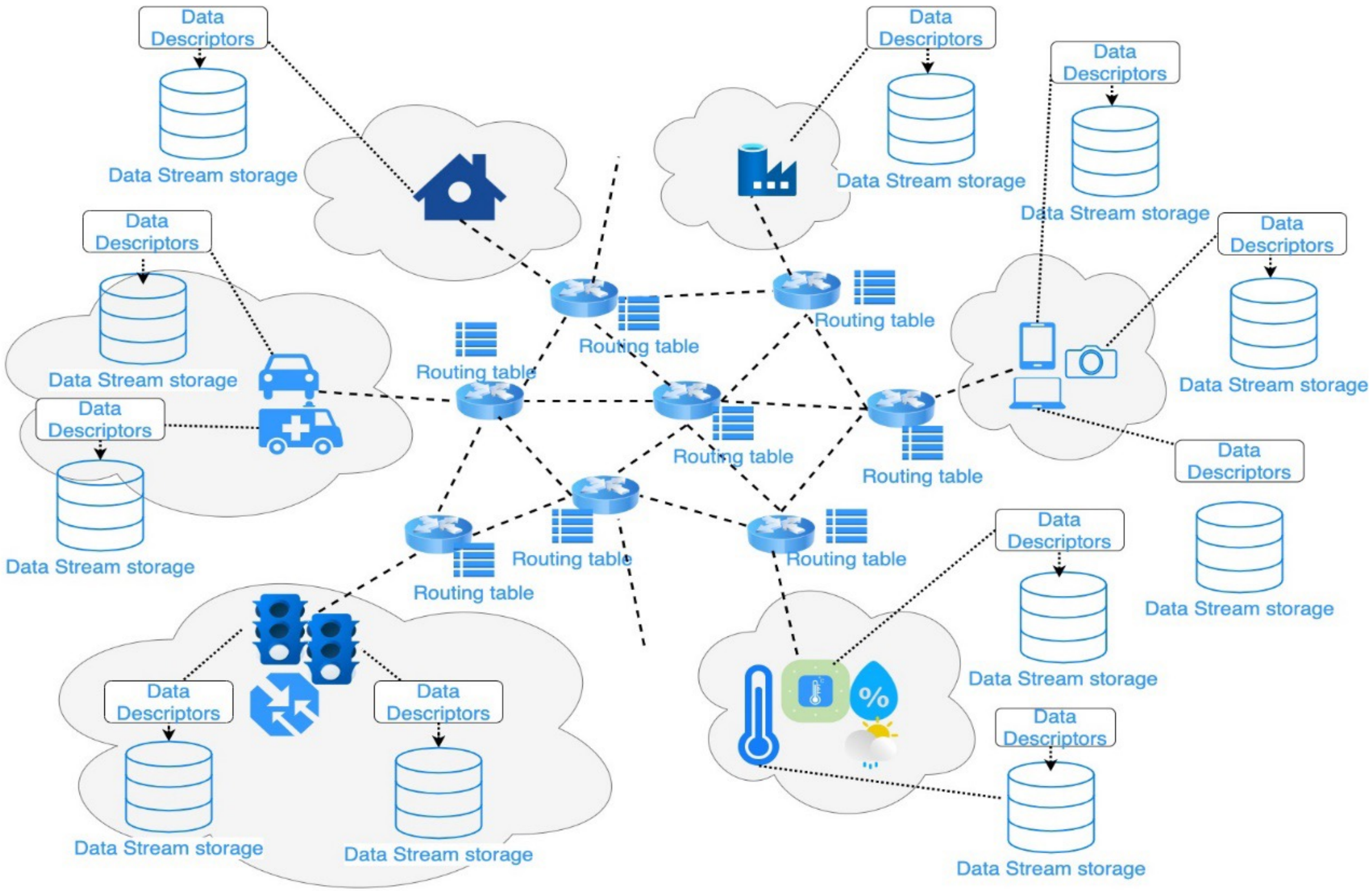


Overview of Our Approach



IoT Database Network

- IoT data are collected in a peer-to-peer manner by sensors on the edge of the Internet
- These data are likely being stored at the edge of the Internet
- Data discovery is to discover the relevant data streams via their descriptions.



Multi-Attribute Annotation Models (MAA)

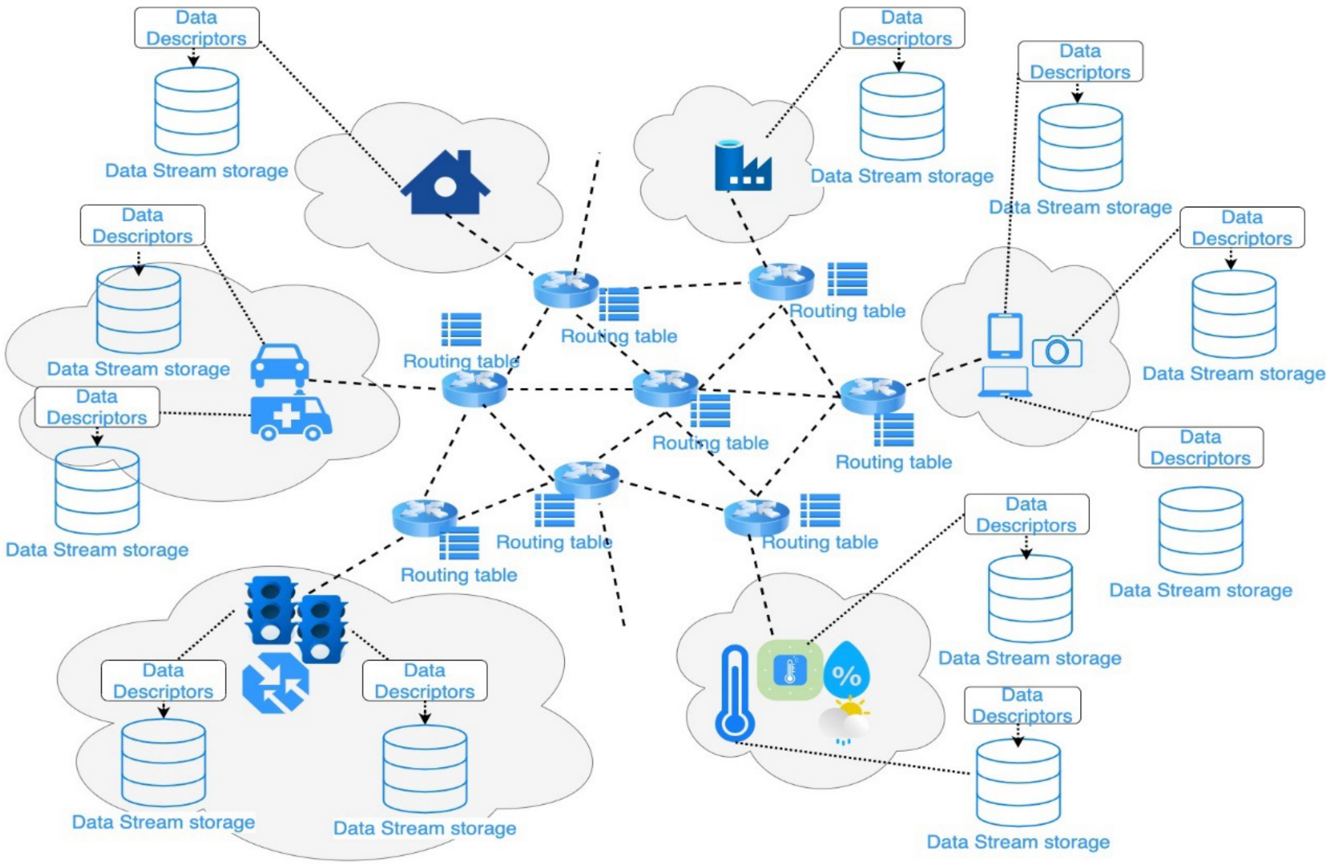
- MAA for the metadata of each datasteam
  - Has been considered widely for data stream annotation. [G.-E. Luis, 2004]
  - $ds = \left(\left(a_1: v_1^{ds}\right), \left(a_2: v_2^{ds}\right), \ldots, \left(a_n: v_n^{ds}\right)\right)$
    - $\left(a_{\mathrm{i}}: v_{\mathrm{i}}^{ds}\right)$: one descriptor with attribute a_{i} and value v_{i}^{ds}
  - Example:

(DataCategory: GPS; Vehicle type: car; City:$Cincinnati$, → descriptor
Region: $Central\ Business\ District$;
Day: $Weekend$; Region traffic volume: v;
Duration: *3/2/18 17:03:20 - 3/2/18 19:10:30*)

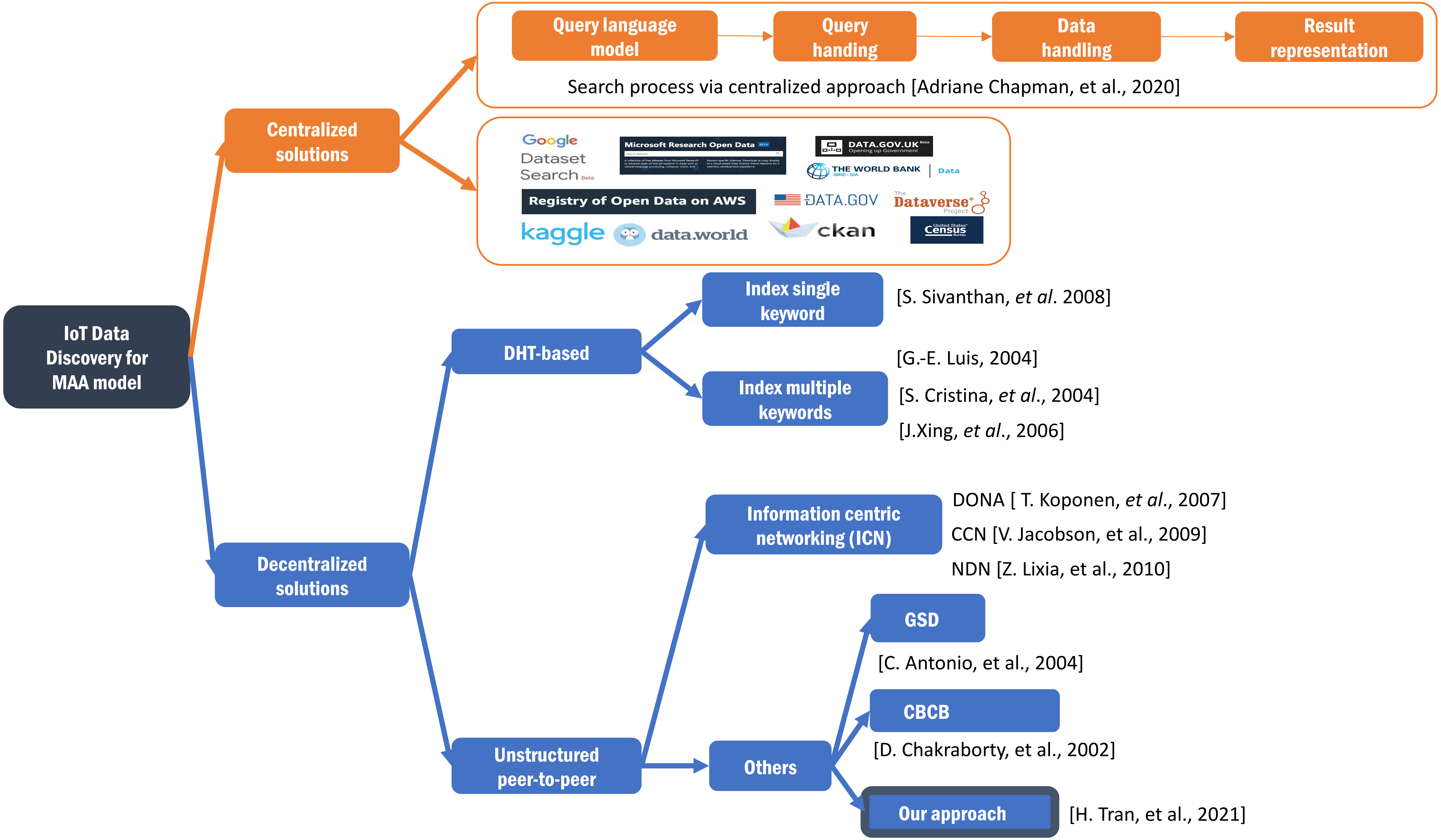
- Query in MAA
  - Subset of attributes
  - Example (DataCategory: GPS; Vehicle type: $ambulance$ || car; Region traffic volume: $[v_1, v_2]$)

Data Discovery Routing in IoT-DBN

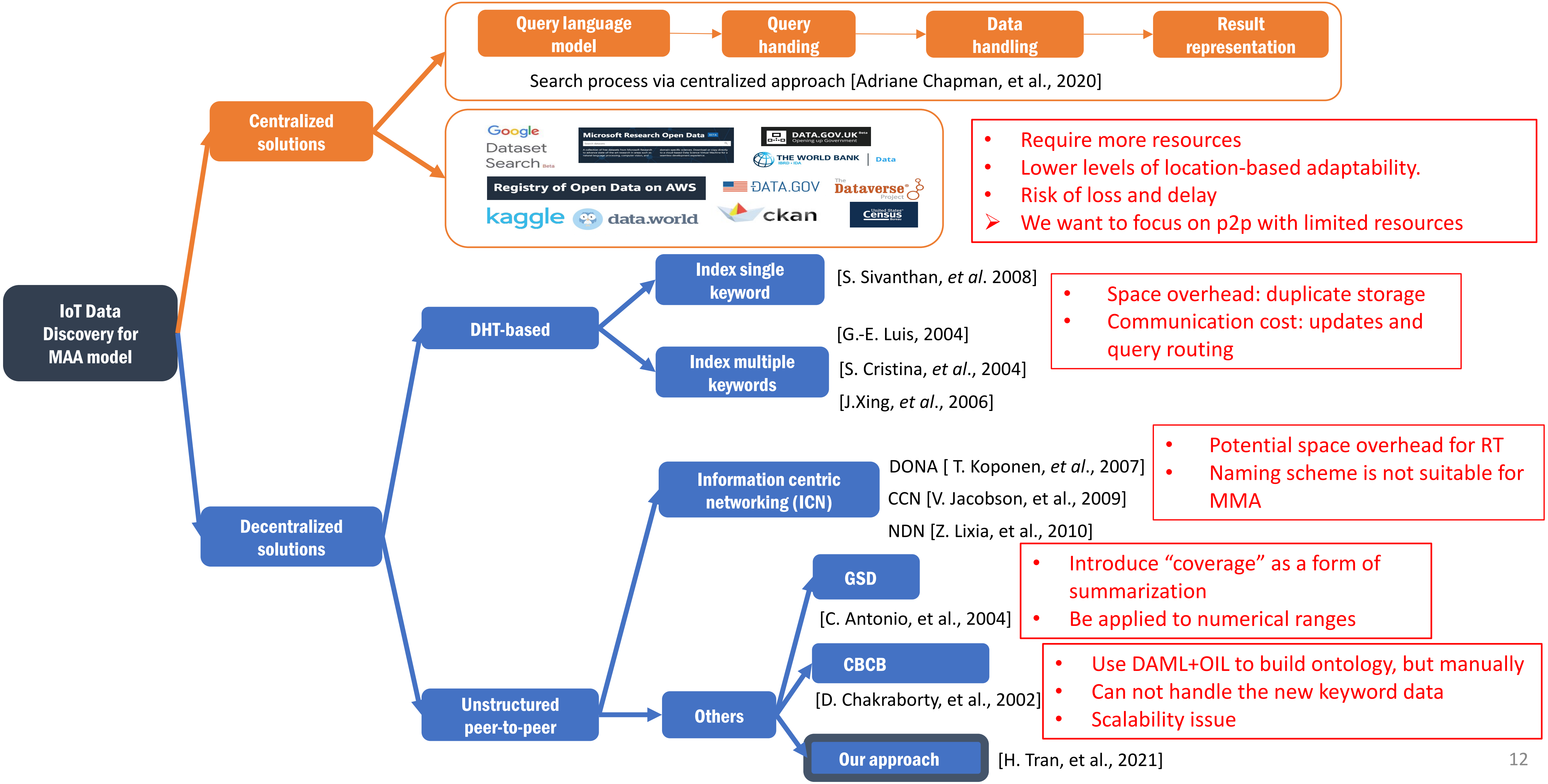
- **Routing table** (RT)
  - Each node builds a routing table to facilitate data discovery
- **Advertisement**
  - Data source sends out data descriptors => Relevant nodes add them in their RTs
- A **data discovery query**
  - Forwarded toward where the data is at based on RTs
  - With the help of RT information
- Like advanced ICN (information centric network)



Overview of Existing Works for Data Discovery



Overview of Existing Works for Data Discovery



Overview of Existing Works for Data Discovery

- Centralized solutions used popularly
  - ➢ We focus on p2p network with limited resource
- Decentralized solutions:
  - ➢ Space overhead issue
  - ➢ Communication cost
  - ➢ Lack of routing table compression for MAA model

Our Proposed Summarization Techniques

Goal: to address the space concerns in the resource-constrained network

- Alphabetical based policy (SP_{alph})
- Hash based policy (SP_{hash})
- Meaning based policy ($SP_{meaning}$)

Our Proposed Summarization Techniques

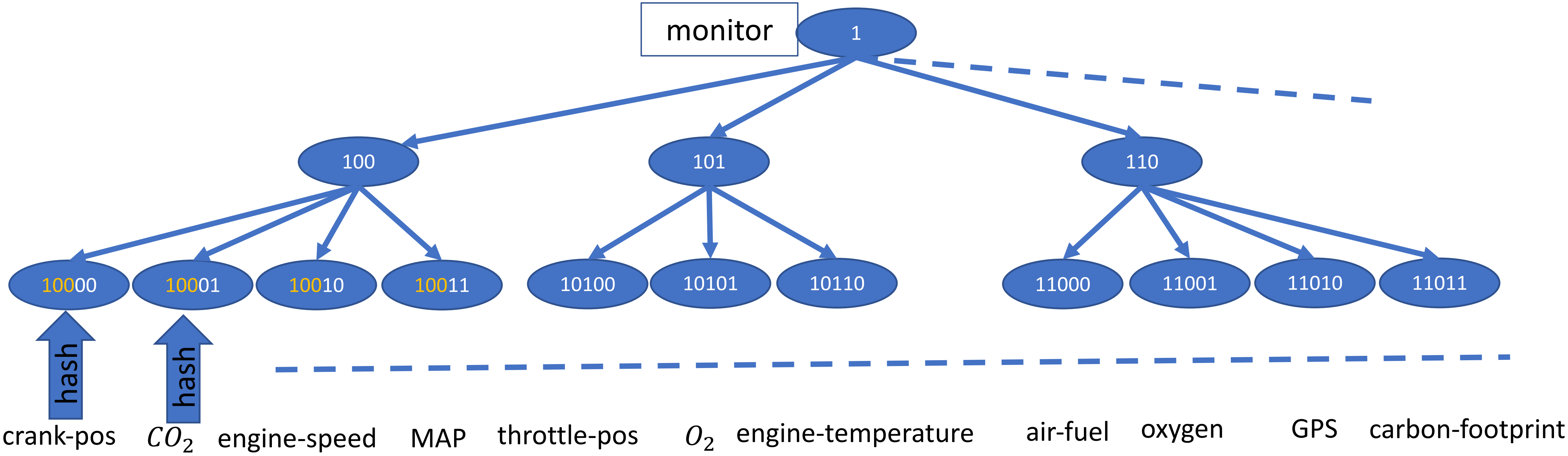
$\boldsymbol{SP_{alph}}$

- Most similar to IP summarization
- Example
  - Monitor:engine
  - Monitor:engine-speed
  - …
  - ⇒ In RTs: Only maintain **monitor:engine**

Our Proposed Summarization Techniques

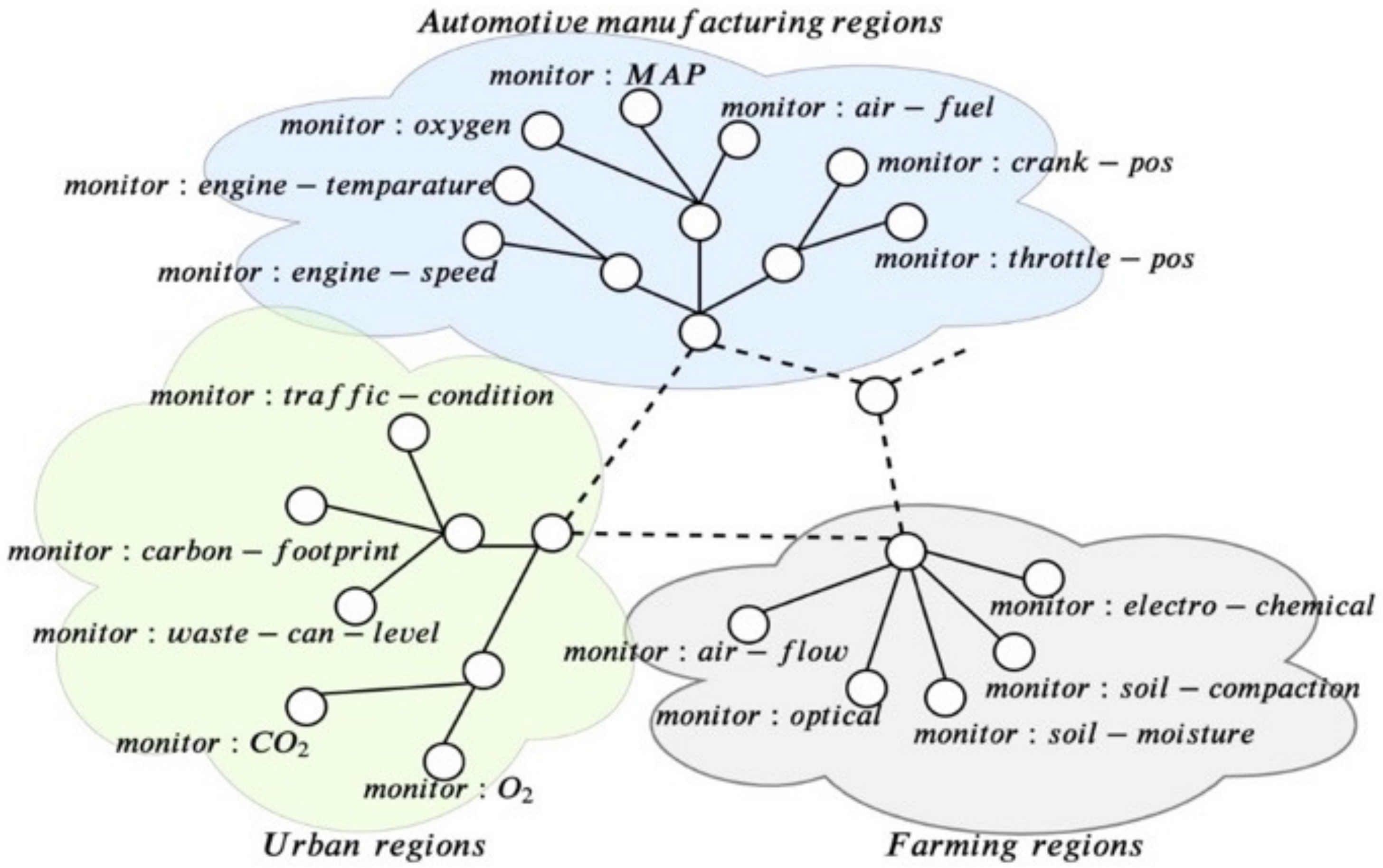
$\boldsymbol{SP_{hash}}$

- Address the space overhead in SP_{alph}
- Hash the keywords and use the hash values as the code
- Naturally summarizes by controlling the hash code length



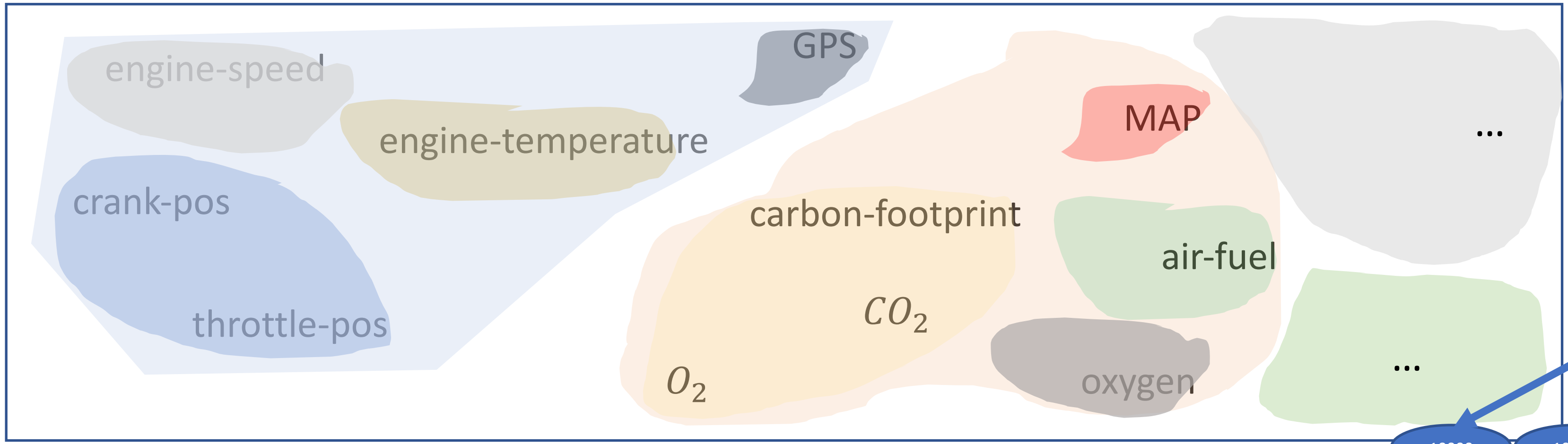
Our Proposed Summarization Techniques

- IoT data streams from a specific system, or a specific environment may have attributed keywords that are **semantically similar**



