TRAFFIC CONGESTION MODELING WITH DEEP ATTENTION HAWKES PROCESS

A Dissertation
Presented to
The Academic Faculty

By

Ruyi Ding

In Partial Fulfillment
of the Requirements for the Degree
Masters' in the
School of Electrical and Computer Engineering

Georgia Institute of Technology

May 2020

TRAFFIC CONGESTION MODELING WITH DEEP ATTENTION HAWKES PROCESS

Approved by:

Dr. Yao Xie
School of Industry and System Engineering
Georgia Institute of Technology

Dr. Mark Davenport
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Matthieu Bloch
School of Electrical and Computer Engineering
Georgia Institute of Technology

Date Approved: April 24, 2020

ACKNOWLEDGEMENTS

I want to thank my parents Guoqing Ding and Zheyan Lu. Thank you for your strong support for my studies abroad!

I want to thank Dr. Yao Xie. Thank you for directing me in my academic and research!

I want to thank the members of Dr. Xie's group Liyan Xie, Shixiang Zhu, Rui Zhang, Yuchi Henry Shaowu, Minghe Zhang. Thank you for your help in research and daily life!

I want to thank Dr. Mark Davenport, Dr. Matthieu Bloch, and all other instructors in my Masters's courses. Thank you for your instruction in my lectures!

I want to thank my roommates Huizong Yang, Sen Wang, and Zirui Xu. Thank you for your care in my daily life!

I want to thank the members of Western Christian Fellowship, Jason Chen, Jasmin Shi, Dr. Sheng Dai, Tingting Wang, and all other students in the group. Thank you for your spiritual guidance and emotional support.

I want to thank the Georgia Institute of Technology and School of Electrical and Computer Engineering. Thank you for your offer me such an excellent learning environment!

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

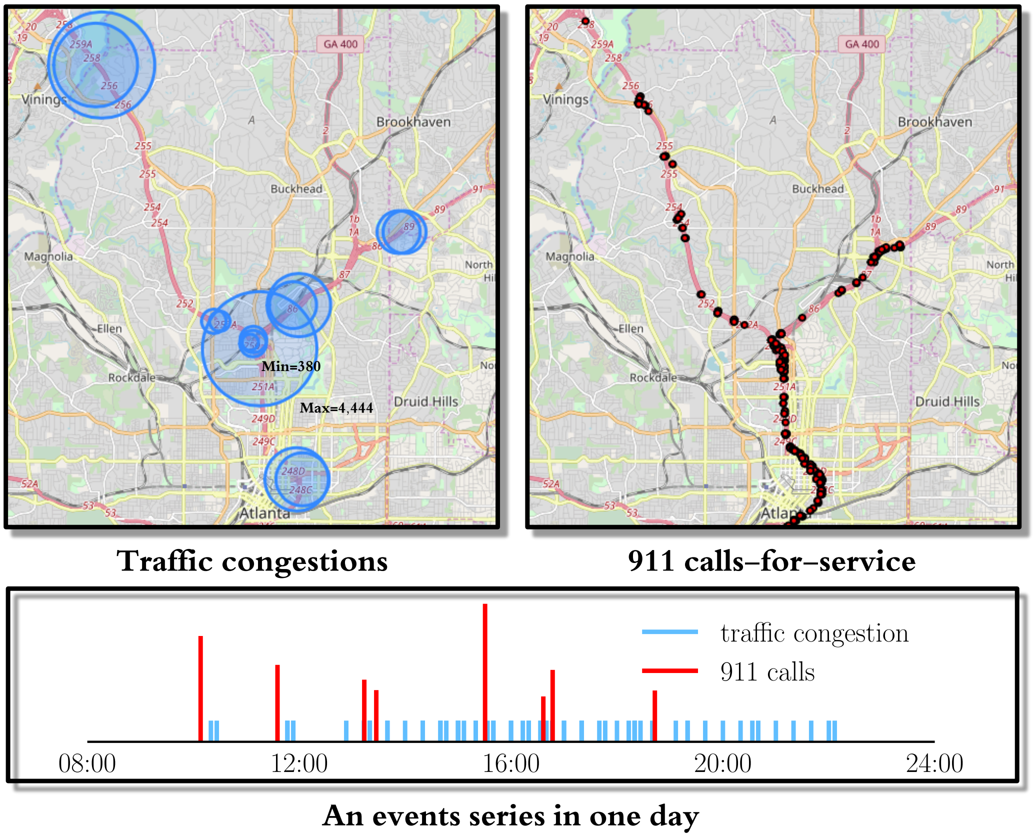
In this thesis, we focus on modeling the traffic congestion in the city of Atlanta. We are trying to predict future congestion events on the main highways in Atlanta. We present a novel framework for modeling traffic congestion events over road networks based on mutually exciting Spatio-temporal point process models. We use multi-modal data by combining traffic sensor networks data with police reports, which contain two types of triggering mechanisms for congestion events. To capture the non-homogeneous temporal dependence of the event on the past, we introduce a novel attention-based approach for the point process model. To incorporate the directional spatial dependence induced by the road network, we adapt the “tail-up” model from the spatial statistics context. We demonstrate the superior performance of our approach compared to the state-of-the-art for both synthetic and real data.

