AN INTEGRATED FRAMEWORK FOR MODELING AND PREDICTING SPATIOTEMPORAL PHENOMENA IN URBAN ENVIRONMENTS

Truc Viet Le

September 21, 2017
By 2050, 67% of the world’s population (6 billion people) would live in urban areas\(^1\).

**Big Data and the (Big) City**

> Use spatiotemporal data to make cities safer and smarter.

What is this thesis about?

“...unlocks the power of big data collected in urban spaces to solve major issues cities face today.”
Problems Addressed: Overview
Rest of the Presentation:

- Data and real-world context
- Problem statement and challenges
- GP: What is it? Why use it?
- Proposed solution framework
- Experiments: Design, evaluation and results

- The problem-solving processes of the proposed solution frameworks are abstracted into a common ‘pipeline’

- Motivation
- Problems Addressed: Overview

- Human Mobility Prediction:
  - Spatial bundle prediction
  - Trajectory prediction

- Traffic Speed and Crime Incident Prediction
  - Gaussian process (GP) models

- The Integrated Framework

- Contributions and Summary
Human Mobility Prediction

PROBLEM I-II
Bundled Pass

Choice Pass
Choose 4 out of 16 attractions (from 9 a.m. to 7 p.m.)

Day Pass
Visit all 14 attractions (from 9 a.m. to 7 p.m.)

Spatial Bundle Prediction

Trajectory Prediction

Reinforcement learning

Revealed preference learning

\[
\max \sum_{i=1}^{n} x_i v_i \\
\text{s.t.} \sum_{i=1}^{n} x_i w_i \leq W.
\]

Knapsack problem

Visitor trajectories

Sentosa Data and Context
Proposed Frameworks
Evaluations and Findings

• **Type 1** visitors arrive earlier, have larger budget $\rightarrow$ less time-sensitive and more well-planned $\rightarrow$ MDP better models this group.
• **Type 2** visitors arrive later, have smaller budget $\rightarrow$ more time-sensitive and prone to myopic decision-making $\rightarrow$ Greedy heuristics better model this group.
Fine-grained Traffic Speed Prediction Using Local Gaussian Processes

PROBLEM III
U.S. Traffic Data and Problem

• Speed reading every 5 minutes on some road segments in Pittsburgh and Washington, D.C. during March – August, 2014

• Spatially infer speed values for the whole network (unobserved locations)

• Temporally infer speeds at future time steps
  - Fine-grained inferences (extensive spatial coverage and short-term horizon) → Needs accuracy and efficiency for real-time use cases

• Main idea: localization – efficient clustering of spatiotemporally correlated sensors, each represents a ‘local’ Gaussian process
  - Train and predict in real-time in response to a traffic speed query
What is a Gaussian Process (GP)?

- Consider linear regression: \( y = \theta_0 + \theta_1 x + \epsilon \)
  - **Bayesian** linear regression finds a (posterior) distribution for the parameters \( \Theta \) that gets updated whenever new data are observed.

- GP is a **non-parametric** approach that finds a distribution over all possible functions \( f(x) \) that are consistent with the observed data:
  - Begins with a **prior** distribution
  - Updates it as new data are observed → **Posterior** distribution over all functions:

  \[
  p(f|D) = \frac{p(f)p(D|f)}{p(D)}
  \]

- **GP defines a prior over functions**, which can be converted into a posterior over functions once we have seen some data, which can then be used for **Bayesian regression**.
Why GP?

• Given a set of data points \( x_1, \ldots, x_n \), GP assumes that \( p(f(x_1), \ldots, f(x_n)) \) is jointly Gaussian with some mean \( \mu(x) \) and covariance \( \Sigma(x) \) given by \( \Sigma_{ij} = K(x_i, y_j) \):
  - \( K \) is a positive-definite kernel function
  - If \( x_i \) and \( x_j \) are close to each other in the input space, the corresponding values in the output space should also be similar.

• GP is a Bayesian (regression) method (it gives the mean and ‘error bar’ estimates). It is also a kernel method:
  - Projects inputs into high-dimensional feature space implicitly and efficiently (via the ‘kernel trick’)
  - Models additive, multiplicative, convolutional, etc. interactions of features via ‘kernel arithmetic’.

\[
K(x, x') = 2^{1-\nu} \frac{(\sqrt{2\nu|x-x'|})^\nu}{\Gamma(\nu)} K_{\nu} \left( \frac{\sqrt{2\nu|x-x'|}}{\lambda} \right)
\]

Matern kernel (geospatial statistics)

\[
K(x, x') = \sigma_0^2 \exp \left[ -\frac{1}{2} \left( \frac{x-x'}{\lambda} \right)^2 \right]
\]

Radial basis function (RBF, aka Gaussian) kernel

• Let \( f \) be the posterior of the observed outputs and \( f_* \) the posterior of the outputs yet to be observed. Because it is a GP:

\[
\begin{pmatrix} f \\ f_* \end{pmatrix} \sim \mathcal{N}\left( \begin{pmatrix} \mu \\ \mu_* \end{pmatrix}, \begin{pmatrix} K & K_* \\ K_*^T & K_{**} \end{pmatrix} \right)
\]