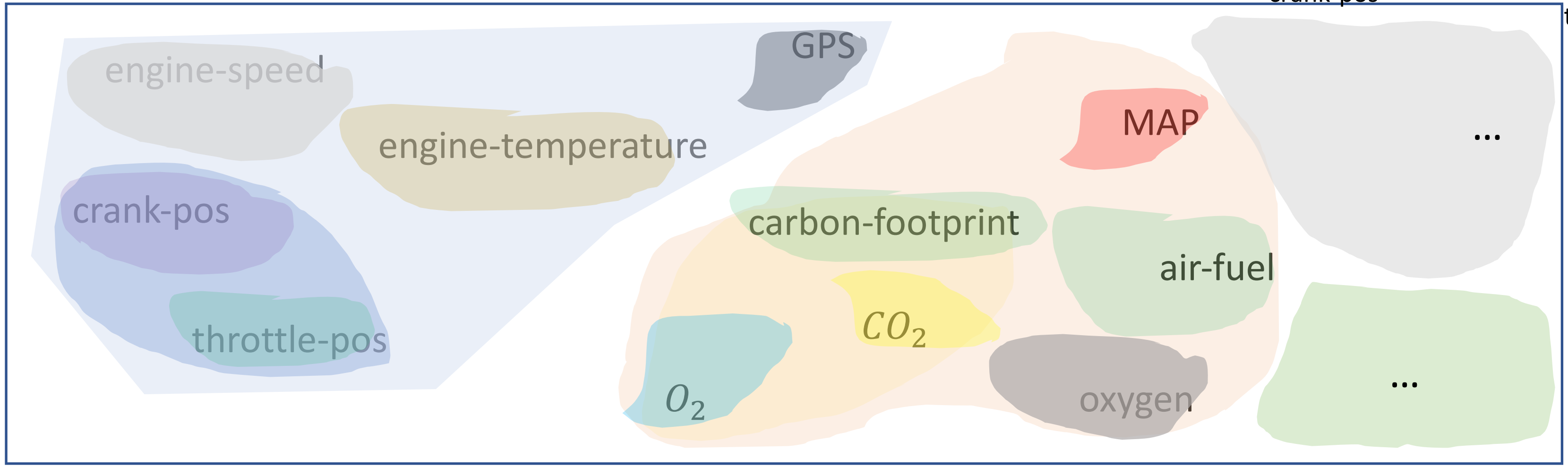
$SP_{meaning}$

1. $Word2Vec(keywords)$ into embedded space

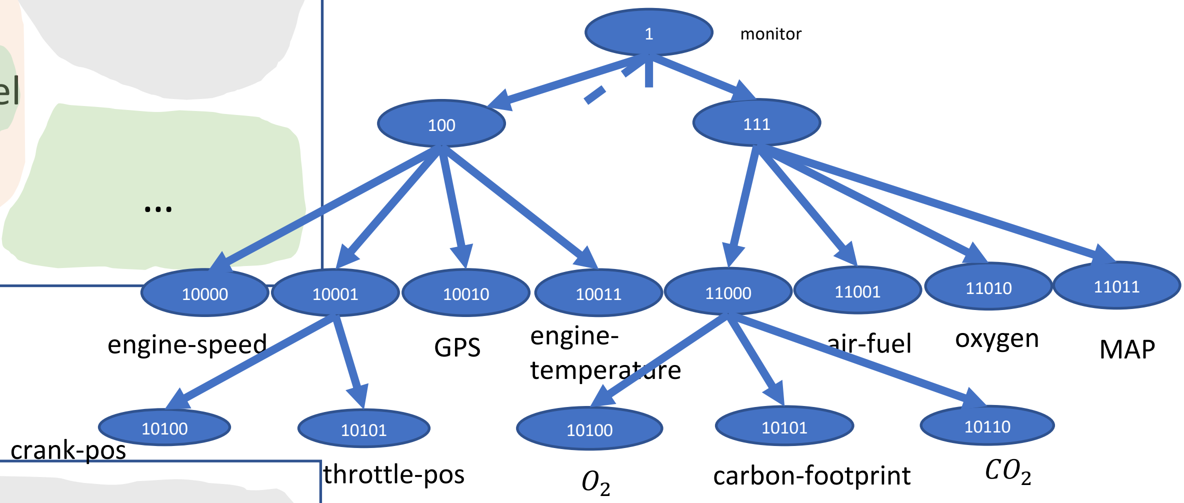


k-Means clustering

2. Clustering with $k = 2^c = 4$

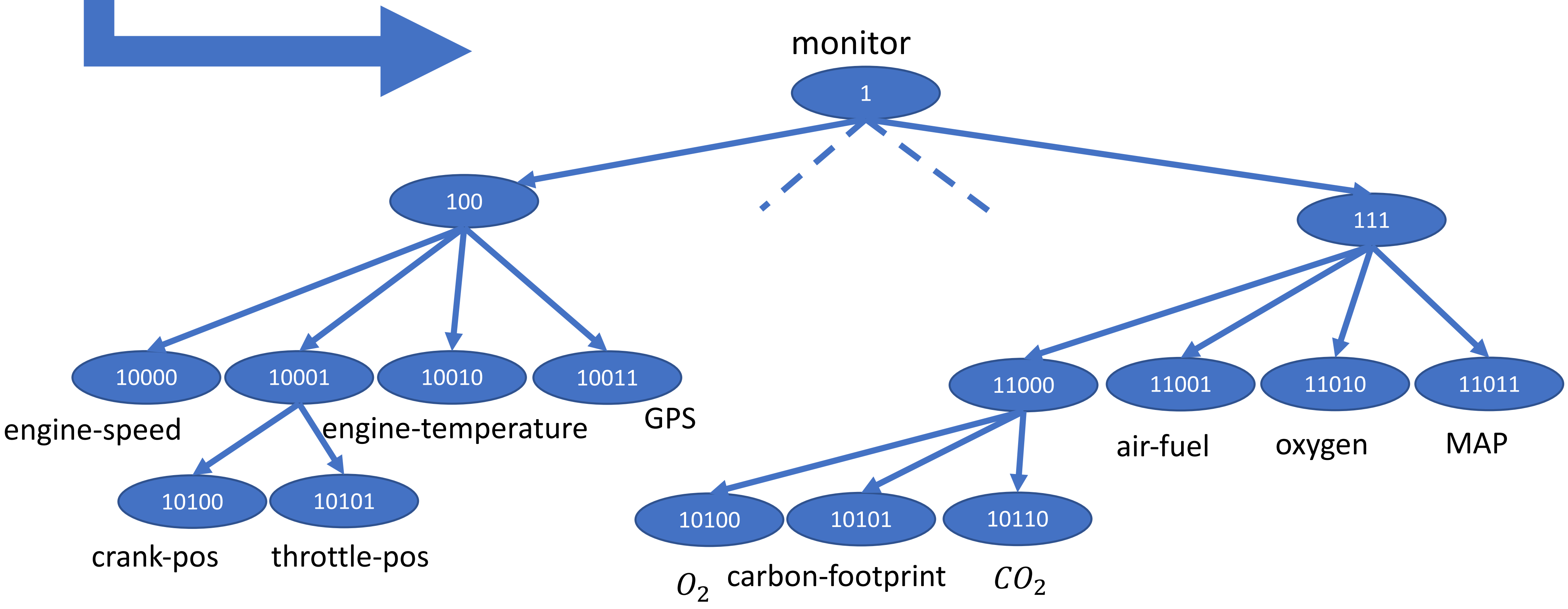


3. Assigned code based on the cluster hierarchy



$SP_{meaning}$

- Monitor sensors = {crank−pos; CO; engine−speed;MAP;throttle−pos; O_2; engine−temperature; air−fuel; oxygen; GPS;carbon−footprint; … }



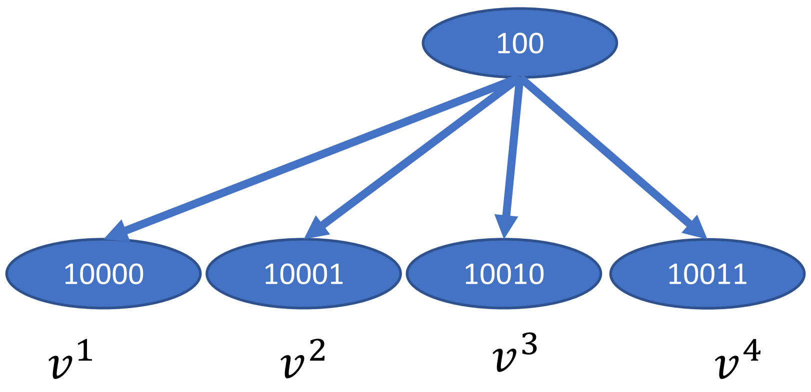
When to summarize

(v^1, v^2, v^3, v^4) – keyword set, corresponding to:

(10000, 10001,10010,10011)- Full sibling code set (FSCS)

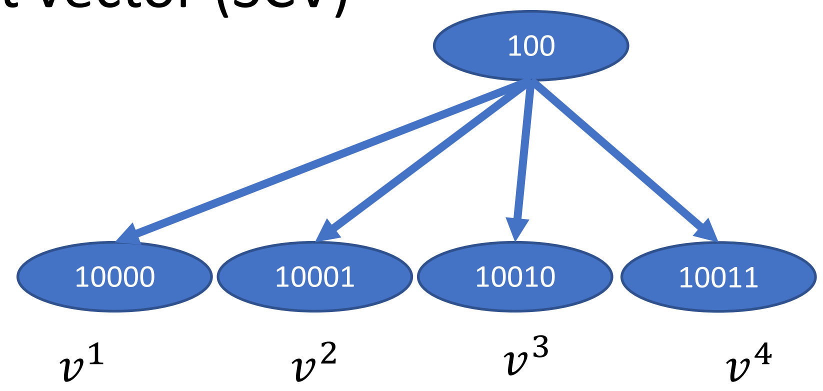
- If a RT has 4 codes in FSCS, they will be summarized into 101
- If it has only 3 codes, what happens if we still summarize?=> misleading

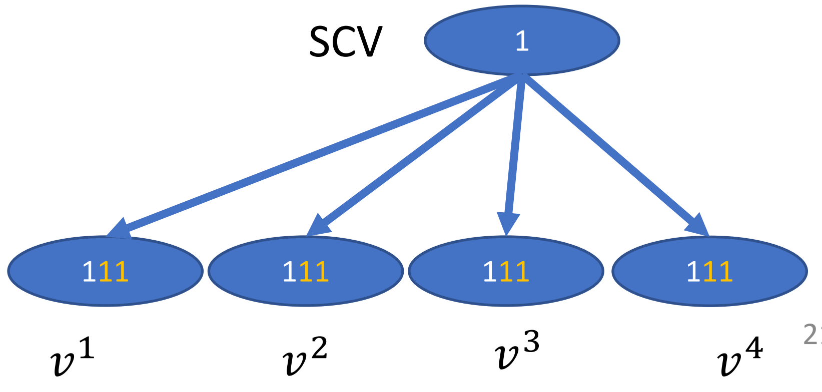
⇒ Need to know the FSCS before summarizing



When to summarize

- How to get FSCS size?
  - Average FSCS Size Estimation
    - Learn the estimation function from dictionary of Wordnet (155K keywords)
    - Derivative from tree configuration
  - FSCS Size Vector
    - Maintain the accurate FSCS size from root to current node in sum-tree: Sibling count vector (SCV)





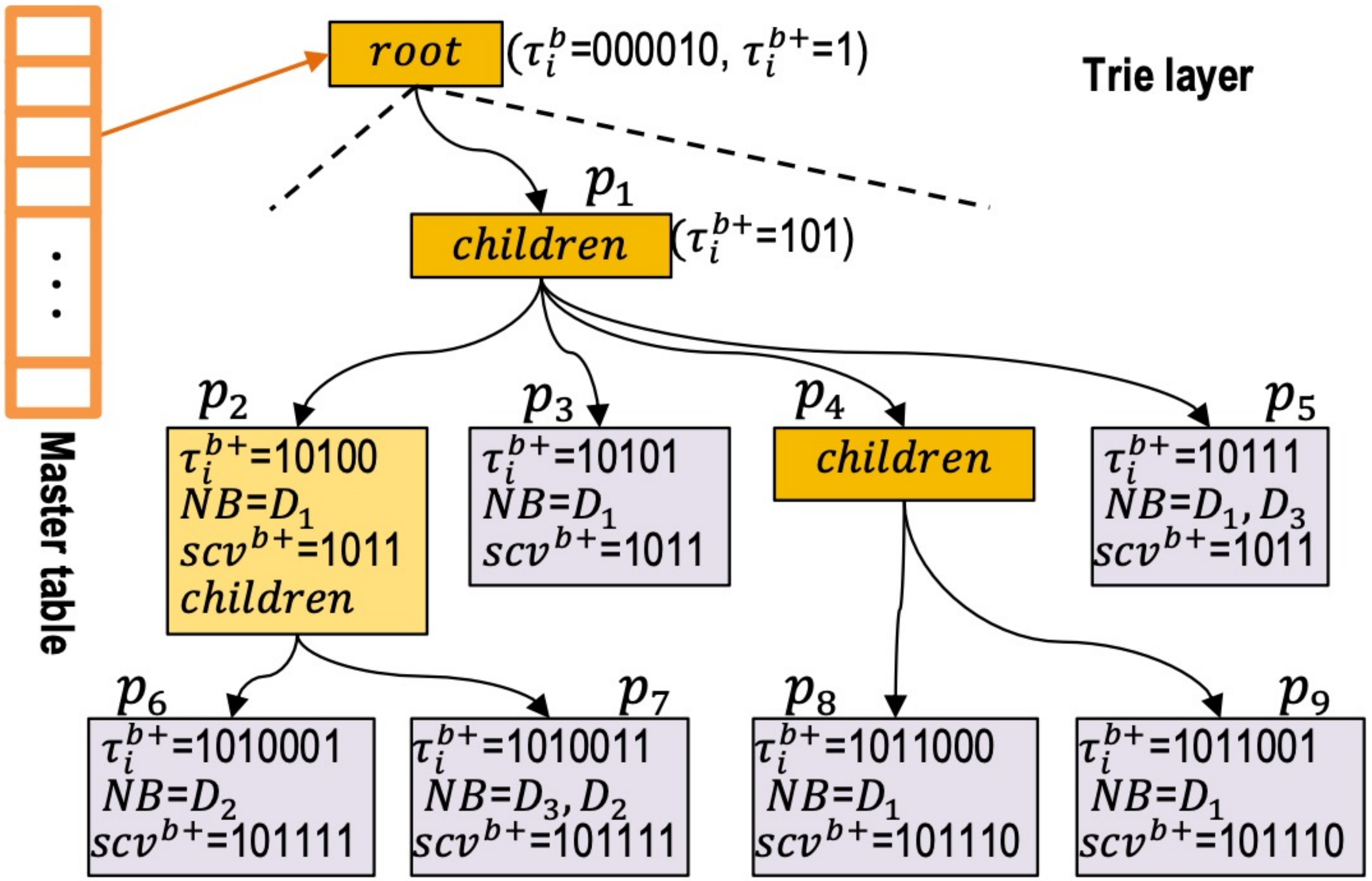
Routing table design for Summarization Tree

hybrid-TableTrie (hTT)

- Master table
- RT-Trie

Other data structures?

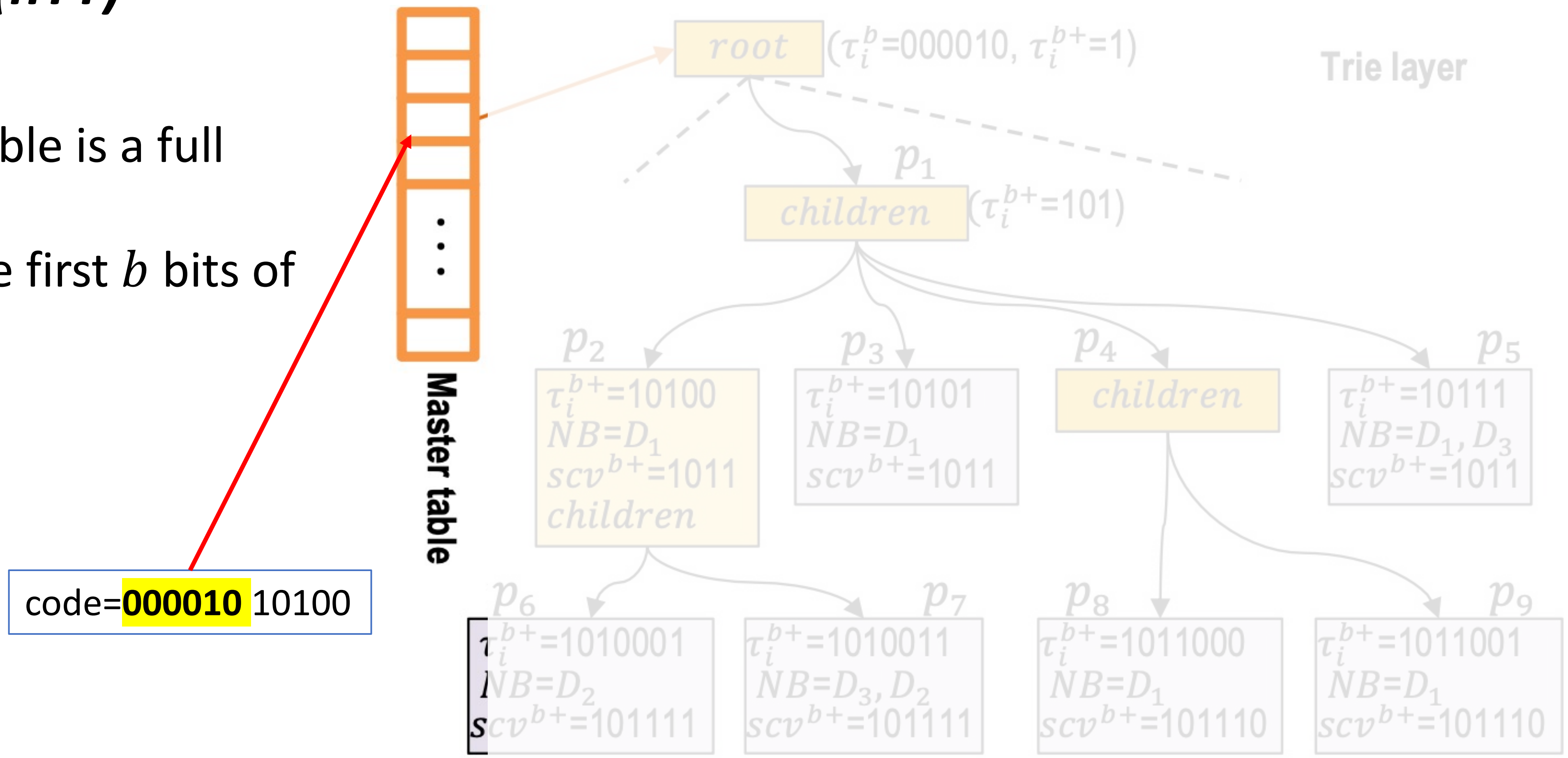
- Binary search tree
- N-ary search tree



Routing table design for Summarization Tree

hybrid-TableTrie (hTT)

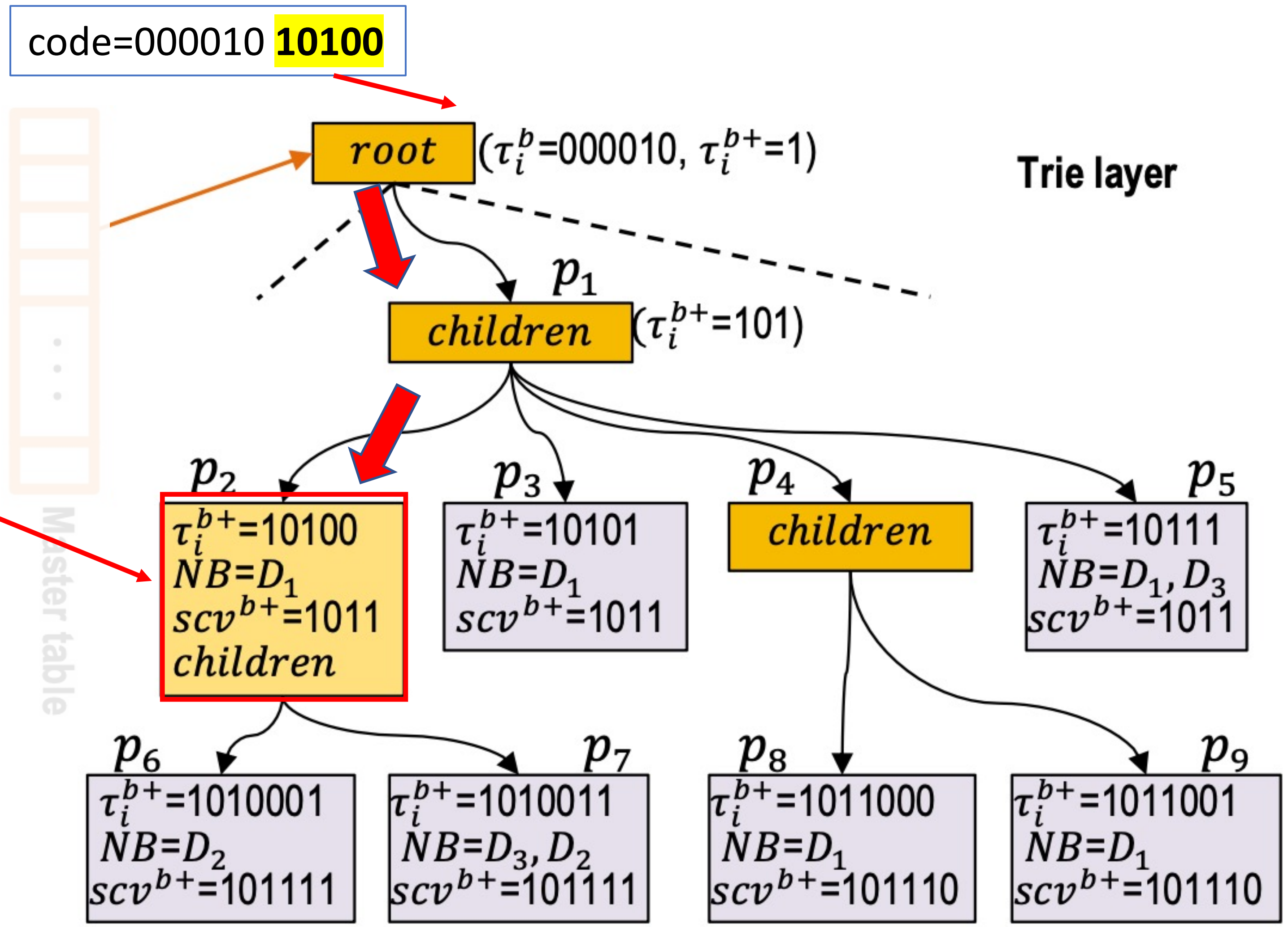
- Master table
  - The master table is a full index table
  - Each index the first b bits of tree code.
- RT-Trie



Routing table design for Summarization Tree

hybrid-TableTrie (hTT)

- Master table
- RT-Trie
  - Each node maintains **code**, **neighbors**, and **sibling count vector**



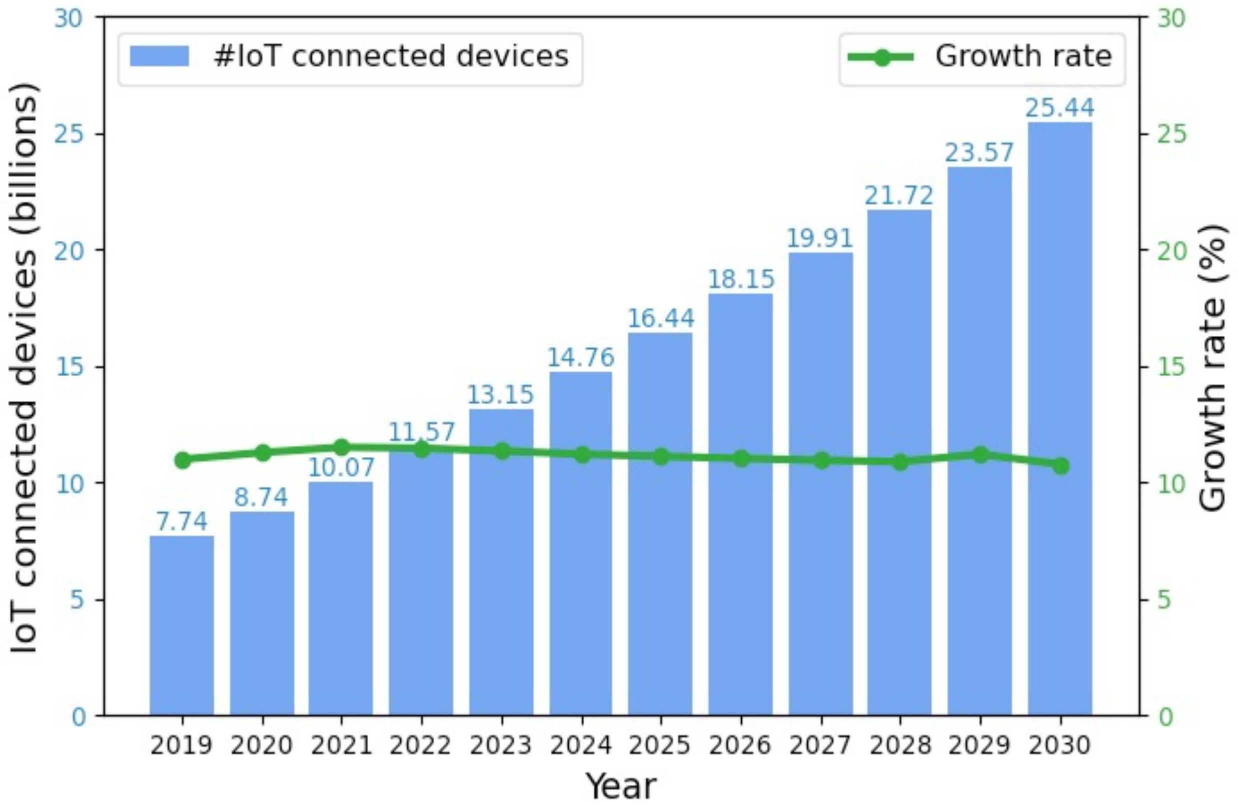
Handling IoT network growth

Growing rate ~ 11% per year

- new data streams => new codes for new descriptors
- When the number of new descriptors increases extensively, => a lot of collisions => misleading routings due to summarization coding

⇒ Need to increase code length

⇒ Need a distributed solution for reestablishment



Growth of the number of IoT devices worldwide (based on prediction) [statista, 2021]

