Spatio-Temporal Analysis of High Volume Crime Patterns: A Spatio-LSTM Framework for Crime Hotspots Forecasting

Yong Zhuang[1], Wei Ding[1], Melissa Morabito[2]

1.Department of Computer Science, University of Massachusetts Boston

1.Department of Criminology and Criminal Justice, University of Massachusetts Lowell

Introduction

Crime is a major social problem in the United States, threatening public safety and disrupting the economy. Understanding patterns in criminal activity allows for the prediction of future crime and enables police precincts to more effectively allocate officers to prevent or respond to incidents.

Our research uses a Spatial Long Short-Term Memory model to detect Spatio-temporal patterns in call-for-service data to try to predict trends in criminal activity in urban areas. In particular, our goal is to design a predictive model to identify high-risk "hot spots" in the near future based upon historical neighborhood crime information of the potential hot region. As a case study, the prediction models are evaluated and compared systematically on the call-for-service data provided by the Portland, Oregon Police Bureau (PPB) for a 5-year period from March 2012 through the end of December 2016.

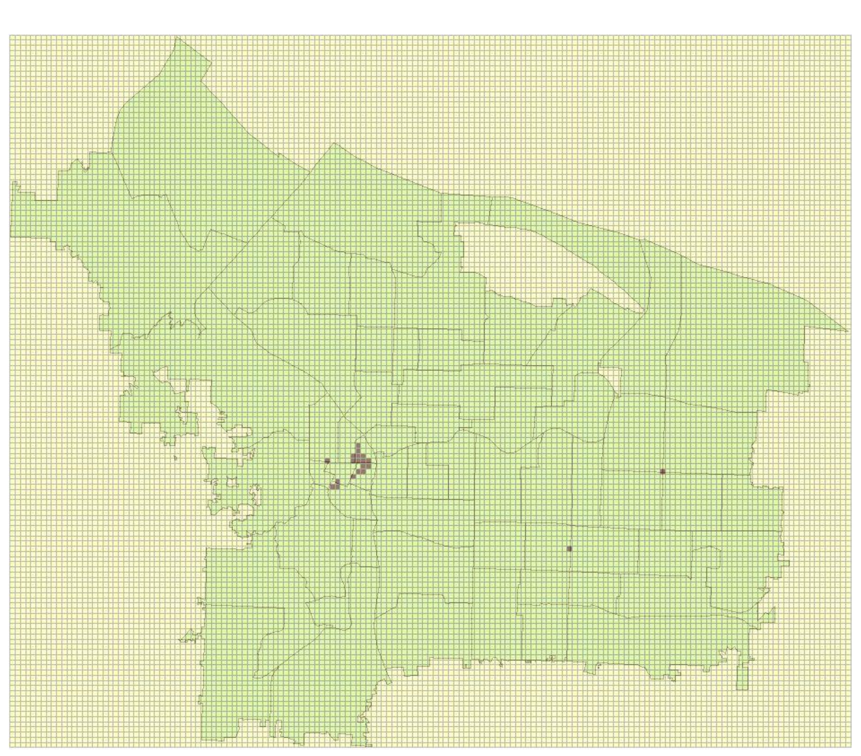
Spatio-LSTM

We formulate the crime hotspot forecasting problem as a binary classification problem as follows:

1. Use a spatial cell's 9-neighbor CFS info to represent its crime activity and that of its neighbors.
2. Use 6 months of historical 9-neighbors CFS info to predict the target cell's class label in the seventh month (1 means hot spot, otherwise, not)
3. The class labels are generated by domain knowledge.

