Spatial AI and Its Applications

Air Quality Prediction and Machines Reading Maps

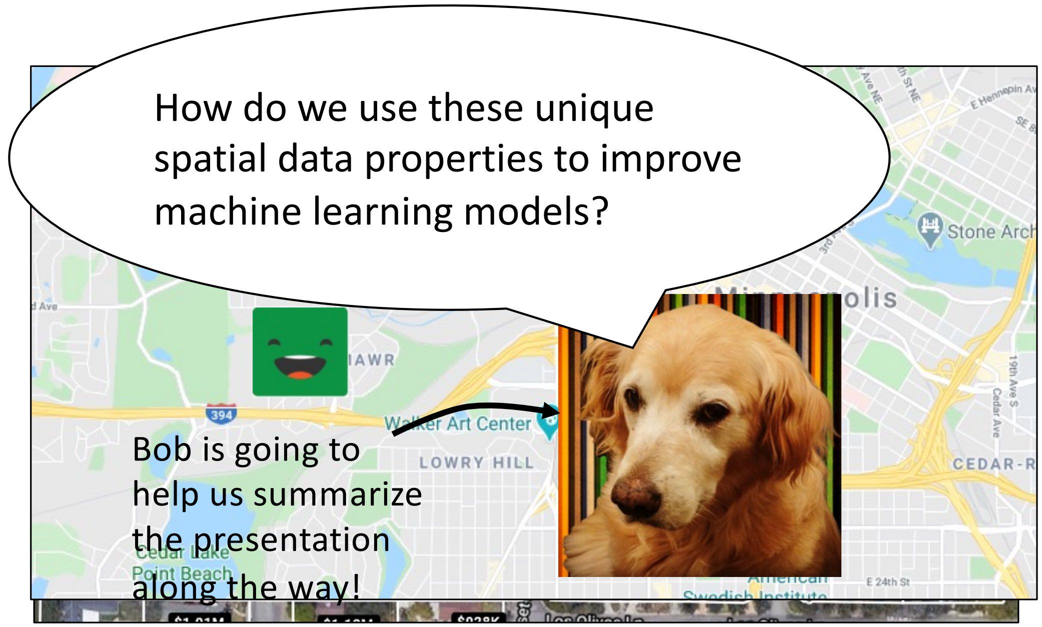
Yao-Yi Chiang

Associate Professor, Computer Science & Engineering Department
University of Minnesota, Twin Cities
yaoyi@umn.edu

Knowledge Computing Lab
https://knowledge-computing.github.io/people.html

What are spatial AI methods?

- Machine learning & data mining methods generally assume independent and identically distributed random variables – i.i.d.
- **But** spatial data are not i.i.d.
- Auto-correlation
  - Nearby things are similar
- Spatial non-stationarity
  - Models are difficult to generalize
- And more…



Air quality near highway I-394 can be very different depending on their locations

Nearby houses have similar prices

Why do we care?

- **Location** is the key to link various types of data
  - e.g. can provide context-rich annotated (training) data (if we do it correctly)

"By using the data in OSM, we were able to collect more than 100 million labeled examples to add to our training data set."
"However, using OSM data for labels presented several challenges that required novel approaches to overcome."

~~Facebook~~ Map with AI

https://ai.facebook.com/blog/mapping-the-world-to-help-aid-workers-with-weakly-semi-supervised-learning/

What are we building?

Real-World Problems & Data

- Health
- Transportation
- Social Sciences

Impact: Tech. Transfer & Open Source Tools In-Use

Metro
GeoSemble
Crosstown
Google AI
The Alan Turing Institute
TERRAGO
CLS Risk Solutions

End-to-End Data Analytics Systems

- **DeepLATTE:** Air Quality Prediction & Forecasting Framework
- **Strabo & mapKurator:** Map Processing Platform
- **ADMS:** Transportation Data Analytics Platform (joint work with USC IMSC)
- …

Fundamental Research

ACM SIGSPATIAL, ACM KDD, IEEE International Conference on Data Mining, IEEE BigData, IEEE Mobile Data Management, Extended Semantic Web Conference, International Journal of Geographical Information Science, Knowledge-Based Systems

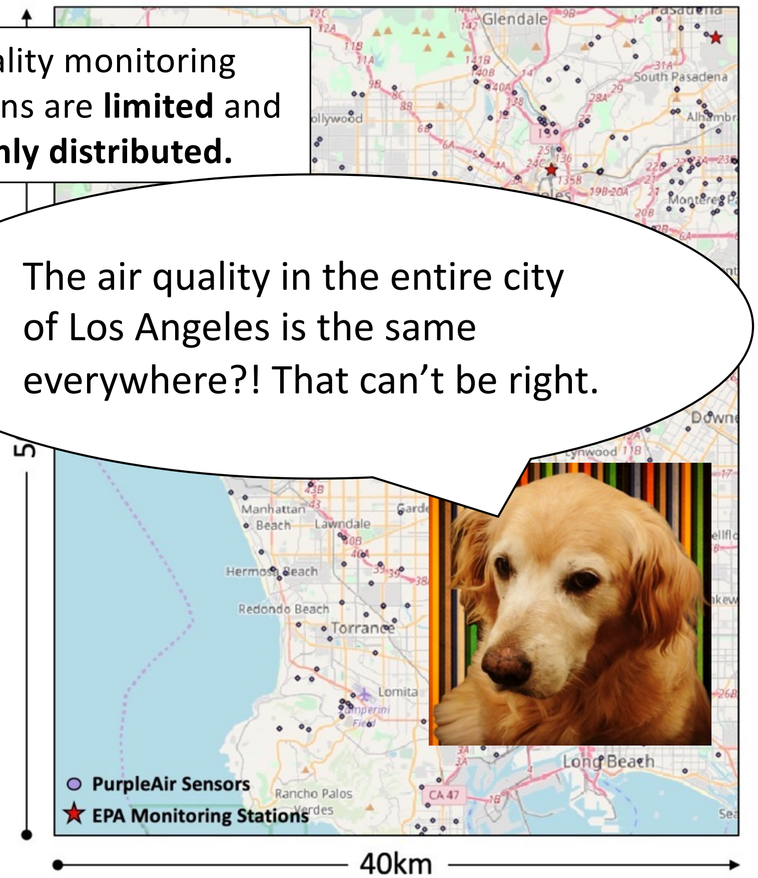
Building Autocorrelation-Aware Representations for Fine-Scale Air Quality Prediction

Lin et al., 2020 (IEEE International Conference on Data Mining)

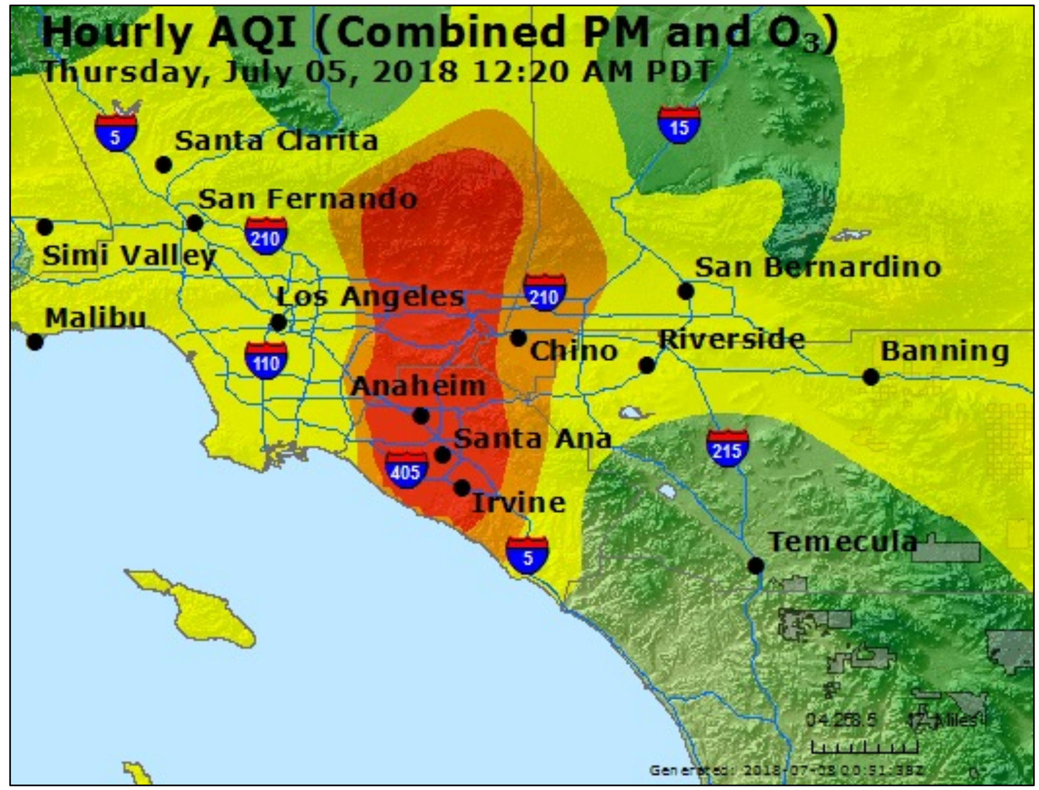
Motivation

Air quality monitoring locations are **limited** and **unevenly distributed.**

The air quality in the entire city of Los Angeles is the same everywhere?! That can't be right.

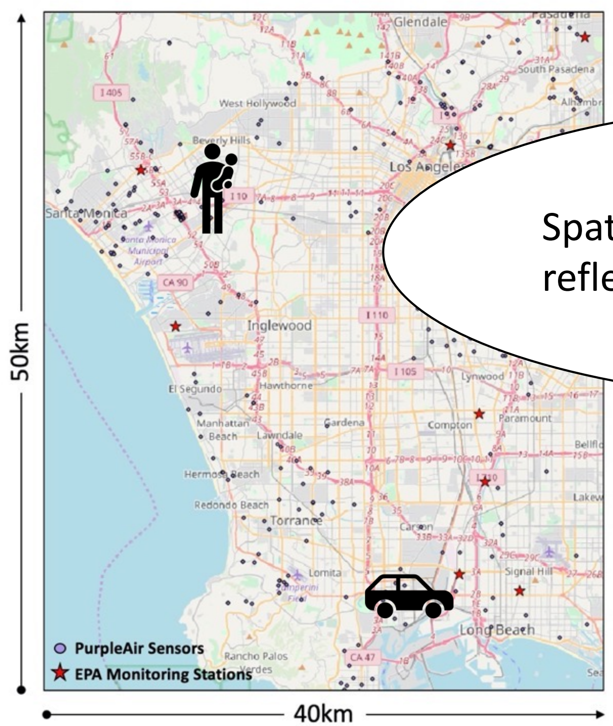


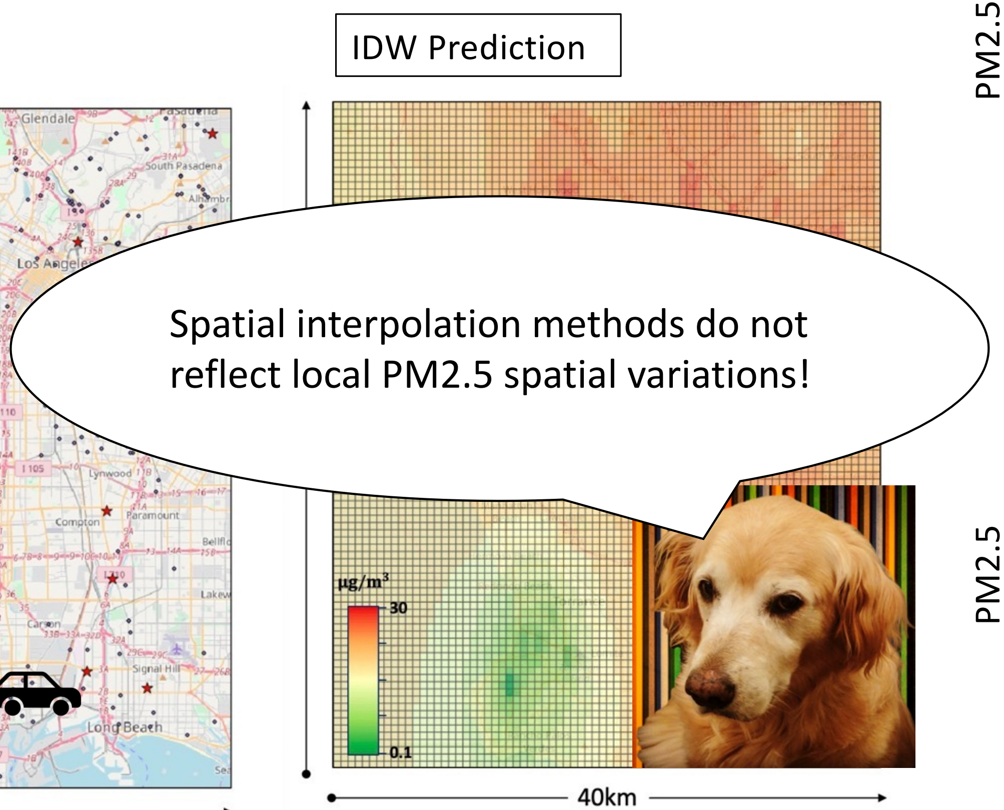
Traditional spatial interpolation methods (e.g., IDW – inverse distance weighting, Kriging) produce smooth results over the space.