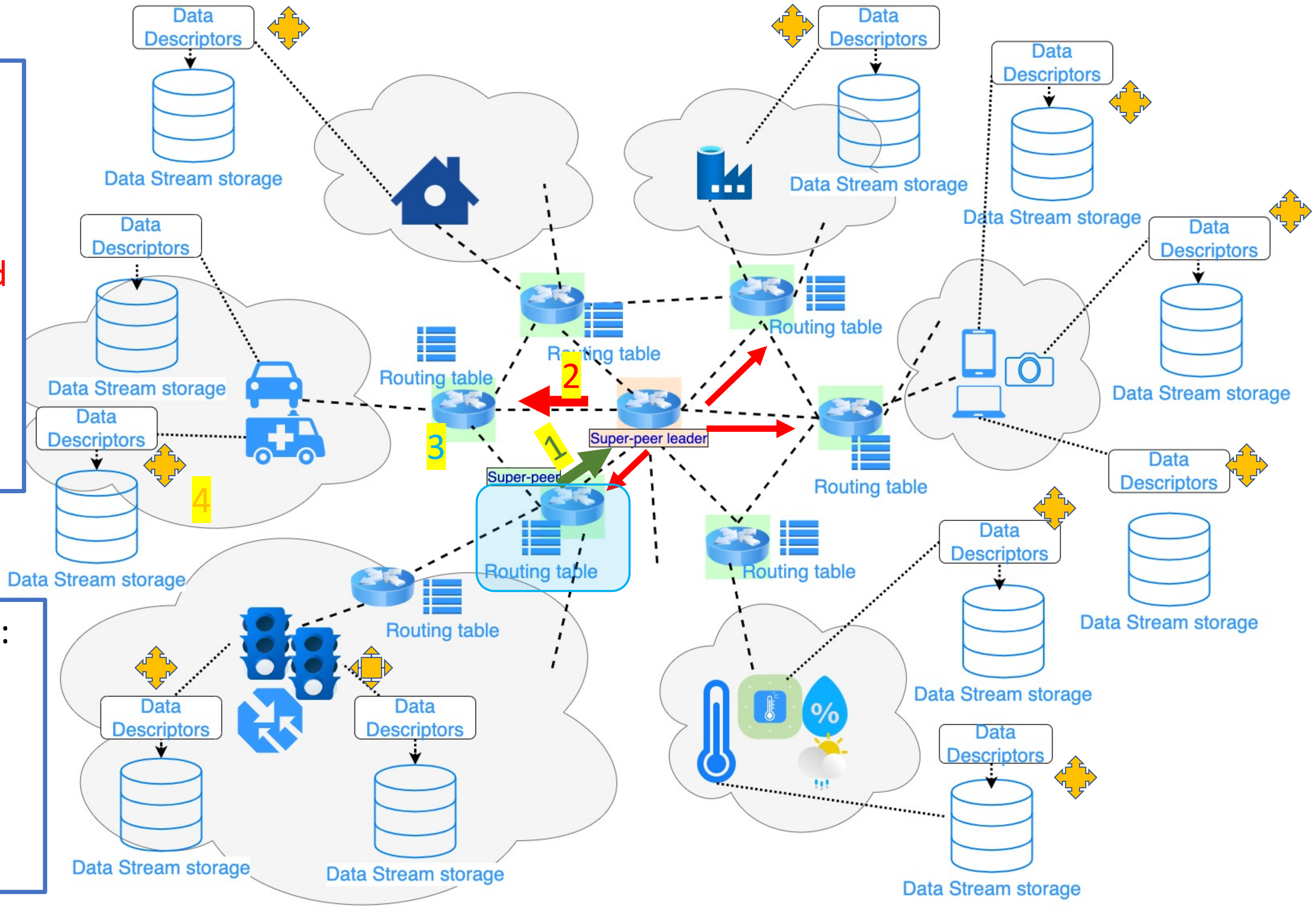
Handling IoT network growth

Reestablishment process:

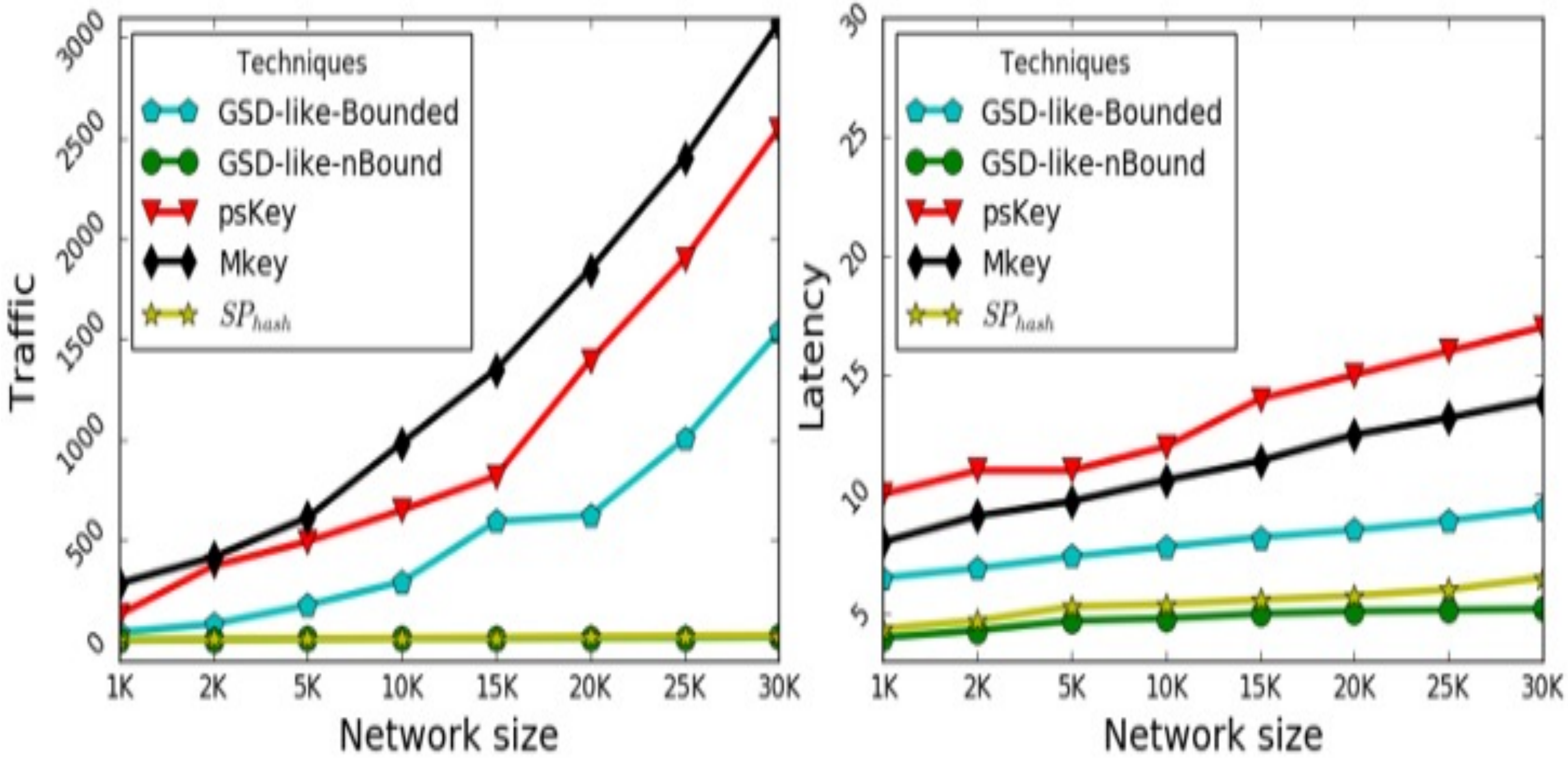
1. **Supper peer** sends a request to super-peer leader
2. **The leader** will trigger the reestablishment process and send the request to all super-peers
3. **Update config** for a larger code
4. All data streams will **re-advertise** their descriptors

Conditions to trigger reestablishment:

- #new descriptors > threshold
- #unique keyword/ hash size > threshold
- the intra-cluster distance > threshold ($ST_{meaning}$)

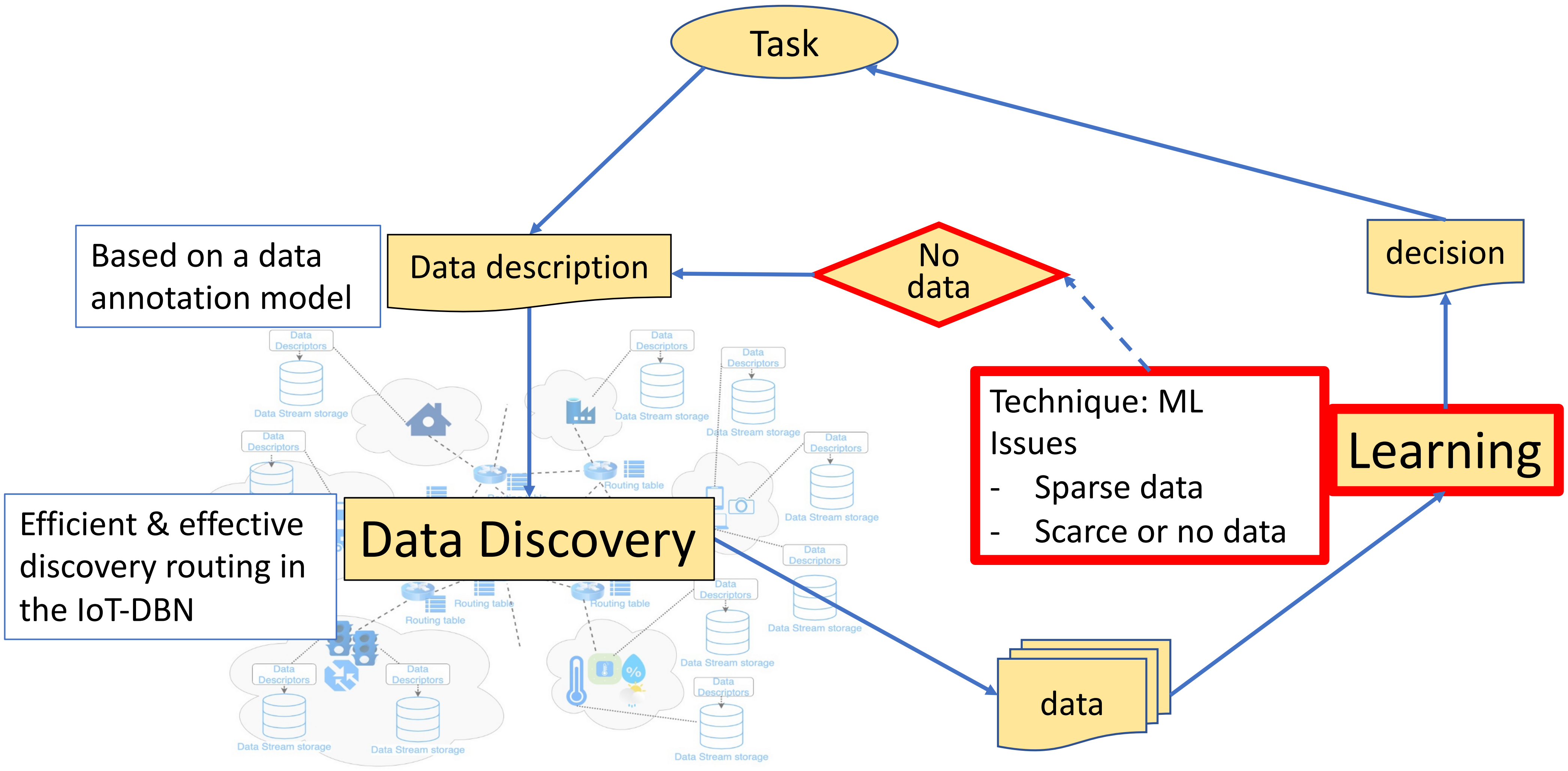


Experimental Study: Compare Data Discovery Approaches



- Reduces the RT size by **20** to **30** folds with **2-5%** increase in latency
- Outperforms DHT based approaches by **2** to **6** folds in terms of latency, traffic.

Overview of Our Approach

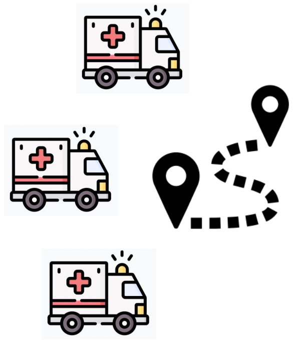


Motivating Example

Hit-and-run case

Police tracks the perpetrator

Requests the soonest ETA ambulance



- Can use Google Map, Apple Map, etc.
- **But estimators are mostly for regular cars, not for ambulances**
- IoT sensors near the incident may capture the data that are useful for identifying and tracking the perpetrator
  - E.g., smart car sensors, roadside cameras

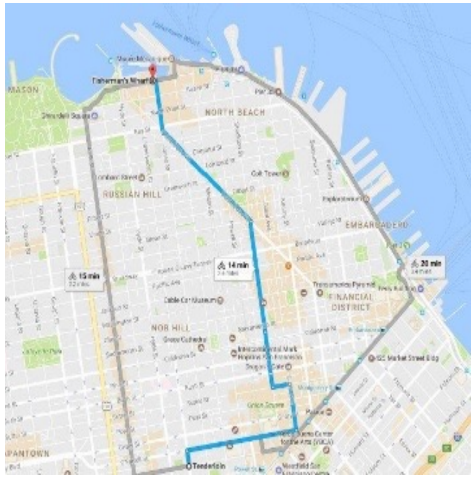
- Need to discover the relevant data
- **Need learning methods to do ETA for ambulances**

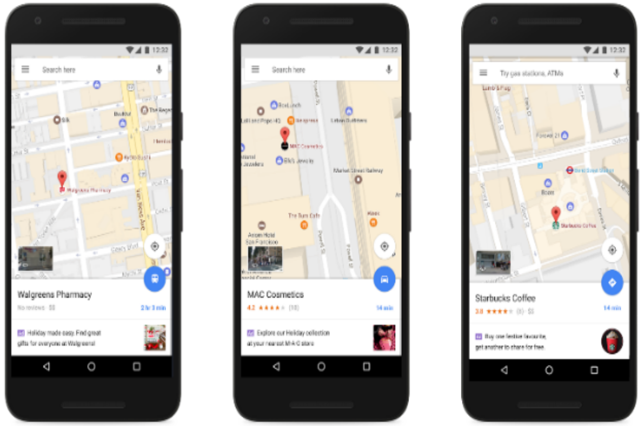
IoT Data Learning-Recap

Which ambulance to order???

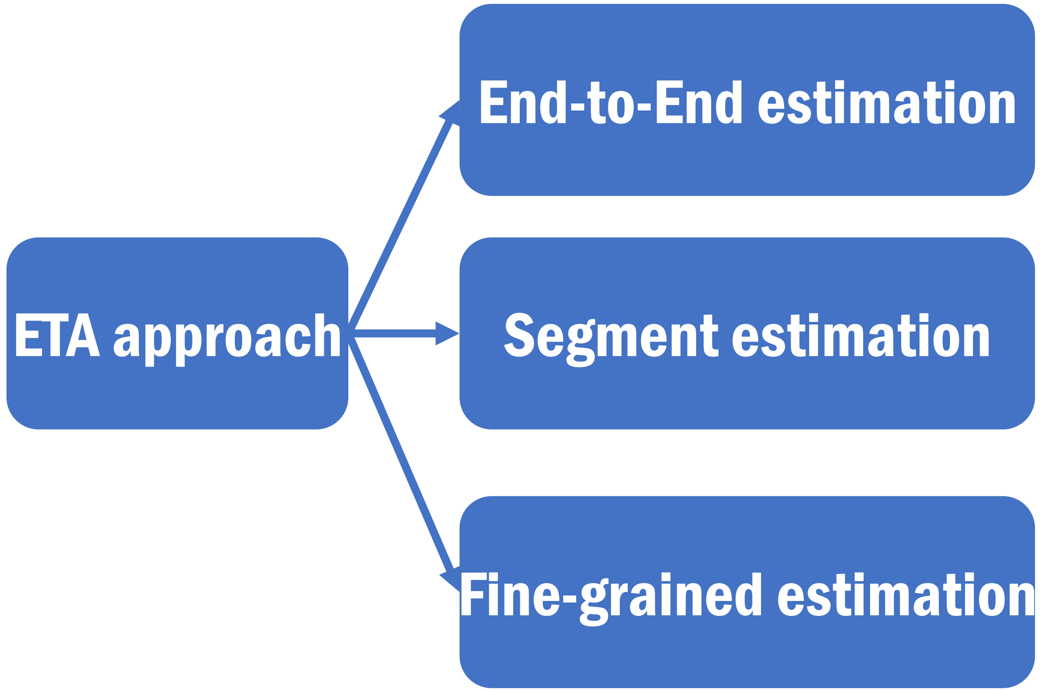
Problems to address:

- How to estimate the arrival time of ambulances at different locations efficiently?
- If the desired data is not available or not sufficient for learning:
  - Can we use transfer learning to learn the data from other data-rich resource?

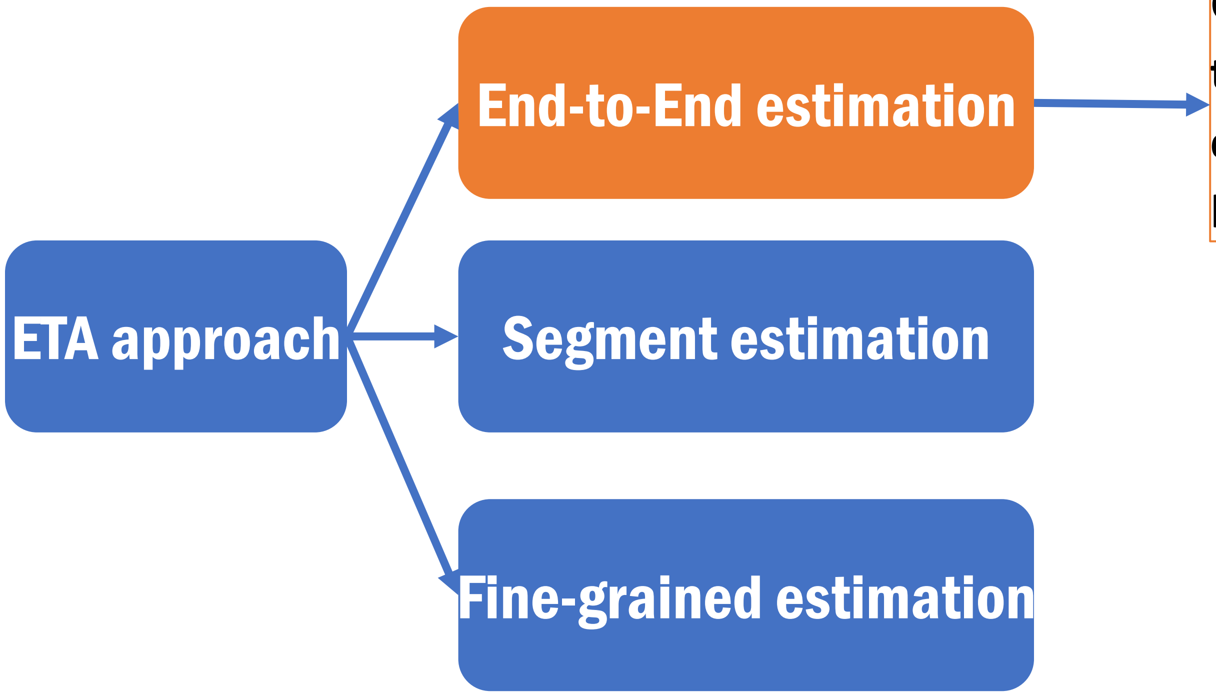




Overview of ETA solution

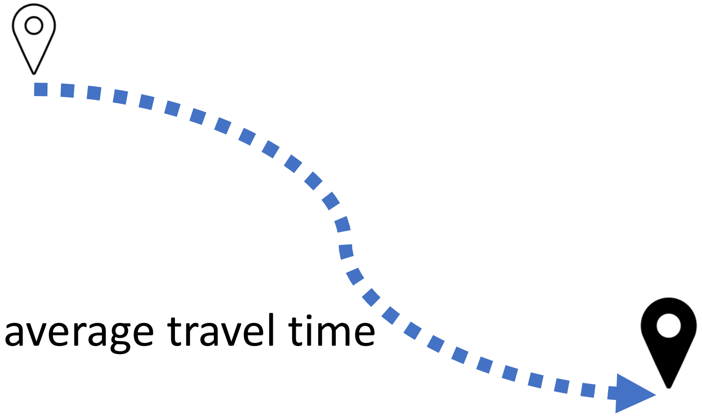


Overview of ETA solution



Calculate the **average travel time from a source to a destination** based on historical GPS trajectory data

[W.-C. Lee, et al., 2012]
[S. Maiti, et al., 2014]
[Z. Wang, et al.,2018]
[H. Wang ,et al.,2019]



Low accuracy

Overview of ETA solution

ETA approach

- End-to-End estimation
- Segment estimation
- Fine-grained estimation

Improve accuracy, but bad performance for long road segment

Partition a trajectory into several segments and obtain different representations for each segment by some embedding methods for various learning models

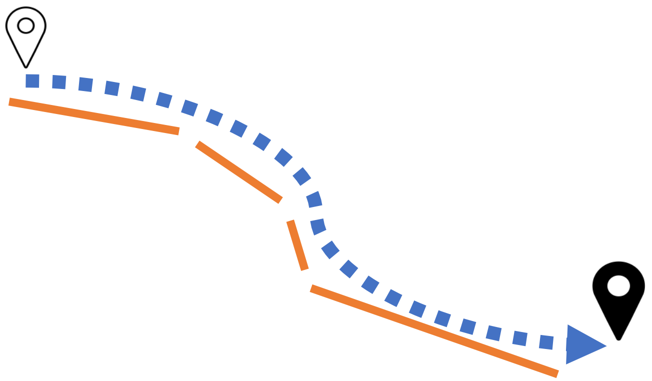
[E. Jenelius, et al., 2013]

[Y. Wang, et al., 2014]

[F. Zhang, et al.,2016]

[K.Fu, et al., 2020]

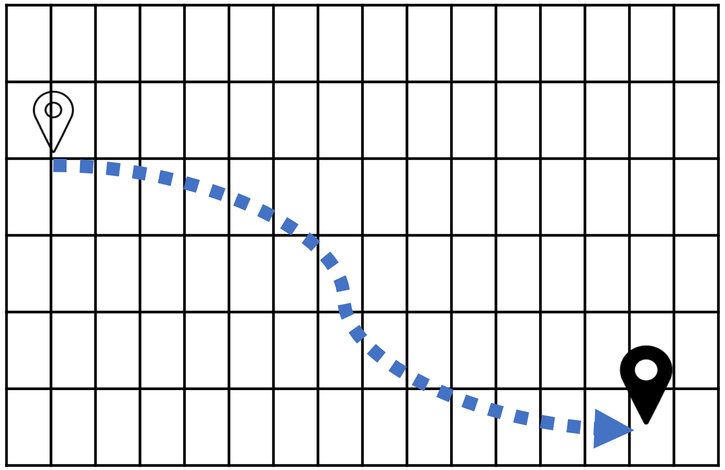
[Y. Sun, et al.,2021]



Overview of ETA solution

ETA approach

- End-to-End estimation
- Segment estimation
- Fine-grained estimation



Focus on **a fine-grained data-driven approach** that constructs models to **learn spatial-temporal knowledge** from small regions of the map where the route passes through

[B. Y. Lin, et al.,2018]

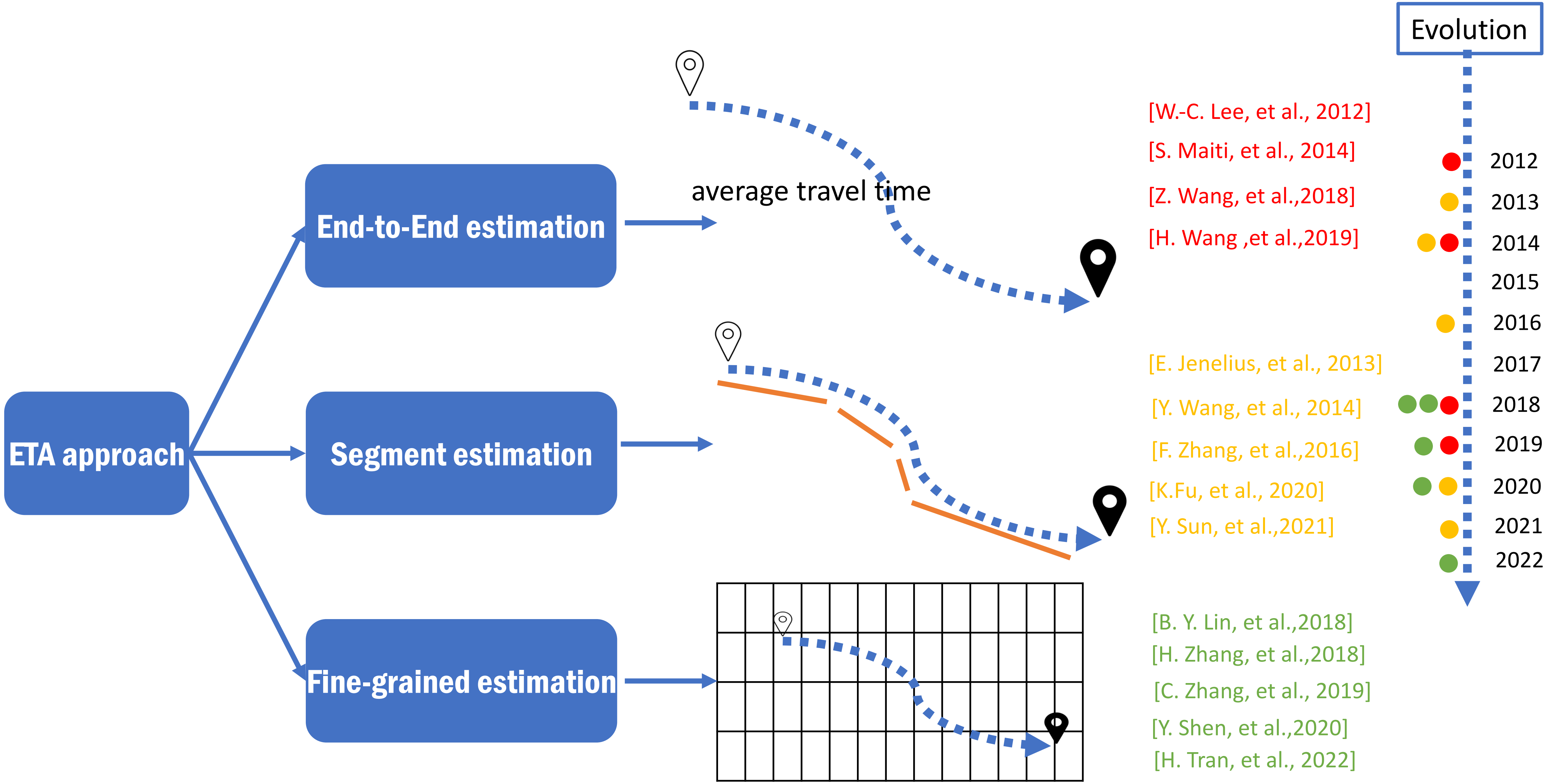
[H. Zhang, et al.,2018]

[C. Zhang, et al., 2019]

[Y. Shen, et al.,2020]

Only consider the impact of individual environmental factors without considering the integrated impact.