CHAPTER 1
INTRODUCTION AND BACKGROUND

Modeling urban traffic is critical to modern city transportation applications, such as route guidance or road planning. In Atlanta, which has over half-million daily commuters, reducing congestion is a top priority. The city spends lots of money on strategies to mitigate the peak traffic flows, including staggered shifts and toll lanes. Therefore, predicting traffic congestion ahead of time is urgent and crucial. However, the complex spatial-temporal dynamics of traffic flow and the influence of real-time random incidents make it challenging and intricate. As a result, understanding and predicting congestion events can help cities to plan traffic more efficiently and plan for future urban development.

Traffic sensors distributed along highways are widely deployed for monitoring the real-time traffic condition: They are key technology which enables to capture the change and congestion in the traffic network. However, such sensors are limited to collect macro-information of vehicles passing by, i.e., counting, average speed, rather than tracking individual cars due to privacy and technological limitations. These data are widely used in traditional traffic modeling, while few of them considers and model traffic congestion events. An essential feature of traffic congestion modeling is the capability to capture the triggering effects. For instance, when traffic congestion happens, it will propagate along the highway and affect the traffic in another place overtime. Also, the influence of police intervention, which is unpredictable and emergent, can be considered into the framework as well as such incidents will also trigger traffic congestion.

We aim to capture both traffic congestion and police intervention, as well as their triggering effect. Self-exciting point processes are a popular model for modeling such a triggering effect, which has been successfully used in many different applications. A Hawkes process models the dependence between events using mutually dependent point processes,



Traffic congestions

911 calls–for–service

An events series in one day

Figure 1.1: An overview of Atlanta traffic dataset. Left shows the distribution of traffic congestions for each traffic sensor. The size of blue bubble represents the total number of traffic congestion events of a specific traffic sensor. Right shows the distribution of traffic incidents reported by 911 calls. Black dots represent the locations of traffic incidents. Bottom An event series in a single day. The height of the red bar indicates the length of the processing time

whose intensities depend on historical events.

There are two main reasons for the knowledge gap between existing point process models and our application in traffic congestion event modeling: (1) Most existing models assume that the influence function *decays monotonically* over time and space and introduce parametric models for the influence function. For instance, this approach is used in methods based on the popular Recurrent Neural Networks (RNNs) [1, 2, 3, 4, 5, 6, 7], which have achieved various successes in modeling complex temporal dependence: e.g., [1] assumes that the influence of an event decreases or increases exponentially over time. However, in traffic modeling settings, the influence of past events may not decay monotonically over time or space. For example, suppose that a bad car accident occurs on the highway. The police will be called to the scene and may need to wait for a specialized unit, like a crane, to come to move the wreckage. This could take several hours. Meanwhile, the whole highway would be shut down, and the influence of the event would not decay at all. (2) We need to consider the specific spatial correlation structure induced by road networks in our modeling. Most Hawkes process models focus on temporal modeling or discretizing space and treat it as a multi-dimensional Hawkes process. However, it is critical to embed the special spatial correlation induced by the road networks in the model. Indeed, the spatial dependence is highly *directional* and what happens "up-stream" will influence what happens"down-stream", and the sensors along the same road (in the same direction) will have higher correlations.

We aim at filling this gap by presenting a novel spatio-temporal attention-based point process (APP) for traffic congestion event modeling. Specifically, we model the influence of the police intervention for 911-call incidents as an exogenous excitation and use the attention mechanism to capture the dynamics of the endogenous self-excitation between traffic congestions. The attention mechanism [8, 9] is originally proposed to capture the non-linear dependence between words in Natural Language Processing. To capture the complex non-homogeneous influence of historical events on the future, we go beyond the assumption that the influence of the historical event fades over time, and leverage the attention mechanism to develop a flexible framework that "focuses" on past events with high importance score on the current event. We introduce an adaptive *score function* to measure the importance between past events and the current event, which extends the conventional dot-product score [9] used in other attention models and is highly interpretable. To tackle the directional spatial correlation induced by road networks, we also adopt the idea of "tail-up" model (developed for spatial statistics for Gaussian processes) to our point process setting. Finally, to achieve constant memory in the face of streaming data, we introduce an online algorithm to implement the attention component efficiently for our APP model, where only the most informative events in the past are retained for computation. Using experiments based on real data, we show that our proposed method outperforms the state-of-the-art both in maximizing the likelihood function of a point process compared with previous approaches and in prediction accuracy on a real-data traffic data set from Atlanta.