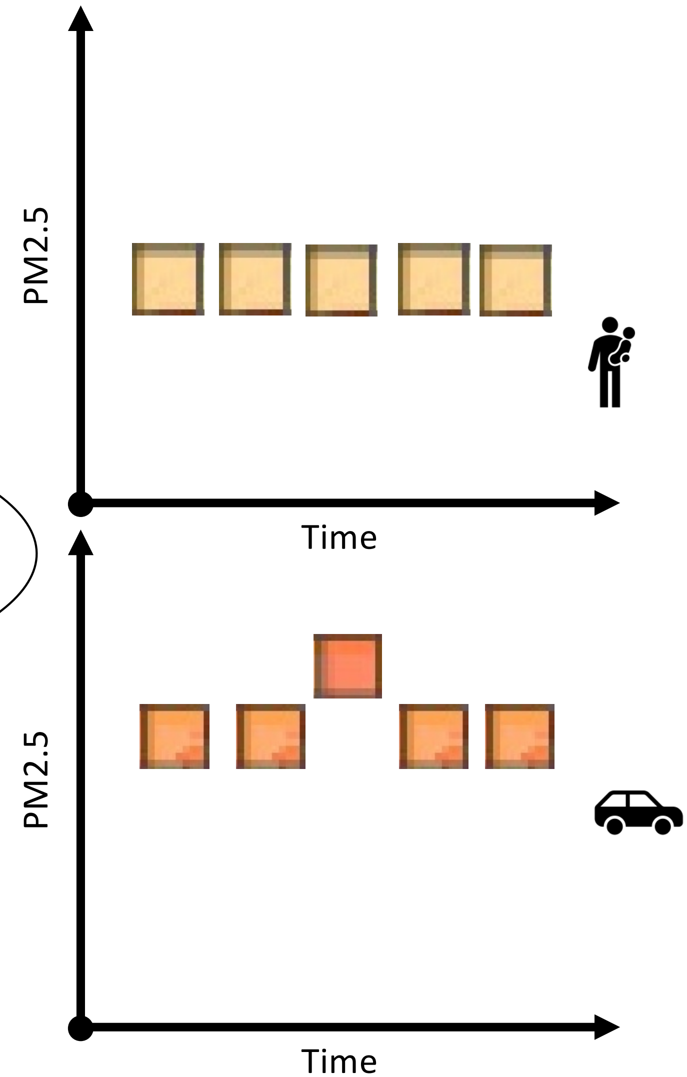


https://airnow.gov/

Not suitable for exposure tracking

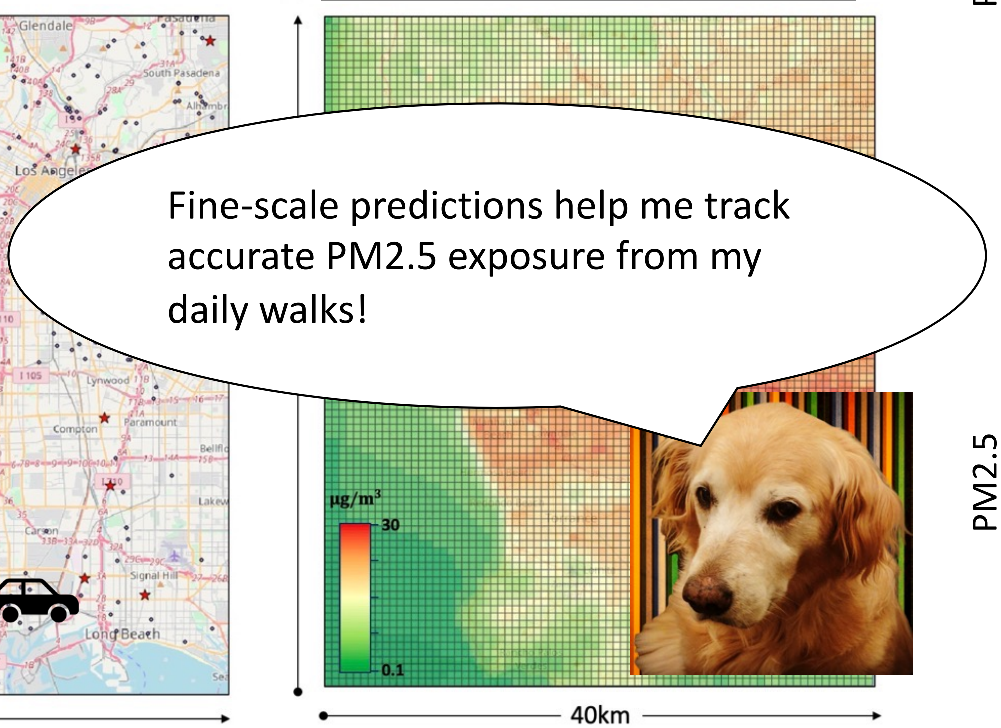


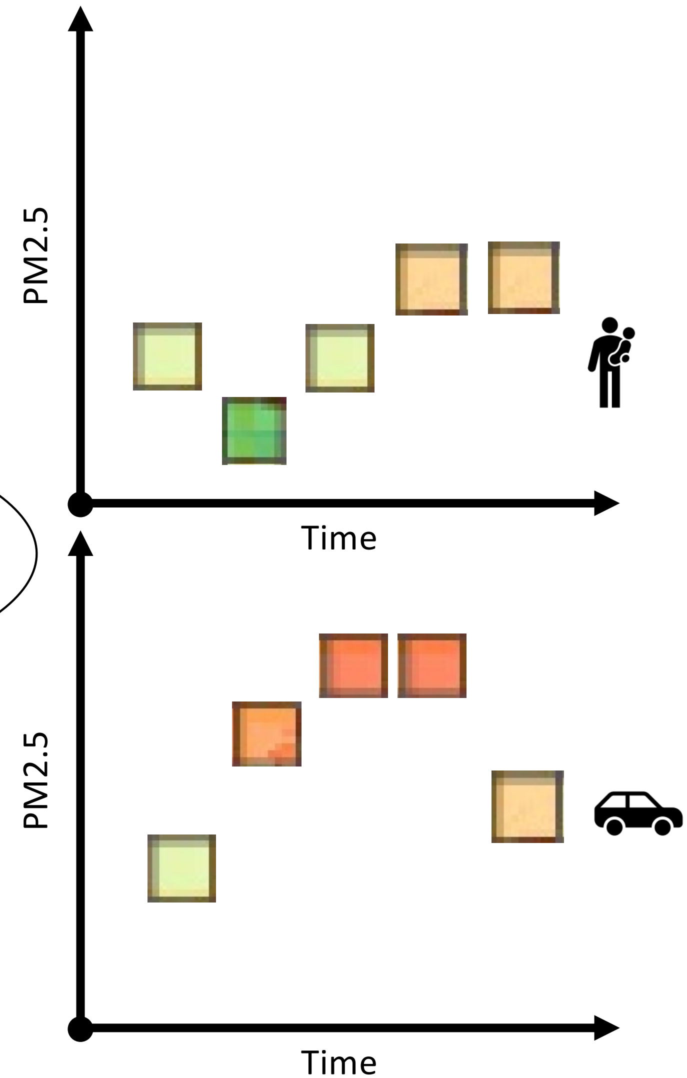




What we need:

Air quality prediction that reflect the environment impact





Hypothesis

- Environmental characteristics significantly impact air quality (e.g., PM2.5)

Authors	Study area	Monitor counts	Dependent variables	Independent variables	Buffer size	(Adjusted) R^2
Briggs et al. (2000)	Huddersfield (UK) Sheffield (UK) Northampton (UK)	20, 28 and 35	NO_2	Road traffic, urban land, and topography (altitudes)	300 m	0.58 to 0.76
Ross et al. (2007)	New York City (US)	28–49	$PM_{2.5}$	Traffic, land use, census	50, 100, 300, 500 and 1000 m	0.607 to 0.642
Su et al. (2008)	Greater Vancouver Regional District, (Canada)	116	NO/NO_2	Road, traffic, meteorology (wind speed, wind direction and cloud cover/insolation)	3000 m	0.53 to 0.60
Mavko et al. (2008)	Portland, (US)	77	NO_2	Traffic-related; Land use-related; Elevation; height from MSL; distance to a river; wind; direction	50, 100, 250, 300, 350, 400, 500, 750 m.	0.66 to 0.81

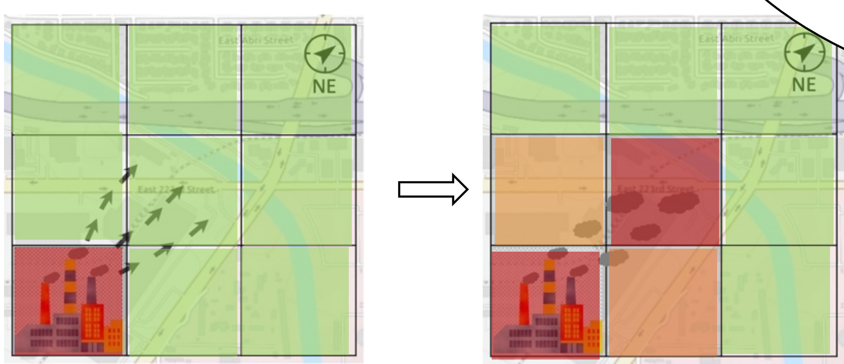
Source: [Liu et al., 2016]

Challenges

- How to learn from thousands of featur
  characteristics with only sparse ar
- How to jointly model spatiotem

Existing approaches either require expert knowledge or does not deal with space and time together and cannot handle sparse and unevenly distributed observations
(e.g., [Briggs et al. 1997; Zheng et al. 2013; Liu et al., 2016; Lin et al. 2017])

Wind blowing towards North East

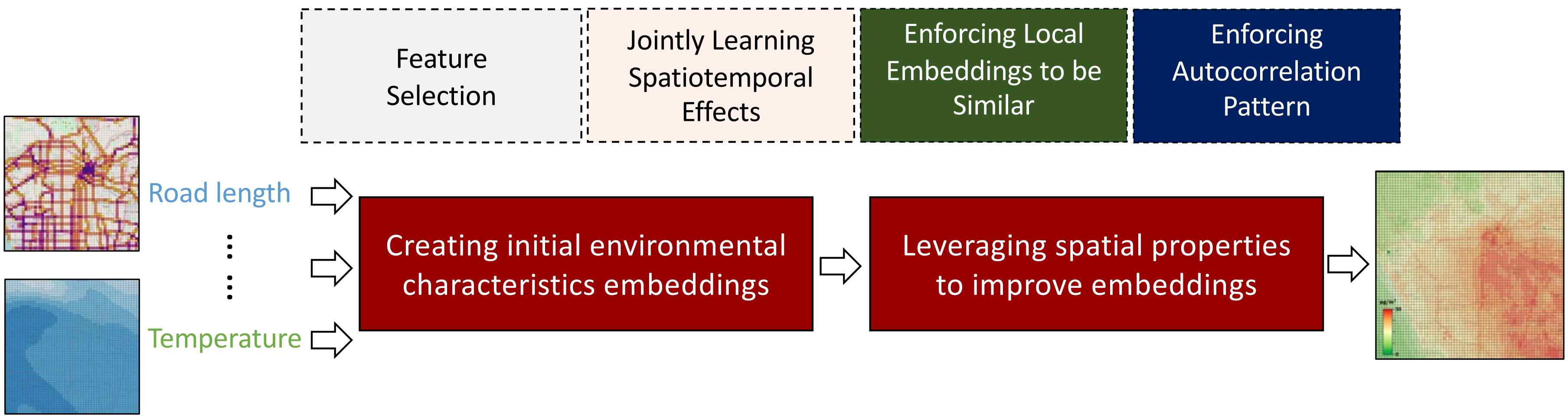


9 am

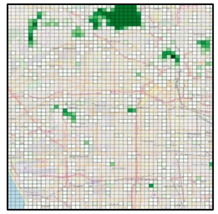
10 am

We build DeepLATTE

- Learn from thousands of features describing the environmental characteristics
- Learn to jointly model spatiotemporal effects
- Learn from sparse and unevenly distributed observations

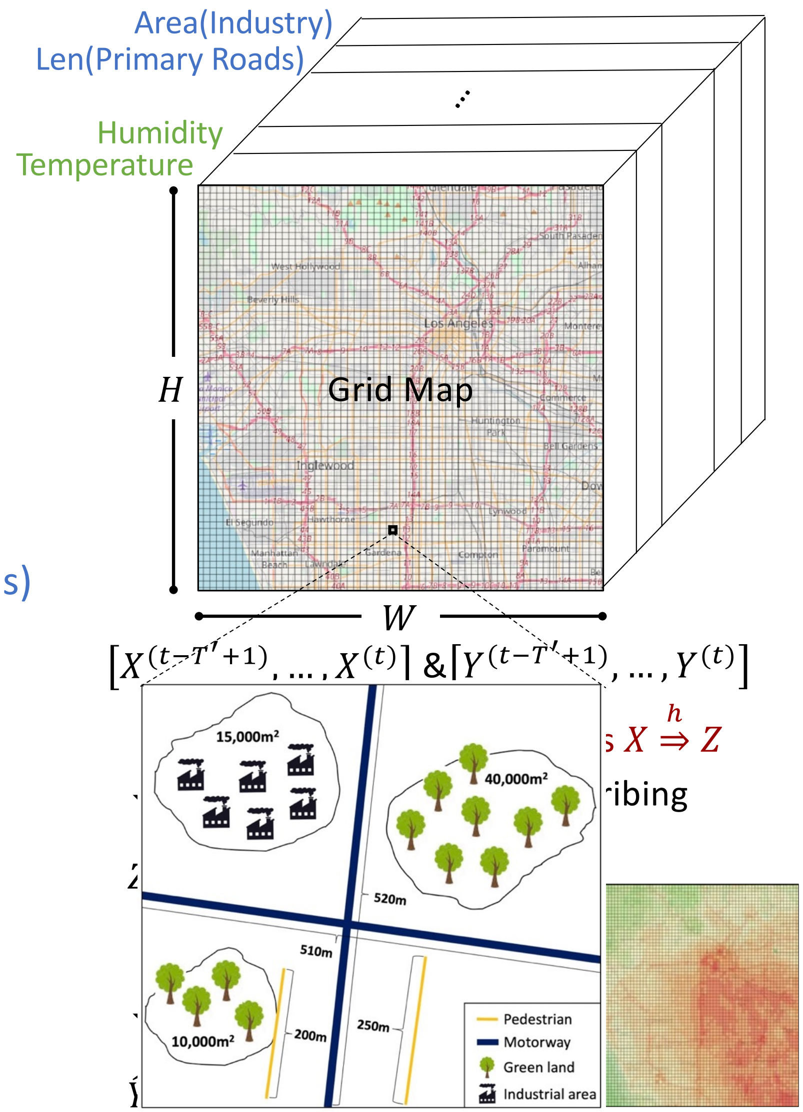


Formally



Green land

Primary road

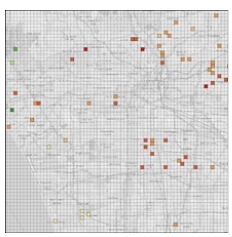


- **Input:** multi-dimension matrix

  $X = (F, H, W)$
  - Each cell in X contains $F = [F_d, F_s]$, describing the environment
  - F_d is dynamic (e.g., weather) and F_s is static (e.g., roads)