Overview of ETA solution

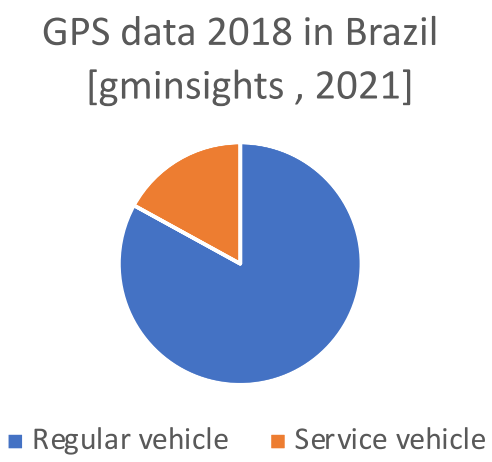


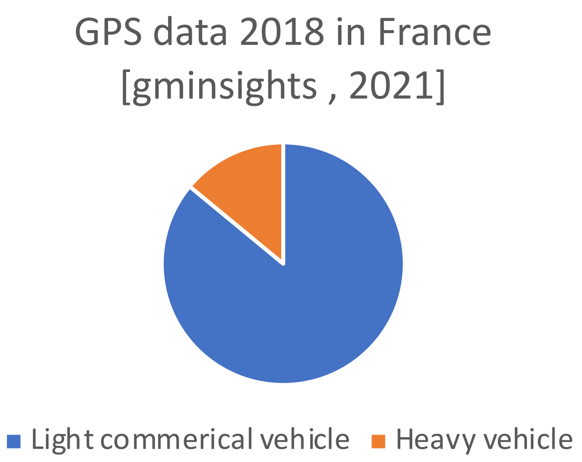
Different Driving Time Among Vehicle Types

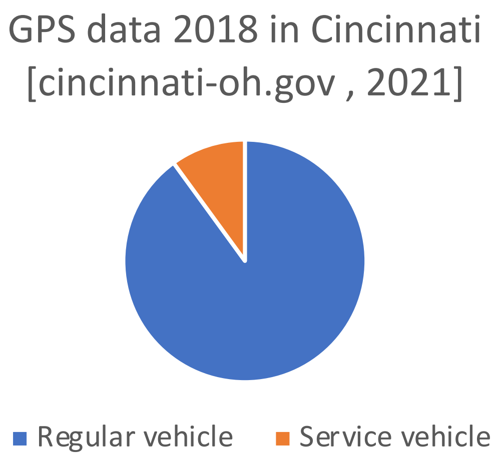
- In the same trajectory, **different type** of vehicles lead to **different driving time**
- But none of the existing approach considers this issue

Challenges

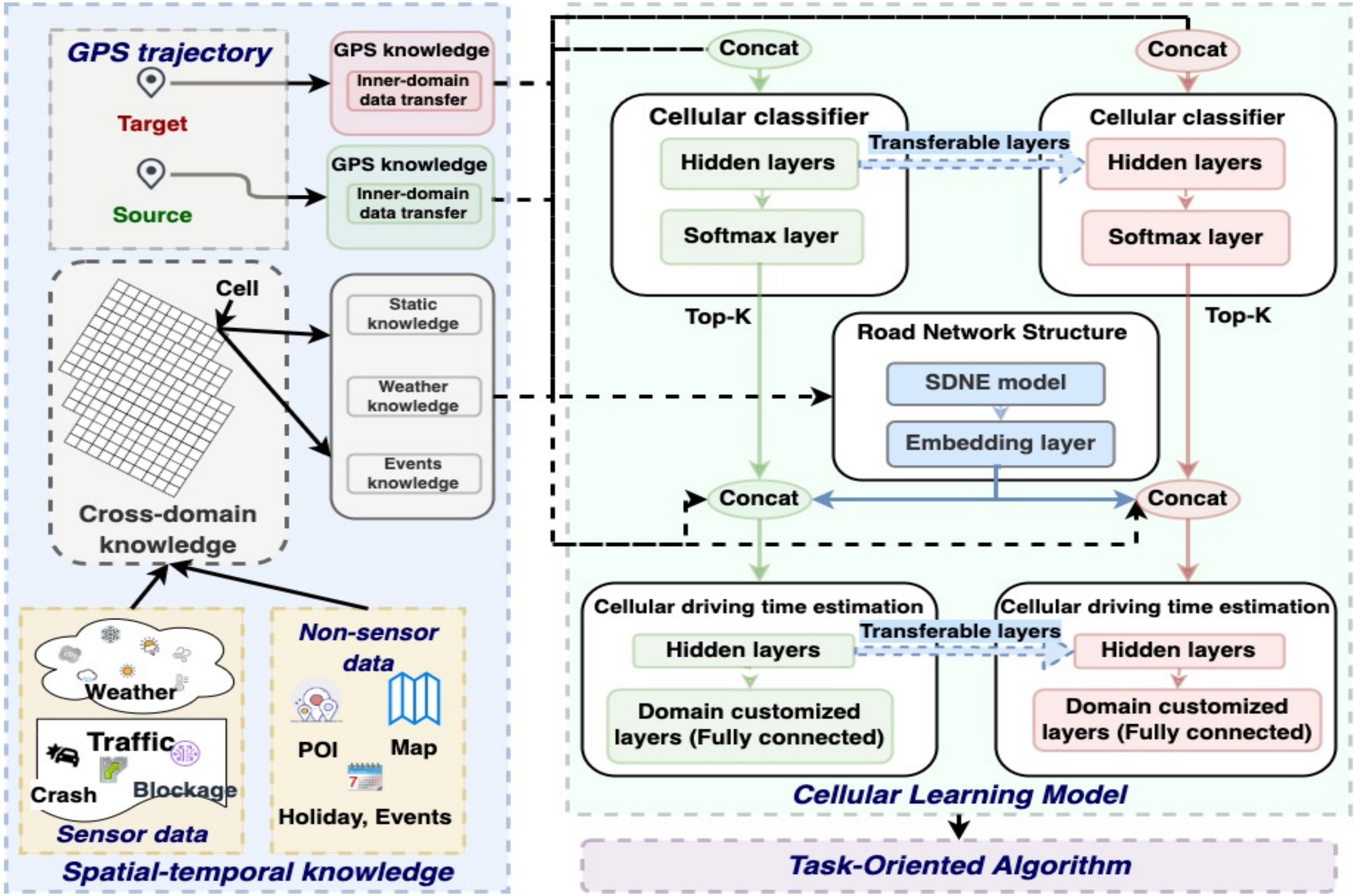
- Using one ETA prediction model for all vehicles
  ⇒ Low prediction accuracy
- Building models for each specific vehicle type (not each vehicle)
  ⇒ Potential data scarceness problem
  ⇒ Data for some special vehicles may not even be available







TLETA-Deep Transfer Learning and Integrated Cellular Knowledge for Estimated Time of Arrival Prediction

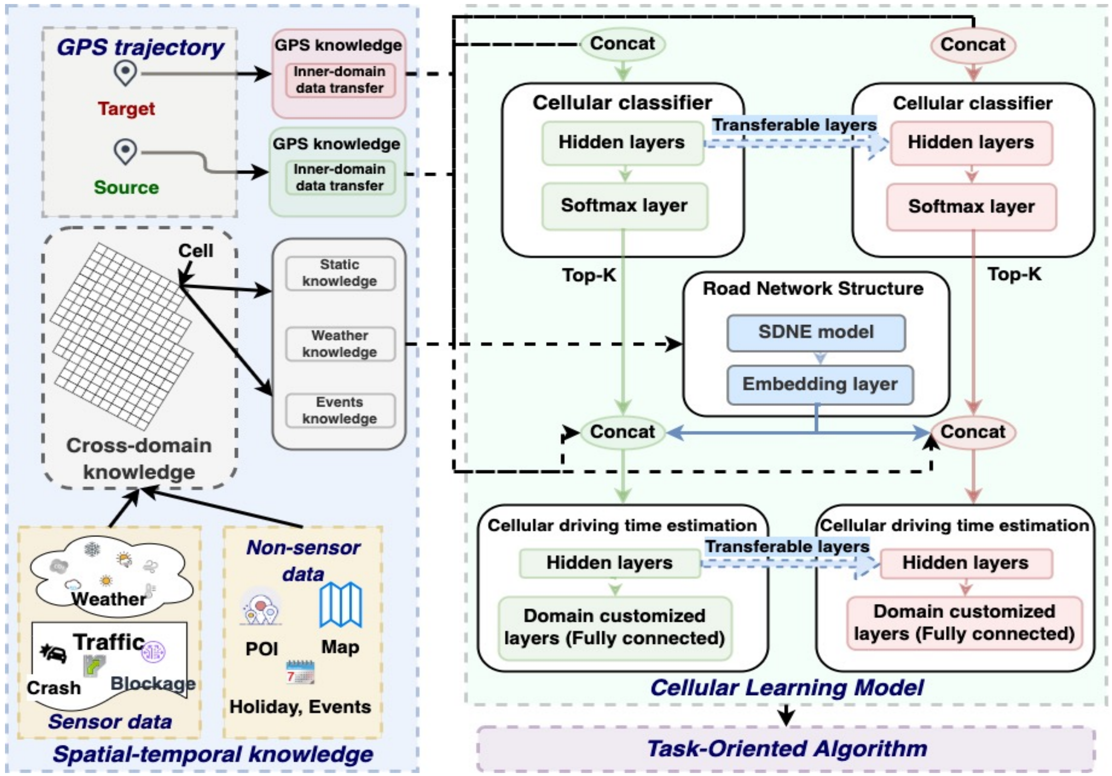


TLETA-Deep Transfer Learning and Integrated Cellular Knowledge for Estimated Time of Arrival Prediction

- **The cellular spatial-temporal knowledge**: domain-specific and cross-domain knowledge

- **The cellular learning module** learns the cellular traffic patterns and includes a classifier, a road network structure embedding scheme, and a cellular ETA algorithm.

- **The task-oriented prediction module** leverages the learned model to predict ETA of a given trajectory

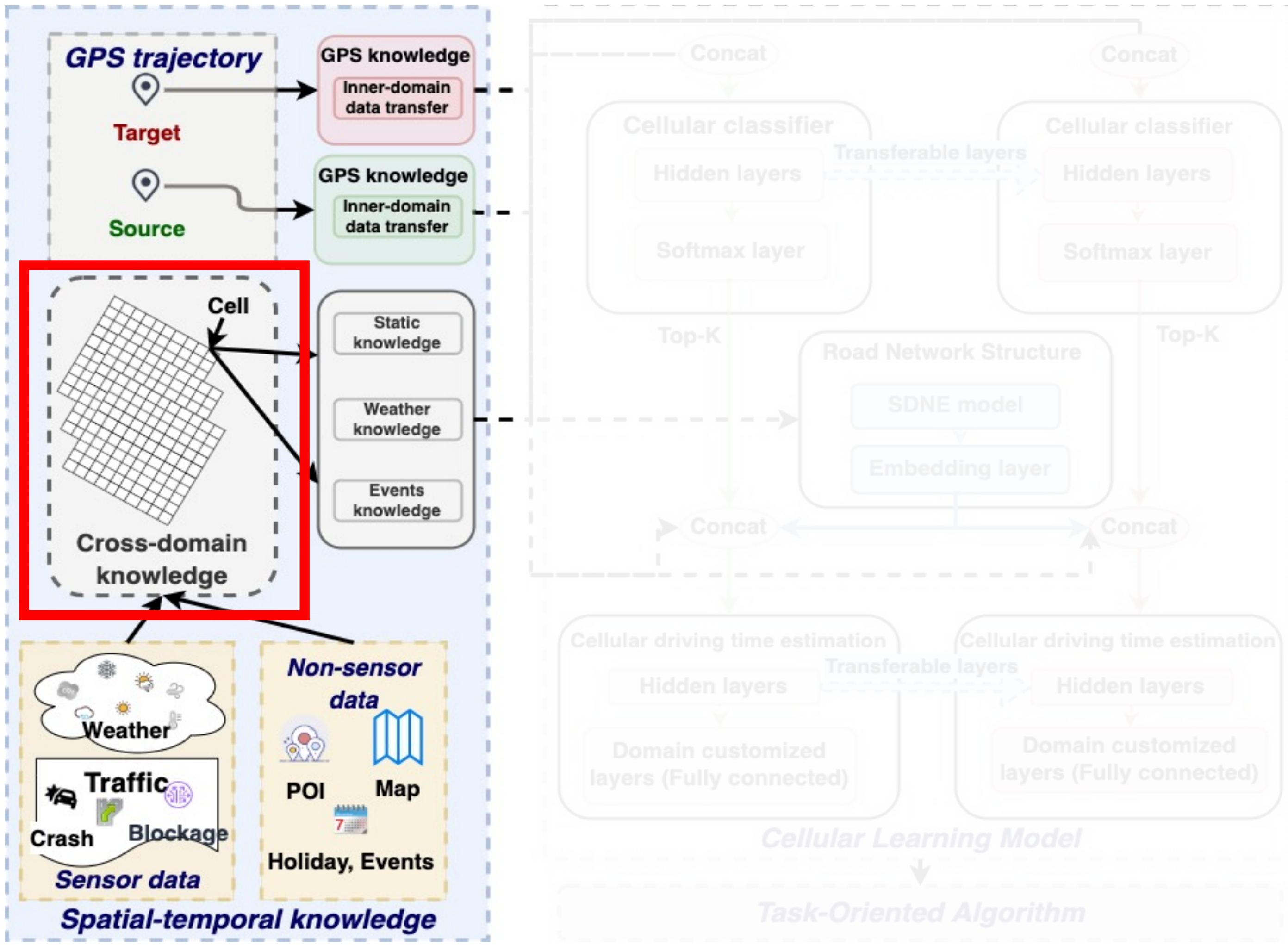


TLETA: Deep Transfer Learning and Integrated Cellular Data for Estimated Time of Arrival Prediction Architecture

TLETA-Cellular spatial-temporal knowledge

Cellular spatial-temporal knowledge

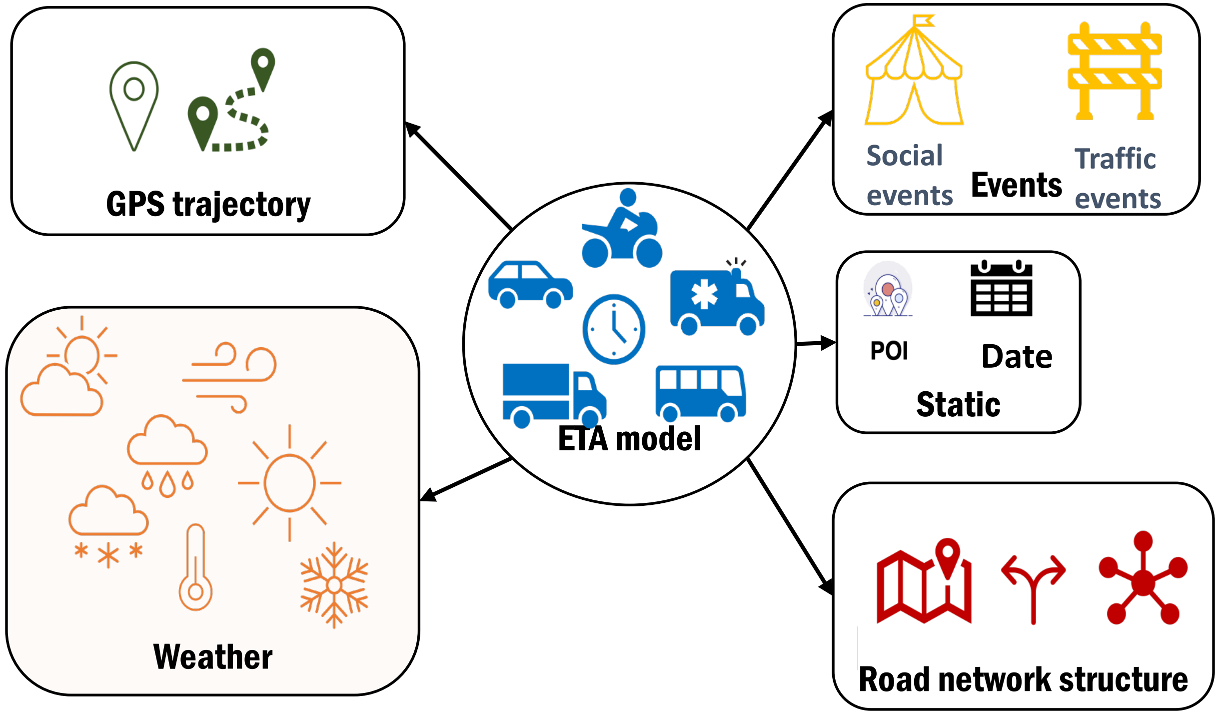
- Partition a city into grid cells

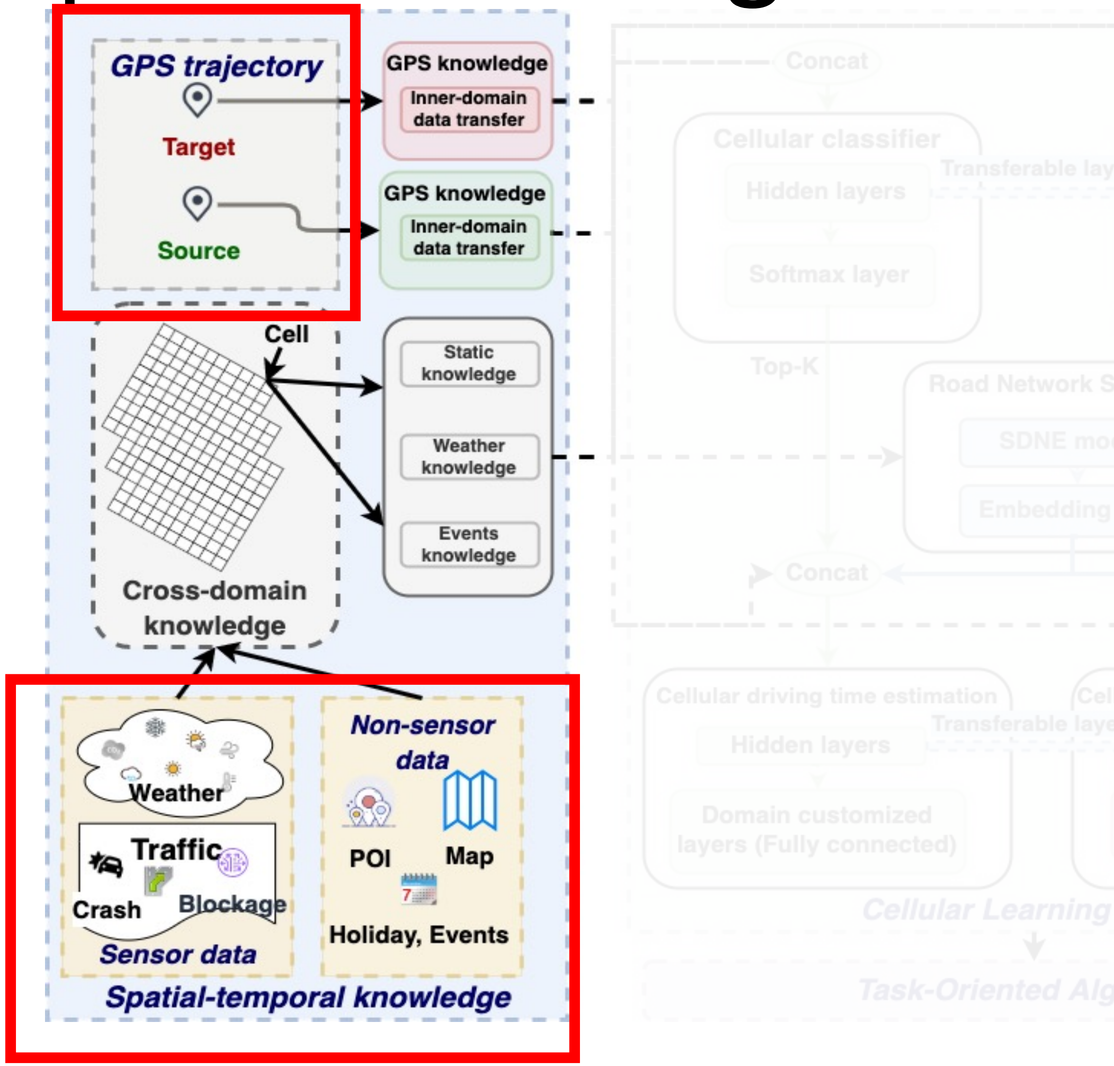


TLETA-Cellular spatial-temporal knowledge

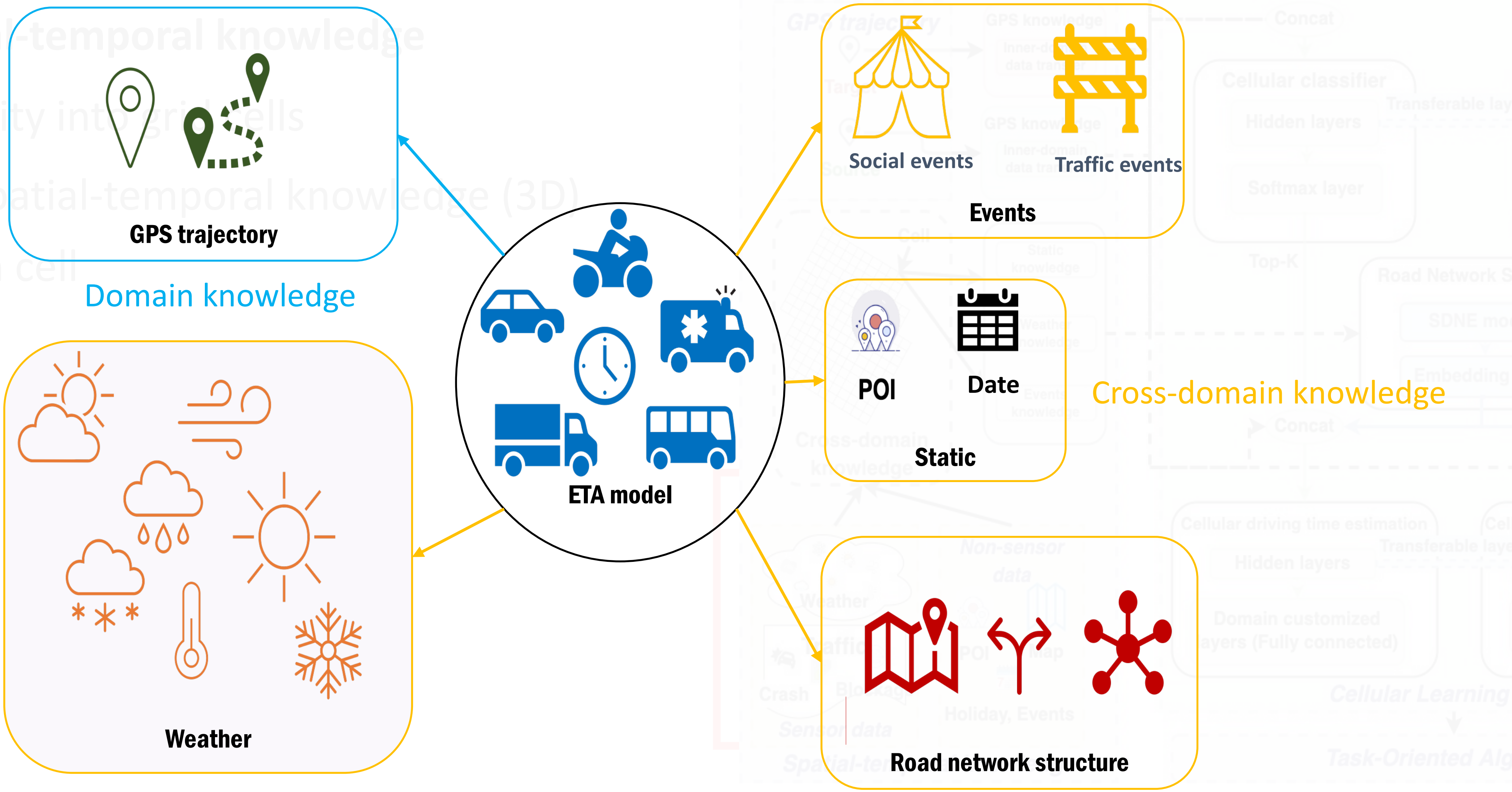
Cellular spatial-temporal knowledge

- Partition a city into grid cells
- Construct spatial-temporal knowledge (3D)
    - For each cell





TLETA-Cellular spatial-temporal knowledge



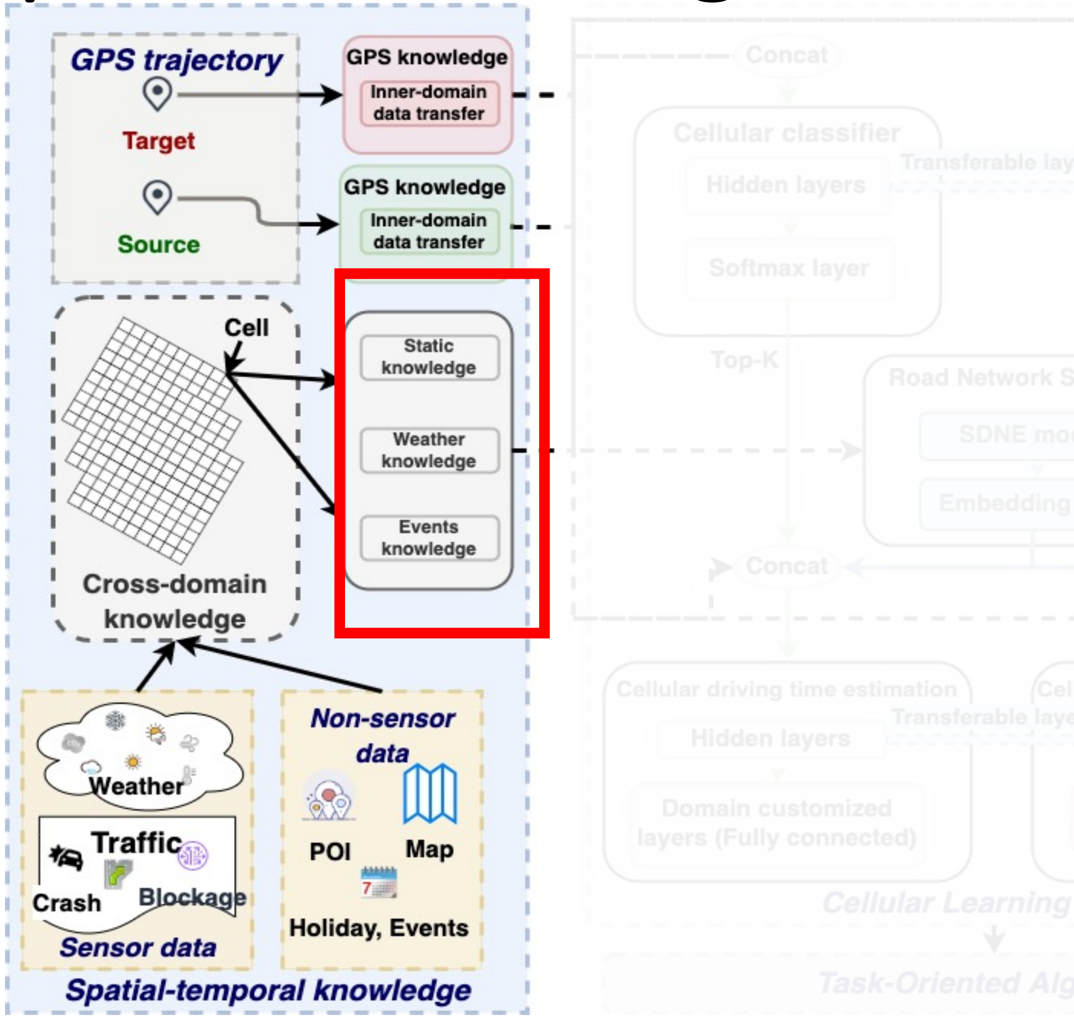
TLETA-Cellular spatial-temporal knowledge

