# A Survey on Transfer Learning for Urban Spatio-temporal Machine Learning


**Yilun JIN**

**Supervisors**: Professor Qiang YANG & Professor Kai CHEN

**Committee:** Professor Wilfred NG (Chair) & Professor Dit-Yan YEUNG


August 18th, 2021


香港科技大學
THE HONG KONG
UNIVERSITY OF SCIENCE
AND TECHNOLOGY

# Outline

- Introduction
- Urban Spatio-temporal Machine Learning
- Transfer Learning
- Urban Spatio-temporal Transfer Learning
    - Spatial Transfer Learning
    - Temporal Transfer Learning
    - Cross-modal Transfer Learning
- Conclusions and Future Work

# Outline

- Introduction
- Urban Spatio-temporal Machine Learning
- Transfer Learning
- Urban Spatio-temporal Transfer Learning
  - Spatial Transfer Learning
  - Temporal Transfer Learning
  - Cross-modal Transfer Learning
- Conclusions and Future Work

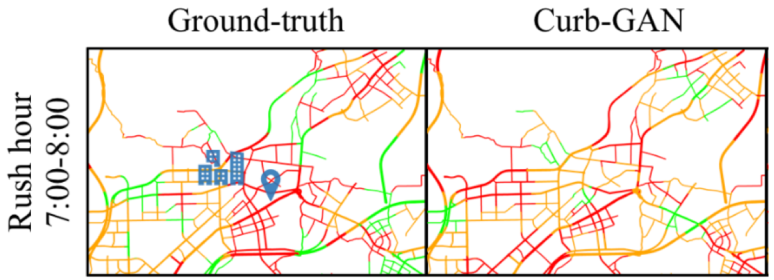
# Cities and Urban Problems

- Urbanization leads to many large cities
  - 54% population live in cities [UN, 2015]
  - Metropolises: New York, London, Tokyo, Hong Kong, etc.

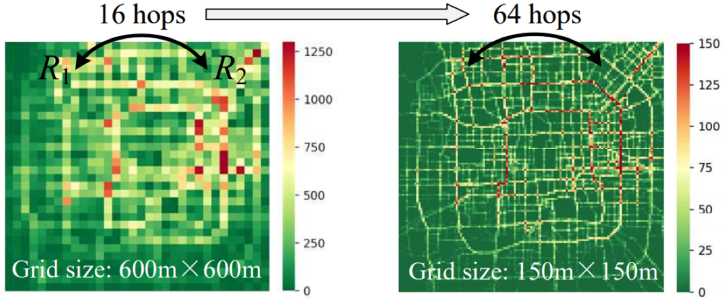
- …and also leads to important urban problems.
  - Environmental pollution
  - Energy consumption
  - Traffic management
  - ……

# Machine Learning for Urban Problems

- **Triggers**: **Sensing** and **machine learning**
  - Smart sensors, GPS, etc. generate **spatio-temporal urban data.**
    - E.g. Trajectories, social networks, environment sensors···
  - Machine learning models capture spatio-temporal information.
    - **Spatio-temporal machine learning (STML):** Convolutional Neural Networks (**CNN**), Recurrent Neural Networks (**RNN**), Graph Neural Networks (**GNN**), ···
- Applications:
  - **Traffic**: Speed, travel time, ···
  - **Human activities**: Human flow, events,
  - **Businesses**: Demand & supply, recommendation, ···



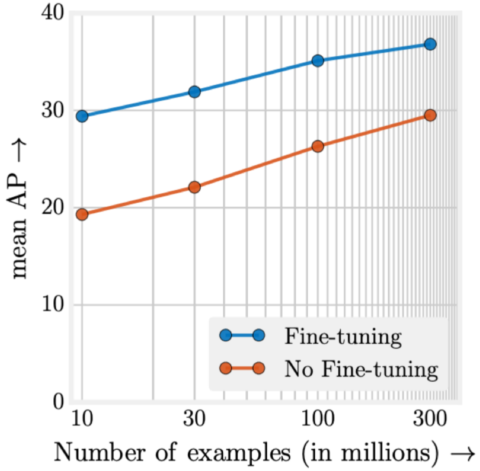

Traffic jams [Zhang et al. 2020]



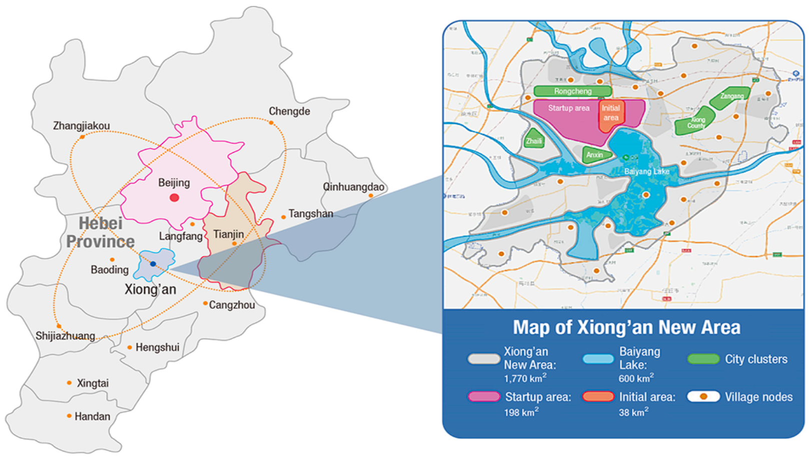

Human Flow [Liang et al. 2021]

# Data scarcity in cities

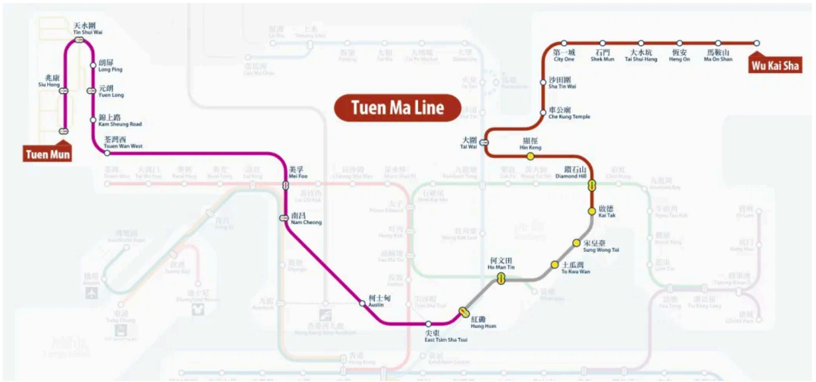
- Machine learning relies on big data.
  - More data leads to better performance.
- Data scarcity is common in cities.
  - Building new cities
  - Planning new urban services
- **Problem: How to mitigate lack of data for urban STML?**




Object detection, Performance VS data size [Sun et al. 2017]



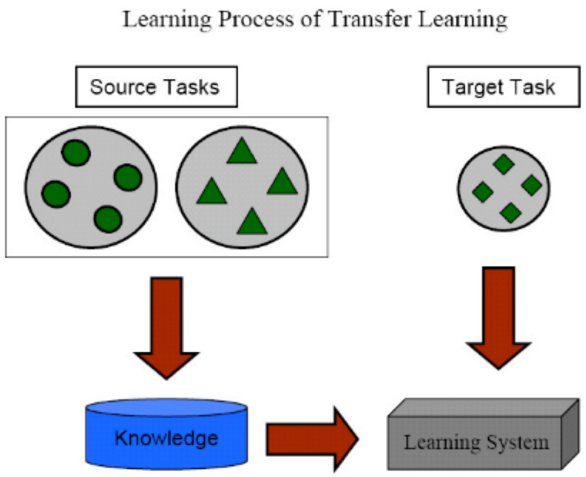

New city: Xiong`an New Area, China



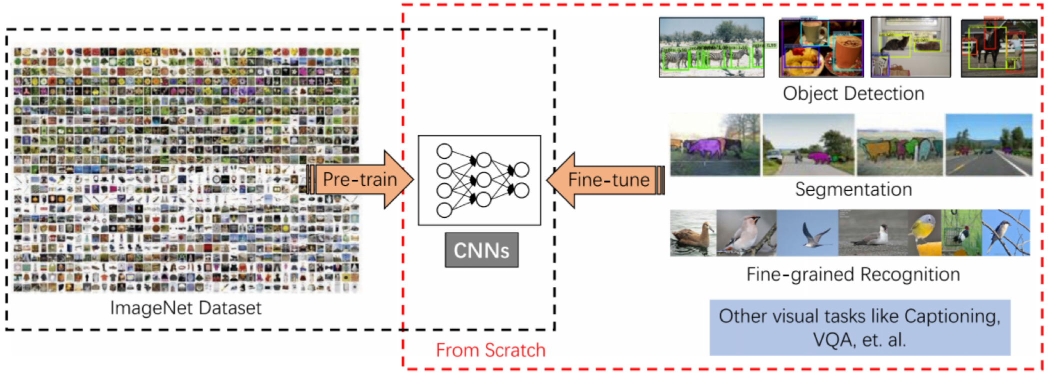

New urban service: Tuen Ma Line, 2021

# Solution: Transfer Learning

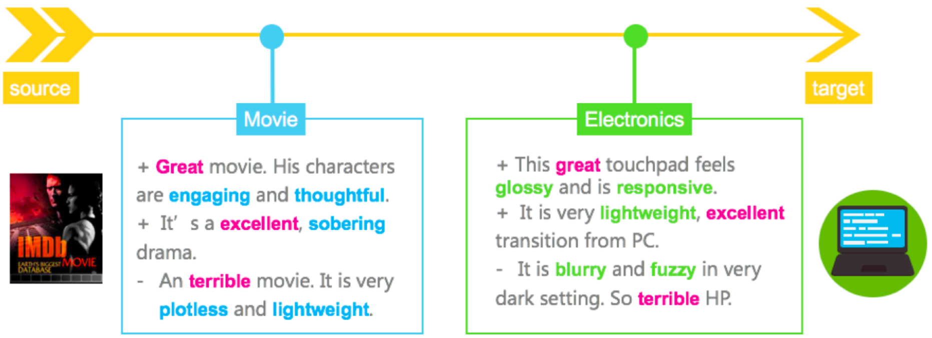
- **Transfer learning**:
  - **Key idea**: Borrow knowledge from **different but related** tasks.
  - Effective in visual recognition, text mining, etc.
    - Pre-training & fine-tuning in computer vision
    - Cross-domain sentiment classification [Li, 2020].
- **Q: How can transfer learning be applied to urban STML?**