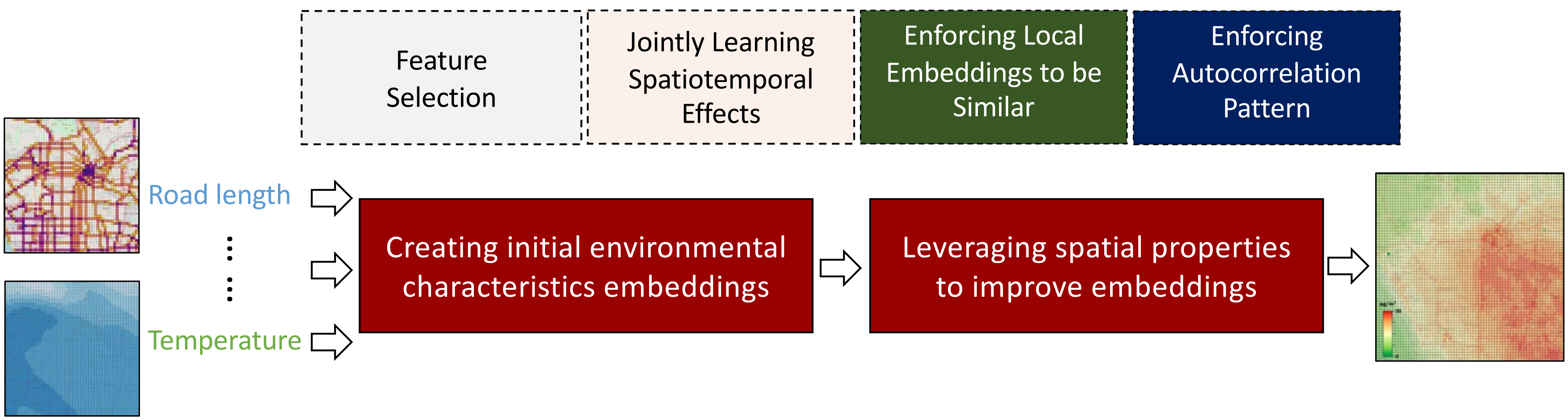
  $Y = (O, H, W)$
  - Each cell in Y contains O, the air quality observation, dimension=1
  - Many empty cells (limited observations)

$Y^{(t)}$



We build DeepLATTE (recap)

- Learn from thousands of features describing the environmental characteristics
- Learn to jointly model spatiotemporal effects
- Learn from sparse and unevenly distributed observations

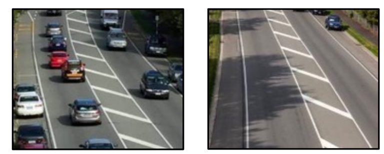


Feature selection and compacting

- Eliminating irrelevant features
- Compacting feature embeddings while capturing important feature interactions

- **Feature Selection**
  - Adding $L1$ regularization $L_{sp} = \sum_{w \in W^{(sp)}} \|w\|$
    i.e., minimizing $(|w_1| + |w_2| + |w_3| + \ \ldots)$

- **Learning Feature Interactions**
  - Minimize $\mathrm{Diff}(X_{sp}, \widehat{X_{sp}})$ to ensure that the condensed feature embeddings effectively captures useful information

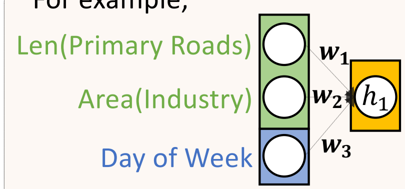


Monday   Sunday

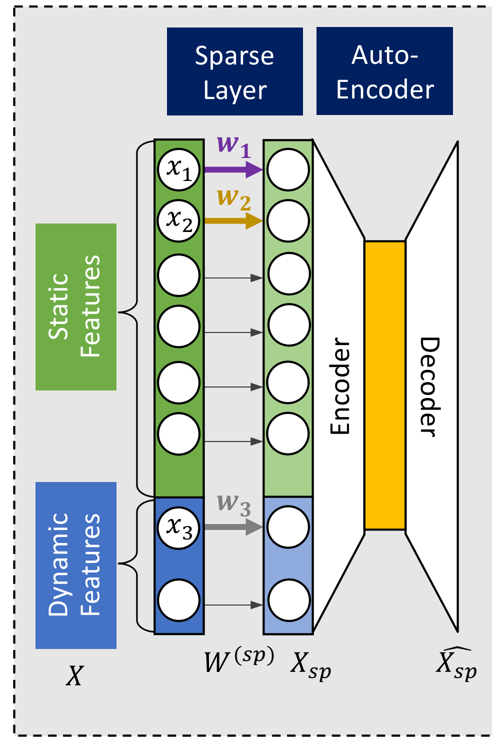
For example,
x_1= Len(Primary Roads)
x_2= Area(Industry)
x_3= Temperature

If w_3 is small, Temperature is not important.

For example,



h_1 - a combination of three features with some weights

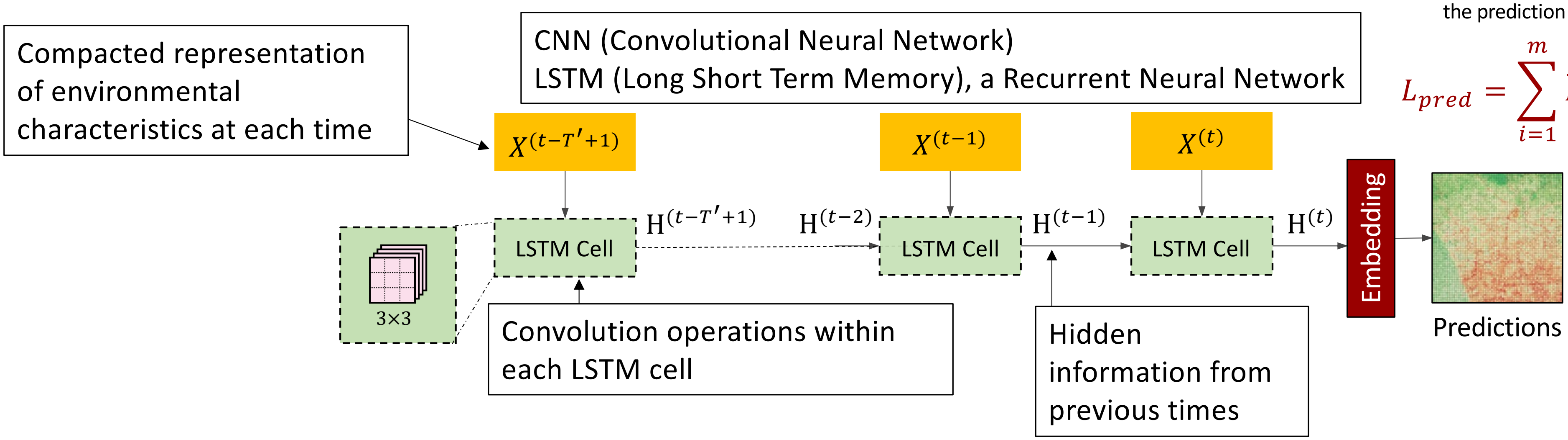


Learning Spatiotemporal Effects

- Capture spatiotemporal effects: current air quality is correlated with the environmental characteristics now, in the past, and from neighboring locations.
- Conv-LSTM layer (Shi et al., 2015)
  - Add the convolution operation directly in the recurrent neural network

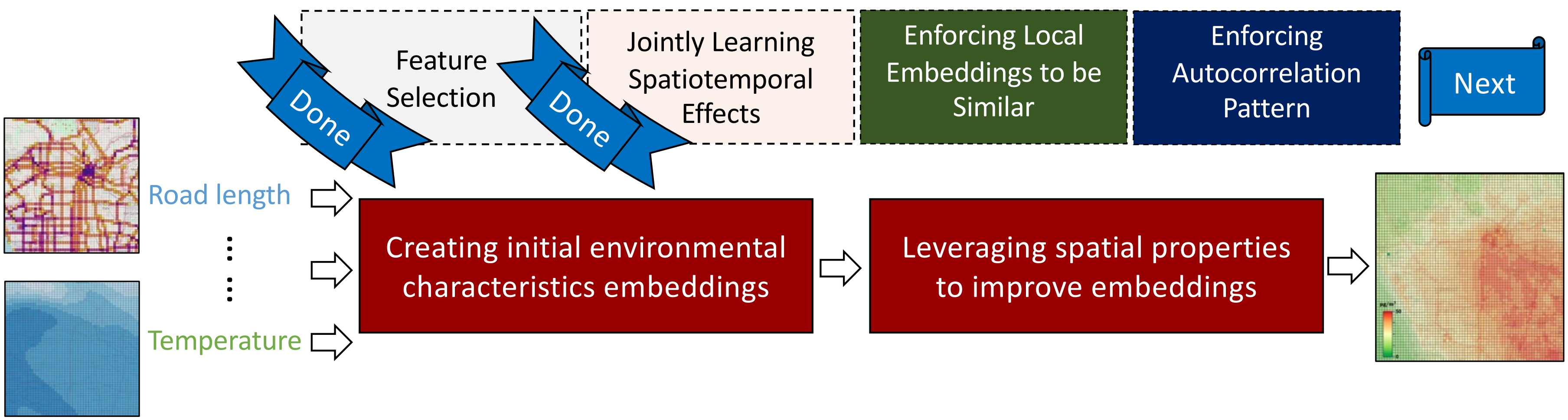
where y_i is the observation and $\hat{y}_i$ is the prediction

$$L_{pred} = \sum_{i=1}^{m} \bar{L}(y_i, \hat{y}_i)$$



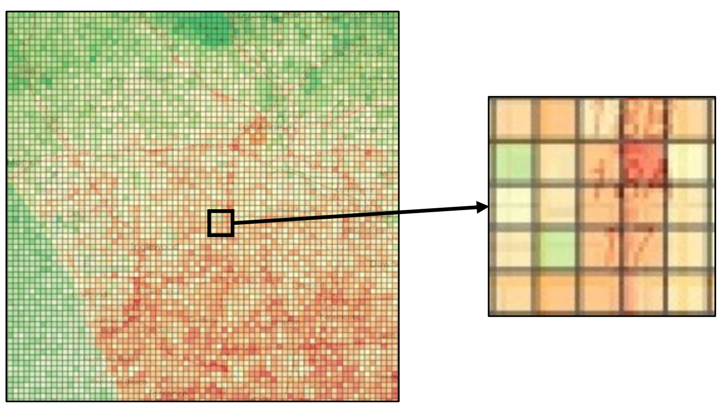
Are we done yet…

- Can compact features and capture important feature interactions
- Can capture spatiotemporal effects
- But we only have sparse and unevenly distributed observations
  - limited variations of environmental characteristics in the training data

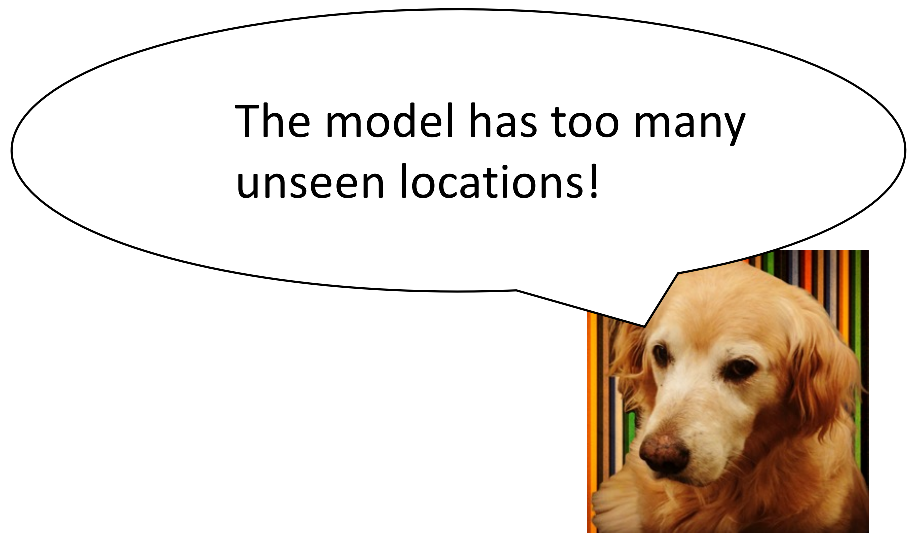


Sparse & unevenly distributed observations

- Sparse & unevenly distributed observations make the model focus on the labeled locations
  - Learned predictions focus on a few locations can fluctuate within a small distance, e.g., 1,000m