• We can then ‘sample’ the posterior: \( f_* \sim \mu + B \times N(0, I) \), where \( BB^T = \Sigma_* \) (called Cholesky decomposition), which takes \( O(N^3) \), where \( N \) is the sample size.
Spatiotemporal GP Kernel for Road Networks

\[ k((r, t), (r', t')) = k_s((u_x, u_y), (u'_x, u'_y)) k_s((v_x, v_y), (v'_x, v'_y)) k_t(t, t') \]


### Feature Description

- **Longitude, latitude**: Longitude and latitude coordinates of the two endpoints (nodes) of a segment.
- **Segment length**: Length (in miles) of a segment.
- **Number of lanes**: The number of lanes a segment has in each direction.
- **Direction**: Direction of a segment: northbound, southbound, eastbound, or westbound.
- **Degree**: Degree of two end nodes of an edge (segment).
- **Betweenness**: Edge betweenness centrality of a segment.
- **One-way**: Is this segment one-way?
- **Road type**: One of the 10 defined types: avenue, boulevard, bridge, lane, place, ramp, road.
GP Framework for Real-time Traffic Speed Prediction

LEARNING

\[ D = N \begin{pmatrix} y_{ij} \end{pmatrix} \approx N \begin{pmatrix} w_i \end{pmatrix} \times \begin{pmatrix} h_j \end{pmatrix} \]

\[ W \quad H \]

\[ \text{Spatial Clustering} \quad \text{Temporal Clustering} \]

\[ \begin{array}{c}
\text{Matrix factorization} \\
\text{Complexity of a local } \\
\text{GP}(i, j) \text{ is } O\left(\frac{NM}{K^2}\right) \\
\text{Complexity of a global } \text{GP} \text{ is } O((NM)^3)
\end{array} \]

\[ \text{Deterministic cluster centroid mapping} \quad \text{Probabilistically} \]

\[ \begin{array}{c}
\text{Speed query for road} \\
\text{segment } r \text{ at time } t
\end{array} \]

\[ \begin{array}{c}
\text{Real-time training} \\
\text{and prediction}
\end{array} \]
Experiments: Design and Evaluation

• ‘Sliding window’ experimental design:
  - Train at time $t$ and test at times $t + i$
  - $W$ is the length of window of observations
  - $\Delta$ is the test duration (e.g., 5 minutes)
  - Evaluate separately for weekdays and weekends
  - Evaluate six (6) different models (local vs. global, with/without ‘side information’)

• Let $\hat{y}$ and $y$ be the estimate and true values, the evaluation metrics are:
  - Mean absolute error (MAE)
  - Mean absolute percentage error (MAPE)
  - Root-mean-square error (RMSE)

\[
\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (\hat{y}_i - y_i)^2},
\]

\[
\text{MAE} = \frac{1}{N} \sum_{i=1}^{N} |\hat{y}_i - y_i|,
\]

\[
\text{MAPE} = \frac{1}{N} \sum_{i=1}^{N} \left| \frac{\hat{y}_i - y_i}{y_i} \right|
\]
Experimental Results: Baselines and Comparison

Incident Prediction for Law Enforcement Resource Optimization
The Law Enforcement Problem

- Densely populated urban areas increasingly puts pressure on law enforcement (LE) agency’s manpower trying to meet ever-rising demands

- Large databases of crime incidents provide fine-grained details:
  - Spatiotemporal (where and when)
  - Context (textual description, urgency classification, type and police response)

- It is possible to make high-precision predictions of incident occurrences using ML

- This serves a larger purpose of a data-driven solution framework for LE resource optimization
  - Time taken to respond to an incident (called “response time”) is a common KPI for many LE agencies

On any given day, no more than $\alpha$ fraction of the incidents ‘fails’ (not responded on time)
Data-driven Framework for Law Enforcement Resource Optimization

\[ \alpha, \gamma: \text{parameters defined the } \% \text{ of incidents unlikely to be responded "on time"} \]

Focus on this part of the framework!

Objective of the optimization

Resource Savings

Jonathan Chase, Jiali Du, Na Fu, Truc Viet Le and Hoong Chuin Lau. Law Enforcement Resource Optimization with Response Time Guarantees. Accepted to the 2017 IEEE Symposium Series on Computational Intelligence (SSCI 2017), Honolulu, Hawaii, USA.
Problem Statement

• Divide the city’s maps and timeline into $|S|$ finite grid squares and $|T|$ intervals
• For each ‘type’ of incident, model the distribution of the count (i.e., number of incidents) within each spatiotemporal combination $|S| |T|
• Each such combination is called a ‘bin’
• Given a query $(x, y, t)$, hash it into a bin index $i$ that has features $f_i$
• Predict the count variable in $i$
• Assuming uniform distribution of the incident occurrence within each bin
Why ‘count’ variable?

• For each sector, given the number of incidents that occur in it at each interval.