[Pan et al. 2009]




ImageNet Pre training in CV [Shen et al. 2019]

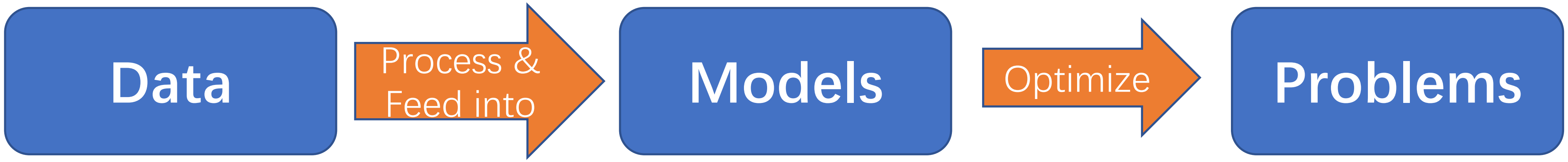



Cross-domain language analysis [Li, 2020]

# Outline

- Introduction
- Urban Spatio-temporal Machine Learning
- Transfer Learning
- Urban Spatio-temporal Transfer Learning
    - Spatial Transfer Learning
    - Temporal Transfer Learning
    - Cross-modal Transfer Learning
- Conclusions and Future Work

# Urban Spatio-temporal Machine Learning




**Q:** How is spatio-temporal information used to organize the data?

**Q:** How to **extract the spatio-temporal information** within the data?

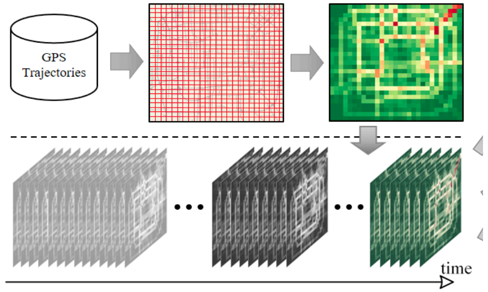
**Q:** How to formulate urban computing tasks into machine learning problems?

# Urban Spatio-temporal Machine Learning: Data

- Data types in urban STML:
  Classified by **spatio-temporal correlations**

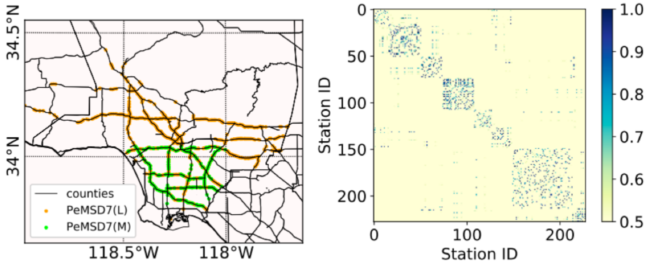
| Data Type | Point | Sequence | Static Map | Static Graph | ST Map | ST Graph |
|---|---|---|---|---|---|---|
| Temporal Variant? | No | Yes | No | No | Yes | Yes |
| Spatial Variant? | No | Optional | Yes | Yes | Yes | Yes |
| Euclidean Spatial Corr.? | / | Optional | Yes | Optional | Yes | Optional |
| Non-Euclidean Spatial Corr. | / | Optional | No | Yes | No | Yes |
| Data Format | $(l, t, a)$ $a \in \mathbb{R}^f$ | $[(t_i, a_i)]_{i=1}^{T}$ or $[(l_i, t_i, a_i)]_{i=1}^{T}$ $a_i \in \mathbb{R}^f$ | $\mathbf{M} \in \mathbb{R}^{W \times H \times f}$ | $G = (V, \mathbf{X}, \mathbf{A})$ $\mathbf{X} \in \mathbb{R}^{\lvert V \rvert \times f}$ $\mathbf{A} \in \mathbb{R}^{\lvert V \rvert \times \lvert V \rvert}$ | $\mathbf{M} \in \mathbb{R}^{T \times W \times H \times f}$ | $G = (V, \mathbf{X}, \mathbf{A})$ $\mathbf{X} \in \mathbb{R}^{T \times \lvert V \rvert \times f}$ $\mathbf{A} \in \mathbb{R}^{T \times \lvert V \rvert \times \lvert V \rvert}$ |
| Example | Incidents events | Weather data Trajectory | POI distribution | Road network | Regional air-quality | Traffic speed sensor readings |
| Models | Feature Engineering | RNN | CNN | GNN | CNN + RNN | GNN + RNN |


Sequence: Trajectories [Wang et al. 2019a]




Converting Trajectories into Video-like Data

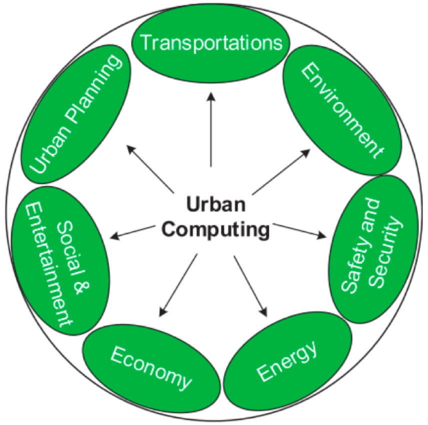
ST Maps: Urban Flow [Zhang et al. 2017]



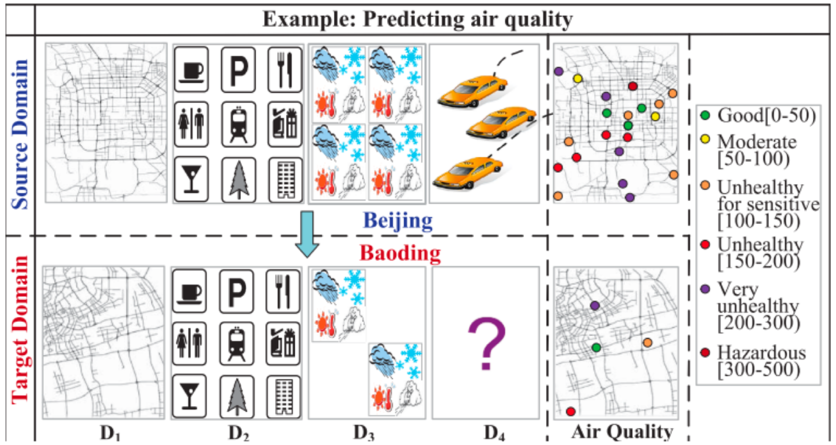

Graph: Traffic network [Yu et al. 2018]

# Urban Spatio-temporal Machine Learning: Data

- **Multi-modal** urban data:
  - e.g. Transport, environment, business [Zheng et al. 2014], etc.
- Leveraging multi-modal data is common in urban STML.
  - e.g. Use **road maps**, **points-of-interests** (POI), **weather**, and **transport** data to predict air quality [Wei et al. 2016a].