P - Predictions



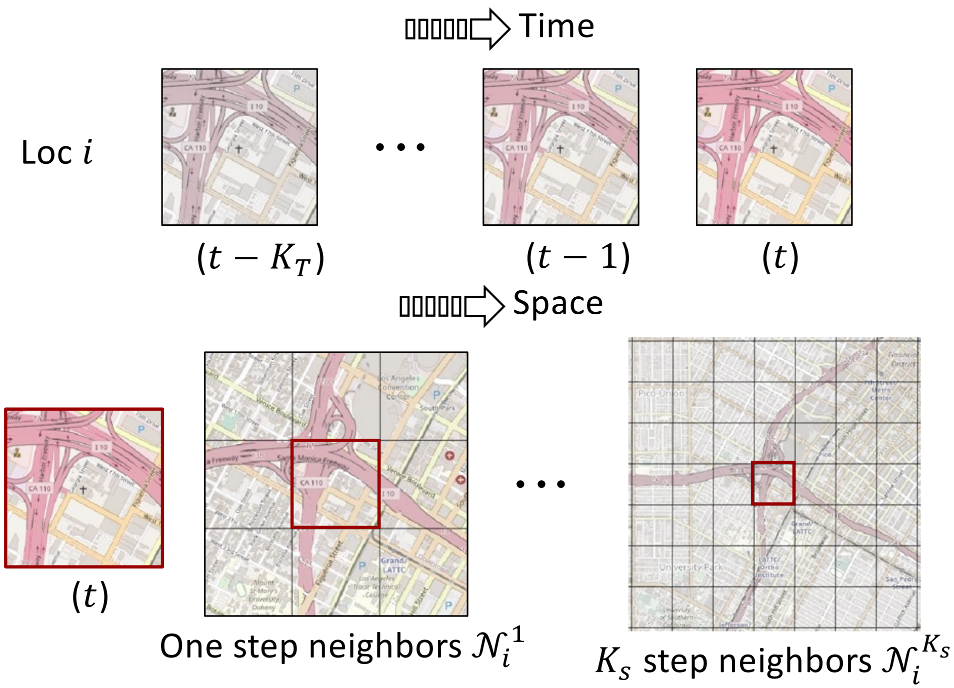
Use spatial data properties to our advantage

- Tobler's First Law of Geography: Everything is related to everything else, but near things are more related than distant things.

$$distance(\underbrace{\boldsymbol{x}_{\boldsymbol{i}}}_{\text{Location A}}, \overbrace{\boldsymbol{x}_{\boldsymbol{j}}}^{\text{Location B}}) < \underbrace{\delta}_{\text{Nearby}} \Rightarrow similarity(\underbrace{\boldsymbol{y}_{\boldsymbol{i}}}_{\text{Air quality at location A}}, \overbrace{\boldsymbol{y}_{\boldsymbol{j}}}^{\text{Air quality at location B}}) > \epsilon$$

Use Tobler's first law of geography

- Enforcing spatially and temporally neighboring embeddings to be similar
  - i.e., environmental characteristics change gradually in space and time



Same location at different times

$$E_i^{(t)} \approx E_i^{(j)} \quad j \in \{t-1, \dots, t-K_T\}$$

Minimizing the Euclidean distance of (E_i, E_j)

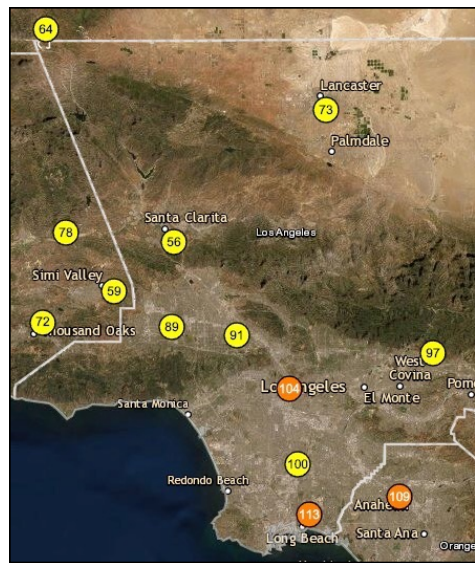
Same time at different locations

$$E_i^{(t)} \approx E_j^{(t)} \quad j \in \left\{\mathcal{N}_i^1, \dots, \mathcal{N}_i^{K_s}\right\}$$

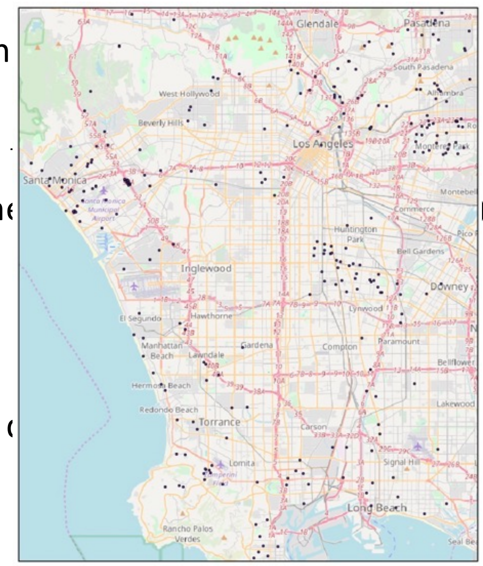
Use Tobler's first law of geography

- Enforcing spatially and temporally neighboring embeddings to be similar
  - i.e., environmental characteristics change gradually in space and time

BUT this only benefits dense and evenly distributed sensor networks. In practice, most sensors do not have nearby neighbors.



South Coast AQMD Monitors
https://gispub.epa.gov/airnow/



PurpleAir Sensors
https://www2.purpleair.com/
(50km×40km geographic area)

Extend Tobler’s first law of geography

- Enforcing neighboring embeddings to have similar air quality
  - Nearby locations have similar air quality implies locations with a similar “environment” have similar air quality

Embedding

Location se[...] Embedding at location B

$$distance(\boldsymbol{x_i}, \boldsymbol{x_j}) < \boxed{\delta} \Longrightarrow similarity(\boldsymbol{z_i}, \boldsymbol{z_j}) > \boxed{\epsilon_1}$$

Location se[...] Embedding at location A

$$similarity(\boldsymbol{z_i}, \boldsymbol{z_j}) > \boxed{\epsilon_1} \Longrightarrow similarity(\boldsymbol{y_i}, \boldsymbol{y_j}) > \boxed{\epsilon_2}$$

Can we learn how to quantify how distance similarity implies environment similarity and then air quality similarity?

Learning autocorrelation pattern in the embedding space

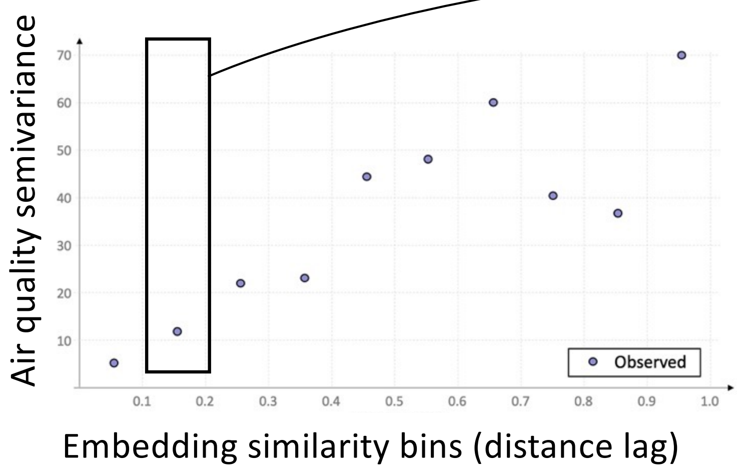
- First, quantifying the spatial autocorrelation pattern that nearby embeddings have a similar air quality
- Use a spatial statistical method – Kriging

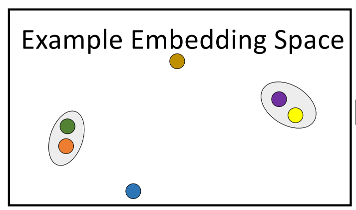
Embeddings

$$similarity(z_i, z_j) > \epsilon_1 \Rightarrow similarity(\boldsymbol{y_i}, \boldsymbol{y_j}) > \epsilon_2$$

In Kriging-like methods, we use geographic distance

$$\left\| E_i - E_j \right\|_2 \in [0.1, 0.2)\ (embedding\ distance)$$





Semivariance Computation

$$\frac{1}{2\times2}\left((Y(\bullet) - Y(\bullet))^2 + (Y(\bullet) - Y(\bullet))^2\right)$$

where $Y(E)$ is the air quality value (label) of embedding E

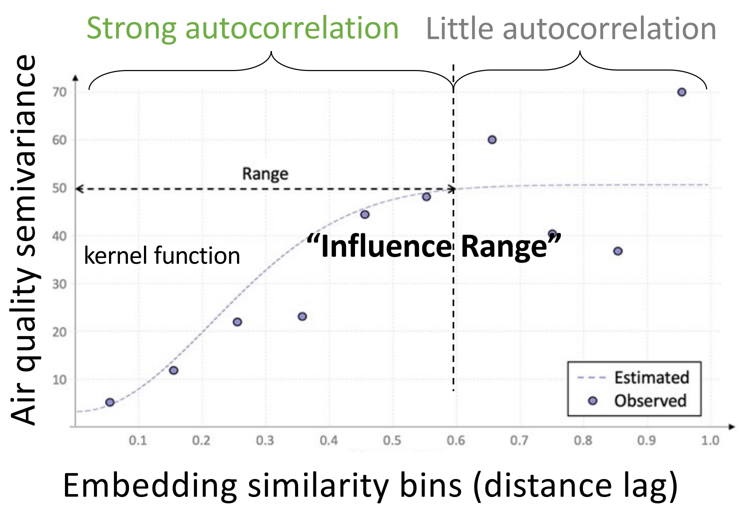
- For every distance lag (bin), computing the semivariance, $\gamma(h)$

$$\gamma(h) = \frac{1}{2N(h)} \sum_i \sum_{j \neq i} \left(Y(E_i) - Y(E_j)\right)^2$$

where $N(h)$ is the number of pairs in a bin

Learning autocorrelation pattern in the embedding space

- First, quantifying the spatial autocorrelation pattern that nearby embeddings have a similar air quality
- Use a spatial statistical method – Kriging
- Quantify the embedding autocorrelation with a kernel function



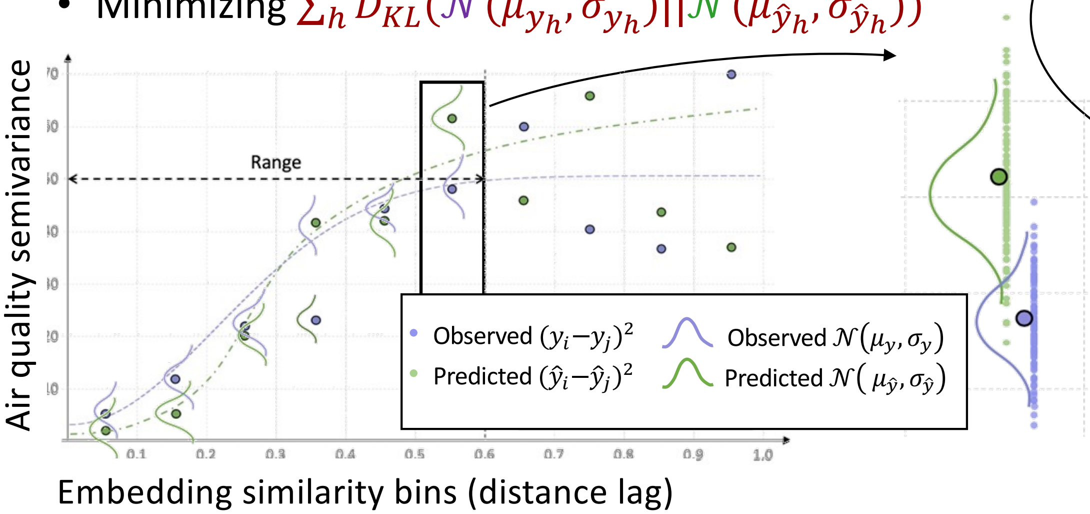
The kernel function tells us: 1) within an influence range, two nearby embeddings would have similar air quality; 2) theoretically how embedding distance implies air quality similarity (the dashed blue line)

Enforcing autocorrelation to refine embeddings

Embeddings

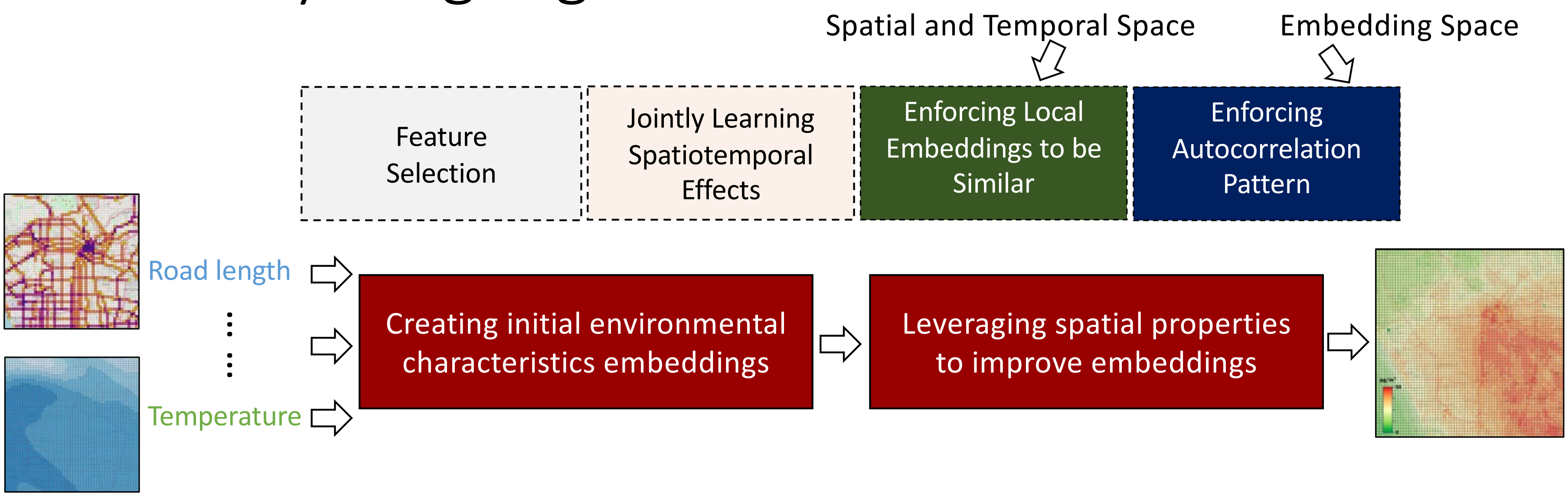
$$similarity(z_i, z_j) < \epsilon_1 \Rightarrow similarity(y_i, y_j) > \epsilon_2$$

- Predictions should have a similar autocorrelation pattern as the observations within the influence range
  - i.e., the purple (observation) and green (prediction) dashed lines (indicati… autocorrelation strength) should be similar
- Represent pairwise embedding distances in each bin as …
- Minimizing $\sum_h D_{KL}(\mathcal{N}(\mu_{y_h}, \sigma_{y_h}) || \mathcal{N}(\mu_{\hat{y}_h}, \sigma_{\hat{y}_h}))$



Encourage the network to learn from unlabeled locations since we can describe each location with an embedding!