Cellular spatial-temporal knowledge

- Partition a city into grid cells
- Construct spatial-temporal knowledge (3D)

$$\mathfrak{J}_{weather,t} = \begin{bmatrix} r^{1,1}_{weather,t} & r^{1,2}_{weather,t} & \cdots & r^{1,\mathcal{J}}_{weather,t} \\ r^{2,1}_{weather,t} & r^{2,2}_{weather,t} & \cdots & r^{2,\mathcal{J}}_{weather,t} \\ \vdots & \vdots & \ddots & \vdots \\ r^{\mathcal{I},1}_{weather,t} & r^{\mathcal{I},2}_{weather,t} & \cdots & r^{\mathcal{I},\mathcal{J}}_{weather,t} \end{bmatrix} \qquad \mathfrak{J}_{POI} = \begin{bmatrix} r^{1,1}_{POI} & r^{1,2}_{POI} & \cdots & r^{1,\mathcal{J}}_{POI} \\ r^{2,1}_{POI} & r^{2,2}_{POI} & \cdots & r^{2,\mathcal{J}}_{POI} \\ \vdots & \vdots & \ddots & \vdots \\ r^{\mathcal{I},1}_{POI} & r^{\mathcal{I},2}_{POI} & \cdots & r^{\mathcal{I},\mathcal{J}}_{POI} \end{bmatrix}$$

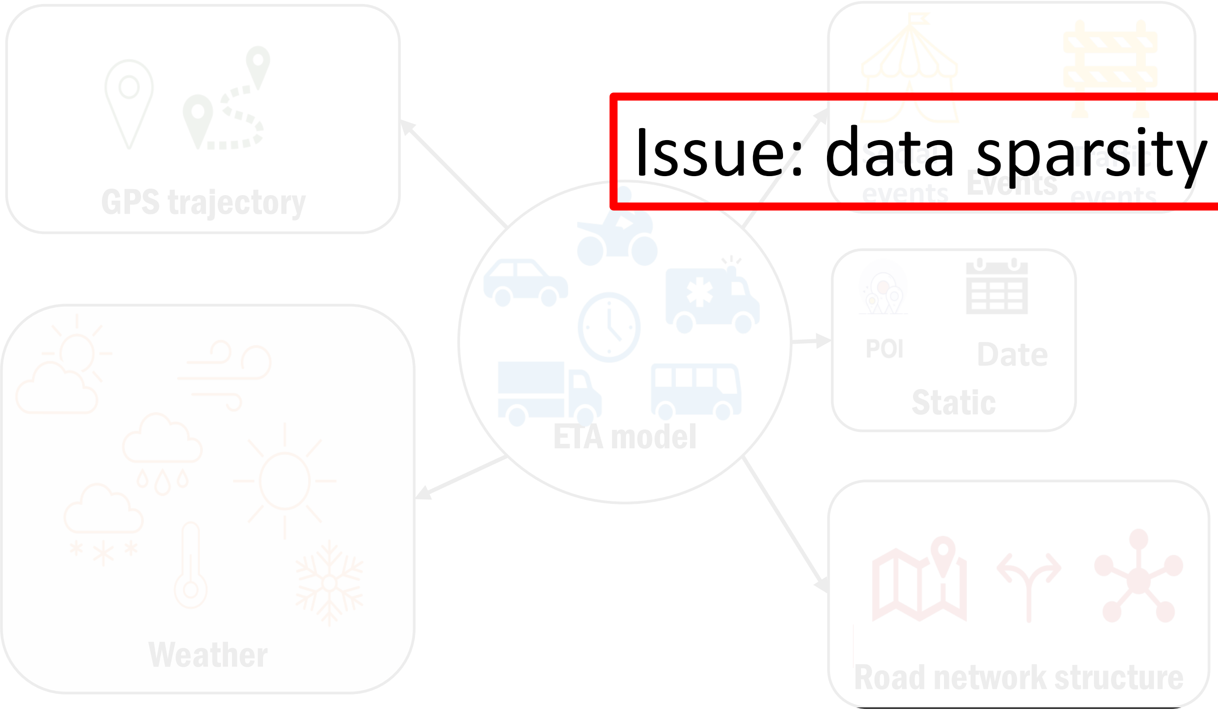
$$\mathfrak{J}_{event,t} = \begin{bmatrix} r^{1,1}_{event,t} & r^{1,2}_{event,t} & \cdots & r^{1,\mathcal{J}}_{event,t} \\ r^{2,1}_{event,t} & r^{2,2}_{event,t} & \cdots & r^{2,\mathcal{J}}_{event,t} \\ \vdots & \vdots & \ddots & \vdots \\ r^{\mathcal{I},1}_{event,t} & r^{\mathcal{I},2}_{event,t} & \cdots & r^{\mathcal{I},\mathcal{J}}_{event,t} \end{bmatrix} \qquad \mathfrak{J}_{GPS,t} = \begin{bmatrix} r^{1,1}_{GPS,t} & r^{1,2}_{GPS,t} & \cdots & r^{1,\mathcal{J}}_{GPS,t} \\ r^{2,1}_{GPS,t} & r^{2,2}_{GPS,t} & \cdots & r^{2,\mathcal{J}}_{GPS,t} \\ \vdots & \vdots & \ddots & \vdots \\ r^{\mathcal{I},1}_{GPS,t} & r^{\mathcal{I},2}_{GPS,t} & \cdots & r^{\mathcal{I},\mathcal{J}}_{GPS,t} \end{bmatrix}$$



TLETA-Cellular spatial-temporal knowledge

Cellular spatial-temporal knowledge

- Partition a city into grid cells
- Construct spatial-temporal knowledge (3D)
  - For each cell



Issue: data sparsity due to fine-grained knowledge

Handling Data Sparseness in ETA

Handling Data data sparsity in ETA

- Collaborative filtering
  [S. Wang,et al.,2017]
- Training intensity modification
  [Y. Sun, et al.,2020]
- Irregular convolution
  [B. Du, et al., 2019]

- Computationally expensive for spatial-temporal knowledge

Handling Data Sparseness in ETA

Handling Data data sparsity in ETA

Collaborative filtering

[S. Wang,et al.,2017]

Training intensity modification

[Y. Sun, et al.,2020]

Irregular convolution

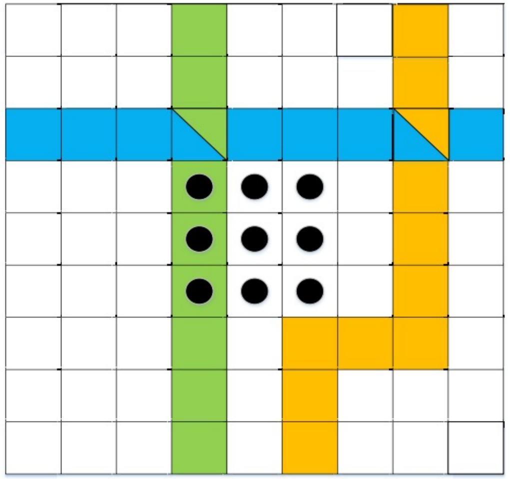
[B. Du, et al., 2019]

- Increase the training intensity of data-sparse cells under the guidance of data-rich cells
- Computationally expensive for spatial-temporal knowledge as needed to address from which cells for guidance

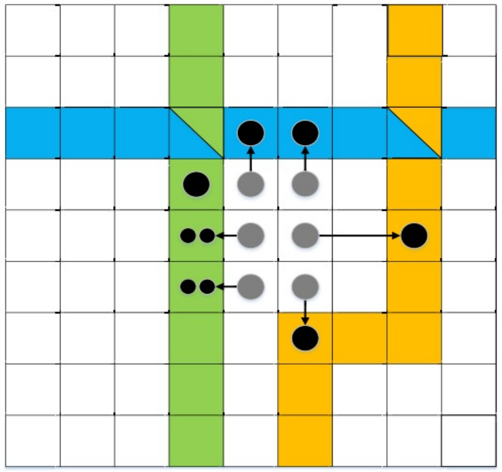
Handling Data Sparseness in ETA

Handling Data data sparsity in ETA

- Collaborative filtering
  [S. Wang,et al.,2017]
- Training intensity modification
  [Y. Sun, et al.,2020]
- Irregular convolution
  [B. Du, et al., 2019]



Regular kernel in CNN

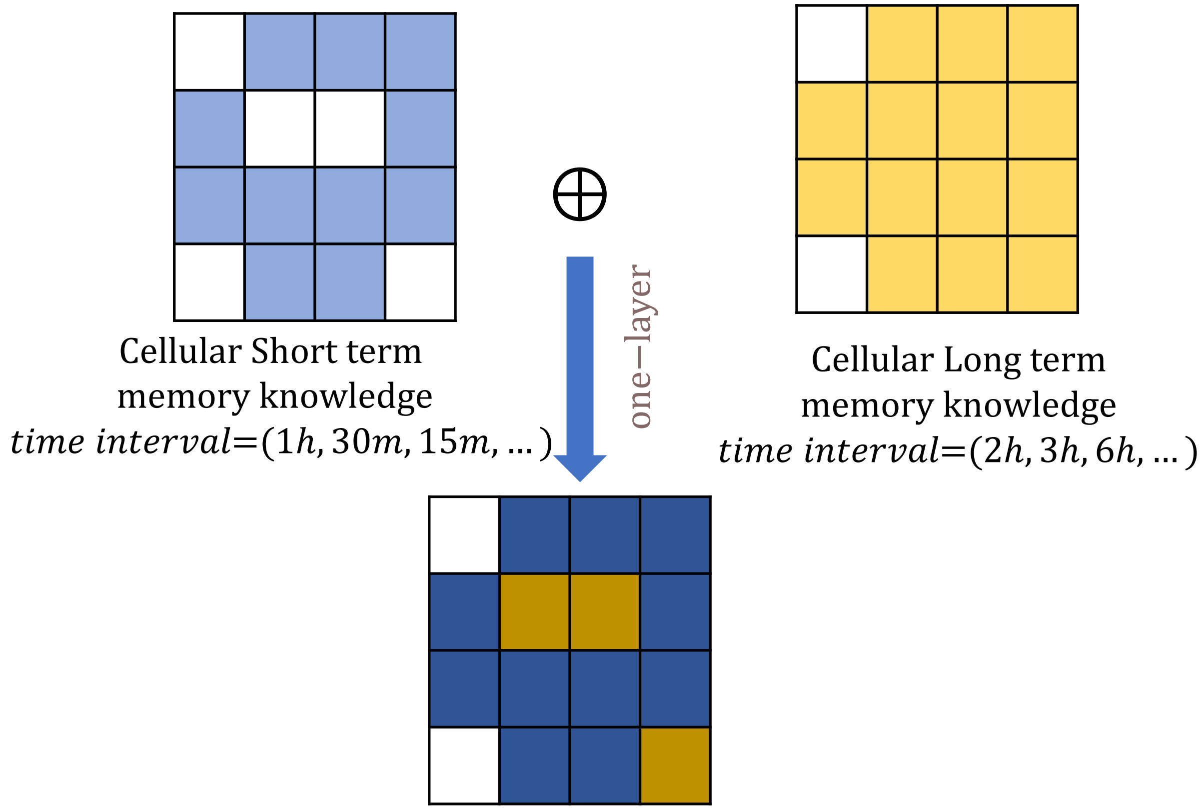


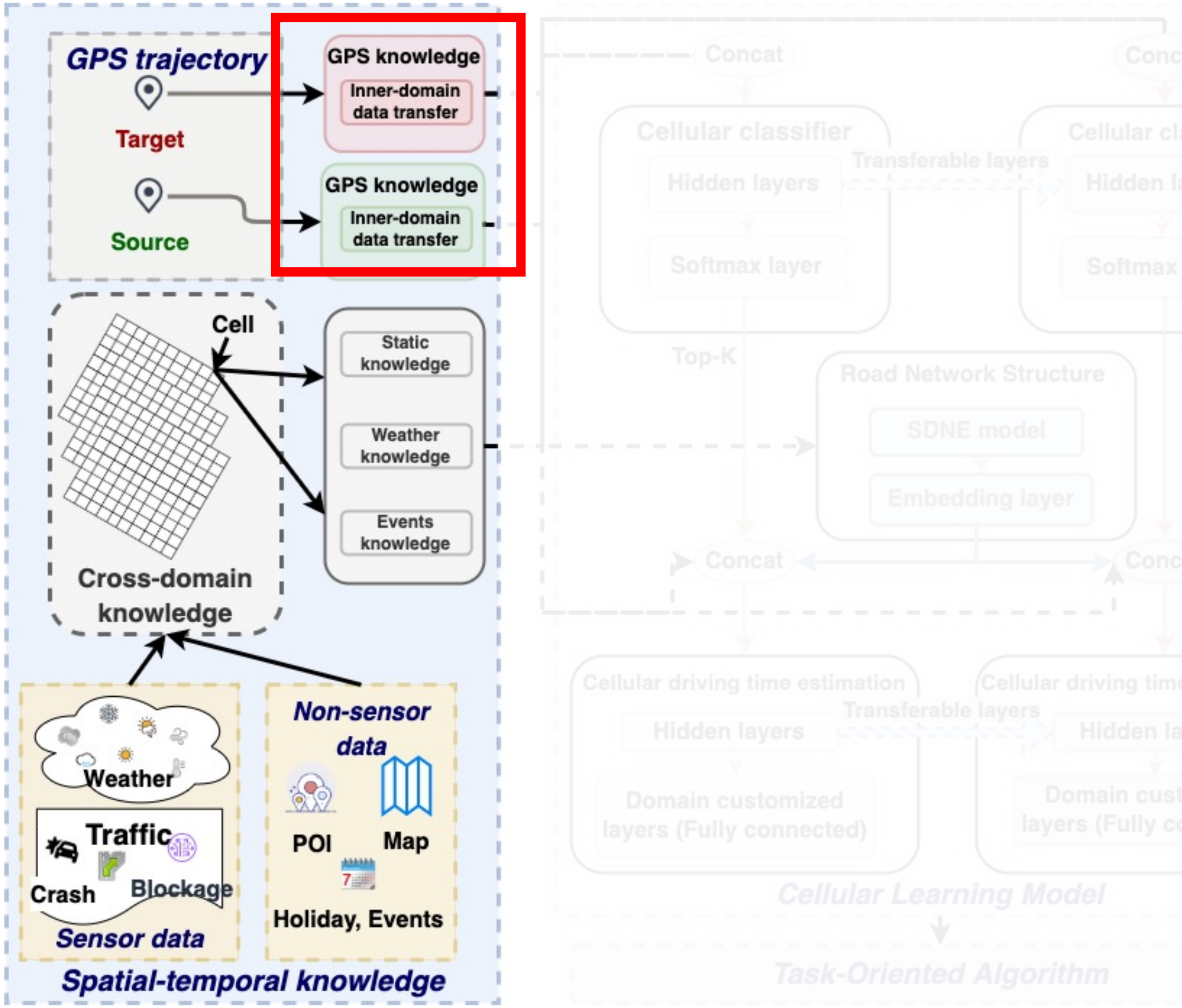
Irregular kernel (proposed)

- If the nearest neighbor cells are too far, cells may have different traffic properties

Inner-domain data interpolation learning

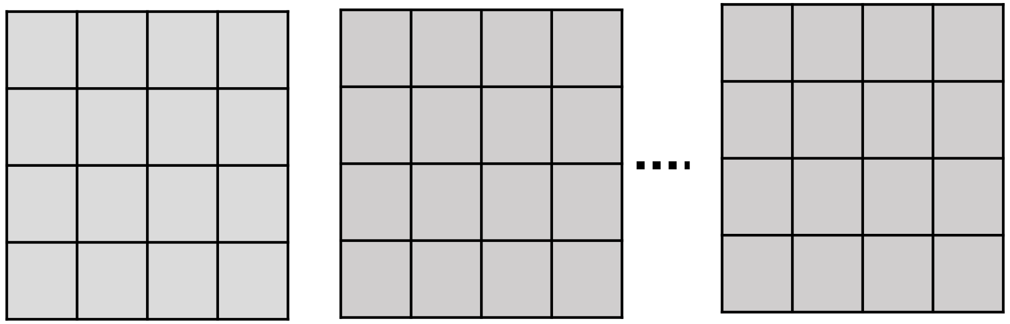
Handling sparse GPS knowledge



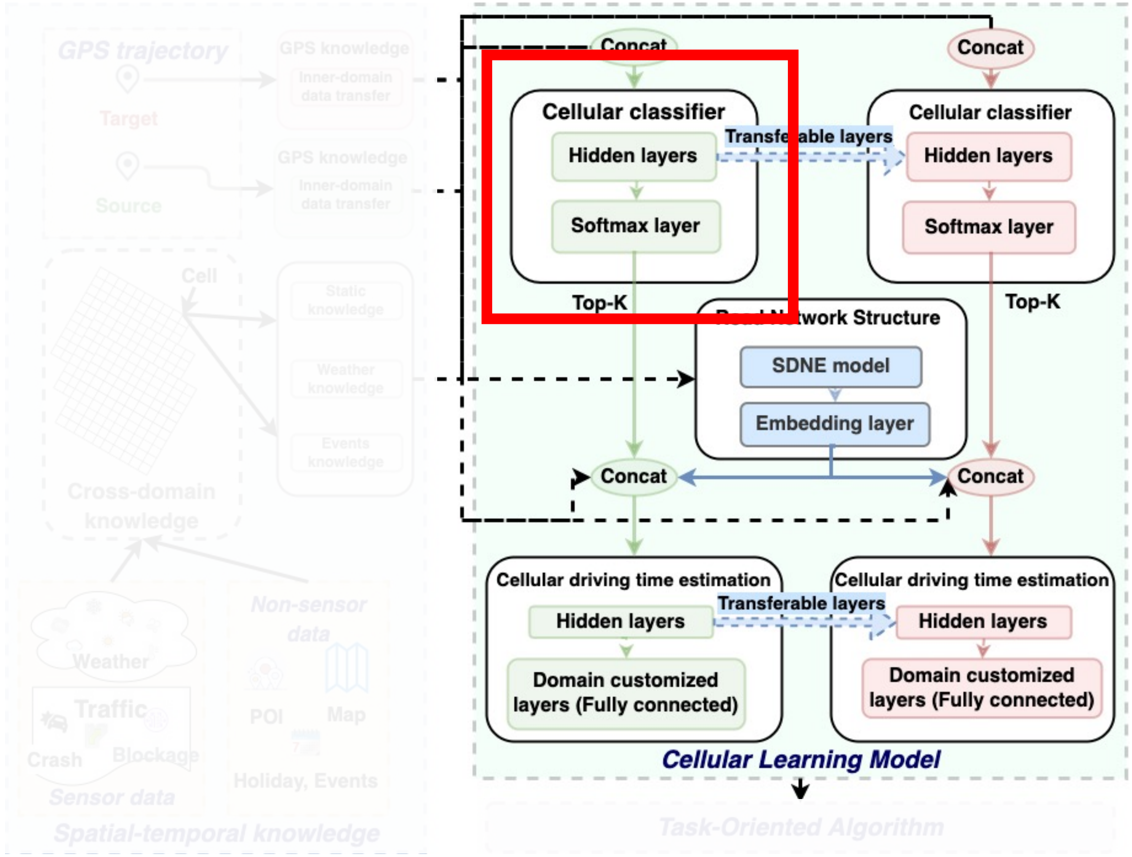


TLETA- Cellular Learning Model

- Classify cellular grids into $\mathcal{N}$ different categories of traffic levels based on driving pattern (average speed and other cellular knowledge) via neural network classification

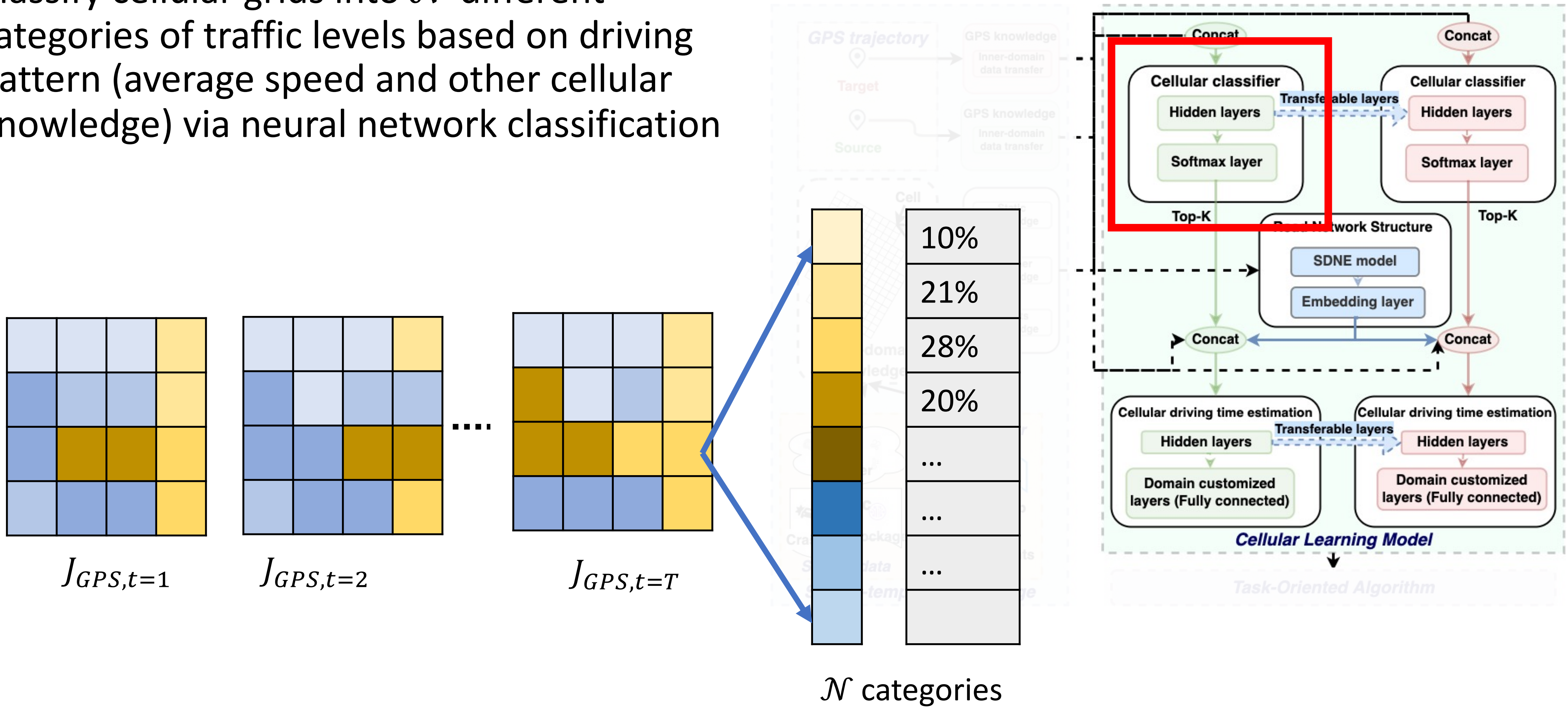


$J_{GPS,t=1}$ $J_{GPS,t=2}$ $J_{GPS,t=T}$



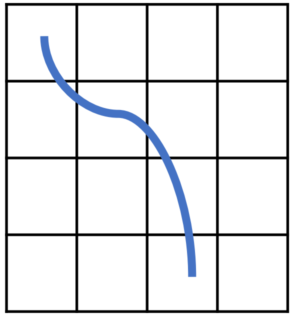
TLETA- Cellular Learning Model

- Classify cellular grids into $\mathcal{N}$ different categories of traffic levels based on driving pattern (average speed and other cellular knowledge) via neural network classification

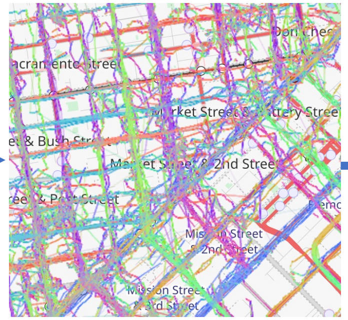


TLETA-SDNE

- Convert each GPS trajectory to list of cells to build road network
- Construct embedding road network via *first-order proximity*

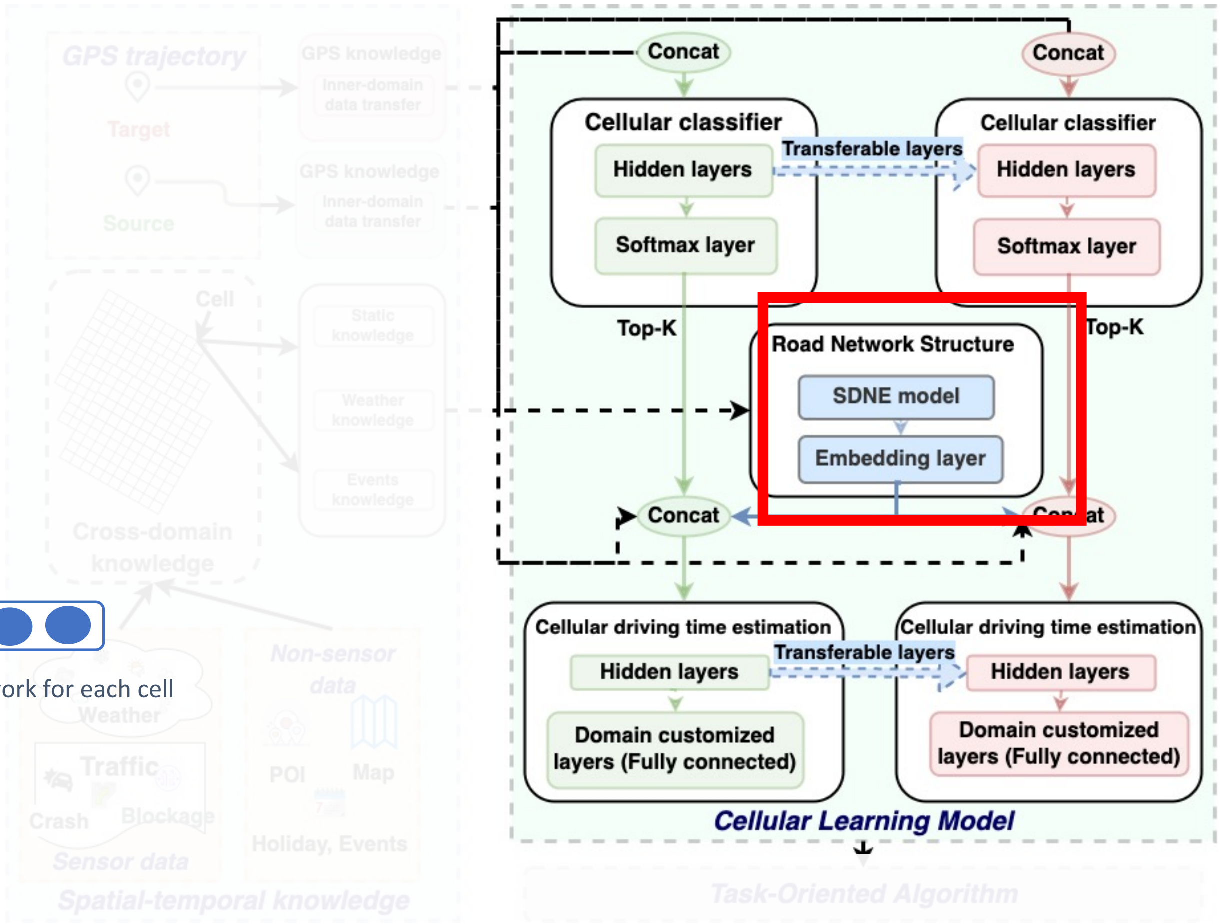


One trajectory



Multiple trajectories

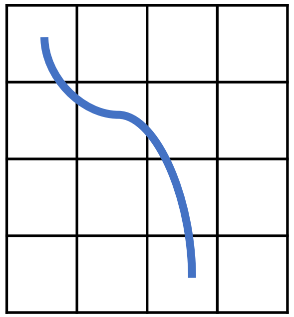
Embedding road network for each cell



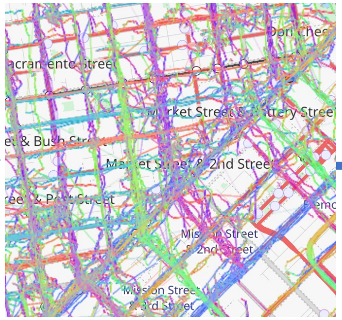
TLETA-SDNE

- Semi-supervised deep model for structural deep network embedding (SDNE) learn the embedding knowledge.