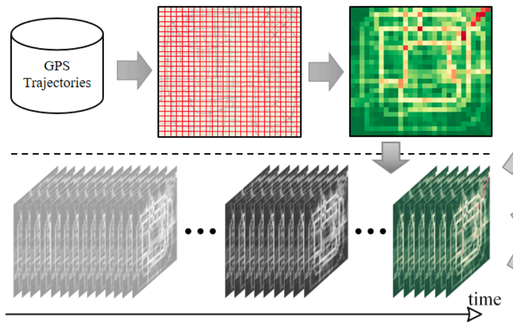

Multi-modal urban data [Zheng et al. 2014]




Use of multi-modal data [Wei et al. 2016a]

# Urban Spatio-temporal Machine Learning: Problems

- Supervised learning: $\min_{f:\mathcal{X}\to\mathcal{Y}} \sum_{i=1}^{|D|} l(f(\mathbf{X}_i), y_i),$
  - **Independent** samples, **label** supervision.
  - Minimize loss function $l(\hat{y}, y)$
    - Squared loss $l(\hat{y}, y) = \|\hat{y} - y\|^2$ for regression
    - Cross-entropy loss for classification $l(\hat{y}, y) = -\sum_i y_i \log \hat{y}_i$
  - Examples:
    - **Forecasting**: $\mathbf{X} = (\mathbf{X}_{t-k} \ldots \mathbf{X}_{t-1}), y = \mathbf{X}_t$
    - **Estimation (e.g. Travel Time)**: $\mathbf{X} = (\text{trajectory, metadata}), y = \text{travel time}$

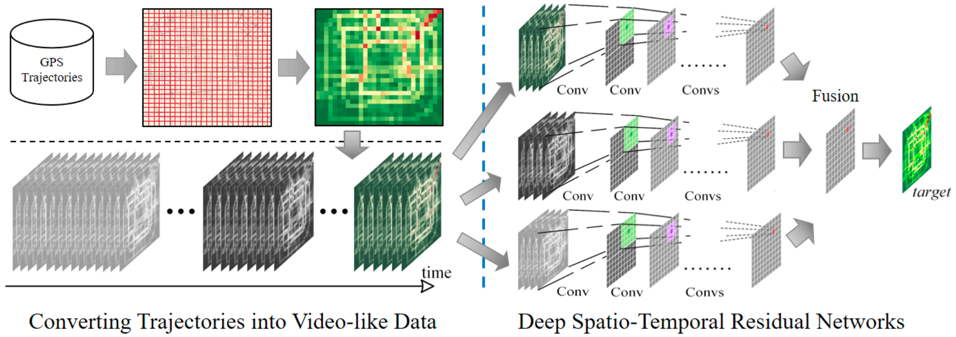


Converting Trajectories into Video-like Data

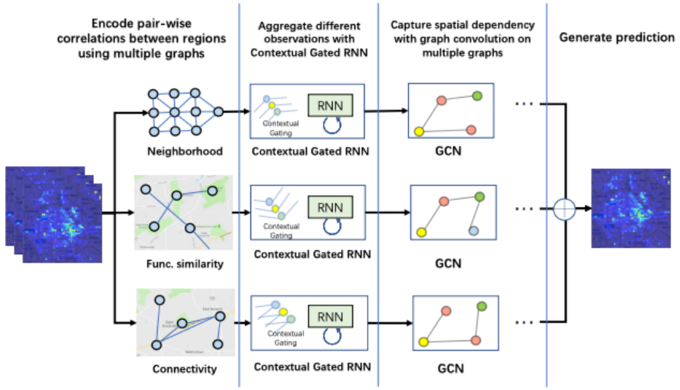
Urban flow forecasting [Zhang et al. 2017]

# Urban Spatio-temporal Machine Learning: Models

- Models capture spatio-temporal relations.
- **Spatial** relations:
  - Convolutional Neural Nets (**CNN**): **Euclidean**, grid data.
    - **Convolution** and **pooling** capture relations within a k*k grid.
    - Applications: Region flow forecasting [Zhang et al. 2017]
  - Graph Neural Nets (**GNN**): **Non-Euclidean**, network data
    - Aggregate information from irregular neighbors via edges.
    - Applications: Traffic forecasting [Geng et al. 2019]



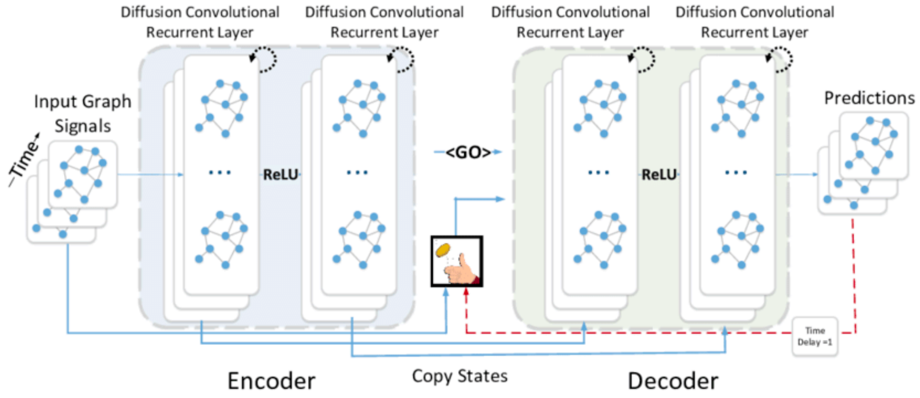

CNN for region flow forecasting [Zhang et al. 2017]




GNN for taxi demand forecasting [Geng et al. 2019]

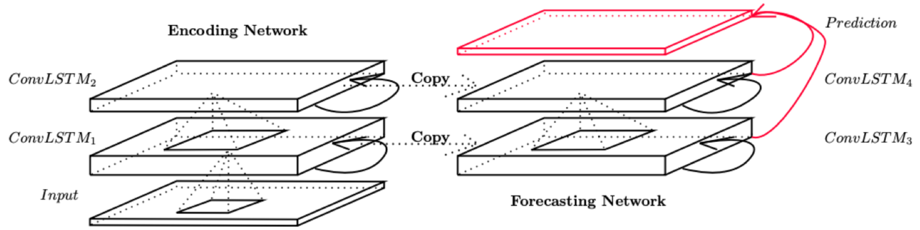
# Urban Spatio-temporal Machine Learning: Models

- **Temporal** correlations:
  - Recurrent Neural Nets (**RNN**)
    - Inputs share a **memory** to remember previous observations.




DCRNN [Li et al. 2018]

- Hybrid Models
  - Jointly model spatio-temporal correlations.
  - Combine RNN with CNN/GNN.
  - e.g. **DCRNN** [Li et al. 2018] for traffic forecasting **ConvLSTM** [Shi et al. 2015] for precipitation.

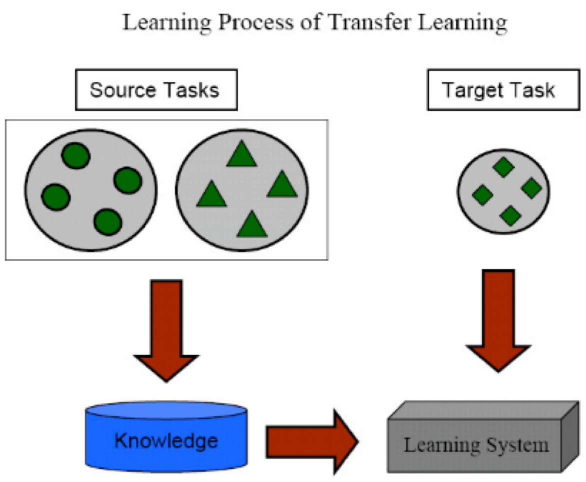



ConvLSTM [Shi et al. 2015]

# Outline

- Introduction
- Urban Spatio-temporal Machine Learning
- Transfer Learning
- Urban Spatio-temporal Transfer Learning
  - Spatial Transfer Learning
  - Temporal Transfer Learning
  - Cross-modal Transfer Learning
- Conclusions and Future Work

# Transfer learning




- **Definitions** [Pan et al. 2009]:
  - **Domain**: $\mathcal{D} = (\mathcal{X}, P(X))$
    Feature space and feature distribution.
  - **Task**: $\mathcal{T} = (\mathcal{Y}, P(y|X))$
    Label space and label conditional distribution.
  - **Transfer learning**: Improve learning on $\mathcal{D}_T, \mathcal{T}_T$ using knowledge from $\mathcal{D}_S, \mathcal{T}_S$
    $\mathcal{D}_S \neq \mathcal{D}_T$ or $\mathcal{T}_S \neq \mathcal{T}_T$
- **Key Challenge**: Identify **domain-invariant** knowledge.
- **Categorization**:
  - Homogeneous: $\mathcal{X}_S = \mathcal{X}_T, \mathcal{Y}_S = \mathcal{Y}_T$
  - Heterogeneous: $\mathcal{X}_S \neq \mathcal{X}_T$ or $\mathcal{Y}_S \neq \mathcal{Y}_T$

# Transfer learning: Methods

- Transfer learning methods: "**What to transfer**"
  - **Instance-based**: Reuse source instances to train on the target domain.
  - **Feature-based**: Learn domain-invariant features for both domains and learn a common downstream model.
  - **Model-based**: Encode knowledge in model parameters and reuse parameters for the target domain.

| Transfer Methodology | Key Assumption | Key Challenge | Categorization | Related Papers | Remarks |
| --- | --- | --- | --- | --- | --- |
| Instance-based | Some source samples are similar to target and can be reused. | Adequately re-weight source samples to train target model. | Non-inductive<br>Inductive | [35, 36, 37]<br>[38, 16] | |
| Feature-based | Source and target data share common latent factors. | Measure and maximize domain invariance of features. | Explicit (domain distance)<br>Implicit (adversarial) | [39, 40, 41, 42]<br>[43, 44, 45, 46] | MMD is a common distance metric. |
| Model-based | Model parameters encode common knowledge across domains. | Identify transferrability of model parameters. | Regularization<br>Fine-tuning | [47, 48, 49]<br>[50, 51, 30] | Fine-tuning popular in deep learning. |

# Transfer learning: Instance-based

- **Assumption**: Some samples from source are similar to target.
- **Key Challenge**: Assign weights to source samples to train target model.
- **Representative Method**: TrAdaBoost [Dai et al. 2007].
  - Intuition:
    - Wrong **target** samples**: Increase** weight
    - Wrong **source** samples**: Decrease** weight (Likely dissimilar from target).
  - Method:
    - Source and target samples: $D_S = \{X_i, y_i\}_{i=1}^{n}, D_T = \{X_j, y_j\}_{j=n+1}^{n+m}, y \in \{0, 1\}$