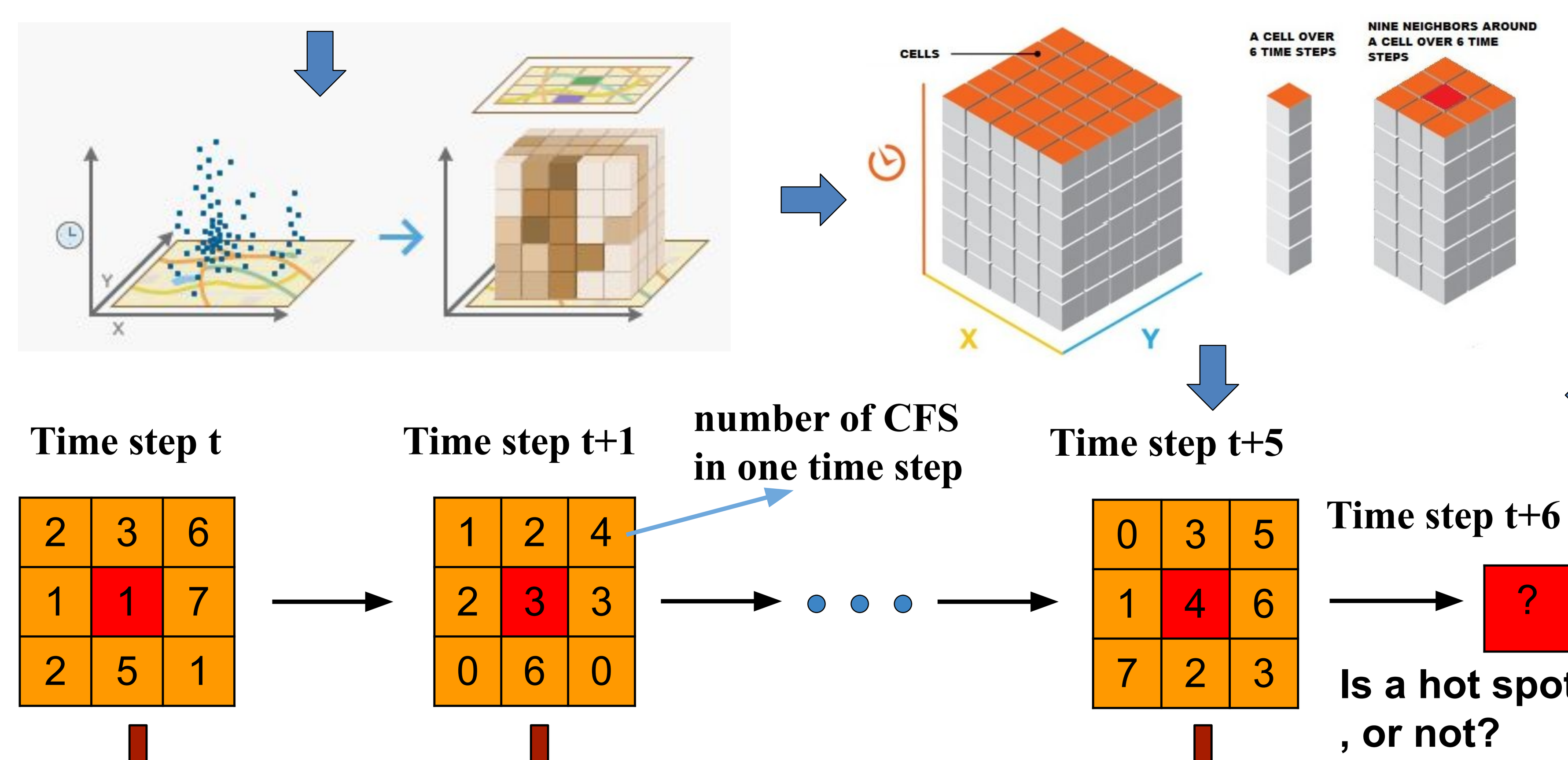
Our model has two networks,

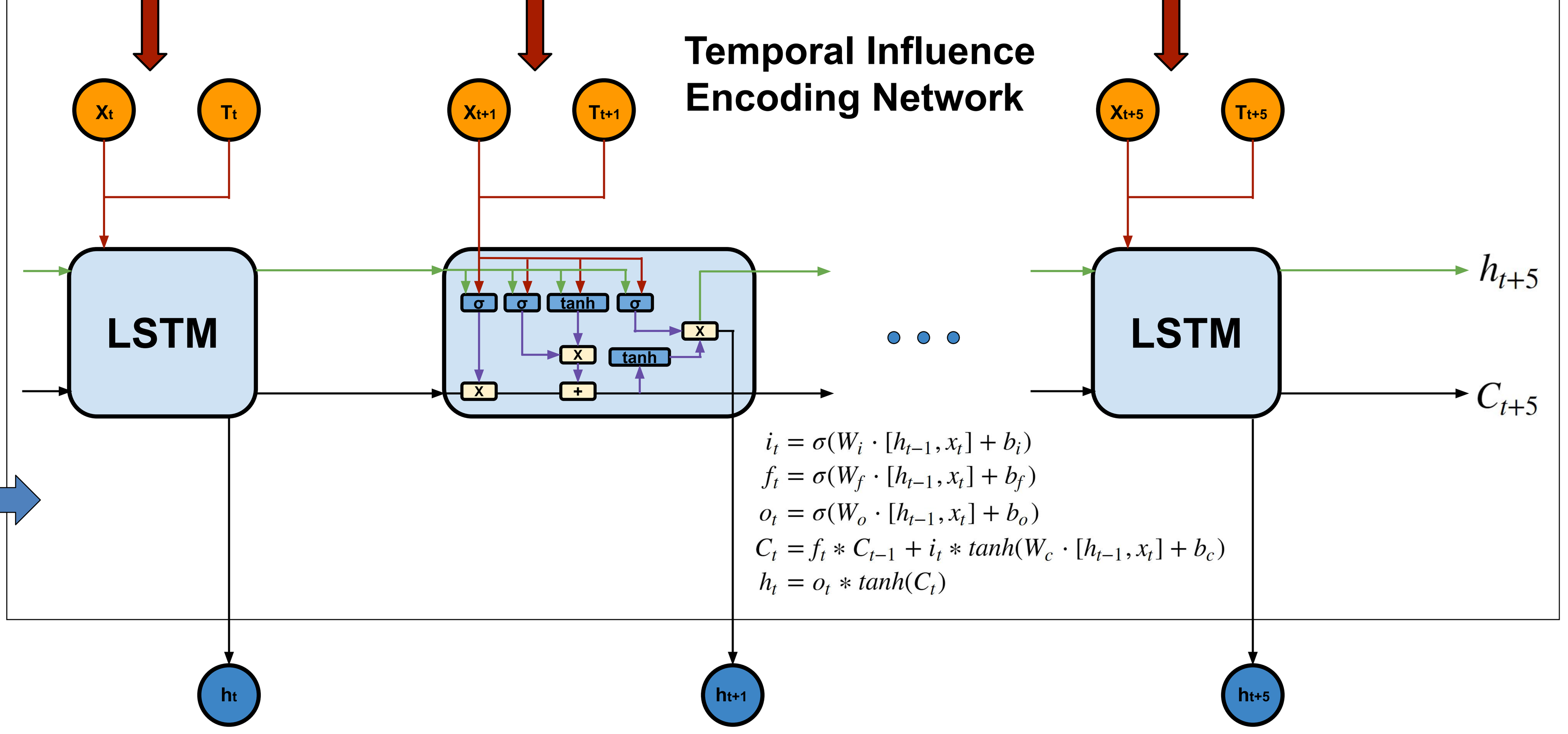
1. The Temporal Influence Encoding Network models the nonlinear dependency over both of the 9-neighbors CFS info and the timings from the past time steps.
2. The Hot Spot Forecasting Network combines the memory (6 time steps) of the influence from timings and the 9-neighbors CFS info with the location info as a input vector, to make the prediction for the next time step.