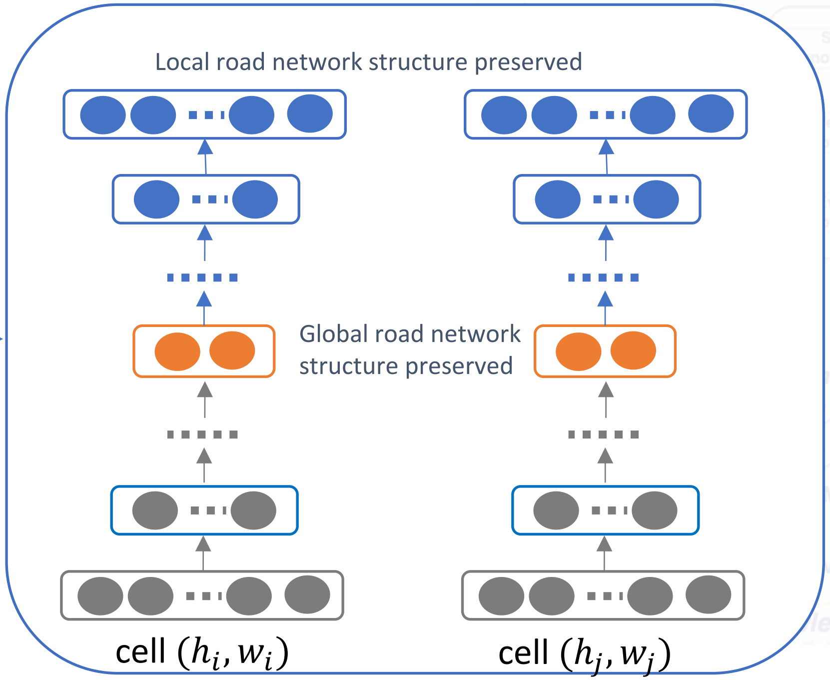
➢ Obtain the embedding vector for each cell that preserves the **local** and **global** structure of the road network.



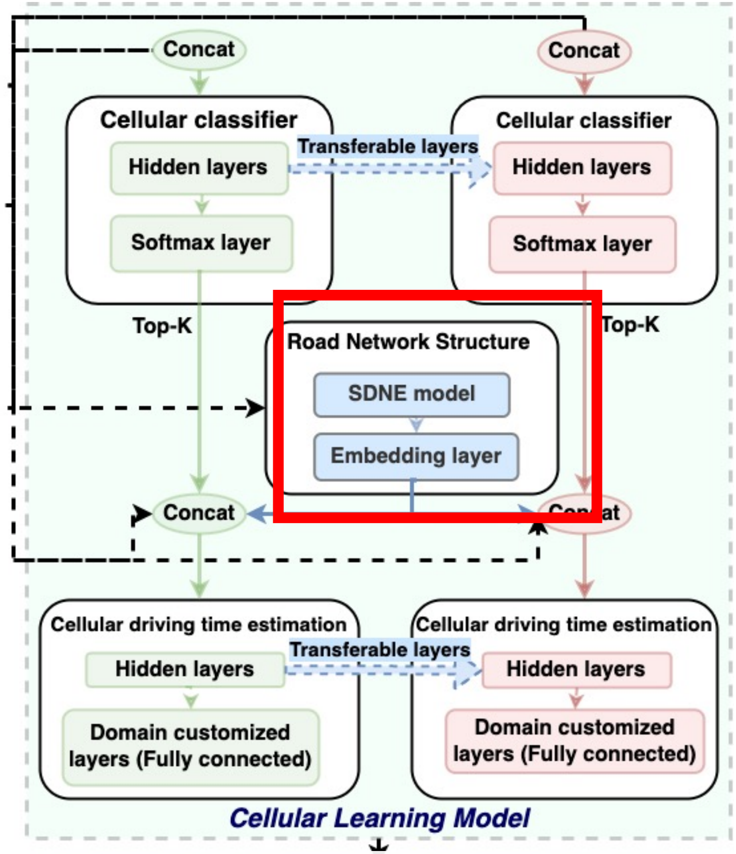
One trajectory



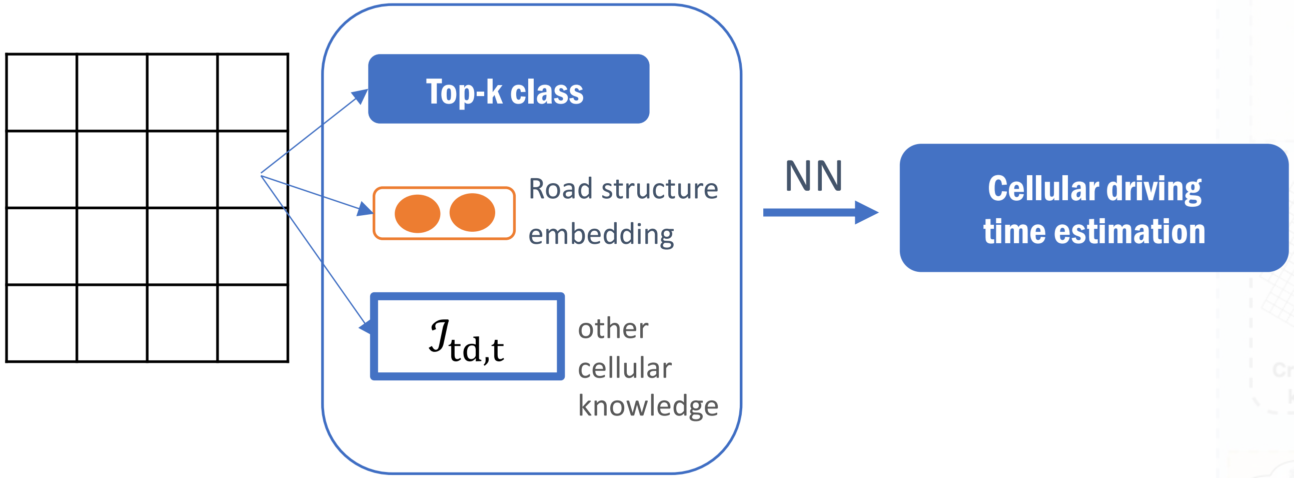
Multiple trajectories

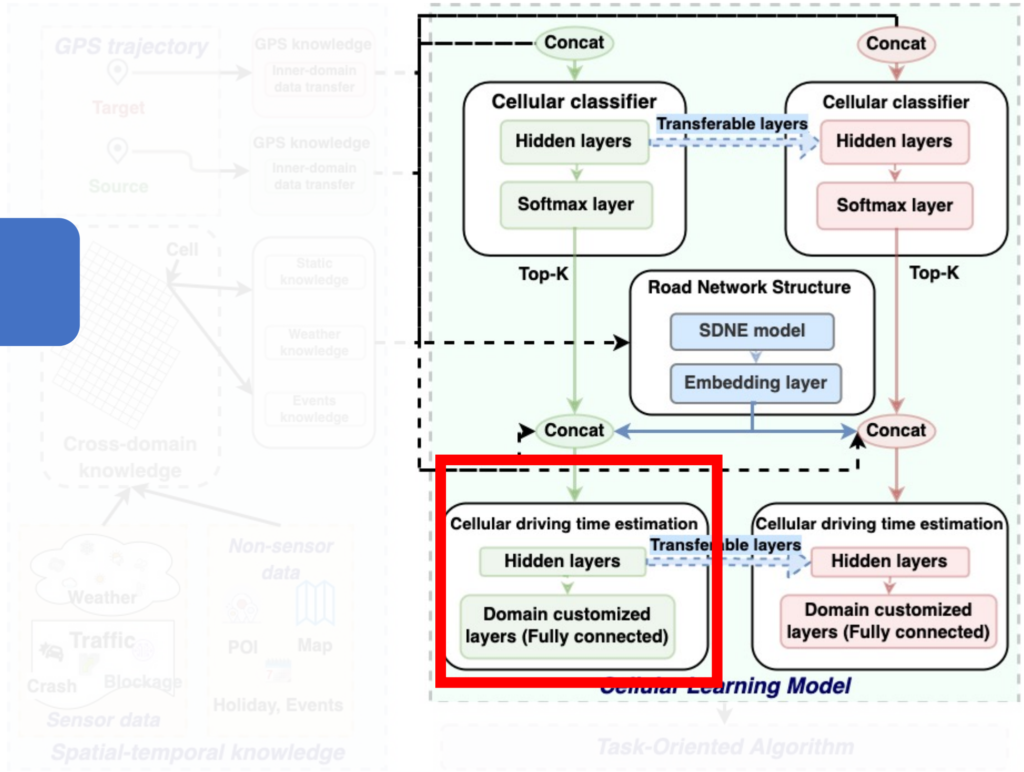


SNDE for each cell in spatial-temporal knowledge



TLETA-Cellular driving time estimation model



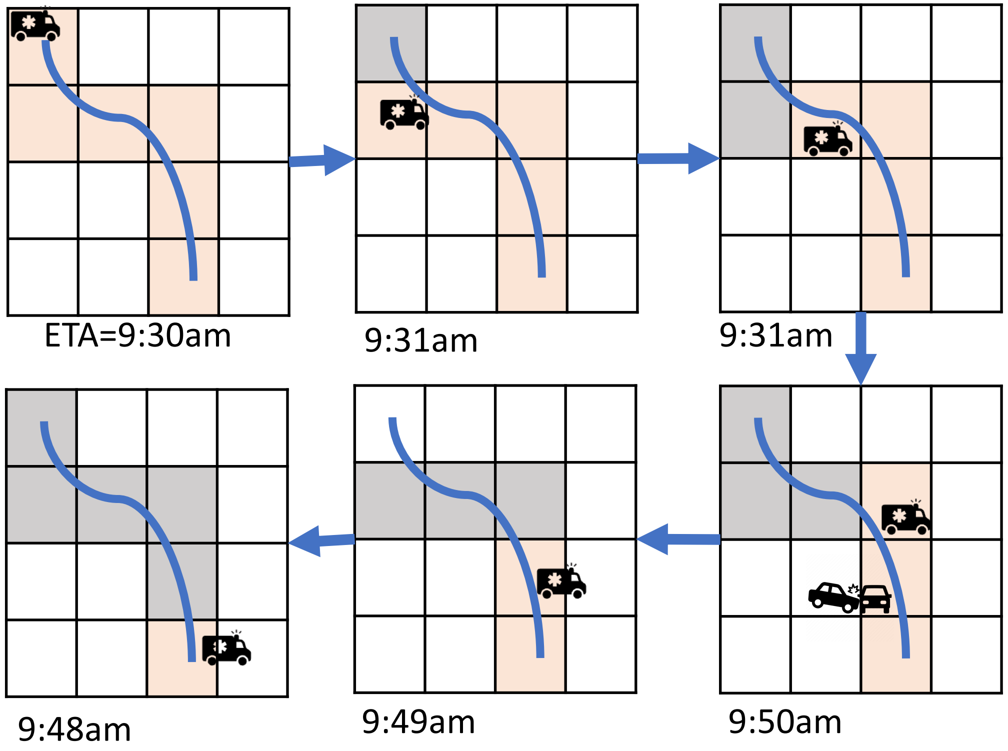


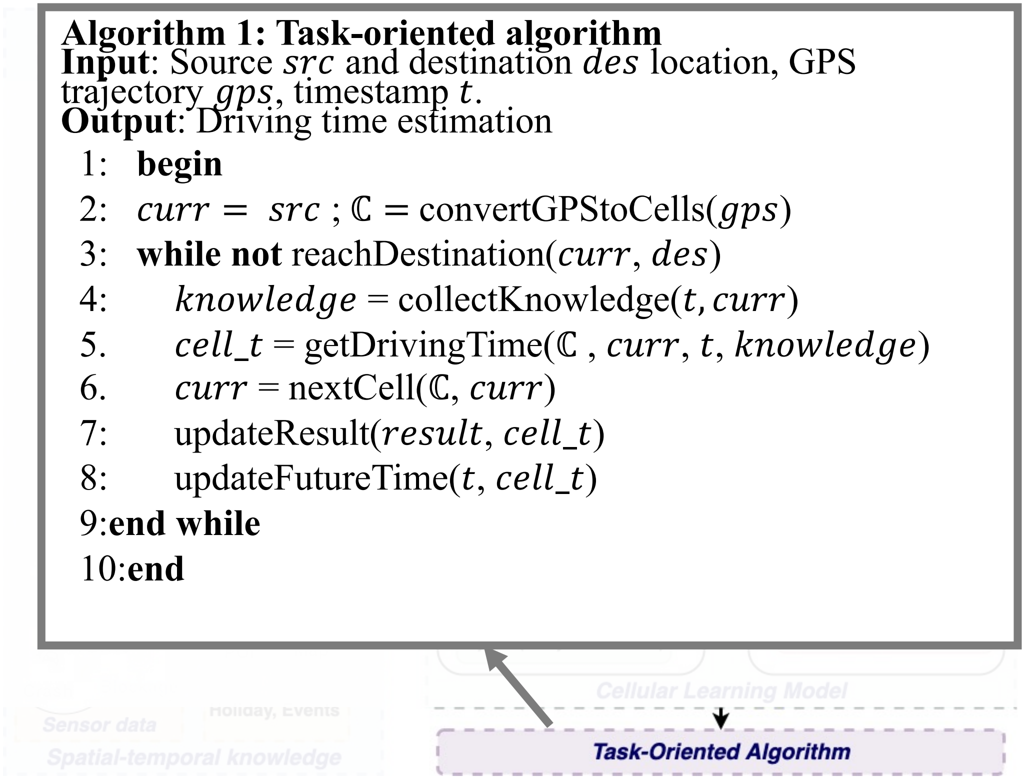
TLETA: Deep Transfer Learning and Integrated Cellular Data for Estimated Time of Arrival Prediction Architecture

TLETA-Task-oriented Algorithm

Task-oriented algorithm

- Given GPS trajectory
- Using the cellular learning output for cellular ETA
- Update ETA in real-time





Algorithm 1: Task-oriented algorithm
Input: Source *src* and destination *des* location, GPS trajectory *gps*, timestamp *t*.
Output: Driving time estimation

```
1:  begin
2:  curr = src ; ℂ = convertGPStoCells(gps)
3:  while not reachDestination(curr, des)
4:      knowledge = collectKnowledge(t, curr)
5.      cell_t = getDrivingTime(ℂ , curr, t, knowledge)
6.      curr = nextCell(ℂ, curr)
7:      updateResult(result, cell_t)
8:      updateFutureTime(t, cell_t)
9:end while
10:end
```

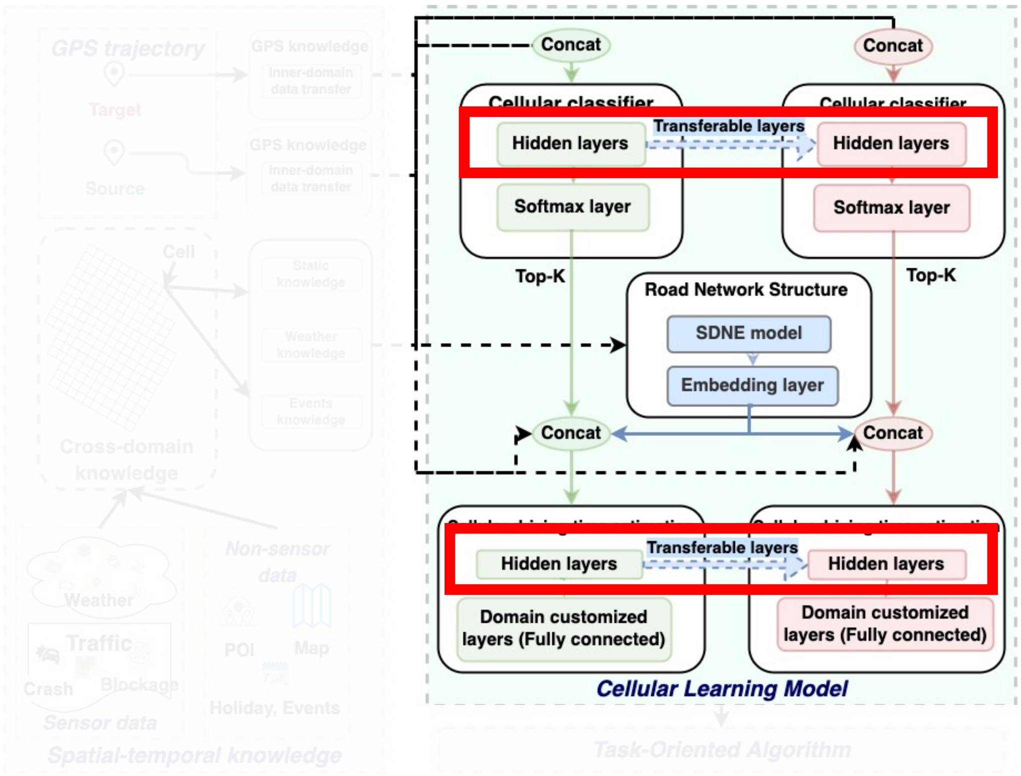

TLETA: Deep Transfer Learning and Integrated Cellular Data for Estimated Time of Arrival Prediction Architecture

A crash will affect the ETA in real-time

TLETA-Transferable Layers

Transfer Learning among vehicle domains

- Transferable hidden layers – fixed while training target domain
- Only train SoftMax layer and Domain customized layers in target domain



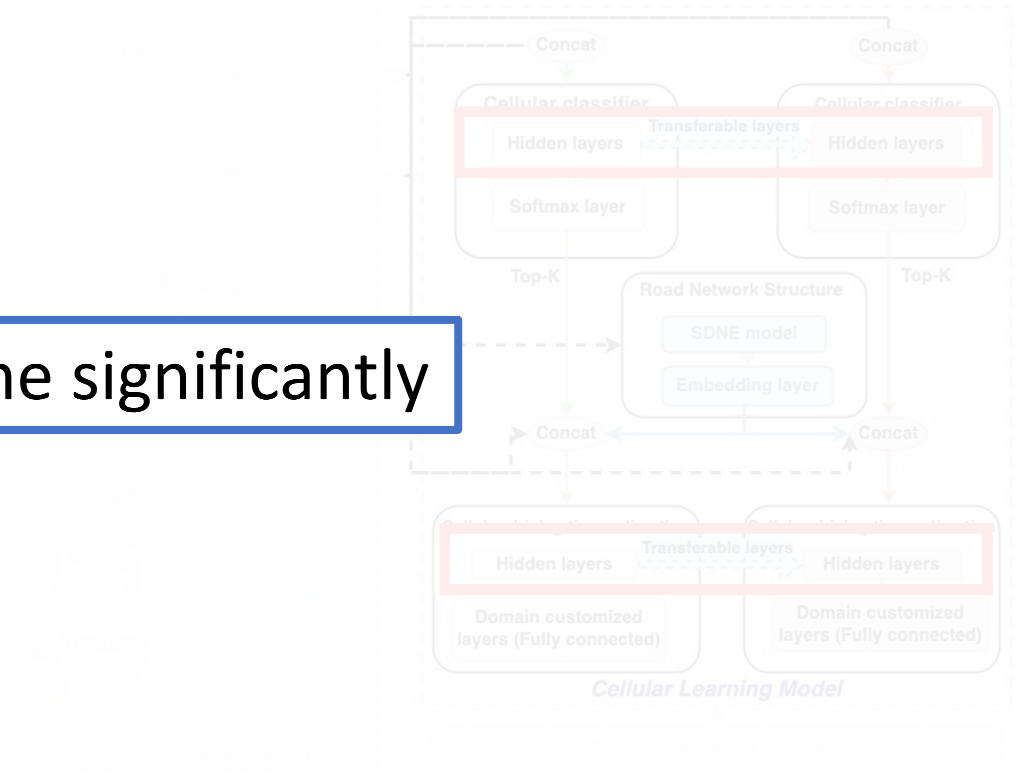
TLETA: Deep Transfer Learning and Integrated Cellular Data for Estimated Time of Arrival Prediction Architecture

TLETA-Transferable Layers

Transfer Learning among vehicle domains

- Transferable hidden layers – fixed while training target domain
- Only train SoftMax / Domain customized layers in target domain

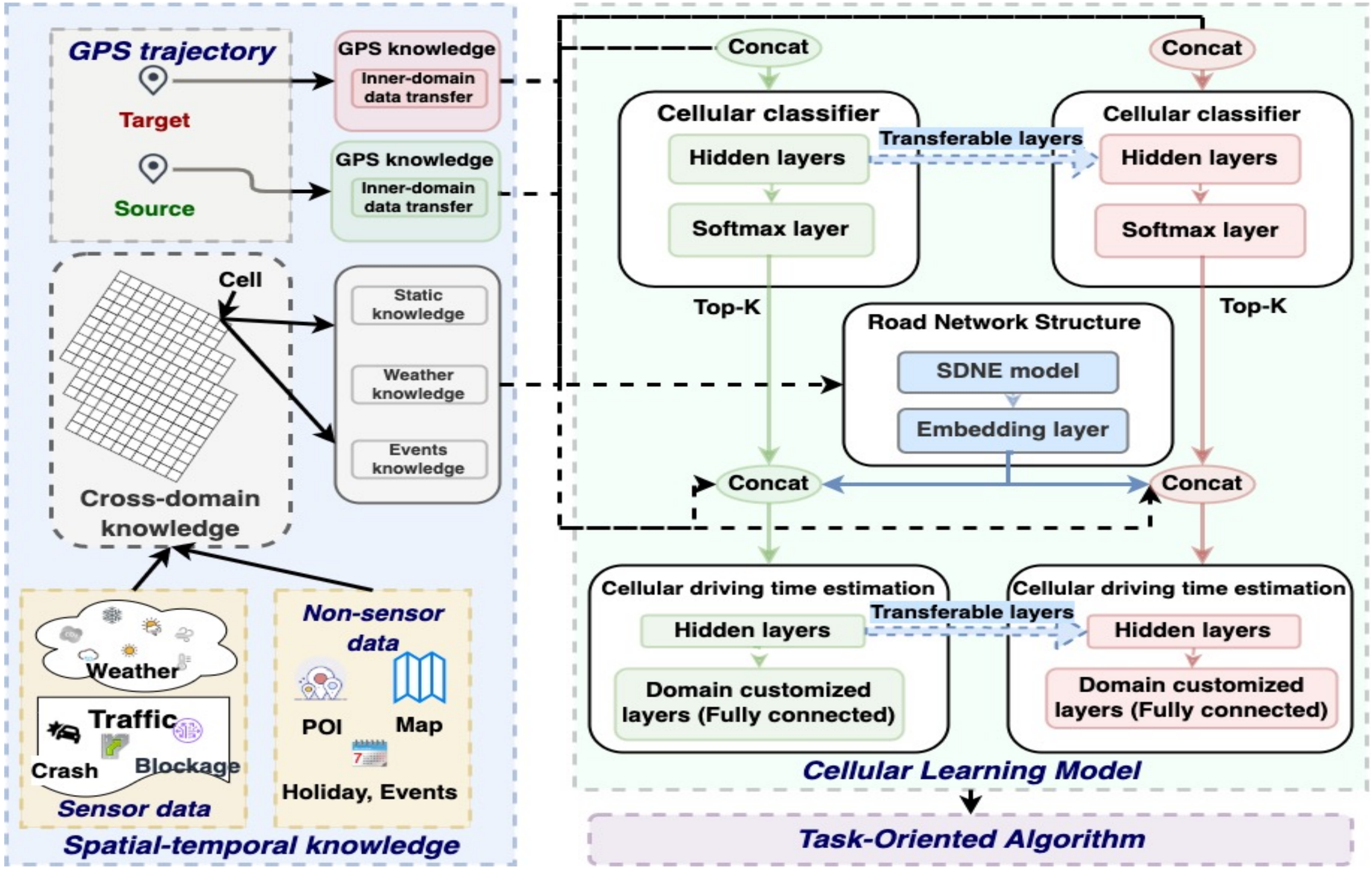
Reduce training time significantly



TLETA: Deep Transfer Learning and Integrated Cellular Data for Estimated Time of Arrival Prediction Architecture