      Sample weights $\mathbf{w}^0 = [w_1^0, \ldots w_{n+m}^0]$.
    - Iterative Reweighting;

$$w_i^{t+1} = \begin{cases} \underbrace{\boxed{w_i^t \beta^{|h_t(X_i) - y_i|}, i \leq n,}}_{\text{Source}} \\ \underbrace{\boxed{w_i^t \beta_t^{-|h_t(X_i) - y_i|}, n+1 \leq i \leq n+m.}}_{\text{Target}} \end{cases} \quad 0 < \beta, \beta_t < 1.$$

# Transfer learning: Feature-based

- **Assumption**: Domains share common latent factors.
- **Challenge:** Measure & minimize domain distance to identify common factors.
- **Methods:** Minimize **M**aximum **M**ean **D**iscrepancy [Borgwardt et al. 2006]

$$\mathrm{MMD}\left(D_S, D_T\right) = \left\| \frac{1}{|D_S|} \sum_{i=1}^{|D_S|} \Phi_K\left(\mathbf{x}_i^S\right) - \frac{1}{|D_T|} \sum_{j=1}^{|D_T|} \Phi_K\left(\mathbf{x}_j^T\right) \right\|_{\mathcal{H}}, \quad \begin{array}{r} K: \text{Kernel function} \\ \Phi_K: \text{Kernel mapping of } K \\ \mathcal{H}: \text{RKHS of kernel } K. \end{array}$$

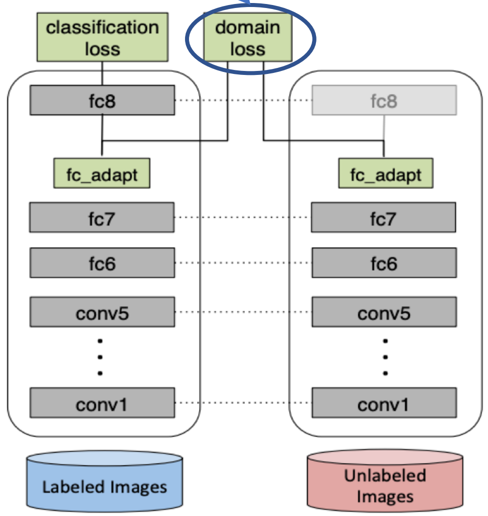
  - E.g. Transfer Component Analysis (TCA) [Pan et al. 2010]

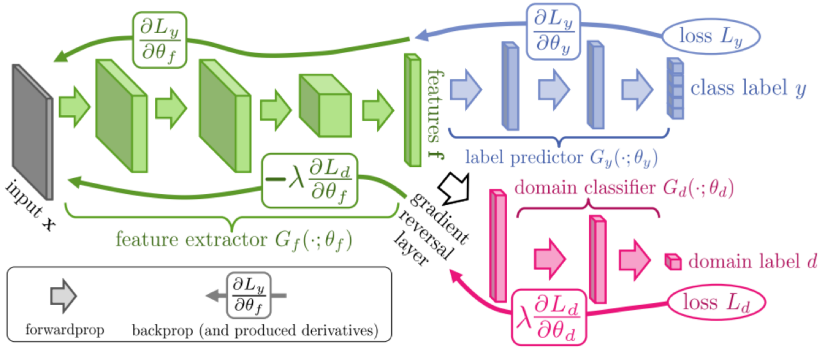
Reformulation of MMD²

$$\min_{\mathbf{W}} \operatorname{tr}(\mathbf{W}^T\mathbf{KLKW}) + \mu \operatorname{tr}(\mathbf{W}^T\mathbf{W})$$
$$\text{s.t.}\, \mathbf{W}^T\mathbf{KHKW} = \mathbf{I}$$

# Transfer learning: Feature-based (Cont)

- **Methods:** Minimize MMD
  - E.g. DDC [Tzeng et al. 2014]
    - Apply MMD regularization on the output features
    - Jointly minimize classification loss and MMD.
- **Methods**: Adversarial learning
  - Intuition: Domain invariant features should be
    1. **Discriminative** w.r.t. **labels**
    2. **Indiscriminative** w.r.t. **source VS target domain**
  - DANN [Ganin et al. 2016]
    - Feature Extractor $G$, Classifier $C$, Domain classifier $D$.
    - Minimize label loss $l_y(G,C)$: Intuition 1
    - Maximize domain loss $l_d(G,D)$: Intuition 2




$$\min_{G,C}\max_{D} V(G,C,D) = \frac{1}{n_s}\sum_{i=1}^{n_s} l_y^i(G,C) - \lambda\left(\frac{1}{n_s}\sum_{i=1}^{n_s} l_d^{i,s}(G,D) + \frac{1}{n_t}\sum_{i=1}^{n_t} l_d^{i,t}(G,D)\right),$$

# Transfer learning: Model-based

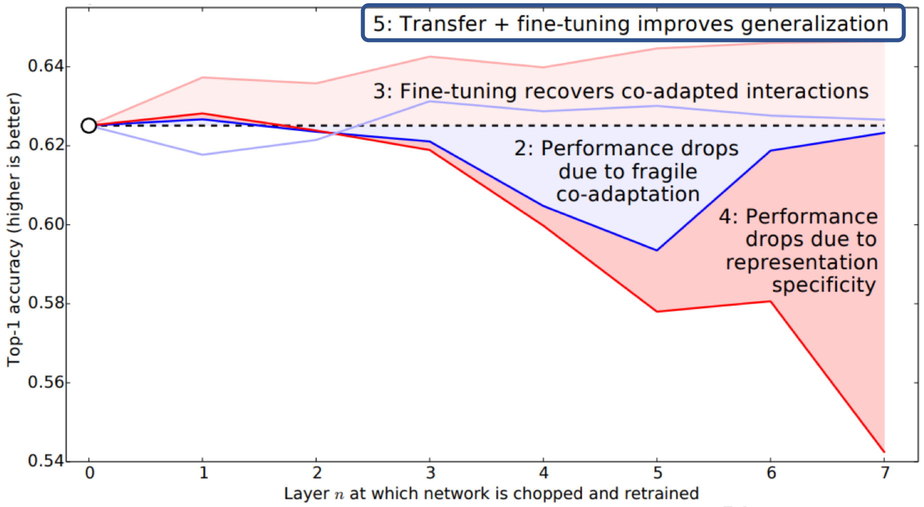
- **Assumption:** Model parameters encode general data structures.
- **Key Challenge:** Find transferrable parameters.
- **Methods:**
  - Train transferrable parameters with regularization
    - e.g. MT-SVM [Evgeniou and Pontil, 2004]
  - Identify transferrable parameters from **well-trained models**
    - [Yosinski et al. 2014]: Initialize from well-trained models improve generalization.
    - **Fine-tuning:** common in deep learning
    - **CV**: ImageNet pre-training.

      **NLP**: BERT [Devlin et al. 2019], GPT-3 [Brown et al. 2020] ···

$$\min_{\mathbf{w}_0,\mathbf{v}_S,\mathbf{v}_T,\xi_{t_i}} \sum_{t\in\{S,T\}} \sum_{i=1}^{n_t} \xi_{t_i} + \lambda_1(\mathbf{v}_S\|_2^2 + \|\mathbf{v}_T\|_2^2) + \lambda_2\|\mathbf{w}_0\|_2^2,$$

$$s.t. y_{S_i}(\mathbf{w}_0 + \mathbf{v}_S)^T\mathbf{X}_{S_i} \geq 1 - \xi_{S_i},$$

$$y_{T_i}(\mathbf{w}_0 + \mathbf{v}_T)^T\mathbf{X}_{T_i} \geq 1 - \xi_{T_i}, \xi_{t_i} \geq 0, t \in S, T.$$

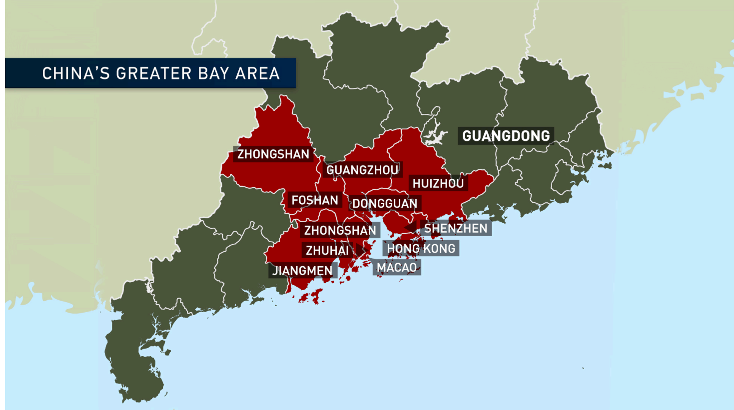


# Outline

- Introduction
- Urban Spatio-temporal Machine Learning
- Transfer Learning
- Urban Spatio-temporal Transfer Learning
  - Spatial Transfer Learning
  - Temporal Transfer Learning
  - Cross-modal Transfer Learning
- Conclusions and Future Work

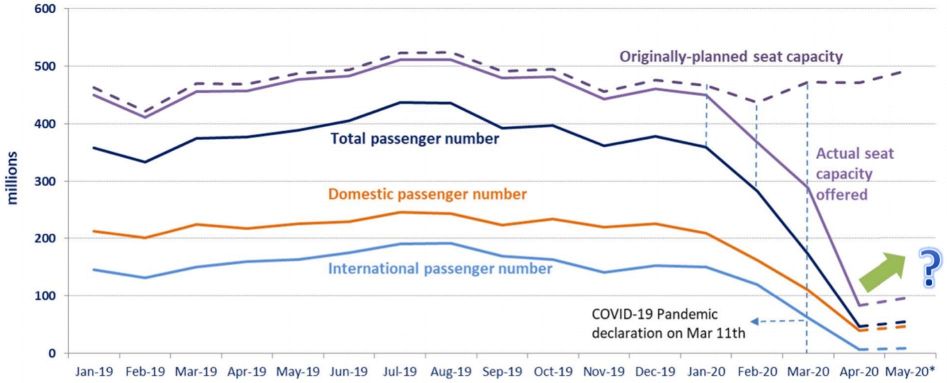
# Urban Spatio-temporal Transfer Learning

- **When to apply, what to transfer** in urban STTL?
  - Planning for new cities.
    - Transfer from existing cities: **spatial STTL.**
  - Planning for new urban services.
    - Transfer from existing urban services: **cross-modal STTL**.
  - Data evolution.
    - Adapt previous knowledge to the present: **temporal STTL**.