• The number of cars allocated in that sector at that interval (prescribed by the optimization) guarantees that the response time KPI can be reached with prob. $\geq 1 - \alpha$.

• Hence, we don’t need to predict the precise location and time of each incident.

• The simulator assumes uniform distribution of incidents within each sector.
Crime Data and Context

- Real-world data provided by a large national law enforcement agency
- Spans over one-year period 2013-14
- Contains more than 200,000 reported incidents (e.g., from emergency calls)

- Each incident has: location, timestamp, type, urgency classification, dispatch and response information (incl. response time)
- Metadata containing neighborhood/sector boundaries and police deployment information
GP Framework for Incident Prediction

Training:
- **Step 1**: Discretize the spatial dimension into grid squares
- **Step 2**: Discretize the temporal dimension into intervals $|S||T|$ bins
  - Compute the feature vector $f_i$ of each bin $i$
- **Step 3**: Learn the count distribution using spatiotemporal GP coupled with the features of each bin

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S,</td>
<td>S</td>
</tr>
<tr>
<td>$T,</td>
<td>T</td>
</tr>
<tr>
<td>$\delta, \tau$</td>
<td>Parameters specifying the granularity of the spatial and temporal dimension</td>
</tr>
<tr>
<td>$x, y, t$</td>
<td>Longitude, latitude and timestamp, respectively</td>
</tr>
<tr>
<td>$x_i, y_i, t_i$</td>
<td>Centroid coordinates of bin $i$ and interval index, respectively</td>
</tr>
<tr>
<td>$f_i$</td>
<td>Feature vector of bin $i$</td>
</tr>
</tbody>
</table>
The spatiotemporal **kernel function** between bins $i$ and $j$:

$$k((x_i, y_i, t_i), (x_j, y_j, t_j)) = k_s((x_i, y_i), (x_j, y_j))k_i(t_i, t_j),$$

**Matern/RBF** kernel: Spatial $\times$ Temporal

$$k((x_i, y_i, t_i, f_i), (x_j, y_j, t_j, f_j)) = k((x_i, y_i, t_i), (x_j, y_j, t_j)) + \sum_f k(f_i, f_j),$$

**Linear** kernel: Additive features

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>location</td>
<td>Longitude and latitude coordinates of the incident</td>
</tr>
<tr>
<td>hours</td>
<td>The integer hours of the incident’s occurrence time (0-23)</td>
</tr>
<tr>
<td>is_weekend</td>
<td>Binary variable whether the incident occurs on the weekend</td>
</tr>
<tr>
<td>neighborhood</td>
<td>Categorical variable specifying the incident’s neighborhood</td>
</tr>
<tr>
<td>sector</td>
<td>Categorical variable specifying the incident’s sector</td>
</tr>
</tbody>
</table>
Experiments: Design and Evaluation

Two types of incidents:
- **Urgent** (33%)
- **Both** (urgent + non-urgent)

“Sliding window” experimental design ($n = 12$):

Evaluation metrics are:
- Mean absolute error (MAE)
- Mean absolute percentage error (MAPE)
- Root-mean-square error (RMSE)

Finally, take the average of each of the metrics for all the test weeks (6).
Experimental Results: Baselines and Comparison

- Linear regression (LM)
- Random forest (RF)
- Support vector machine (SVM)
- Gradient boosting regression (GBR)

Comparing the predicted and actual number of incidents (both for weekday + weekend) for one particular test week.
THE INTEGRATED FRAMEWORK:

Modeling and Predicting Spatiotemporal Phenomena in Urban Environments
The Framework

LEARNING

S

f

1. Spatiotemporal Clustering*

2. Environment Modeling

3. Machine Learning Model $M_j$

PREDICTION

i ∈ T

$f_i$

1. Classification*

2. Environment Mapping

3. $M_k$’s Parameters $\Theta_k$

4. SOLVE

- Revealed Preference (RP)
- Reinforcement Learning (RL)
- Gaussian Process (GP)
- Knapsack
- Decision Models
- GP Regression

- Trajectory clustering
- Matrix factorization
- Incident types

j ∈ {1, ..., $K$}

- Logistic regression
- k-NN

Frame of reference

(HMMs)

> Road networks
> Urban area boundaries

Spatiotemporal Data

> Trajectory clustering
> Matrix factorization
> Incident types

> Logistic regression
> k-NN

Transitions between Attractions in Sentosa

> Road networks
> Urban area boundaries

Frame of reference

(Kernel functions)

Longitude

Latitude

1.25

103.81

103.82

28

(HMMs)
Contributions and Summary

• Extends the “Urban Data Analytics” box (data mining, machine learning, visualization) of the Urban Computing Framework (Zheng et al., ’14)

• The framework combines ML methods to solve a diverse set of real-world problems in urban environments using spatiotemporal data
  - Human mobility prediction
  - Traffic speed prediction
  - Crime incident prediction

• The framework abstracts features of the individual solutions into a common problem-solving process that is highly generalizable
  - Clustering and classification
  - Modeling of the environment
  - Environmental mapping and ML parameters retrieval
Thank You and Questions?