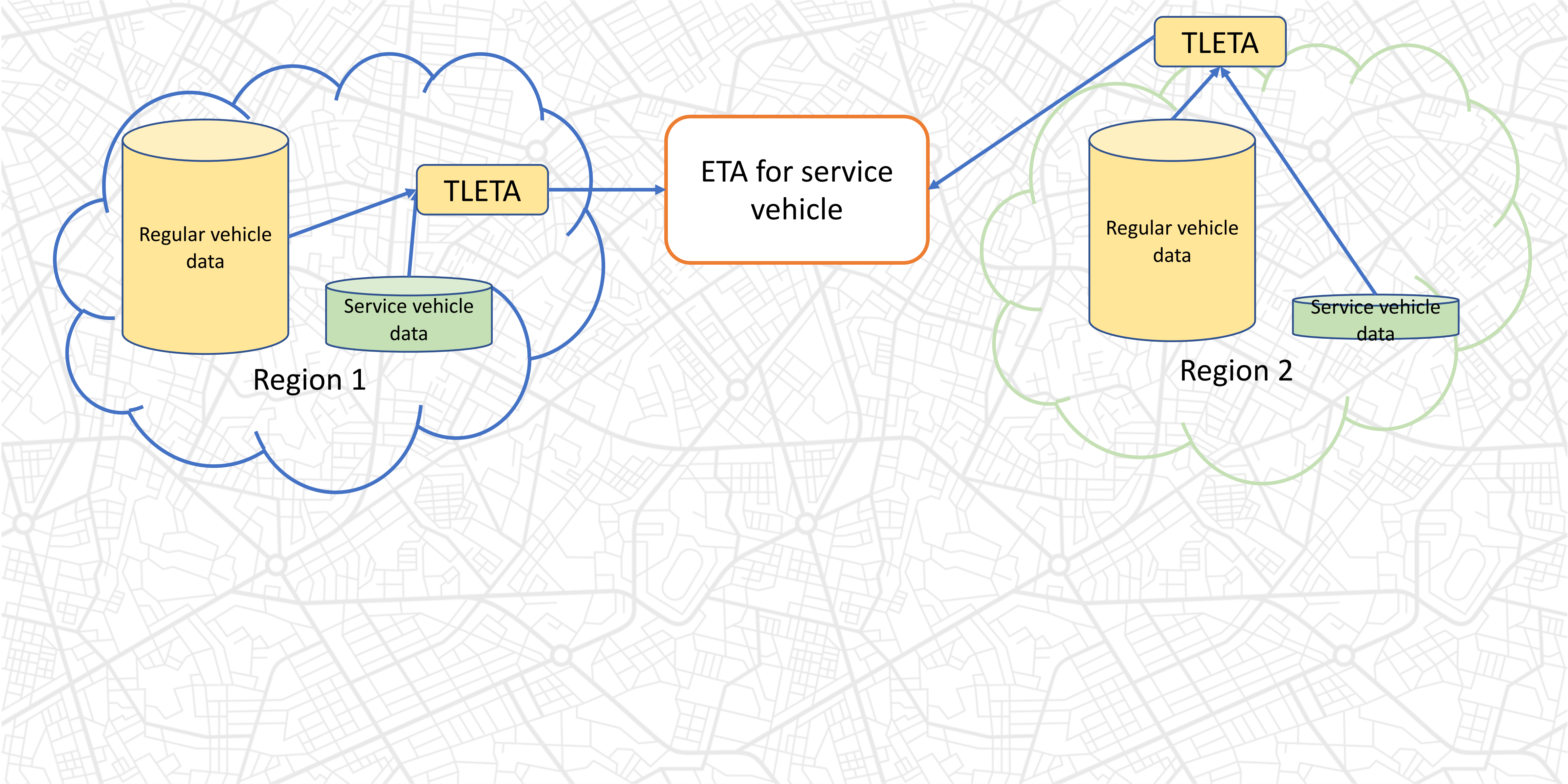
Summary of TLETA

- **The cellular spatial-temporal knowledge**: domain-specific and cross-domain knowledge
- **The cellular learning module** learns the cellular traffic patterns and includes a classifier, a road network structure embedding scheme, and a cellular ETA algorithm.
- **The task-oriented prediction module** leverages the learned model to predict ETA of a given trajectory

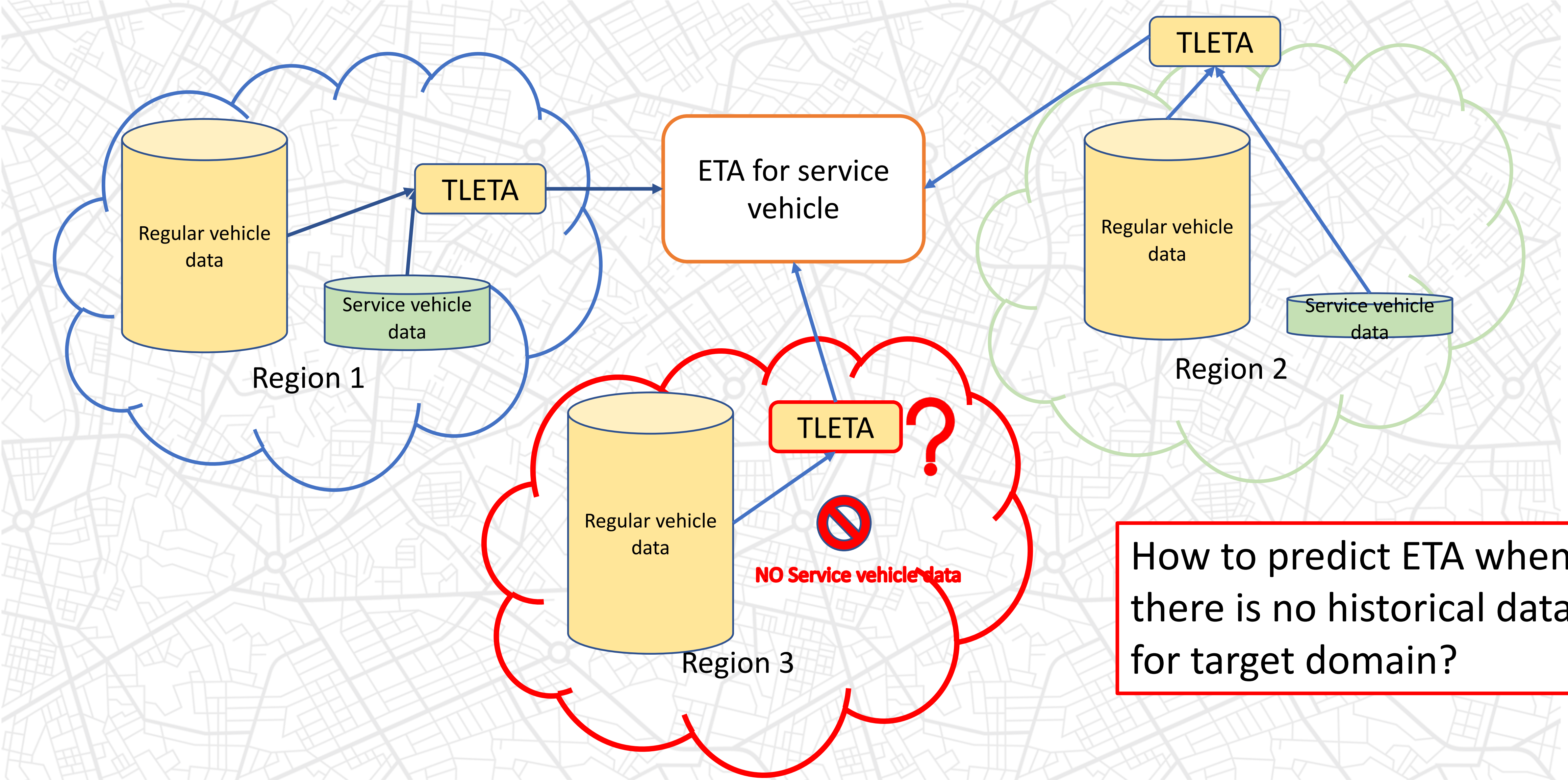


TLETA: Deep Transfer Learning and Integrated Cellular Data for Estimated Time of Arrival Prediction Architecture

Unavailable data for target domain



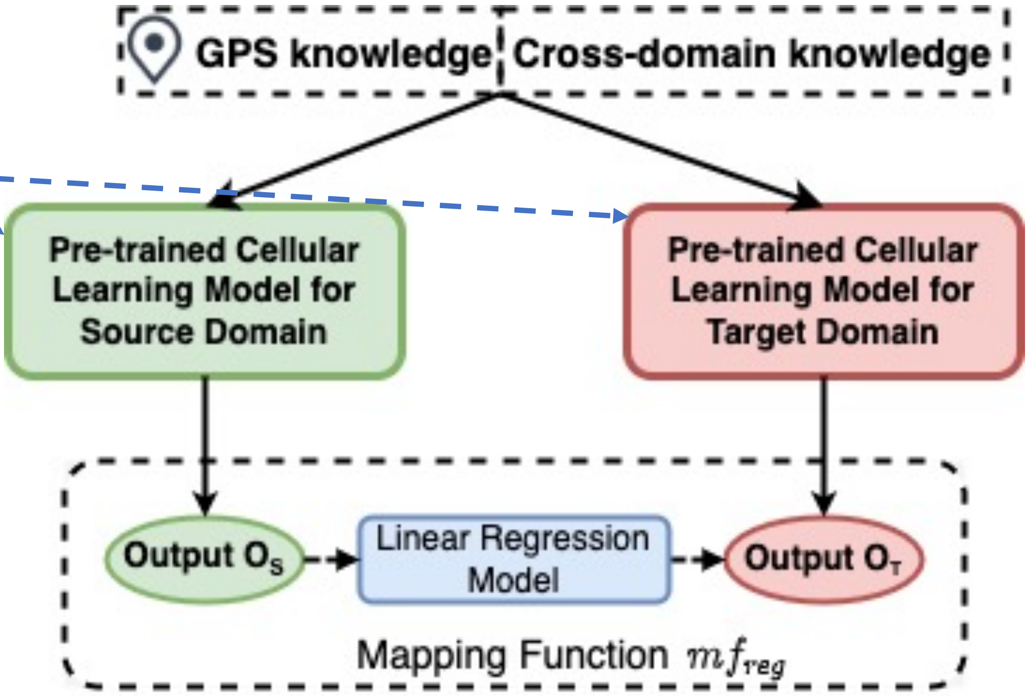
Unavailable data for target domain



How to predict ETA when there is no historical data for target domain?

In-region mapping function learning for each region

Learned from TLETA

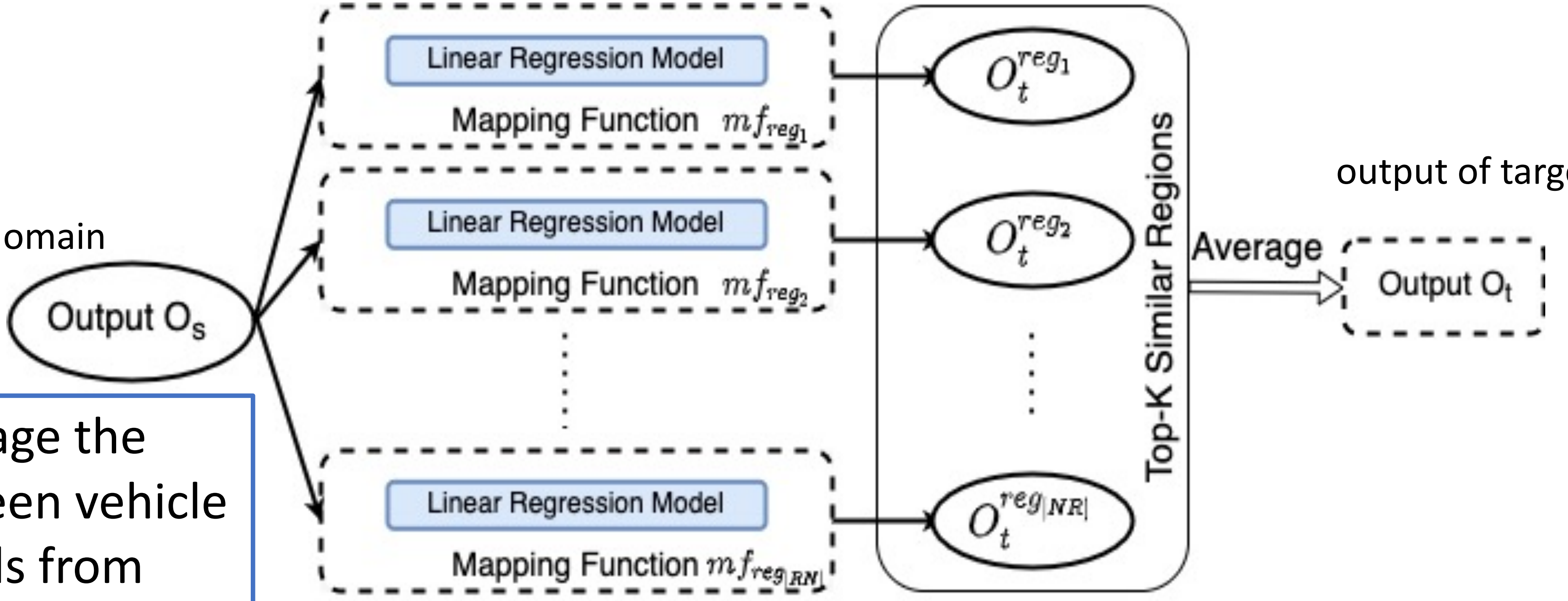


In-region mapping function learning in region *reg*

Goal: To learn the relation between vehicle domain models in regions where data is available

Inter-region transfer learning

output of source domain

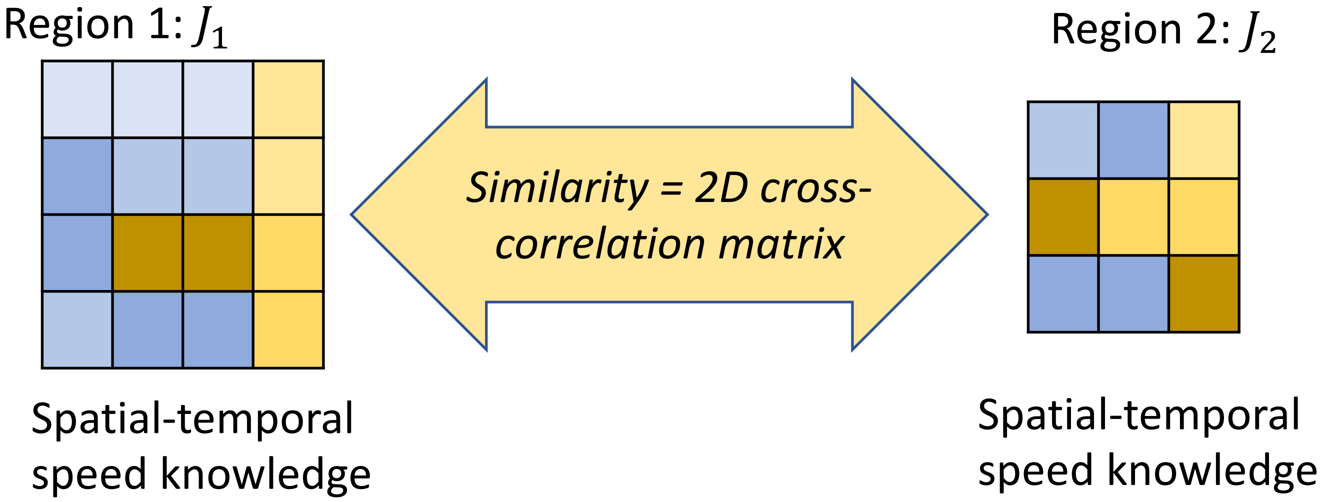


output of target domain

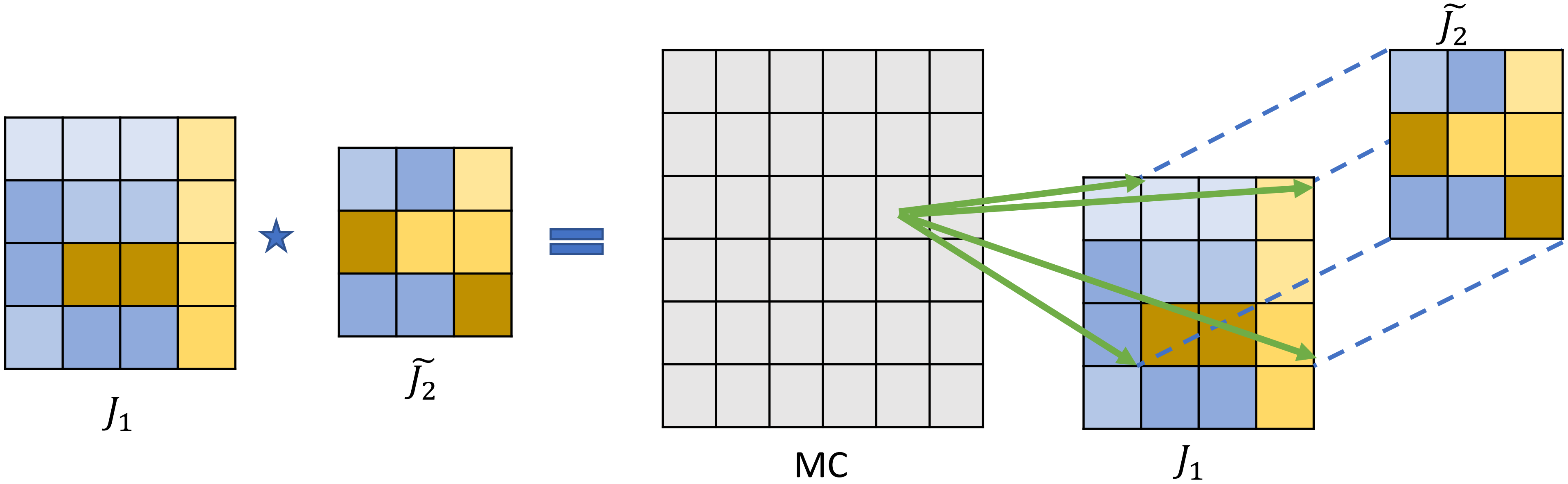
Goal: To leverage the relation between vehicle domain models from other regions to predict a target domain model in a region where the data is unavailable

Inter-region transfer learning

Region spatial-temporal similarity



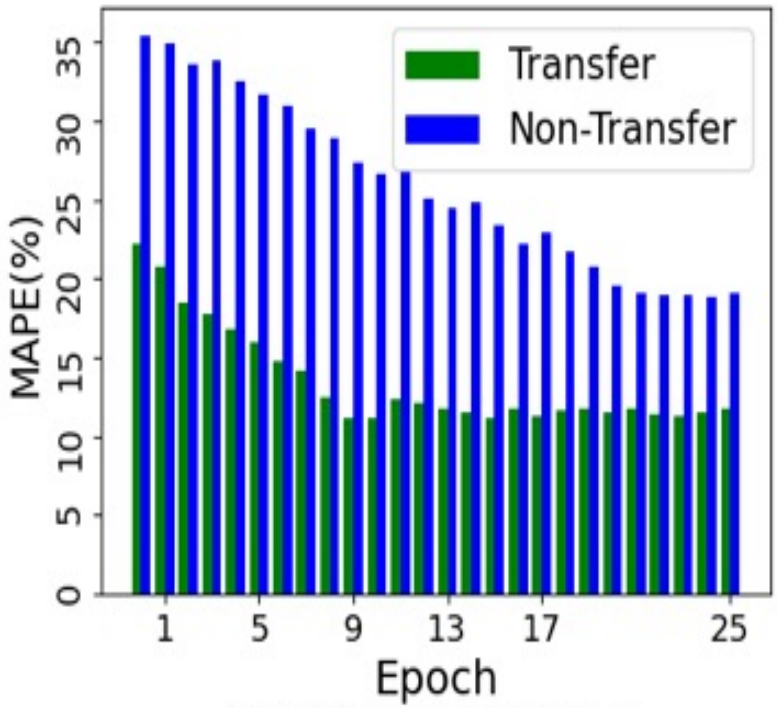
- two-dimensional cross-correlation matrix $MC = J_1 \times \widetilde{J_2}$ (where $\widetilde{J_2}$ - complex conjugation)
- *Mean(MC)* – denote the similarity level
- higher *Mean(MC)* presents a higher similarity



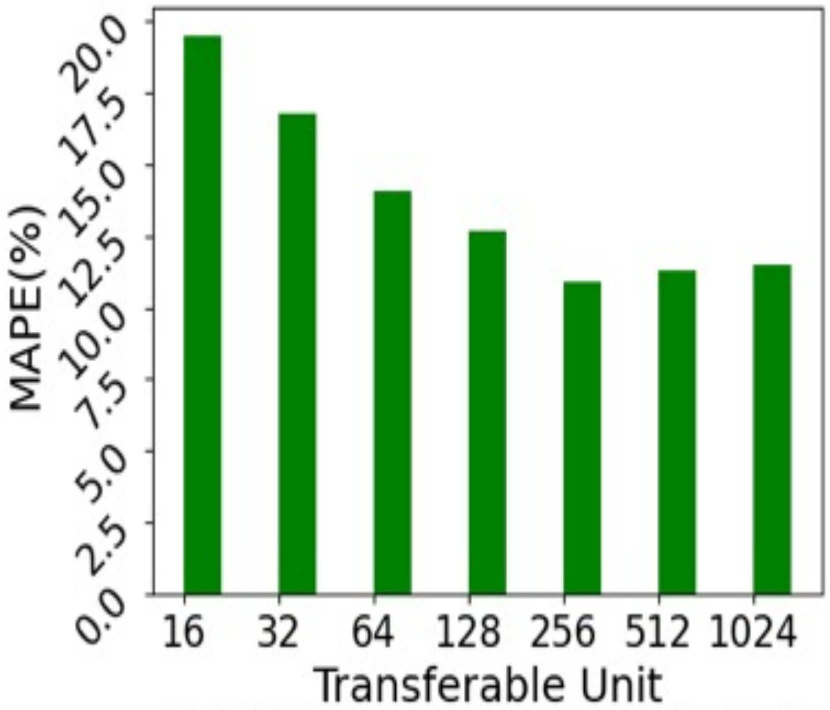
Different ETA Approach Comparison

Approach	Category	Consider Impact Factors							Handle data sparsity			Consider different vehicle types
		GPS	Weather	Event	POI	Time	Local road network structure	Global road network structure	Sparse	TL	Unavailable	
[W.-C. Lee, et al., 2012]	Segment	x										
[S. Maiti, et al., 2014]	End-to-End	x										
[Z. Wang, et al.,2018]	End-to-End	x										
[H. Wang ,et al.,2019]	End-to-End	x					x					
[A. Hofleitner, et al., 2012]	Segment	x					x		x			
[E. Jenelius, et al., 2013]	Segment	x	x			x	x					
[Y. Wang, et al., 2014]	Segment	x					x		x			
[F. Zhang, et al.,2016]	Segment	x	x						x			
[K.Fu, et al., 2020]	Segment	x	x			x	x		x			
[Y. Sun, et al.,2021]	Segment	x							x	x		
[B. Y. Lin, et al.,2018]	Segment	x			x	x			x	x		
[P. Krishnakumari, et al.,2018]	Fine-grained	x					x		x	x		
[H. Zhang, et al.,2018]	Fine-grained	x										
[C. Zhang, et al., 2019]	Fine-grained	x		x	x		x		x	x		
[Y. Shen, et al.,2020]	Fine-grained	x				x	x	x				
[S. Wang,et al.,2017]	Segment	x	x	x	x	x	x		x			
[Y. Sun, et al.,2020]	Segment	x							x	x		
[B. Du, et al., 2019]	Fine-grained	x	x	x					x			
[L. Wang, et al., 2019]	Fine-grained	x	x			x			x	x		
[S. Elmi , et al., 2020]	Segment	x	x			x	x		x	x		
[Y. Huang, et al., 2021]	Segment	x					x		x	x		
[T. Mallick, et al.,2021]	Segment	x					x		x	x		
[J. Wang, et al., 2016]	Segment	x				x	x		x	x		
TLETA	**Fine-grained**	**x**	**x**	**x**	**x**	**x**	**x**	**x**	**x**	**x**	**x**	**x**