Development in Greater Bay Area:
**Spatial STTL**


Constructing a new MTR line:
**Cross-modal STTL**




Adaptation to COVID-19: **Temporal STTL**
[ICAO, 2020]

# Urban Spatio-temporal Transfer Learning

- **Key Challenges:**
  - Follows from urban STML & transfer learning.
  - Common **spatio-temporal** patterns
  - Common **multi-modal** knowledge

# Spatial STTL

- **Formulation**:
  - Target: city with limited data
  - Source: city with abundant data of the same type
  - Homogeneous TL: $P(X)_S \neq P(X)_Y \text{ or } P(y|X)_S \neq P(y|X)_T$
    - Differences in urban layout, etc.
- **Coarse** VS **Fine**-grained methods:
  - **Coarse**-grained methods: Treat each city as a whole and apply transfer learning
  - **Fine**-grained methods: Divide cities as small regions. Find **similar region pairs** and transfer between them.

# Spatial STTL: Coarse-grained

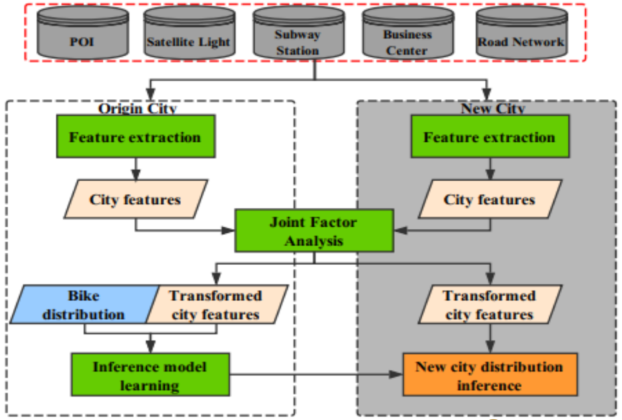
- **Feature-based**: Extract features and project into common space.
  - Commonly **multi-modal** features: POI, land usage, transportation, etc.

CoFA [Liu et al. 2018a]

- **Problem**: Inferring dockless bike distribution
- **Transfer method**: Factor Analysis,

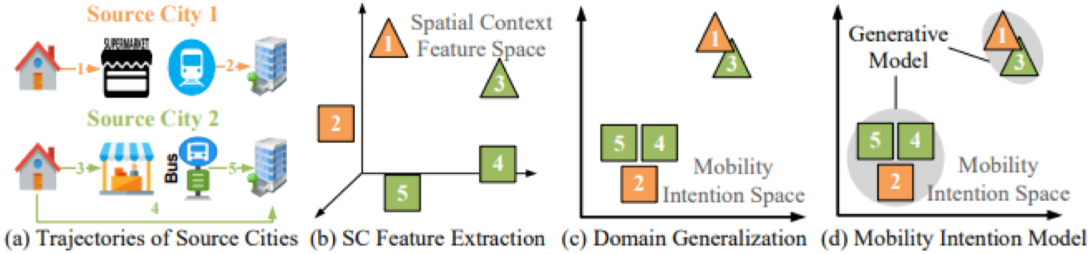
$$\mathbf{H}^*_{s:t}, \mathbf{W}^*, \mu^* = \arg \min_{\mathbf{H}_{s:t}, \mathbf{W}, \mu} \|\mathbf{M}_{s:t} - \mathbf{W}\mathbf{H}_{s:t} - \mu\|_F^2,$$

  - Project original features into a unified feature space by **minimizing reconstruction error**.




[Pang et al. 2020] and [He et al. 2020]

- **Problem**: Inferring human mobility in new cities
- **Transfer method**: TCA [Pan et al. 2010]



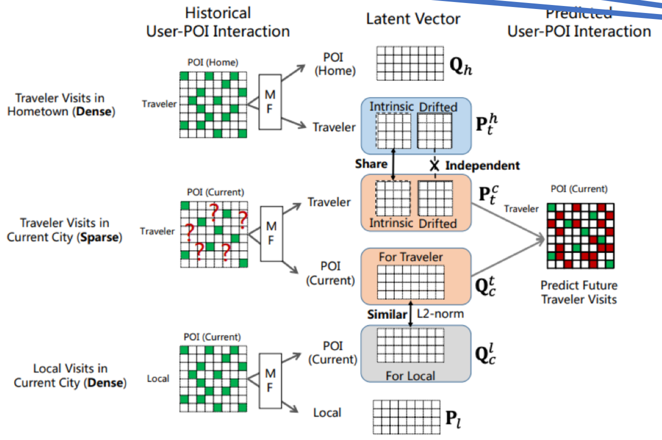
# Spatial STTL: Coarse-grained (Cont)

- **Model-based methods:** Train a model on source, re-use on target.
  - Two parts of models: **general** VS **city-specific**.

PR-UIDT [Ding et al. 2019]

- **Problem**: Cross-city, cross-user POI recommendation
- **Transfer method**: Regularization
  - Split both user and POI embeddings into **general** and **non-local** parts.
  - Regularize on the non-local part.

$$L_{PR} = \min_{\mathbf{P}_t^c, \mathbf{Q}_c^t, \mathbf{P}_t^h, \mathbf{Q}_h, \mathbf{P}_l, \mathbf{Q}_c^l} L_1(\mathbf{P}_t^c, \mathbf{Q}_c^t) + \alpha L_2(\mathbf{P}_t^h, \mathbf{Q}_h) + L_3(\mathbf{P}_l, \mathbf{Q}_c^l) + \beta(||\mathbf{Q}_c^t - \mathbf{Q}_c^l||_F^2),$$




Matrix Factorization Loss

Regularization on non-local part



香 港 科 技 大 學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

# Spatial STTL: Fine-grained

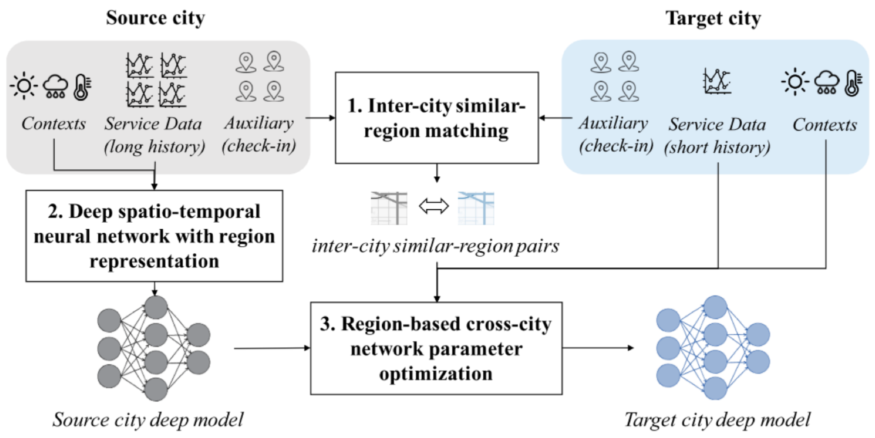
- Transfer may lead to higher error than non-transfer:
  - **Negative Transfer**: when domains differ a lot.
  - e.g. DC → NYC.
    DC is far less populated than NYC.

[Wang et al. 2019b]

| | D.C.→Chicago | | Chicago→D.C. | | D.C.→NYC | | NYC→D.C. | |
|---|---|---|---|---|---|---|---|---|
| | 1-day | 3-day | 1-day | 3-day | 1-day | 3-day | 1-day | 3-day |
| **Target Data Only** | | | | | | | | |
| *ARIMA* | 0.740 | 0.694 | 0.707 | 0.661 | 0.360 | 0.341 | 0.707 | 0.661 |
| *DeepST* | 0.771 | 0.711 | 1.075 | 0.767 | 0.350 | 0.359 | 1.075 | 0.767 |
| *ST-ResNet* | 0.914 | 0.703 | 0.869 | 0.738 | 0.376 | 0.349 | 0.869 | 0.738 |
| **Source & Target Data** | | | | | | | | |
| *DeepST (FT)* | 0.652 | 0.611 | 0.672 | 0.619 | 0.363 | 0.369 | 0.713 | 0.711 |
| *ST-ResNet (FT)* | 0.667 | 0.615 | 0.695 | 0.623 | 0.385 | 0.349 | 0.696 | 0.691 |

- Solution: Fine-grained methods
  - **Idea**: Cities may be dissimilar, but they must have **similar parts** (e.g. residential areas, business areas). These parts share **common knowledge**.
  - **Methodology**: Divide-and-match
    - Divide cities into smaller regions.
    - Obtain **similar region pairs**.
    - Transfer between similar regions instead of whole cities.