Put everything together



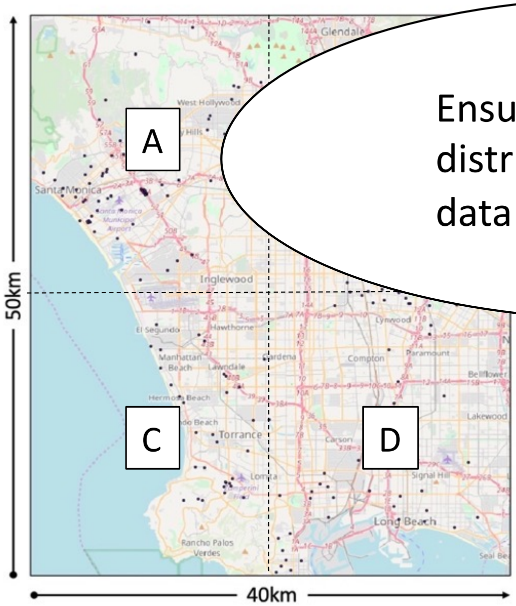
Total Loss = $\alpha \times$ Loss[Sparse Layer] + $\beta \times$ Loss[Auto-Encoder]
+ $\gamma \times$ Loss[Local Autocorrelation] + $\eta \times$ Loss[Global Autocorrelation]

Experiment settings

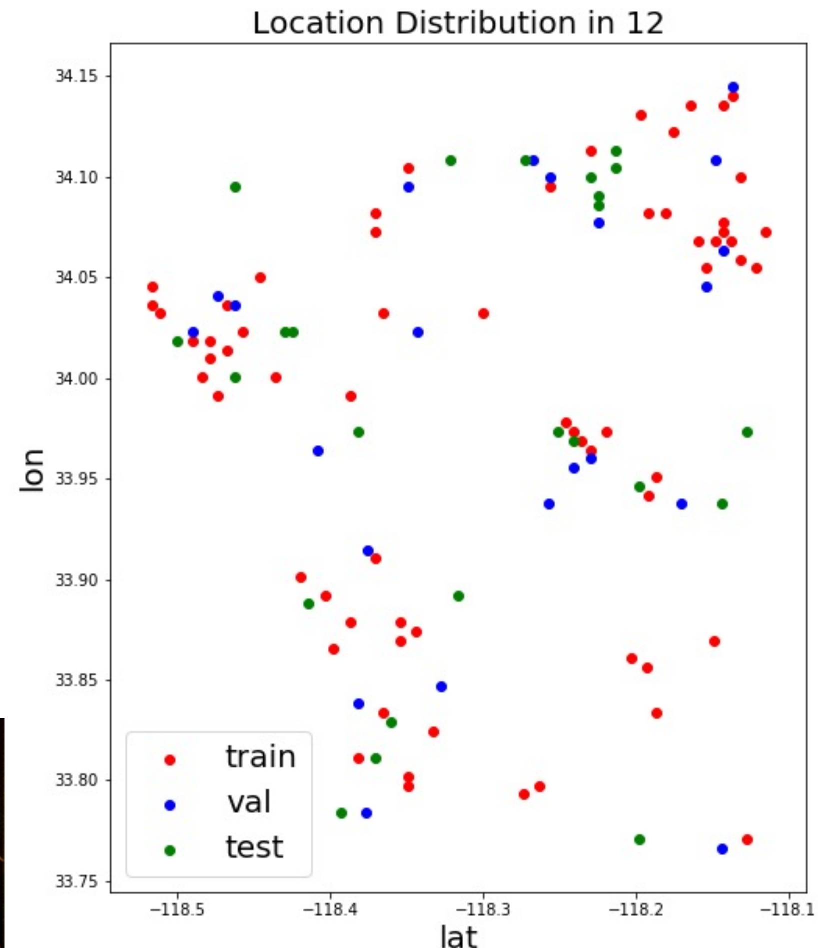
- Create a grid surface with cell size 500m×500m
  - 50km×40km in Los Angeles
- Input Data
  - Air quality data: PurpleAir
    - hourly $PM_{2.5}$ measurements (2018)
  - Meteorological data: DarkSky
    - hourly weather information, e.g., temperature, visibility, pressure, humidity
  - Geographic data: OpenStreetMap
    - 82 features, e.g., length(primary roads), area(green land), count(hotels)
  - Other features: hour of day, day of week, day of year, longitude and latitude

Experiment settings

- Training settings
  - Dividing the area into four parts, each selecting randomly 60% locations for training, 20% for validation. and 20% for testing

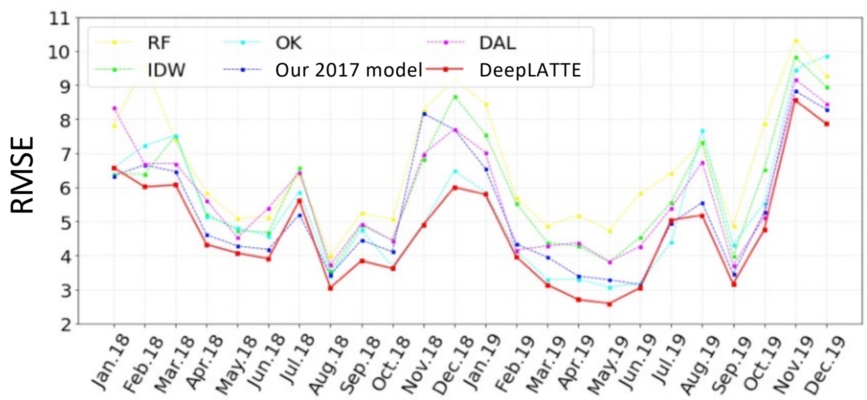


Ensure similar spatial distribution of the training data and validation data



Quantitative analysis

- Built one predictive model per month
- DeepLATTE (red line) outperformed all baseline methods in RMSE and R2



But, for the locations that do not have a ground observation, how do we know we are correct?

- Ablation studies
  - Without the feature selection module underperforms 1.8%-5.1% in RMSE
  - Without learning autocorrelation underperforms 4.1%- 8.3% in RMSE

Evaluating geo-features

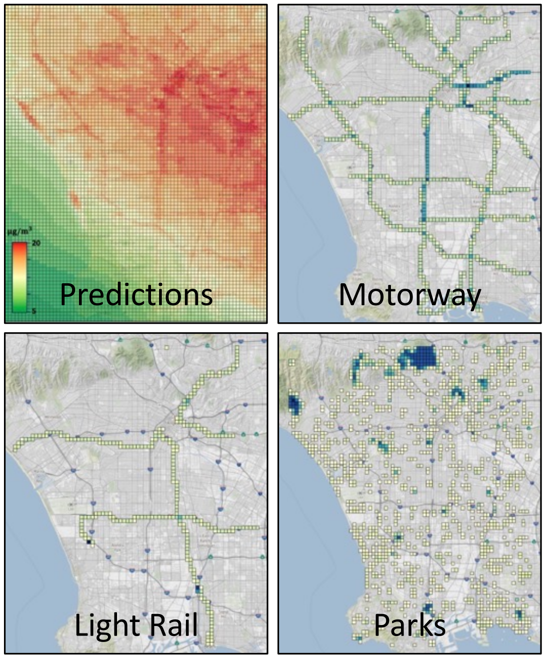
- Evaluating the relationships between predictions and geographic features
  - Showing the result in Oct. 2018, i.e., computing monthly average

Away from major transportation networks improve air quality

(m)	Motorway	Light Rail
0	16.9374	17.2880
Distance <= 500	16.7656	17.1838
Distance <= 1,000	16.6616	17.0828

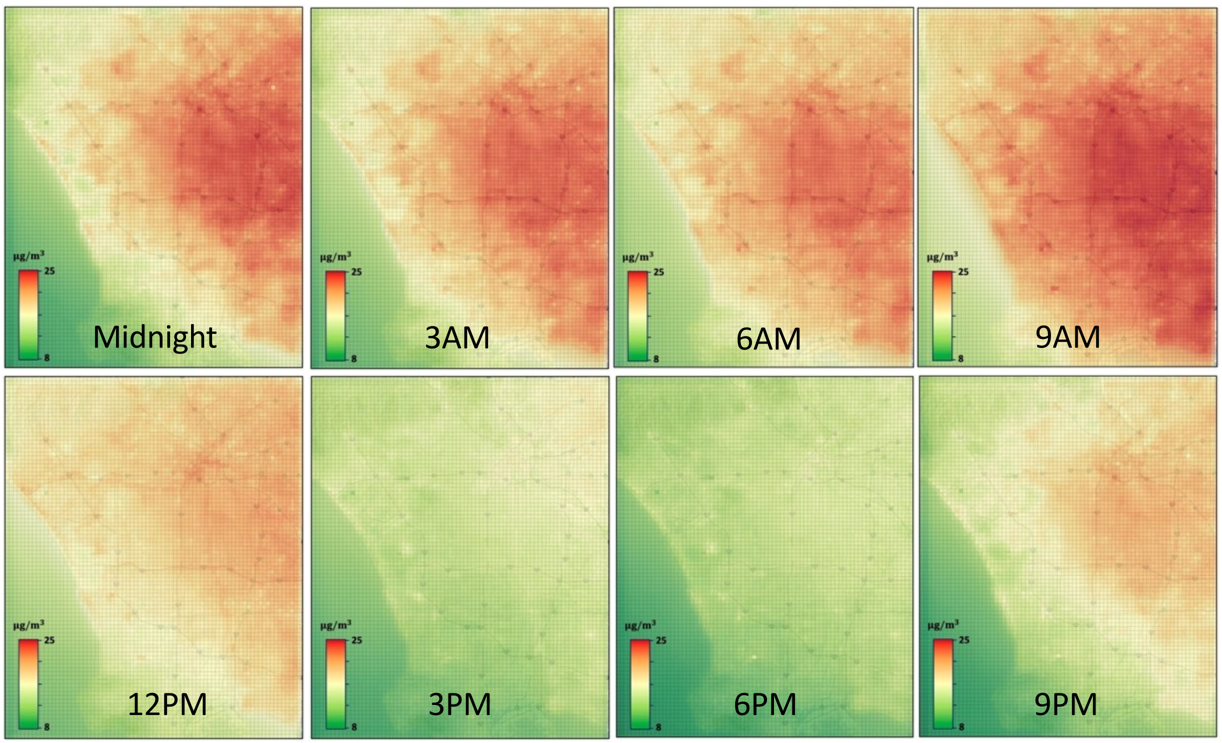
Close to parks improve air quality

(m)	Park
0	16.4163
Distance > 500	16.6054
Distance > 1,000	17.0344

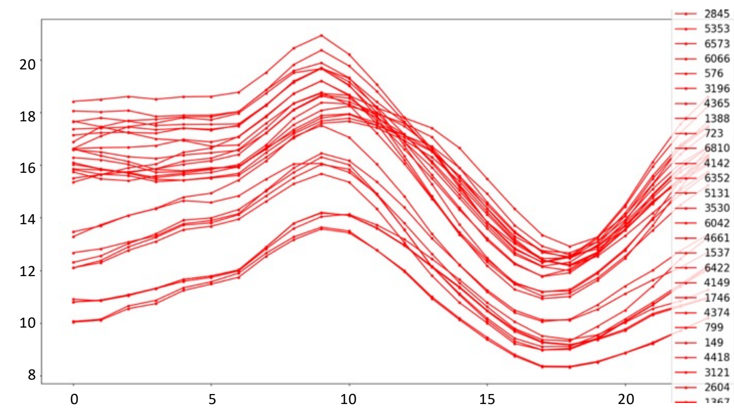


Visualizing hourly average prediction patterns

Same-scale legend

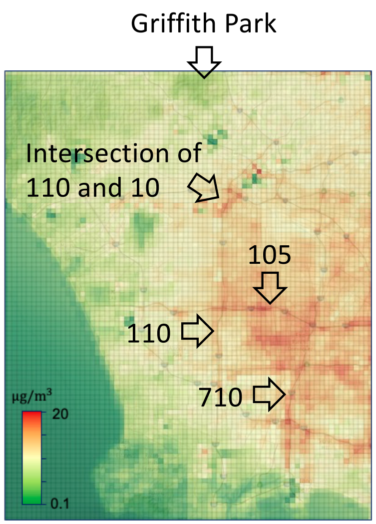


- For each grid, computing the hourly average over a year

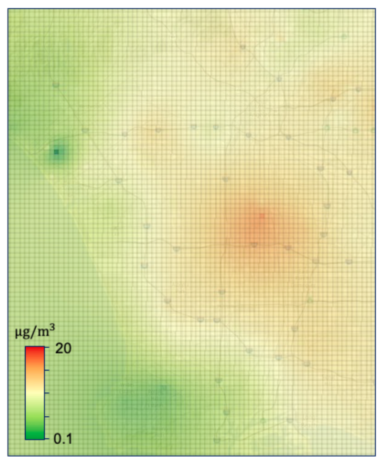


Large temperature differences in a day create a thick layer in the air preventing PM2.5 to escape