Experimental studies of parameter analysis for TLETA



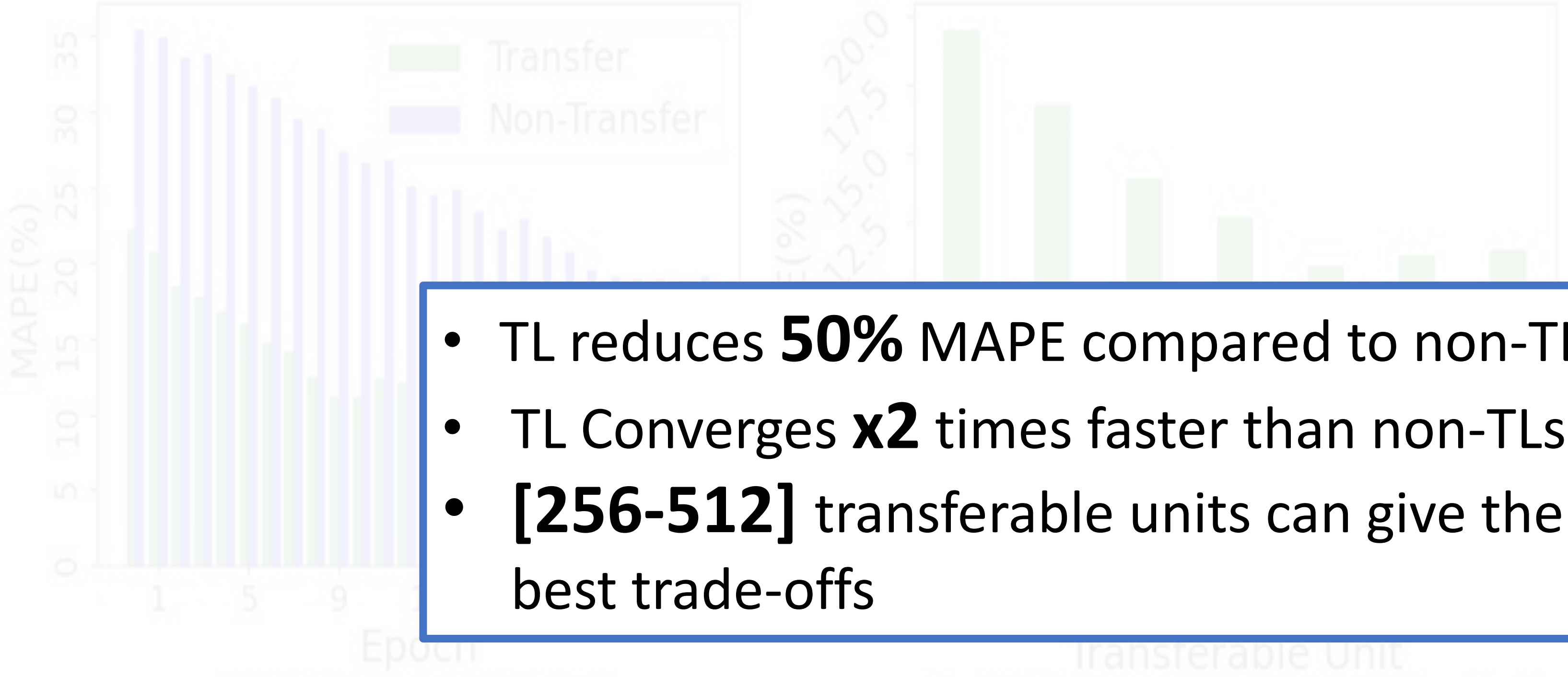
(a) MAPE and Epoch



(b) MAPE and Transferable Unit

MAPE - Mean absolute percentage error

Experimental studies of parameter analysis for TLETA



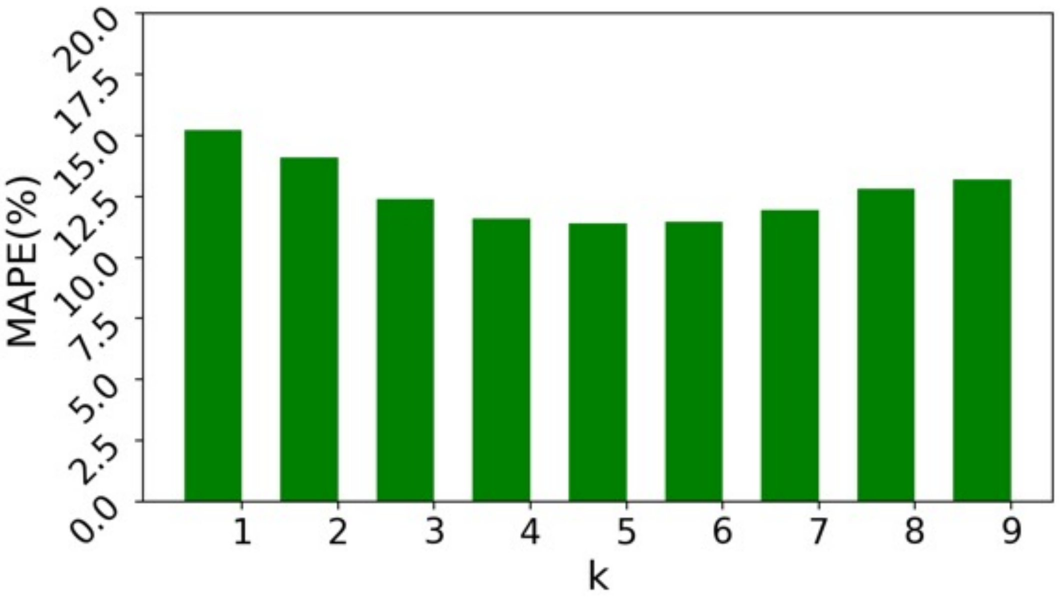
(a) MAPE and Epoch

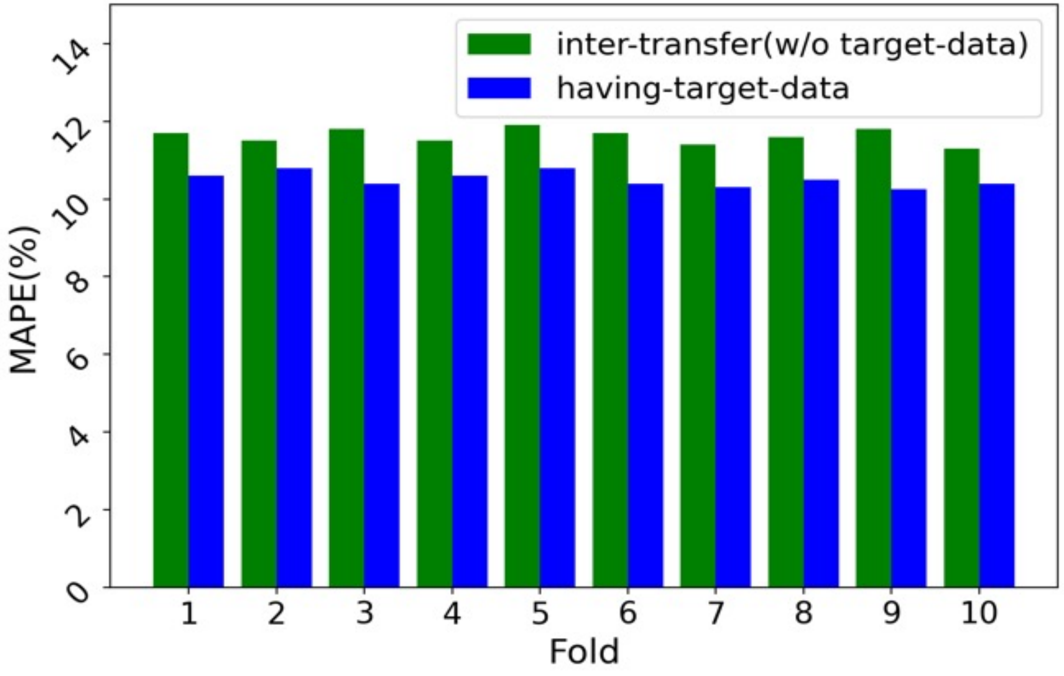
(b) MAPE and Transferable Unit

- TL reduces **50%** MAPE compared to non-TL
- TL Converges **x2** times faster than non-TLs
- **[256-512]** transferable units can give the best trade-offs

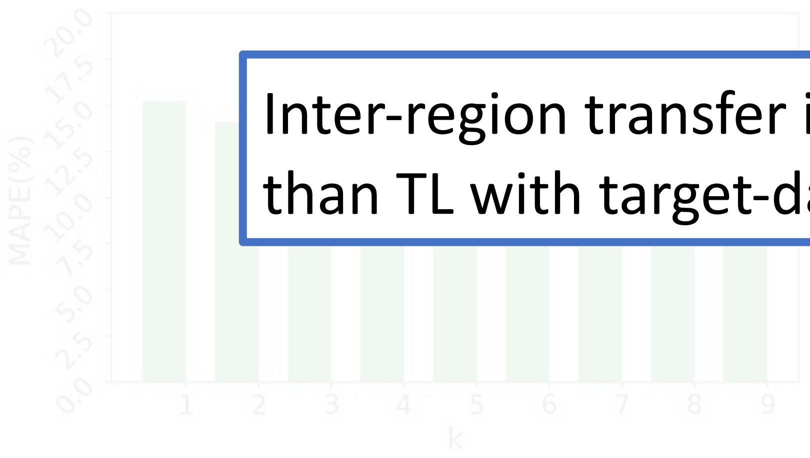
MAPE - Mean absolute percentage error

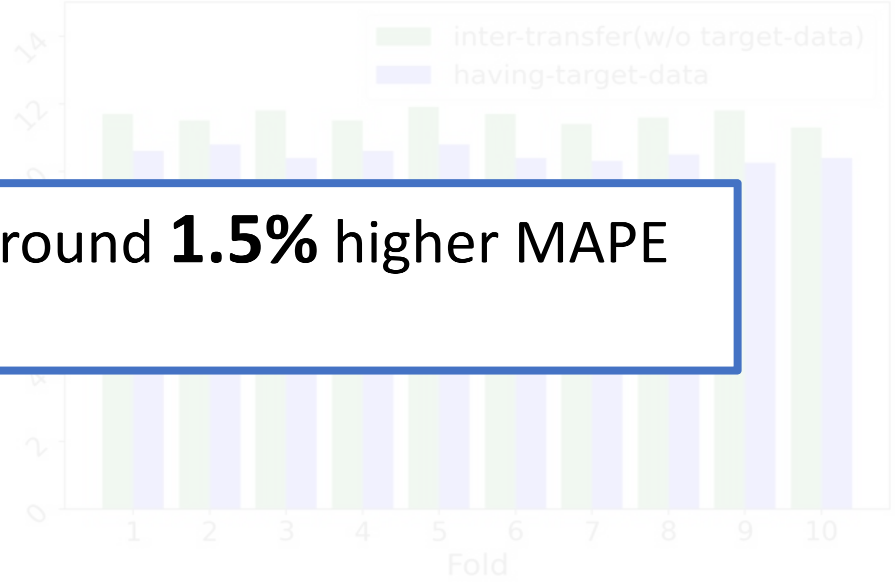
Experimental studies for Inter-region transfer





Experimental studies for Inter-region transfer





Inter-region transfer is only around **1.5%** higher MAPE than TL with target-data

Experimental studies for ETA prediction

G – GPS knowledge
S – Static
W – Weather
E – Event
R – Road Network Structure

MAPE - Mean absolute percentage error
RMSE - Root Mean Square Error

Approach	Knowledge					Metrics	
	G	S	W	E	R	MAPE	RMSE
[Y. Shen, et al.,2020]	x				x	19.57%	76s
[B. Du, et al., 2019]	x		x	x		18.44%	74s
[K.Fu, et al., 2020]	x		x		x	17.92%	71s
[H. Wang ,et al.,2019]	x					25.32%	102s
Reduced knowledge categories	x					23.78%	86s
	x	x				19.64%	72s
	x		x			21.44%	81s
	x			x		20.85%	80s
	x				x	18.23%	70s
	x		x	x		18.20%	70s
	x		x		x	16.88%	66s
	x		x	x	x	14.35%	58s
	x	x		x	x	13.66%	53s
	x	x	x		x	14.02%	56s
	x	x	x	x		15.71%	62s
	x	**x**	**x**	**x**	**x**	**12.37%**	**36s**

Experimental studies for ETA prediction

G – GPS knowledge
S – Static
W – Weather
E – Event
R – Road Network Structure

MAPE - Mean absolute percentage error
RMSE - Root Mean Square Error

Approach	Knowledge					Metrics	
	G	S	W	E	R	MAPE	RMSE
[Y. Shen, et al.,2020]	x				x	19.57%	76s
[B. Du, et al., 2019]	x		x	x		18.44%	74s
[K.Fu, et al., 2020]	x		x		x	17.92%	71s
[H. Wang ,et al.,2019]	x					25.32%	102s
[illegible]	x	[illegible]	[illegible]	[illegible]	[illegible]	23.78%	86s
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	19.64%	72s
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	21.44%	81s
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	20.85%	80s
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	18.23%	70s
R	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	18.20%	70s
categories	x		x		x	16.88%	66s
	x		x	x	x	14.35%	58s
	x	x		x	x	13.66%	53s
	x	x	x		x	14.02%	56s
	x	x	x	x		15.71%	62s
	x	**x**	**x**	**x**	**x**	**12.37%**	**36s**

Improved at least **1.3%** (MAPE) and **6%** (RMSE) with the same configuration for ETA prediction

Experimental studies for TL performance

Cincinnati datasets in 2018

information statistics

Properties	Dataset	
	Urban	Suburban
#GPS points for regular vehicles	1.2M	1M
#GPS points for service vehicles	1M	175K
POI	16K	8K
Splitting factor ε°	0.001°	0.001°

Transfer learning performance comparison of TLETA and other traffic forecasting methods

Dataset	Urban			Suburban		
Approach	MAPE	RMSE	Time	MAPE	RMSE	Time
STCNet [C. Zhang, et al., 2019]	17.65%	62s	104m	18.01%	76s	91m
RegionTrans [L. Wang, et al., 2019]	16.23%	56s	110m	17.88%	67s	93m
TL-DCRNN [T. Mallick, et al.,2021]	16.98%	57s	72m	17.21%	65s	56m
Lin et al. [B. Y. Lin, et al.,2018]	22.58%	73s	53m	24.32%	97s	45m
FBTL [S. Elmi , et al., 2020]	23.29%	81s	122m	25.78%	105s	99m
TEEPEE[Y. Huang, et al., 2021]	20.96%	70s	88m	22.43%	85s	72m
Non-transfer	14.78%	55s	49m	18.51%	74s	39m
TLETA	**10.54%**	**34s**	**31m**	**11.81%**	**38s**	**29m**

Experimental studies for TL performance

Cincinnati datasets in 2018 information statistics

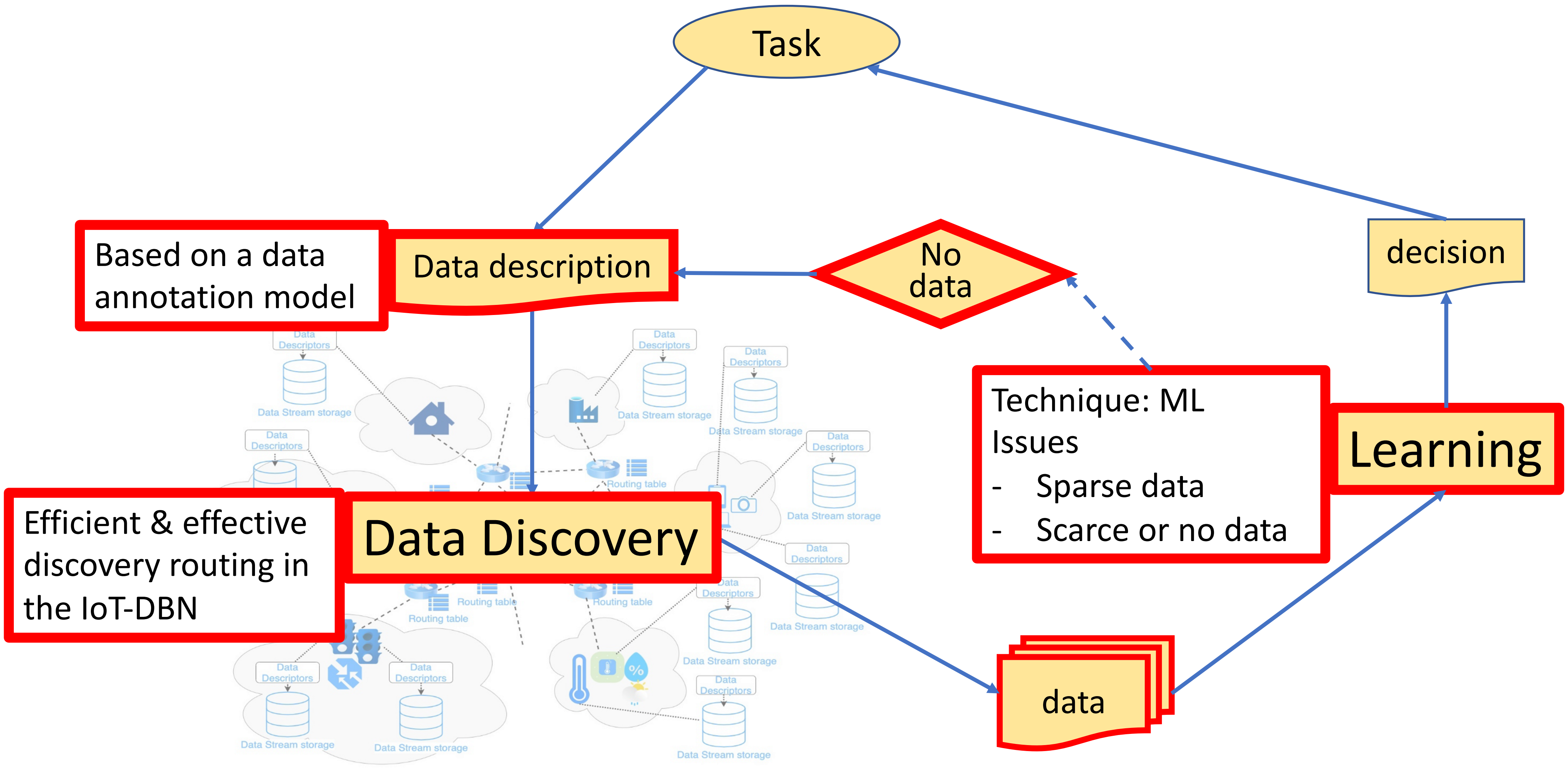
Properties	Dataset	
	Urban	Suburban
#GPS points for regular vehicles	1.2M	1M
#GPS points for service vehicles	1M	175K
POI	16K	8K
Splitting factor ε°	0.001°	0.001°

Transfer learning performance comparison of TLETA and other traffic forecasting methods

Dataset	Urban			Suburban		
Approach	MAPE	RMSE	Time	MAPE	RMSE	Time
[illegible]Net [S. Zh[illegible] et al., 2019]	[illegible]	[illegible]	[illegible]	18.01%	76s	91m
RegionTrans [L. Wang, et al., 2019]	[illegible]	[illegible]	110m	17.88%	67s	93m
TL-DCRNN [T. Mallick, et al.,2021]	16.98%	57s	72m	17.21%	65s	56m
Lin et al. [B. Y. Lin, et al.,2018]	22.58%	73s	53m	24.32%	97s	45m
FBTL [S. Elmi , et al., 2020]	23.29%	81s	122m	25.78%	105s	99m
TEEPEE[Y. Huang, et al., 2021]	20.96%	70s	88m	22.43%	85s	72m
Non-transfer	14.78%	55s	49m	18.51%	74s	39m
TLETA	**10.54%**	**34s**	**31m**	**11.81%**	**38s**	**29m**

Improved at least **35%** (MAPE) and **39%** (RMSE) and **41%** (training time) compared to the state-of-the-art approaches

Summary


Task
Based on a data annotation model
Data description
No data
decision
Technique: ML
Issues
- Sparse data
- Scarce or no data
Learning
Efficient & effective discovery routing in the IoT-DBN
Data Discovery
data
Data Descriptors
Data Stream storage
Routing table

Future Research

- Considering provenance-based data discovery
- Focus on adapting the applications of our learning techniques to address different questions in ITS and other applicable areas.

Selected Publications

- Tran, Hieu, Son Nguyen, I-Ling Yen, and Farokh Bastani. "Into Summarization Techniques for IoT Data Discovery Routing." 2021 IEEE 14th International Conference on Cloud Computing (CLOUD). IEEE, 2021.
- Tran, Hieu, Son Nguyen, I. Yen, and Farokh Bastani. "IoT Data Discovery: Routing Table and Summarization Techniques." *arXiv preprint arXiv:2203.10791* (2022).
- Tran, Hieu, Son Nguyen, I. Yen, and Farokh Bastani. TLETA: Deep Transfer Learning and Integrated Cellular Knowledge for Estimated Time of Arrival Prediction. 25th IEEE International Conference on Intelligent Transportation Systems 2022 (accepted)
- Tran, Hieu, Son Nguyen, I. Yen, and Farokh Bastani. "IoT Data Discovery: Routing Table and Summarization Techniques.". IEEE Transactions on Cloud Computing. (to be submitted)

References

- [statista, 2021] statista, "Number of Internet of Things (IoT) connected devices worldwide from 2019 to 2030," 8.25.2021. [Online]. Available: https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/.
- [statista, 2022] Data volume of internet of things (IoT) connections worldwide in 2019 and 2025. [Online]. Available: https://www.statista.com/statistics/1017863/worldwide-iot-connected-devices-data-size/
- [S. Sivanthan, et al. 2008] S. Sivanthan and et al., "Efficient multiple-keyword search in DHT-based decentralized systems," IEEE International Symposium on Performance Evaluation of Computer and Telecommunication Systems, 2008
- [G.-E. Luis, 2004] G.-E. Luis, "Data Indexing in Peer-to-Peer DHT Networks," in IEEE 24th International Conference on Distributed Computing Systems, 2004.
- [S. Cristina, et al., 2004] S. Cristina and M. Parashar, "Enabling flexible queries with guarantees in P2P systems," in IEEE Internet Computing, 2004.
- [J.Xing, et al., 2006] J. Xing, Y . Y .P .K and C. S.H.G, "Supporting multiple-keyword search in a hybrid structured peer-to-peer network," in IEEE International Conference on Communications, 2006.
- [T. Koponen, et al., 2007] T. Koponen and et al., "A data-oriented (and beyond) network architecture," in ACM SIGCOMM Computer Communication Review 37.4, 2007
- [ccn project] "Content Centric Networking project," [Online]. Available: http://www.ccnx.org/.
- [V. Jacobson, et al., 2009] V. Jacobson and et al., "Networking named content," in the 5th international conference on Emerging networking experiments and technologies, 2009.
- [Z. Lixia, et al., 2010] Z. Lixia and et al., "Named data networking (ndn) project.," Relatório Técnico NDN-0001, Xerox Palo Alto Research Center- PARC 157, 2010.
- [D. Chakraborty, et al., 2002] D. Chakraborty and et al. , "GSD: A novel group-based service discovery protocol for MANETS," in 4th International workshop on mobile and wireless communications network. IEEE, 2002.

References

- [Adriane Chapman, et al., 2020] Chapman, Adriane, et al. "Dataset search: a survey." The VLDB Journal 29.1 (2020): 251-272.
- [gminsights , 2022] "Vehicle tracking device market," [Online]. Available: gminsights.com/industry-analysis/vehicle-tracking-market.
- [S. Maiti, et al., 2014] S. Maiti and et al., "Historical data based real time prediction of vehicle arrival time," in IEEE ITSC, 2014.
- [H. Wang ,et al.,2019] H. Wang and et al., "A simple baseline for travel time estimation using large-scale trip data," in ACM TIST, 2019.
- [Z. Wang, et al.,2018] Z. Wang and et al., "Learning to estimate the travel time," in 24th ACM SIGKDD, 2018.
- [W.-C. Lee, et al., 2012]W.-C. Lee and et al., "HTTP: a new framework for bus travel time prediction based on historical trajectories," in ACM SIGSPATIAL, 2012.
- [Y. Wang, et al., 2014] Y. Wang, Y. Zheng and Y. Xue, "Travel time estimation of a path using sparse trajectories," in ACM SIGKDD, 2014.
- [K.Fu, et al., 2020] K. Fu and et al., "Compacteta: A fast inference system for travel time prediction," in ACM SIGKDD, 2020.
- [E. Jenelius, et al., 2013] E. Jenelius and H. Koutsopoulos, "Travel time estimation for urban road networks using low frequency probe vehicle data," in Transportation Research Part B: Methodological, 2013.
- [A. Hofleitner , et al., 2012] A. Hofleitner and et al., "Learning the dynamics of arterial traffic from probe data using a dynamic Bayesian network," in Transactions on Intelligent Transportation Systems, 2012.
- [F. Zhang, et al.,2016]F. Zhang and et al., "Urban link travel time prediction based on a gradient boosting method considering spatiotemporal correlations," in ISPRS international journal of geo-information, 2016.
- [Y. Sun, et al.,2021] Y. Sun and et al., "Road network metric learning for estimated time of arrival," in ICPR, 2021.
- [T. Mallick, et al.,2021] T. Mallick and et al., "Transfer learning with graph neural networks for short-term highway traffic forecasting," in ICPR, 2021.
- [J. Wang, et al., 2016] J. Wang and et al., "Traffic speed prediction and congestion source exploration: A deep learning method," in IEEE ICDM, 2016.

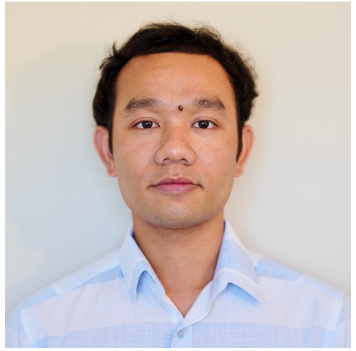
References

- [C. Zhang, et al., 2019] C. Zhang and et al., "Deep transfer learning for intelligent cellular traffic prediction based on cross-domain big data," in J-SAC, 2019.
- [P. Krishnakumari, et al.,2018] P. Krishnakumari and et al., "Understanding network traffic states using transfer learning," in ITSC, 2018.
- [B. Y. Lin, et al.,2018] B. Y. Lin and et al., "Transfer learning for traffic speed prediction: A preliminary study.," AAAI Workshops, 2018.
- [Y. Shen, et al.,2020] Y. Shen and et al., "TTPNet: A neural network for travel time prediction based on tensor decomposition and graph embedding," in IEEE Transactions on Knowledge and Data Engineering, 2020.
- [H. Zhang, et al.,2018]H. Zhang and et al., "Deeptravel: a neural network based travel time estimation model with auxiliary supervision," in IJCAI, 2018.
- [S. Wang, et al.,2017] S. Wang and et al., "Computing urban traffic congestions by incorporating sparse GPS probe data and social media data," in ACM Transactions on Information Systems, 2017.
- [B. Du, et al., 2019] B. Du and et al., "Deep irregular convolutional residual LSTM for urban traffic passenger flows prediction," in T-ITS, 2019.
- [Y. Sun, et al.,2020] Y. Sun and et al., "CoDriver ETA: Combine driver information in estimated time of arrival by driving style learning auxiliary task," in IEEE Transactions on Intelligent Transportation Systems, 2020.
- [L. Wang, et al., 2019] L. Wang and et al., "Cross-City transfer learning for deep spatio-temporal prediction," in IJCAI-19, 2019.
- [S. Elmi , et al., 2020] S. Elmi and K.-L. Tan, "Travel time prediction in missing data areas: feature-based transfer learning approach," in IEEE HPCC/SmartCity/DSS, 2020.
- [Y. Huang, et al., 2021] Y. Huang and et al., "Transfer learning in traffic prediction with graph neural networks," in ITSC, 2021.
- [C. Antonio, et al., 2004] C. Antonio, M. J. Rutherford and A. L. Wolf, "A routing scheme for content-based networking," in IEEE INFOCOM, 2004.

References

- [synopsys,2022] https://www.synopsys.com/designware-ip/technical-bulletin/memory-options.html
- [Marica Amadeo, et al., 2016] Amadeo, Marica, et al. Amadeo, Marica, et al. "Information-centric networking for the internet of things: challenges and opportunities." IEEE Network 30.2 (2016): 92-100.
- [Antonio Carzaniga, et al., 2004] Carzaniga, Antonio, Matthew J. Rutherford, and Alexander L. Wolf. "A routing scheme for content-based networking." IEEE INFOCOM 2004. Vol. 2. IEEE, 2004.
- [Scott Empson, 2020] Route Summarization, Cisco Press., 2020
- [Dipanjan Chakraborty, et al.,2002] Chakraborty, Dipanjan, et al. "GSD: A novel group-based service discovery protocol for MANETS." 4th International workshop on mobile and wireless communications network. IEEE, 2002.

Biographical Sketch



TRUNG HIEU TRAN,

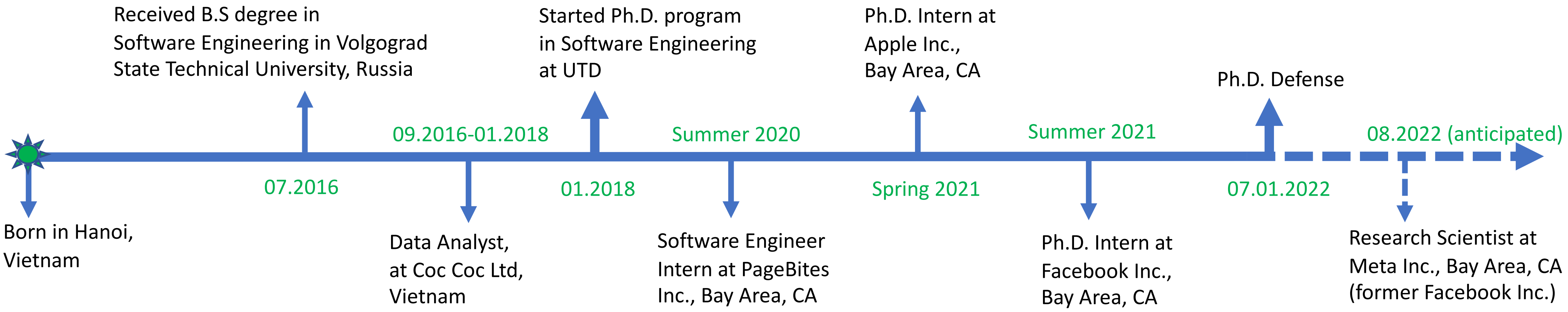
Ph.D. in Software Engineering – The University of Texas at Dallas (UTD)

www.linkedin.com/in/trunghieu-tran

https://scholar.google.com/citations?user=qN7mGjsAAAAJ

https://trunghieu-tran.github.io

trunghieu.tran@utdallas.edu or trantrunghieu7492@gmail.com



Thank you for your attention
Q&A