# Spatial STTL: Fine-grained




- Option 1: Using **fixed** similarity of **raw features $M_s, M_t$**.
  - Compute region-wise similarity $\rho_{r_s,r_t}$.
  - Minimize domain distance for matched regions $\Delta = \{(r_t, r_s), \forall r_t\}$,

CityTransfer [Guo et al. 2018]

- **Problem**: Cross-city site recommendation
  - Similarity Measure: Pearson correlation
  - Train autoencoder $f$ to minimize

$$\sum_{(r_t,r_s)\in\Delta} \rho_{r_t,r_s} \|f(\mathbf{M}_{r_s}) - f(\mathbf{M}_{r_t})\|^2$$

and use $f(M_s), f(M_t)$ for recommendation.

RegionTrans [Wang et al. 2019b]

- **Problem**: Spatio-temporal forecasting
  - Similarity Measure: Cosine similarity
  - Pre-train CNN-LSTM $f_2(f_1(x))$ on source.
  - Fine-tune $f_2(f_1(x))$ on target:

$$\sum_{r_t} \|y_{r_t} - f_2(f_1(x_{r_t}))\|^2 + \sum_{(r_s,r_t)\in\Delta} \rho_{r_s,r_t} \|f_1(x_{r_t}) - f_1(x_{r_s})\|^2$$

Target label

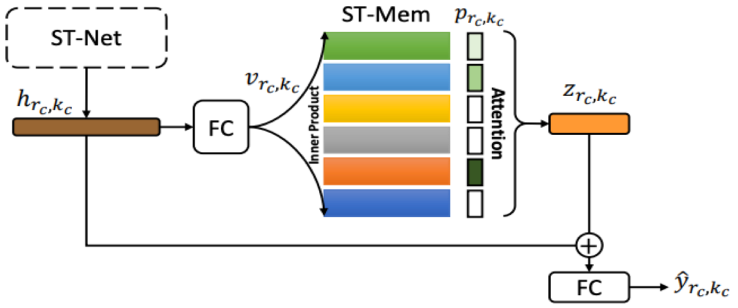
Matched domain distance

# Spatial STTL: Fine-grained (Cont)

- Option 2: Using **trainable** similarity of **output features**.

MetaST [Yao et al. 2019]

- **Problem**: Spatio-temporal forecasting.
- **Methods**: Transfer **attention** values
  - Cluster source regions using k-means
  - Set memory $M \in \mathbb{R}^{k \times f}$ for each cluster.
  - For source region $r_c$, use output $h_{r_c,k_c}$ to query $M$, get weights $p_{r_c,k_c}$. Match $p_{r_c,k_c}$ with its cluster id.
  - Use attention output $z_{r_c,k_c}$ as complement features for target regions.




WANT [Liu et al. 2019]

- **Problem**: Cross-city site recommendation
- **Methods**: Transferability **re-weighting**.
  - Architecture: DANN [Ganin et al. 2016]
  - Weight each source sample with target via:
    - **Domain similarity**: According to $D$.
    - **Data quality**: According to $(G, C)$.
  - Minimize **weighted** DANN loss:

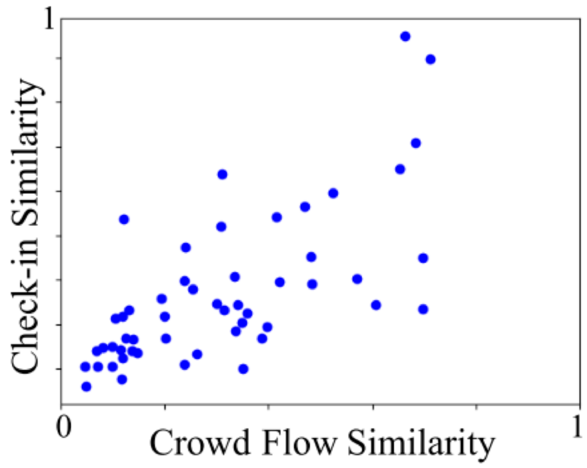
$$\min_{G,C} \max_{D} V(G,C,D) = \frac{1}{n_s} \sum_{i=1}^{n_s} w_i l_y^i(G,C) - \lambda \left( \frac{1}{n_s} \sum_{i=1}^{n_s} w_i l_d^{i,s}(G,D) + \frac{1}{n_t} \sum_{i=1}^{n_t} l_d^{i,t}(G,D) \right),$$

# Spatial STTL: Fine-grained (Cont)

- Summary of fine-grained methods

| Related Work | Task | Matching Data | Matching Metric | Trainable Matching |
|---|---|---|---|---|
| [Wang et al. 2019b] | Spatio-temporal Forecasting | Raw features | Cosine Similarity | No |
| [Yao et al. 2019] | | Output features | Implicitly via Attention | **Yes** |
| [Guo et al. 2018] | Cross-city site recommendation | Raw features | Pearson correlation | No |
| [Liu et al. 2021] | | Output features | Implicitly via domain and label classifier | **Yes** |

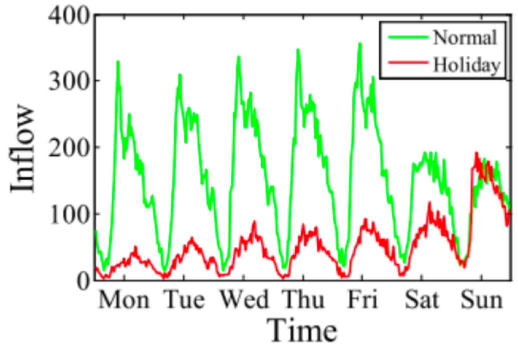
- Raw feature matching:
  - **Pro**: Stable. Incorporates multi-modal information.
  - **Con**: Performance relies on quality of features.
- Output feature matching;
  - **Pro**: Flexible, trainable matching.
  - **Con**: Limited target data → overfit.



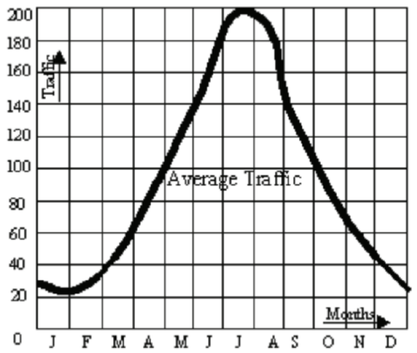

[Wang et al. 2019b] **Check-in** (auxiliary) is related to **crowd flow** (limited).

# Temporal STTL

- Temporal STTL tackles data **distribution shifts**.
  - Two kinds of distribution shifts
- **Formulation**:
  - Homogeneous TL
  - Source: Previous periods
  - Target: Current period




**Sudden changes,** e.g. holiday [Zhang et al. 2017]




**Long-term (monthly) changes** [Karacasu et al. 2011]

- **Related Concept**: **Continual** learning [Chen and Liu, 2018].
  - A series of domains and tasks $\mathcal{D}_i, \mathcal{T}_i, i = 1 \ldots N,$
  - For each $n$, learn task $\mathcal{T}_n$ with knowledge from $\mathcal{T}_i, i < n$ .
  - In temporal STTL, each timestamp defines a task.



香 港 科 技 大 學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

# Temporal STTL for Indoor Localization

- Indoor localization: signal strength $\mathbf{s} = (s_1, \ldots s_m) \rightarrow$ location $(x, y)$
- Temporal TL for indoor localization:
  - **Goal**: At $t_k$, adapt $f_1$ to $f_k$ using $\{\mathbf{s}_{ki}, l_{ki}\}_{i=1}^{l}$, $\{\mathbf{s}_{kj}\}_{j=l+1}^{l+u}$
  - The first $l$ labeled data are from **fixed** positions: **Reference points.**

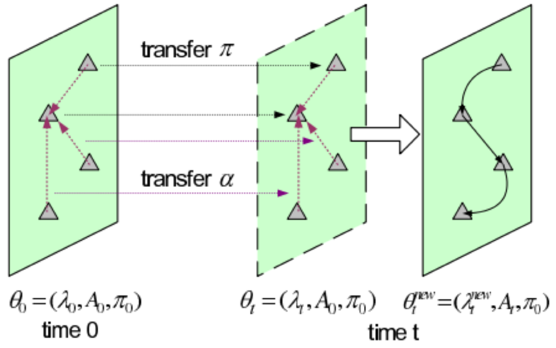
- **Feature-based** method: LeManCoR [Pan et al. 2007]
  - **L**ocalization with **e**xtended **Man**ifold **Co**-**R**egularization
  - **Assumption**: Models from different times **agree on reference points.**

$$f_1^*, f_k^* = \arg\min_{f_1 \in H_{K_1}, f_k \in H_{K_k}} \frac{\mu}{l_1} \sum_{i=1}^{l_1} V(\mathbf{s}_{1i}, l_{1i}, f_1) + \gamma_A^{(1)} \|f_1\|_{H_{K_1}} + \gamma_B^{(1)} \|f_1\|_I$$
$$+ \frac{1}{l} \sum_{i=1}^{l} V(\mathbf{s}_{ki}, l_{ki}, f_k) + \gamma_A^{(2)} \|f_k\|_{H_{K_k}} + \gamma_B^{(2)} \|f_k\|_I + \frac{\gamma_I}{l} \sum_{i=1}^{l} [f_1(\mathbf{s}_{1i}) - f_k(\mathbf{s}_{ki})]^2,$$

$|f|_H$: Complexity regularization
$|f|_I$: Manifold regularization

# Temporal STTL for Indoor Localization (Cont)

- **Model-based** method: TrHMM [Zheng et al. 2008]
  - Transferring Hidden Markov Models: $(L, O, \lambda, A, \pi)$
    - $L, O$: Location & observation space
    - $\lambda = P(o|l) = N(\mu, \Sigma), l \in L, o \in O$
    - $A$: Transition matrix between locations $L$. $\pi$: Initial distribution over $L$




  - **Assumption**: **Fixed relations** between reference (r) and other points (k) across $t$.
  - Method:
    1. Estimate $\lambda_1$ at time $t_1$.
    2. Obtain relations between reference and other points via regression.