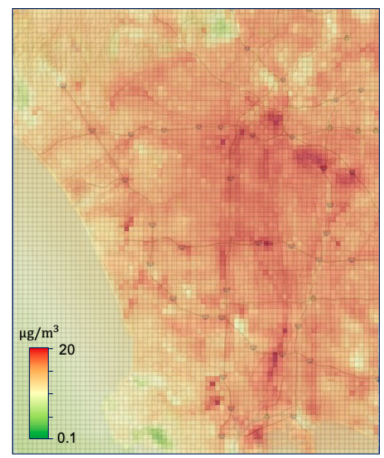
Spatial Visualizations
Monthly Average Predictions



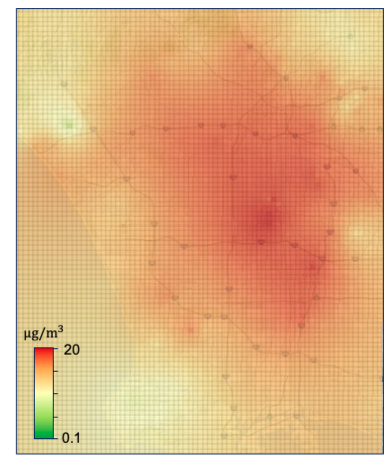
Feb. 2019, Our Model



Feb. 2019, Ordinary Kriging

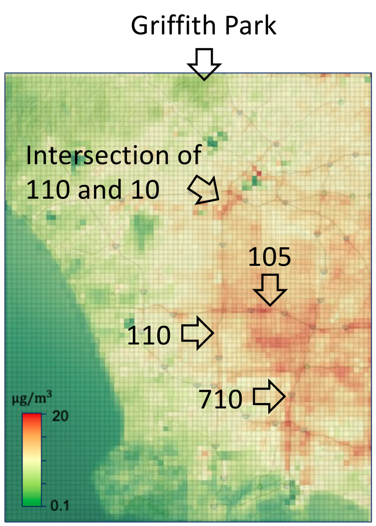


Oct. 2019, Our Model

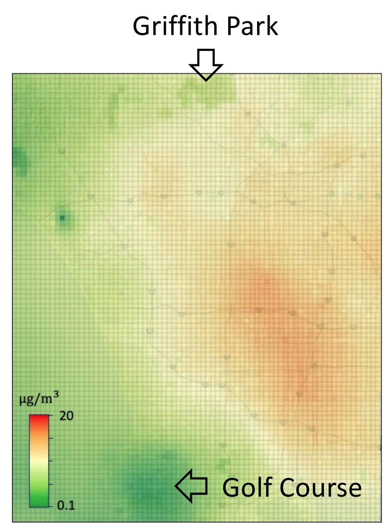


Oct. 2019, Ordinary Kriging

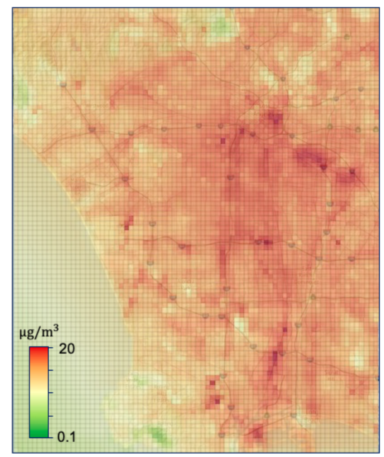
Spatial Visualizations
Monthly Average Predictions



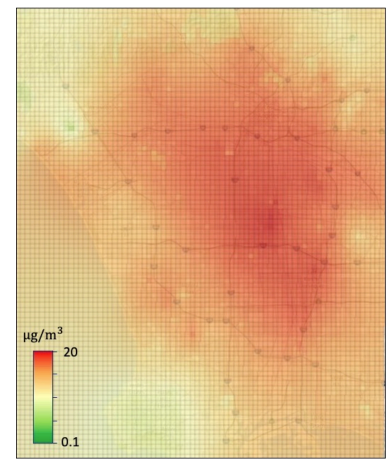
Feb. 2019, Our Model



Feb. 2019, Universal Kriging



Oct. 2019, Our Model



Oct. 2019, Universal Kriging

* Universal Kriging can leverage covariates (e.g., temperature) to estimate a trend in the spatial process

Evaluating selected dynamic features

- Selected dynamic features (10 out of 14), Oct. 2018 monthly average

  - Visibility
  - Day of week
  - Dew point
  - Day of year
  - Temperature
  - Wind speed
  - Wind direction
  - Cloud cover
  - Pressure
  - Hour of day

"Atmospheric visibility has been found to correlate well with PM2.5 concentrations" [Zhang et al. (2006)]

Dynamic Features	Pearson Correlation with Predictions
Visibility	-0.7395
Dew point	0.3070
Temperature	-0.2417
Wind speed	-0.2608
Wind direction	0.1794
Cloud cover	0.25413
Pressure	-0.0613

Evaluating selected static features

- Selected static features (21 out of 84, showing top 12)
  - Latitude
  - Longitude
  - Land use: commercial
  - Roads: motorway
  - Traffic: stops
  - Land use: residential
  - Traffic: fuel
  - Roads: secondary
  - Roads: service
  - Waterways: river
  - Railways: light rail
  - Railways: rail

Moore et al. show that industrial areas, arterial roads, open areas are statistically significantly associated with PM2.5 in Los Angeles (R-value is approximately 0.4 to 0.6 respectively) using LUR approach.

Kam et al. demonstrated that the light-rail lines and subways are strongly associated with ambient PM levels in Los Angeles ($R2=0.61$) by personally monitoring the air quality at the stations.

Summary

- Presented a novel spatial-enabled machine learning approach that predicts fine-scale air quality, support interpretable results

- Future Work
    - Apply our model to other location-dependent time-series data, e.g., remotely sensed thermal imagery over time
    - Improve the interpretability (the selected features)
    - Model the uncertainty in the contextual data
    - Improve handling the spatial non-stationarity problem

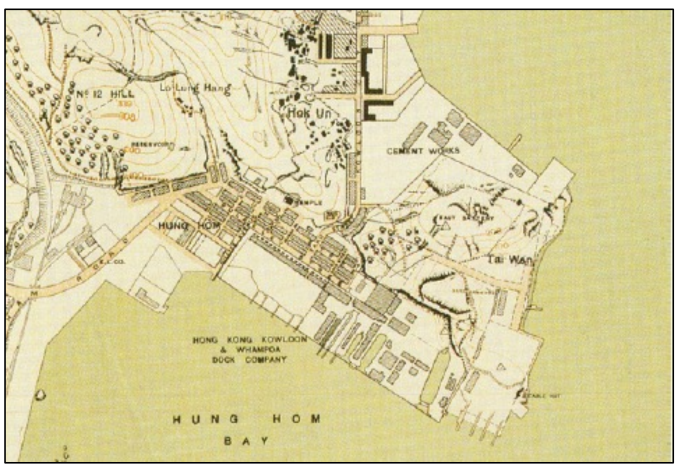
Generating Linked Historical Maps

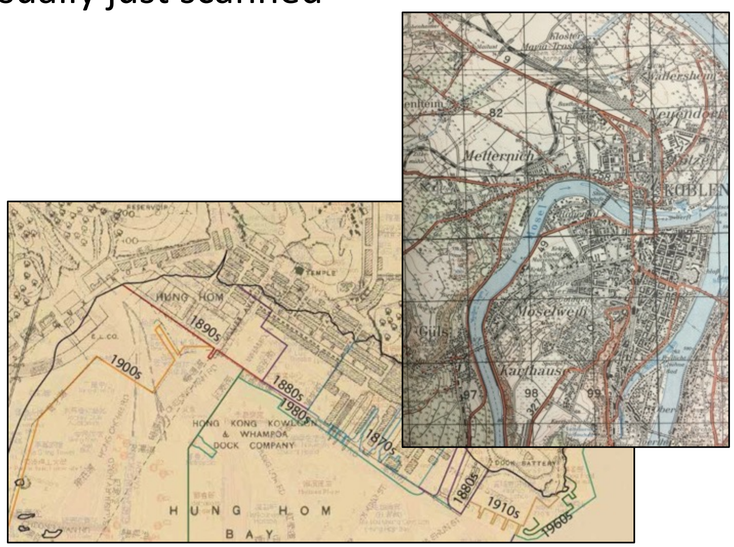
Li et al. KDD (2020)

Why historical maps?

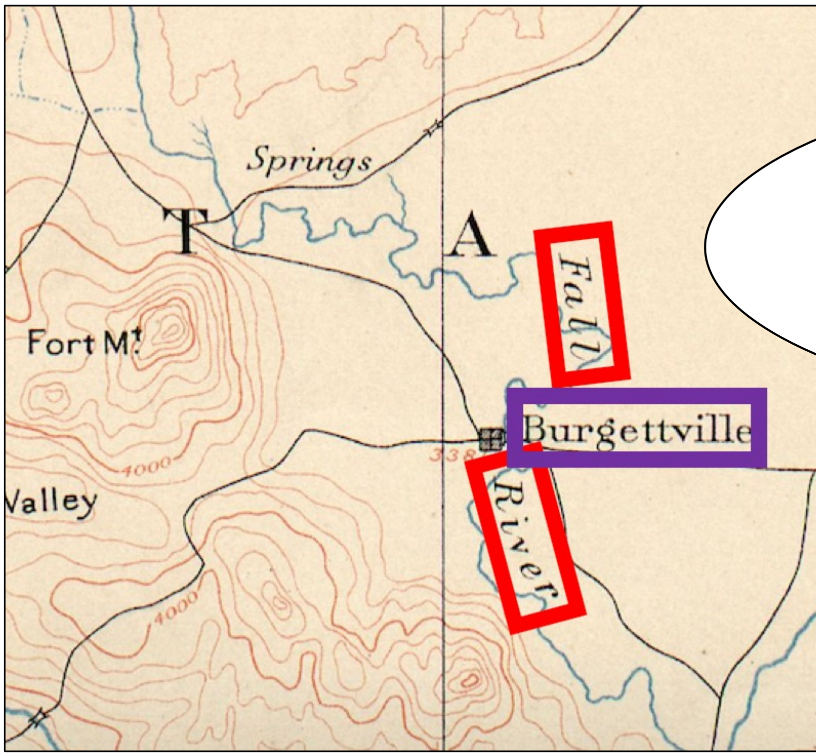
- Existing data sources typically contain only contem datasets, e.g., present place names
- Thousands of historical maps contain detailed geographi information at various times in the past
- Most of the historical maps are usually just scanned images with limited metadata

How can we find relevant maps and make them useful if manual metadata curation is not possible?



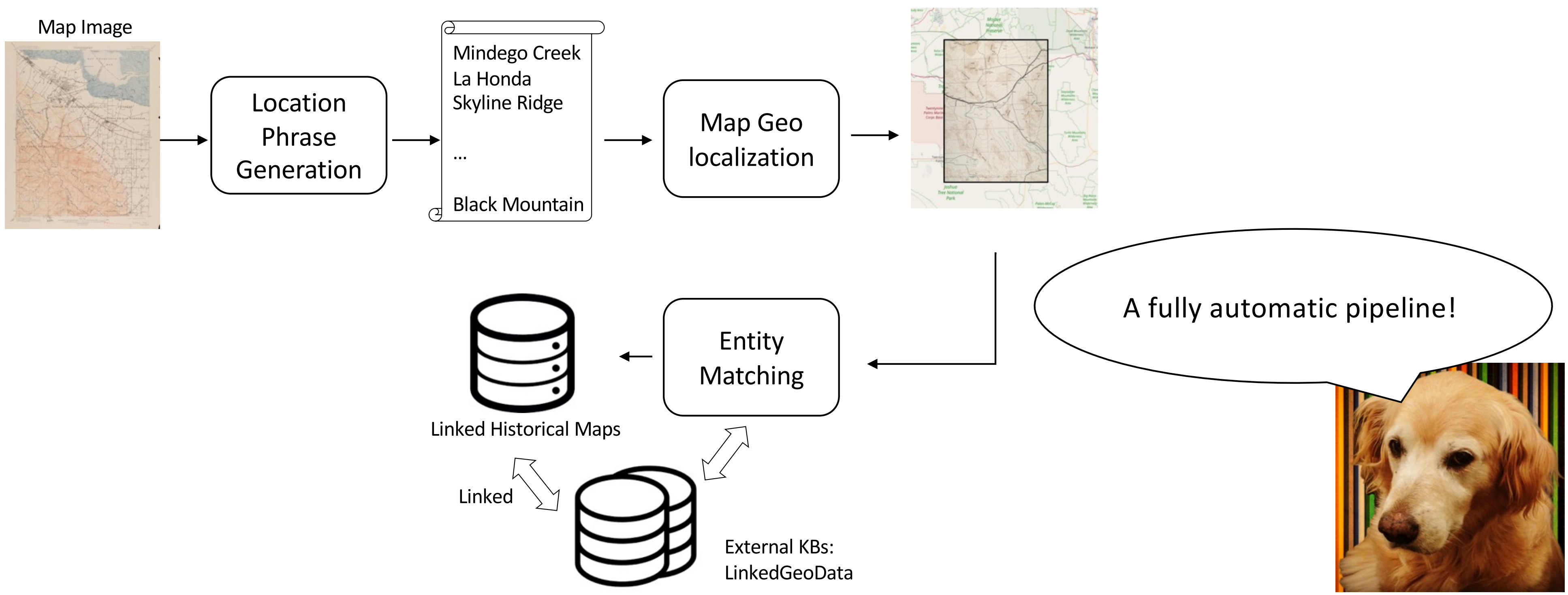


Text on maps are useful but complex



OCR (optical character recognition) tools, such as Google Vision API, would only generate "Fall" and "River" but not "Fall River"