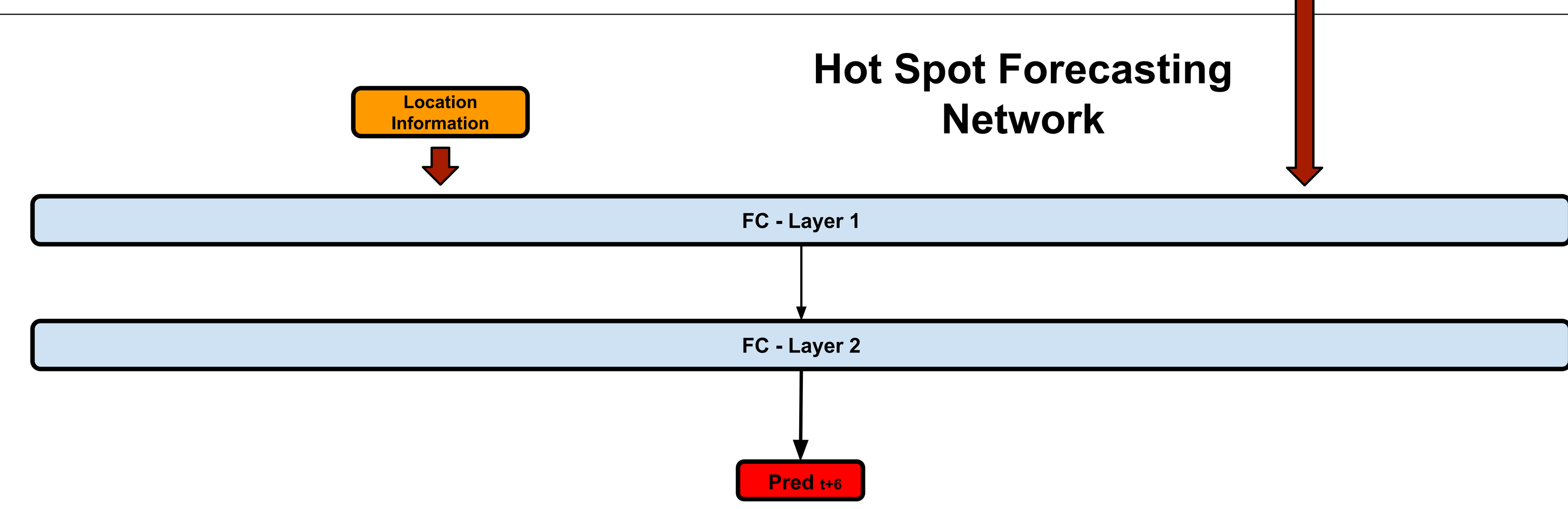
Spatio-Temporal Data Construction

Variables of call-for-services data				
category	occurrence date	x coordinate	y coordinate	census tract





$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$
$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$
$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$
$$C_t = f_t * C_{t-1} + i_t * tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$$
$$h_t = o_t * tanh(C_t)$$



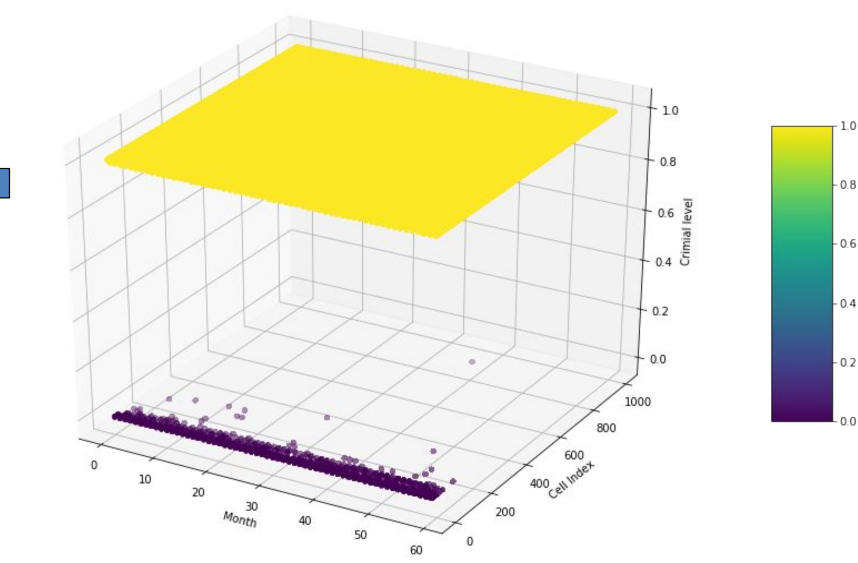
$$Pred_{t+6} = P(Y_{t+6} \mid [h_t, h_{t+1}, h_{t+2}, h_{t+3}, h_{t+4}, h_{t+5}, coordinate_x, coordinate_y]) = f([h_t, h_{t+1}, h_{t+2}, h_{t+3}, h_{t+4}, h_{t+5}, coordinate_x, coordinate_y])$$

Acknowledgements

Funding for this research was provided in part by the Oracle Education Foundation grants to the College of Science and Mathematics.

Potential Hot Cells Selection

One of the major questions that presented itself during our analysis was that of the mobility of the hot spots. How likely was it that the areas with the highest crime numbers would be the same month-to-month and year-to-year? If socioeconomic factors remain largely constant over the course of the data, it may be difficult for these patterns to change. The following plot presents this problem.



We sorted all the cells by the number of CFS occurred in March 2012, and indexed them as #1 ~ #N (N = total number of cells). Then we assigned tiers of criminal activity to the cells, where the top 20 hottest cells were labeled level 0 and the others level 1, for each of the 60 months of data in the dataset. Finally, 73 cells are chosen as potential hot spots..

Experiments

Map size	138 x 163
Cell size	600ft x 600ft

Training set	3006
Validation set	438
Testing set	438

Temporal Influence Encoding Network Parameters	
Time Steps	6
Hidden Neurons	20

Hot Spot Forecasting Network Parameters	
Hidden Layer 1	122
Hidden Layer 2	50
Output	2

Results

Model	Accuracy	F-measure
Decision Tree	0.86	0.768
Random Forests	0.88	0.79
Gaussian Navie Bayes	0.78	0.67
Linear Discriminant Analysis	0.88	0.795
Nearest Neighbor	0.85	0.726
Support Vector Classification	0.877	0.80
Multilayer Perceptron	0.84	0.71
Spatio-LSTM	0.90	0.83