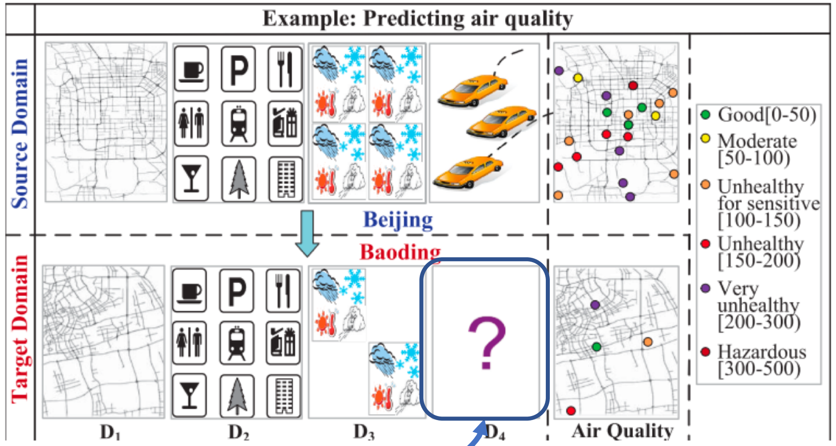
$$s_j^k = \alpha_{0j}^k + \alpha_{1j}^k r_{1j} + \ldots \alpha_{lj}^k r_{lj} + \varepsilon_j,$$

    3. At time $t$, **re-use** the **regression** model to reconstruct data, and **fine-tune** $\lambda_t$

$$\mu_t = \beta \mu_1 + (1 - \beta) \mu_t^{reg},$$

$$\Sigma_t = \beta(\Sigma_1 + (\mu_t - \mu_1)(\mu_t - \mu_1)^T) + (1 - \beta)(\Sigma_t^{reg} + (\mu_t - \mu_t^{reg})(\mu_t - \mu_t^{reg})^T).$$

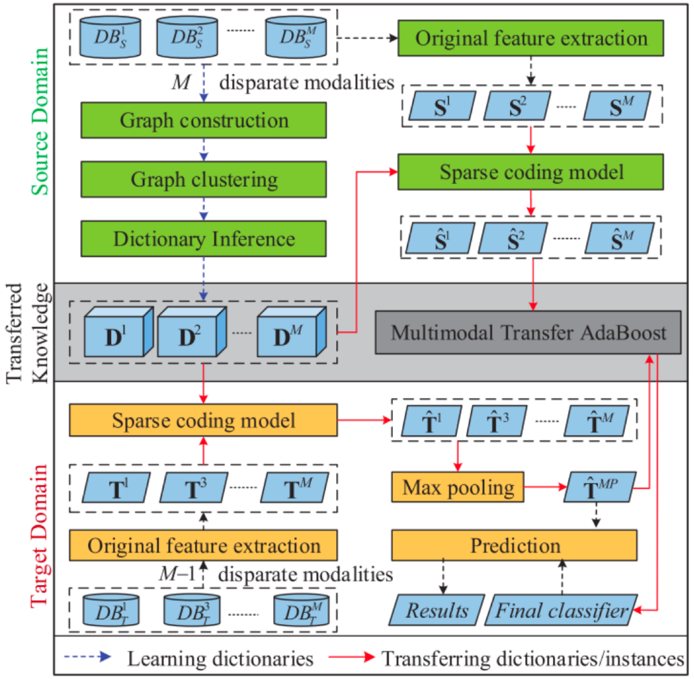
# Cross-modal STTL




- **Two scenarios** of cross-modal STTL:
  - Missing feature modality:
    - e.g. Predict air quality using **road maps**, **POI, weather**, and **transport** data
      transport data is missing in a city.
    - **Solution**: Learn the **relation** between **existing** modalities and the **missing** one.
    - Heterogeneous TL: $\mathcal{X}_S \neq \mathcal{X}_T$
  - Missing label modality:
    - e.g. Detect ride-sharing car trajectories with **no labeled trajectories**.
    - **Solution**: Find related data (e.g. taxis), and **link source labels with target ones**.
    - Heterogeneous TL: $\mathcal{Y}_S \neq \mathcal{Y}_T$

# Cross-modal STTL: Missing feature modality

- FLORAL [Wei et al. 2016a]
  - **Assumption**: **Relations between modalities** are invariant across domains.
  - **Key Idea**: Extract and transfer such relations
  - **Methods**:
    - **Relation encoding:** Build sample-modality graph
      **Nodes**: Each sample in each modality
      **Intra-modality edges:** feature distance
      **Inter-modality edges**: sample proximity
    - **Relation learning:** Cluster sample-modality graph. Each cluster forms **base vectors** in **dictionaries**.
    - **Transfer**: Share dictionaries across domains
      **Sparse coding** to obtain domain-invariant features.
      **Multi-modal TrAdaBoost** to reweight modalities.



# Cross-modal STTL: Missing label modality

- CoHTL [Wei et al. 2016b]
  - **Target**: sensor data with few labels. **Source**: Social messages
  - **Key Idea**: **Link samples** & **labels** from social messages to sensors.
  - **Methods**:
    - **Label linking**: Topic model & word embedding.
    - **Sample linking**: Spatio-temporal proximity.
    - **Feature-based transfer:** Link regularization.

| eat | exercise | travel |
|---|---|---|
| have dinner | exercise | go outdoor |
| break-fast | exercise | buy tickets |

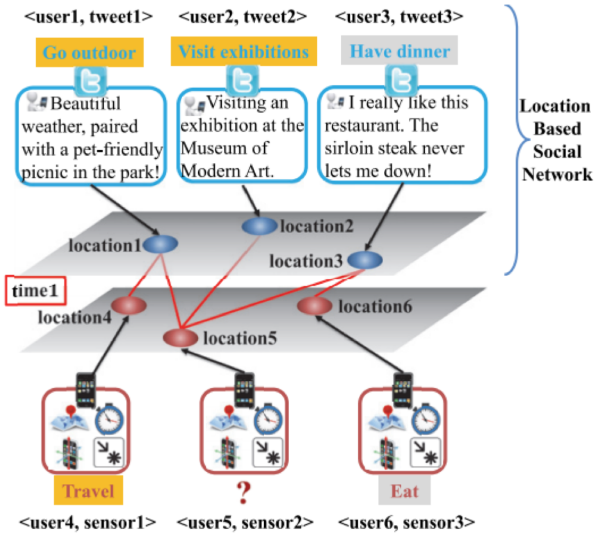
Results of label linking by word embeddings.

$$\mathcal{O} = \boxed{\|\mathbf{P} - \mathbf{U}\mathbf{V}_1\|_F^2 + \|\mathbf{Q} - \mathbf{W}\mathbf{V}_2\|_F^2}$$
$$\boxed{+ \beta \sum_{i=1}^{m} \sum_{j=1}^{n} S_{ij} \|\mathbf{u}_i - \mathbf{w}_j\|_2^2} + \gamma R(\mathbf{U}, \mathbf{W}, \mathbf{V}_1, \mathbf{V}_2),$$

Reconstruction error

Regularization by sample links ($S_{ij}$).