Generating linked historical maps



Generating location phrases

Google Cloud
Vision API

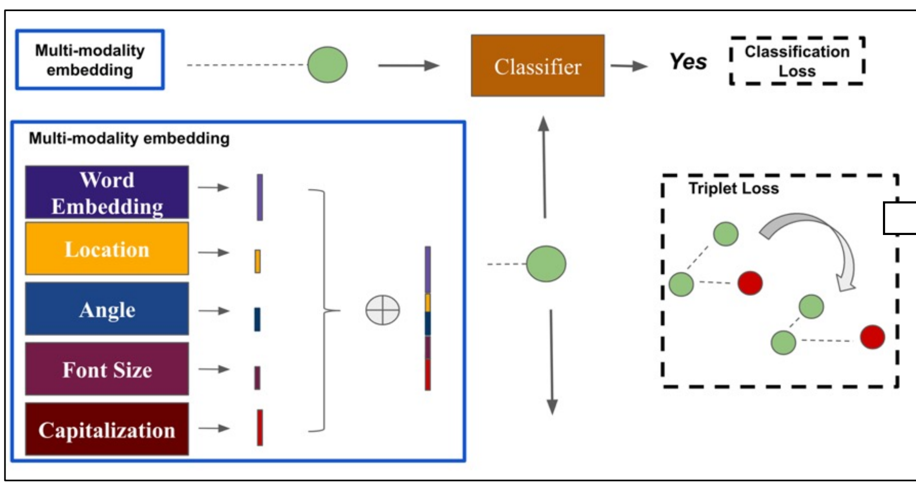
...

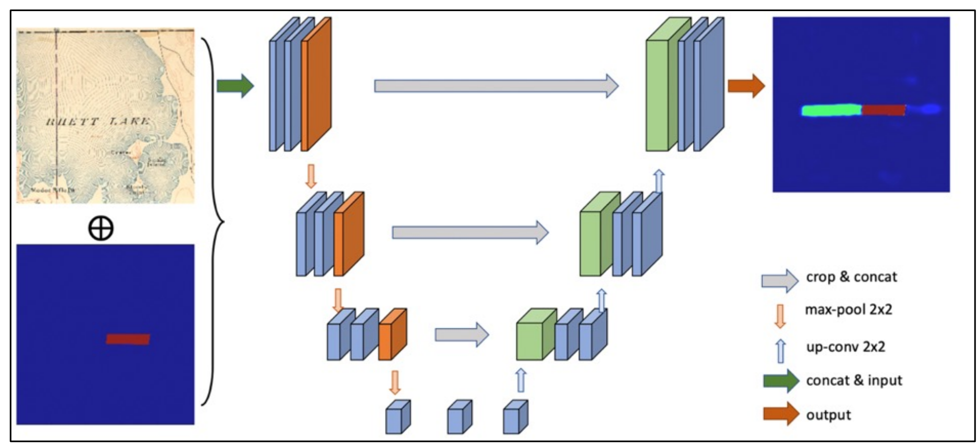
Textual Model: Labels in the same phrase should share similar textual features

High Recall/Low Precision
Help to determine a search neighborhood

Visual Model: Labels in the same phrase could have similar nearby geographic features

Low Recall/High Precision
Help to refine the results





U-Net semantic segmentation

Geolocalization using location phrases

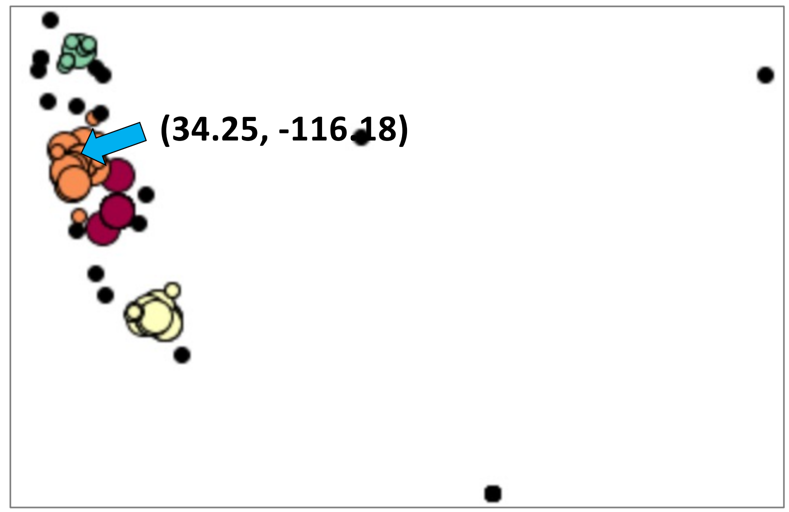
We use Google Geocoding API as the geo-coder: text to lat/long

Location Phrases → **Geo-coder** → List of Candidate Coordinates → **Clustering** (DBScan) →

Mindego Creek
Providence Mountains
Cooks Wall
Clipper Mountain
…
…
Banana Spring

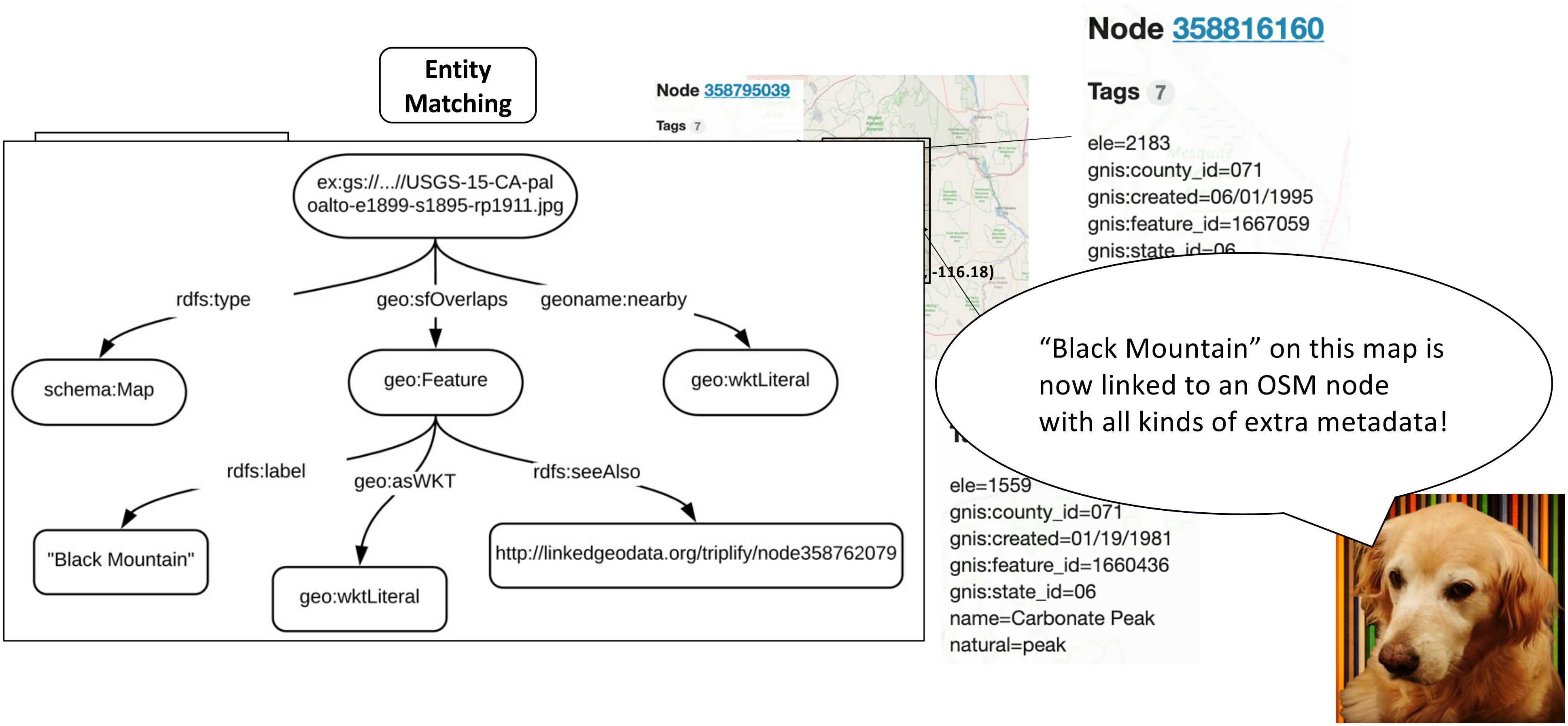
Mindego Creek
(36.77, -119.48)
(36.54, -119.72)
…

Providence Moutains
(38.72, -120.53)
(36.74, -119.42)
…



The cluster center of the largest component is the map geolocalization result.

Entity linking using map geolocation



Experimental setting and metrics

- Datasets
  - United States Geological Survey maps (USGS)
    - 15 maps with 4,375 text regions and locations
  - Ordnance Survey maps (OD)
    - 10 maps with 2,197 text regions
  - New York Public Library maps (NYPL)
    - 500 maps without annotated text regions but with locations

Crowdsourced by NYPL

Geolocalization results for small dataset

Geocoding using all text as a paragraph

Geocoding using individual words and then spatial clustering

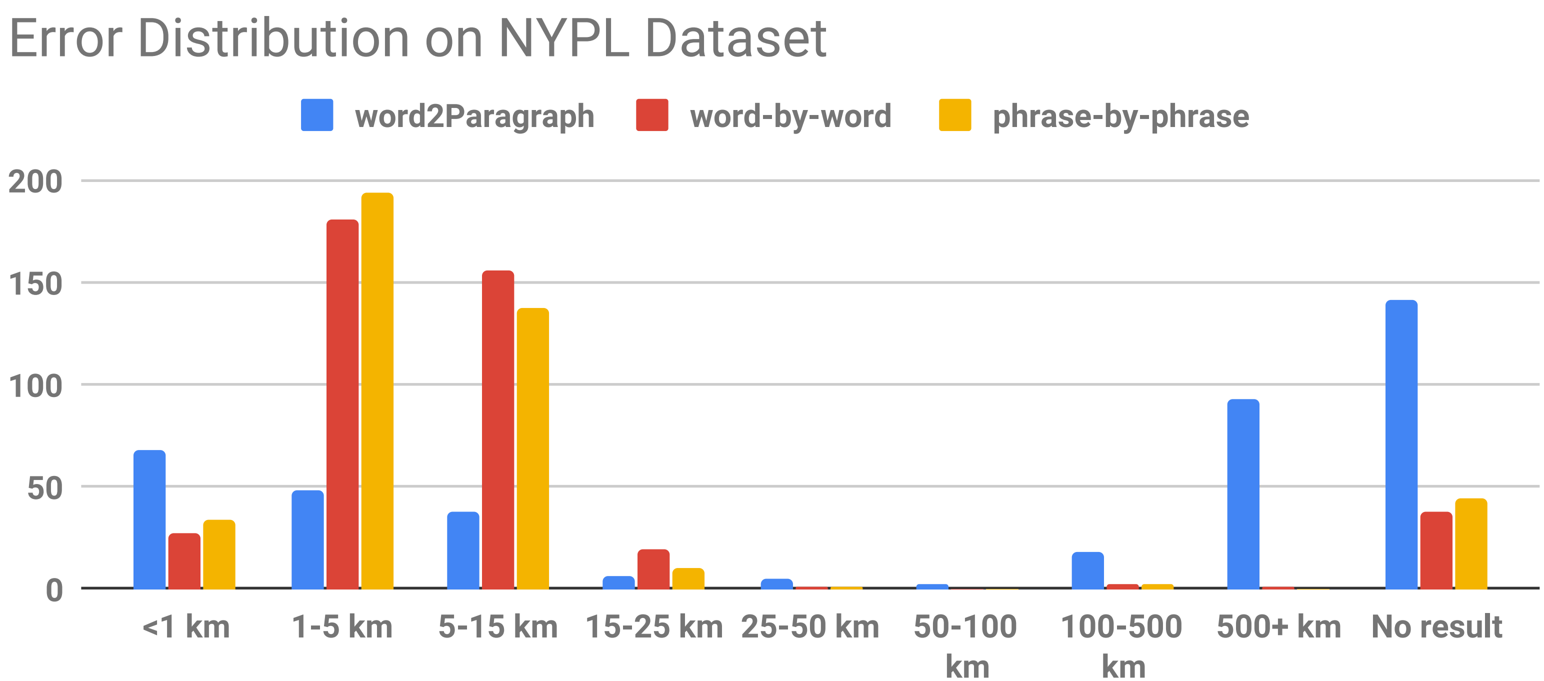
Geocoding using individual phrases and then spatial clustering

	Map Name	**Wrd2Paragraph**		**WrdByWrd(Ours)**		**PhrasByPhras(Ours)**		**Ground Truth**	
		Lat.	Lng.	Lat.	Lng.	Lat.	Lng.	Lat.	Lng.
Pred. (°)	60-CA-amboy-e1942	36.78	-119.42	33.94	-116.83	34.25	-116.24	34.50	-115.50
	60-CA-amboy-e1943-rv1943	36.75	-121.77	33.97	-116.78	34.24	-116.18	34.50	-115.50
	60-CA-modoclavabed-e1886	41.37	-121.02	37.99	-121.80	41.16	-121.54	41.50	-121.50
		km	scale	km	scale	km	scale	km	scale
Error	60-CA-amboy-e1942	435.55	3.02	137.23	0.95	**73.38**	**0.51**	N/A	N/A
	60-CA-amboy-e1943-rv1943	619.31	4.30	131.60	0.91	**68.78**	**0.48**	N/A	N/A
	60-CA-modoclavabed-e1886	42.55	0.31	391.14	2.82	**37.95**	**0.27**	N/A	N/A

USGS dataset contains 15 images, we used 12 for training and 3 for testing.