Example of spatio-temporal sample linking



# Summary of urban STTL

- **Spatial**: Homogeneous TL
  - **Coarse** VS **Fine**-grained methods: transfer between **similar** parts
- **Temporal:** Homogeneous TL,
  - **Example:** Temporal STTL for indoor localization (via reference points)
- **Cross-modal: Heterogeneous** TL
  - **Cases:** Missing feature/label modality
  - **Key:** Relations between modalities.

| Urban STTL Setting | Existing Works | Common ST Patterns | | | | | Common Multi-modal Knowledge | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | Coarse-grained | | Fine-grained | | | Feature | Computing | Sparse | Co- |
| | | Feature | Model | Instance | Feature | Model | Concatenation | Similarity | Coding | training |
| Spatial Transfer Learning | Liu et al. [63] | ✓ | | | | | ✓ | | | |
| | Liu et al. [64] | ✓ | | | | | ✓ | | | |
| | Pang et al. [65] | ✓ | | | | | ✓ | | | |
| | He et al. [66] | ✓ | | | | | ✓ | | | |
| | Li et al. [67] | ✓ | | | | | | | | |
| | Ding et al. [68] | | ✓ | | | | | | | |
| | Wang et al. [69] | | ✓ | | | | | | | |
| | Wang et al. [70] | | | | ✓ | ✓ | | ✓ | | |
| | Yao et al. [71] | | | | | ✓ | | | | |
| | Song et al. [72] | | | | ✓ | ✓ | | ✓ | | |
| | Mallick et al. [73] | | | | | ✓ | | | | |
| | Guo et al. [74] | | | | ✓ | | ✓ | ✓ | | |
| | Liu et al. [75] | | | ✓ | ✓ | | ✓ | | | |
| | Wang et al. [22] | | | ✓ | ✓ | | ✓ | | | |
| Temporal Transfer Learning | Pan et al. [55] | | | | ✓ | | | | | |
| | Zheng et al. [76] | | | | | ✓ | | | | |
| Cross-modal Transfer Learning | Wei et al. [77] | | | ✓ | ✓ | | | | ✓ | |
| | Wang et al. [78] | | | ✓ | | ✓ | | | | ✓ |
| | Wei et al. [79] | | | | ✓ | | | ✓ | | |

# Conclusion

- **Motivation**: **Urban computing** + **machine learning** meets **lack of data**.
- **Challenge**: Common ST patterns & Common multi-modal knowledge
- **Categorization**: Spatial, temporal and cross-modal
- **Future directions**:
  - Transfer learning with **effective multi-modal fusion**:
    - Existing works mainly use feature concat or feature similarity.
  - Transfer learning with **dynamics**:
    - Temporal STTL with detection of data shifts.
    - Adaptive knowledge transfer at different periods.
  - Transfer learning with **privacy**
    - Spatio-temporal data may contain user privacy.

# References


- [UN, 2015] UN, DESA. "World urbanization prospects: The 2014 revision." *United Nations Department of Economics and Social Affairs, Population Division: New York, NY, USA* 41 (2015).
- [Zhang et al. 2020] Zhang, Yingxue, et al. "Curb-GAN: Conditional Urban Traffic Estimation through Spatio-Temporal Generative Adversarial Networks." Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020.
- [Liang et al. 2021] Liang, Yuxuan, et al. "Fine-Grained Urban Flow Prediction." Proceedings of the Web Conference 2021. 2021.
- [Sun et al. 2017] Sun, Chen, et al. "Revisiting unreasonable effectiveness of data in deep learning era." Proceedings of the IEEE international conference on computer vision. 2017.
- [Pan et al. 2009] Pan, Sinno Jialin, and Qiang Yang. "A survey on transfer learning." IEEE Transactions on knowledge and data engineering 22.10 (2009): 1345-1359.
- [Li 2020] Li, Zheng. Neural knowledge transfer for low-source sentiment analysis: cross-domain, cross-task & cross-lingual. Doctoral Dissertation, HKUST. 2020.
- [Shen et al. 2019] Shen, Zhiqiang, et al. "Object detection from scratch with deep supervision." IEEE transactions on pattern analysis and machine intelligence 42.2 (2019): 398-412.
- [Zhou et al. 2020] Zhou, Zhengyang, et al. "RiskOracle: A Minute-Level Citywide Traffic Accident Forecasting Framework." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 34. No. 01. 2020.

# References


- [Wang et al. 2019a] Wang, Leye, et al. "Ridesharing car detection by transfer learning." Artificial Intelligence 273 (2019): 1-18.
- [Xu et al. 2018] Xu, Zhe, et al. "Large-scale order dispatch in on-demand ride-hailing platforms: A learning and planning approach." Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2018.
- [Zhang et al. 2017] Zhang, Junbo, Yu Zheng, and Dekang Qi. "Deep spatio-temporal residual networks for citywide crowd flows prediction." Thirty-first AAAI conference on artificial intelligence. 2017.
- [Li et al. 2018] Li, Yaguang et al. , Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting, International Conference on Learning Representations, 2018.
- [Shi et al. 2015] Shi, Xingjian et al. "Convolutional LSTM network: A machine learning approach for precipitation nowcasting." Advances in neural information processing systems. 2015.
- [Dai et al. 2007] Dai, Wenyuan et al. Boosting for transfer learning. Proceedings of the 24th international conference on Machine learning (ICML '07). 2007.
- [Pan et al. 2010] Pan, Sinno Jialin, et al. "Domain adaptation via transfer component analysis." IEEE transactions on neural networks 22.2 (2010): 199-210.
- [Borgwardt et al. 2006] Borgwardt, Karsten M., et al. "Integrating structured biological data by kernel maximum mean discrepancy." Bioinformatics 22.14 (2006): e49-e57.
- [Tzeng et al. 2014] Tzeng, Eric, et al. "Deep domain confusion: Maximizing for domain invariance." arXiv preprint arXiv:1412.3474 (2014).

# References


- [Yu et al. 2018] Yu, Bing, Haoteng Yin, and Zhanxing Zhu. "Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting." Proceedings of the 27th International Joint Conference on Artificial Intelligence. 2018.
- [Zheng et al. 2014] Zheng, Yu, et al. "Urban computing: concepts, methodologies, and applications." ACM Transactions on Intelligent Systems and Technology (TIST) 5.3 (2014): 1-55.
- [Wei et al. 2016a] Wei, Ying, Yu Zheng, and Qiang Yang. "Transfer knowledge between cities." Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016.
- [Ganin et al. 2016] Ganin, Yaroslav, et al. "Domain-adversarial training of neural networks." The journal of machine learning research 17.1 (2016): 2096-2030
- [Evgeniou and Pontil, 2004] Evgeniou, Theodoros, and Massimiliano Pontil. "Regularized multi--task learning." Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. 2004.
- [Yosinski et al. 2014] Yosinski, Jason, et al. "How transferable are features in deep neural networks?." Advances in Neural Information Processing Systems 27 (2014): 3320-3328.
- [Devlin et al. 2019] Devlin, Jacob, et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). 2019.
- [Brown et al. 2020] Brown, Tom, et al. Language Models are Few-Shot Learners, Advances in Neural Information Processing Systems 33 (2020)
- [Rosenstein et al. 2005] Rosenstein, Michael T., et al. "To transfer or not to transfer." NIPS 2005 workshop on transfer learning. Vol. 898. 2005.

# References


- [Liu et al. 2018a] Liu, Zhaoyang, Yanyan Shen, and Yanmin Zhu. "Inferring dockless shared bike distribution in new cities." Proceedings of the eleventh ACM international conference on web search and data mining. 2018.
- [Liu et al. 2018b] Liu, Zhaoyang, Yanyan Shen, and Yanmin Zhu. "Where Will Dockless Shared Bikes be Stacked? ---Parking Hotspots Detection in a New City." Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2018.
- [Pang et al. 2020] Pang, Yanbo, et al. "Intercity Simulation of Human Mobility at Rare Events via Reinforcement Learning." Proceedings of the 28th International Conference on Advances in Geographic Information Systems. 2020.
- [He et al. 2020] He, Tianfu, et al. "What is the human mobility in a new city: Transfer mobility knowledge across cities." Proceedings of The Web Conference 2020. 2020.
- [Ding et al. 2019] Ding, Jingtao, et al. "Learning from hometown and current city: Cross-city POI recommendation via interest drift and transfer learning." Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 3.4 (2019): 1-28.
- [Wang et al. 2018] Wang, Zhaodong, et al. "Deep reinforcement learning with knowledge transfer for online rides order dispatching." 2018 IEEE International Conference on Data Mining (ICDM). IEEE, 2018.
- [Wang et al. 2019b] Wang, Leye, et al. "Cross-city Transfer Learning for Deep Spatio-temporal Prediction." IJCAI International Joint Conference on Artificial Intelligence. 2019.

# References


- [Guo et al. 2018] Guo, Bin, et al. "Citytransfer: Transferring inter-and intra-city knowledge for chain store site recommendation based on multi-source urban data." Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 1.4 (2018): 1-23.
- [Yao et al. 2019] Yao, Huaxiu, et al. "Learning from multiple cities: A meta-learning approach for spatial-temporal prediction." The World Wide Web Conference. 2019.
- [Liu et al. 2021] Liu, Yan, et al. "Knowledge Transfer with Weighted Adversarial Network for Cold-Start Store Site Recommendation." ACM Transactions on Knowledge Discovery from Data (TKDD) 15.3 (2021): 1-27.
- [Karacasu et al. 2011] Karacasu, Murat, et al. "Variations in traffic accidents on seasonal, monthly, daily and hourly basis: Eskisehir case." Procedia-Social and Behavioral Sciences 20 (2011): 767-775.
- [Chen and Liu 2018] Chen, Zhiyuan, and Bing Liu. "Lifelong machine learning." Synthesis Lectures on Artificial Intelligence and Machine Learning 12.3 (2018): 1-207.
- [Pan et al. 2007] Pan, Sinno Jialin, et al. "Adaptive localization in a dynamic WiFi environment through multi-view learning." AAAI. Vol. 7. 2007.
- [Zheng et al. 2008] Zheng, Vincent Wenchen, et al. "Transferring Localization Models over Time." AAAI. 2008.
- [Wei et al. 2016b] Wei, Ying, et al. "Instilling social to physical: Co-regularized heterogeneous transfer learning." Thirtieth AAAI Conference on Artificial Intelligence. 2016.