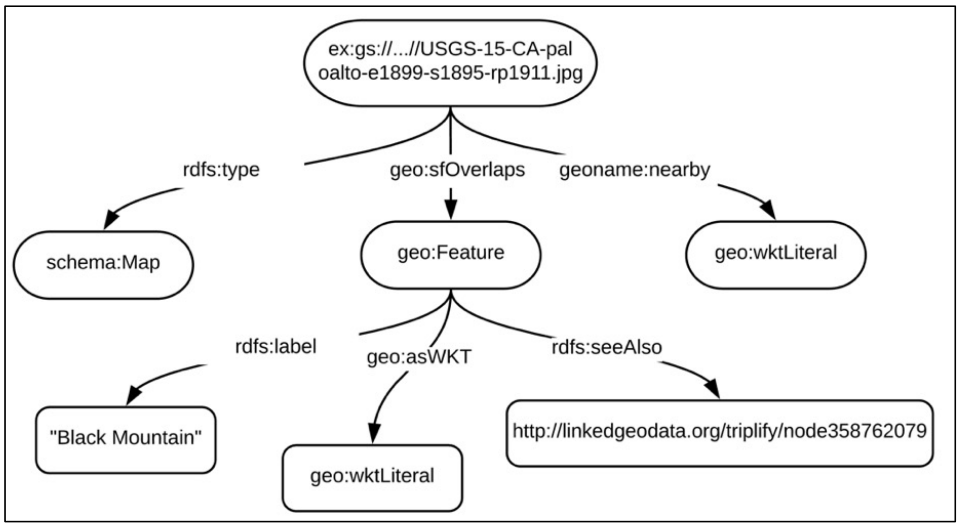
If smaller than 1, the map image covers the predicted geocoordinates

Geolocalization results for large dataset



NYPL dataset contains 500 images.

Summary with a query sample



More than just keyword search!

Sample query: search for historical maps that contain mountains higher than 1,000 meters

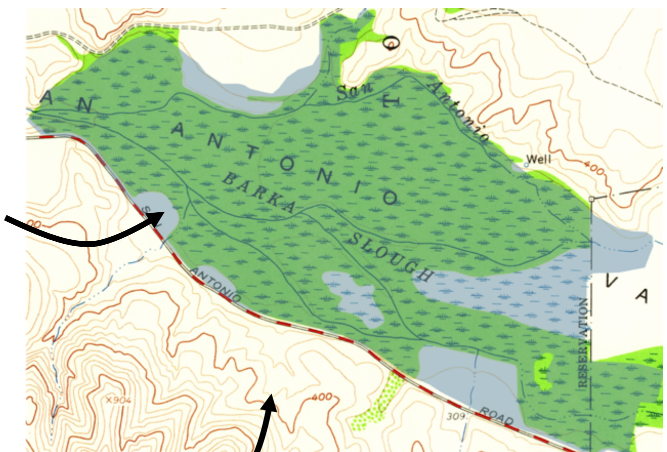
```
PREFIX geo: <http://www.opengis.net/ont/geosparql#>
PREFIX rdfs: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX geoname: <http://linkedgeodata.org/ontology/>

SELECT ?map
WHERE {
  ?map geo:sfOverlaps
       [ rdfs:seeAlso ?lgd_uri ] .
  SERVICE <http://linkedgeodata.org/sparql> {
     ?lgd_uri geoname:elevation ?h .
  }
  FILTER (?h > 1000)
}
GROUP BY ?map
```

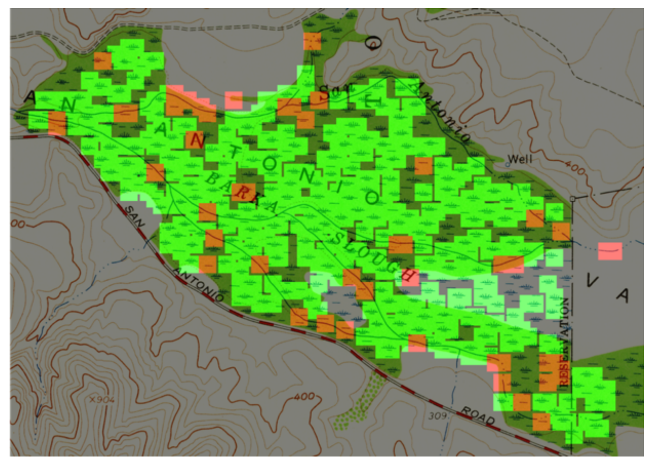
Ongoing: Object detection with weak annotation

- Object detection with limited training data

Blue area – wetland boundaries from the United States Geological Survey



Scanned historical map



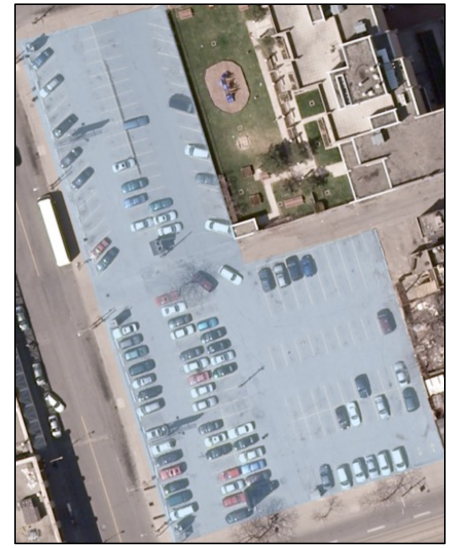
Identified locations of wetland symbols

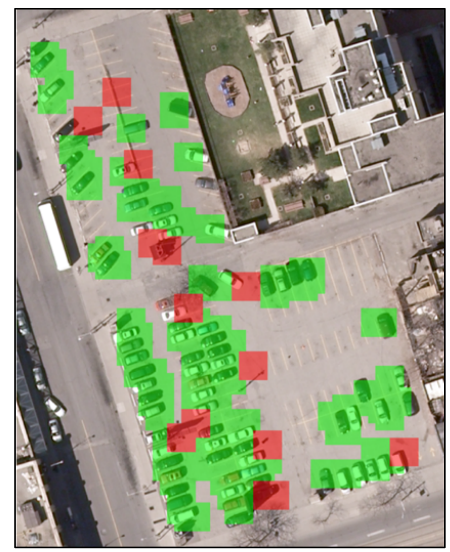
Green: true positives

Red: false Positives

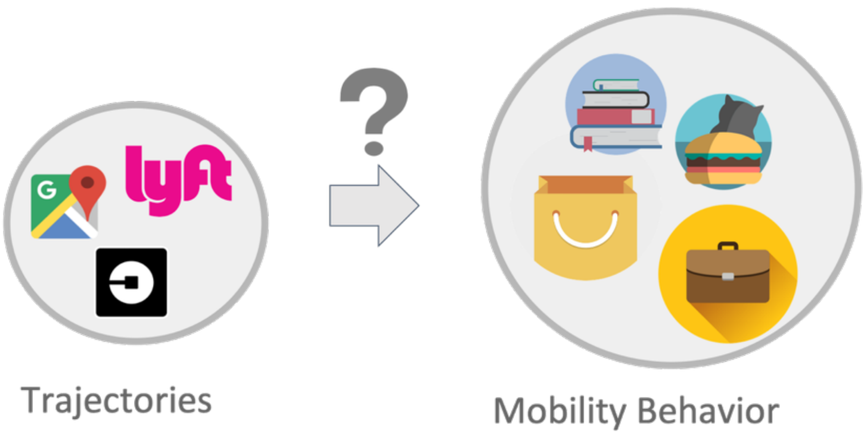
Results visualizations (cars)

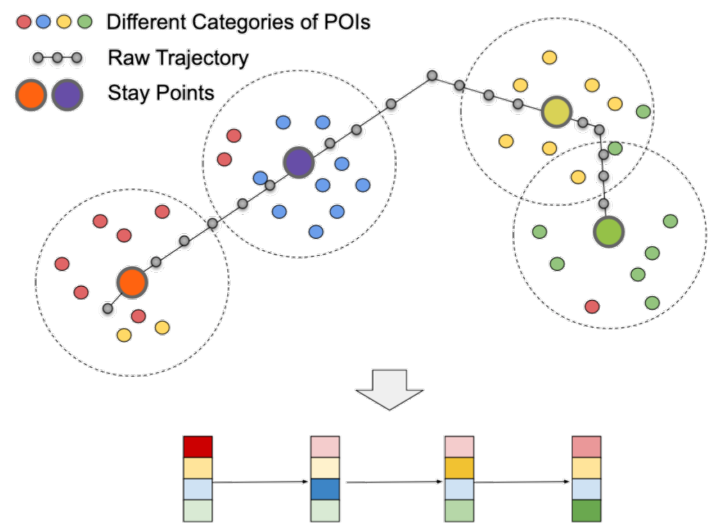
- We know there are many target objects within the boundary
  - Spatial auto-correlation
  - Spatial co-occurrence of cars and parking lots
- But we only have one sample…
- Build a generative model to learn a more “relaxed” representation (a distribution) than descriptive models
- So that we can iteratively improve the representation when we find more objects like the target sample





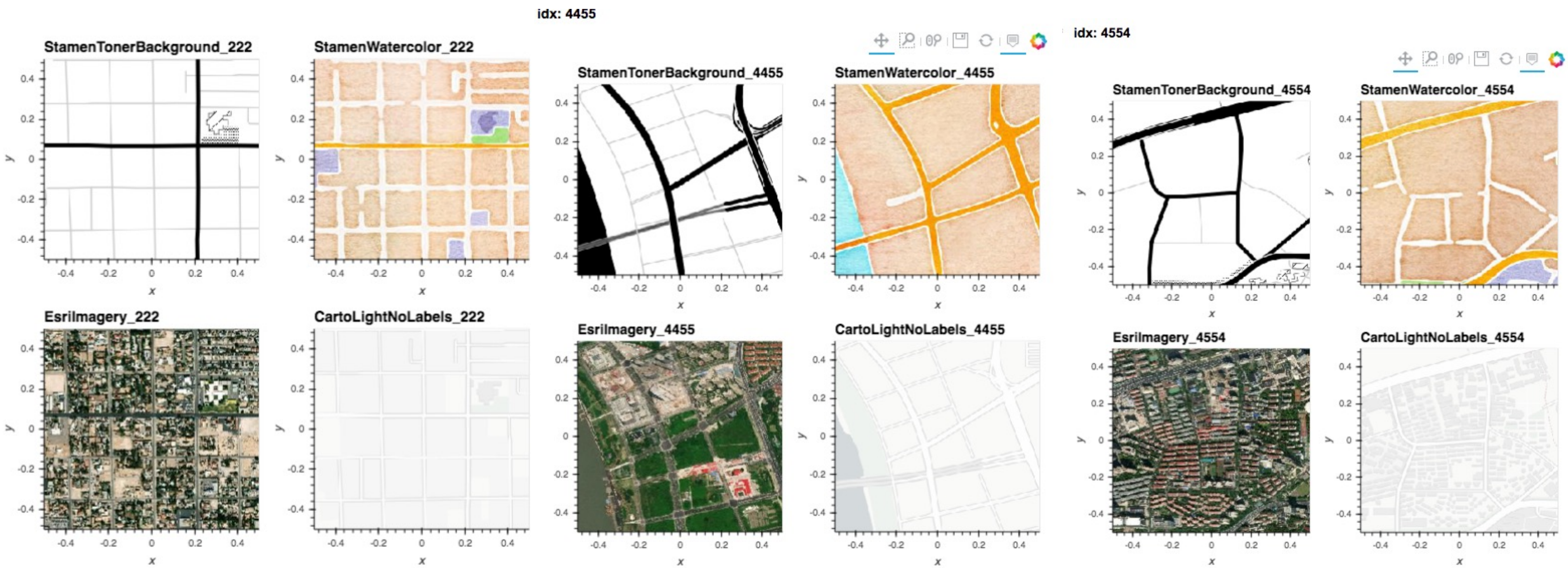
Ongoing: Detecting Trajectory Moving Behaviors Limited and Uncertain Contextual Data





- Manual labeling is expensive
- Various spatial & temporal **scales** in the trajectories
- **Geographical context can be incomplete**

Ongoing: Encoding the World's Geospatial Data



Overall summary

- We can exploit spatial and te
  relationships across data t
  amounts of information-
  data
- We can use the contextual da
  many types of analytics about spatial
  things, e.g.,
  - **Predicting** fine-scale air quality
  - **Detecting** objects from geo-Images using weakly annotated data
  - **Geolocating** and linking scanned map images with external knowledge bases

And many, many other real-world applications that need the investigation of fundamental computer & spatial science research to solve!

Acknowledgement

- Stefan Leyk, Johannes Uhl (CU Boulder)
- Yijun Lin, Weiwei Duan, Zekun Li, Jina Kim, Jose Luis Ambite, Craig Knoblock, Hayley Song, Dan Feldman (USC CS), Meredith Franklin, Sandy Eckel (USC Keck Medical School)
- Sasan Tavakkol (Google AI)