The main contributions are as follows: (1) To the best of our knowledge, our APP model is the first attempt to combine traffic sensor count data with police reports for traffic event modeling; (2) In terms of methodology, our APP model includes a novel attention-based mechanism to capture a non-homogeneous spatio-temporal dependence of the event on the past; (3) the APP model includes a novel approach to capture the directional spatial dependence by adapting similar idea used for the "tail-up" model which was used to model spatial correlation for hydrology systems such as rivers and streams; and (4) experimental results demonstrate the benefits of the APP model both on synthetic and real case studies.

Related work. Most of the previous works [10, 11, 12, 13, 14, 15, 16, 17, 18, 19] on traffic modeling focus on predicting speed, volume, density and travel time, which have achieved remarkable success in this field. Other works [20, 21, 22] target at modeling traffic congestion based on the speed, density of vehicle stream, which gives good mathematical descriptions for traffic flow. However, dynamic traffic event modeling is a new approach and still in nascent stage. Existing work in discrete event modeling using point processes, such as [23, 24, 25, 26], often make strong assumptions and specify a parametric form of the intensity functions. Such methods enjoy good interpretability and are efficient. However, parametric models are not expressive enough to capture the event dynamics in some applications.

Recent interest has focused on improving the expressive power of point process models. There are have been attempts on RNNs based point process models [1, 2, 5, 7], which use RNNs to memorize the influence of historical events. However, the conditional intensity is assumed to be some specific functional forms. There are other attempts [3, 6] in using RNNs to model event dependence without specifying the conditional intensity function explicitly. These works only use RNN as a generative model where the conditional function is not available. They focus on studying different learning strategies since maximum likelihood estimation is not applicable here.

Another recent work [27] has aimed at looking for a more general way to model point processes, where no parametric form is assumed. It uses a neural network to parameterize the hazard function, where the conditional intensity can be further derived by taking the derivative of the hazard function. This approach is highly flexible and easy to compute since no numerical integral calculation is involved. However, the model is only specified using a neural network, which reduces interpretability. In addition, this model only works for temporal events.

A recent work [28] also uses attention to model the historical information in point processes. However, their proposal differs from our APP model because it is still a parametric form and assumes a decaying exponential assumption on the conditional intensity function, which may not capture distant events although they are important. We do not make such assumptions in our APP model and can capture important events as long as their "importance score" is high. Moreover, [28] focuses on temporal point processes while we also consider spatio-temporal point processes; they use the conventional dot-product score function to measure the similarity of two events while we introduce the more flexible score function based on neural networks which are learned from data. Another related work [18] uses two individual attention structure where the temporal and spatial dependences are captured via two attention structures.

CHAPTER 2
METHODOLOGY

In this section, we propose an attention-based point process model2.1 to predict the appearance of traffic congestion and consider the reported 911-call incidents as exogenous excitation.

2.1 Spatio-temporal Point Process

A spatio-temporal point process (**STPP**) is a random collection of points, where each point represents the time and location of an event.[29] In our case, traffic congestion $\{x_1, \ldots, x_{N_x(T)}\}$ is treated as a set of events, where $N_x(T)$ represents the total number of events happen in time horizon $[0, T)$ and in the location space $\mathcal{K}$. Let $\mathcal{H}_t = \mathcal{X}_t$ denote the collection of events taking place before time t, where $\mathcal{X}_t = \{x_i\}_{x_i<t}$. This congestion **STPP** model is characterized via the conditional intensity function $\lambda(t, k|\mathcal{H}_t)$, which is the conditional probability of observing a traffic congestion $(t, k) \in [0, T) \times \mathcal{K}$ given the history $\mathcal{H}_t$.

$$\lambda(t, k|\mathcal{H}_t) = \mathbb{E}[N_k(t, t + dt)|\mathcal{H}_t]/dt \tag{2.1}$$

where $N_k(t, t + dt)$ is the counting measure of events on sensor k in $[t, t + dt]$

2.2 Police Intervention

The police intervention, which in forms of 911-call incidents, will impose additional pressure to the road system. Usually, such pressure will spread via the highway from where the incident occurred. The volume of this exogenous intensity is related with the spatial correlation between two locations u, v, denoted as $\alpha_t(u, v)$, which is time-variant as the urban traffic intensity is changing over time. We will discuss an estimation of this spatial correlation in **next section**

Now we consider a set of 911-call incidents $\mathcal{Y} = \{y_j\}$, where $y_j = t_j, r_j, z_j$ denotes time, location and duration of one incident respectively. When the time of a congestion (t, k) is in the middle of a 911-call incident, i.e. $t \in [t_j, t_j + z_j)$, we add an exogenous excitation to the original intensity function.

$$\mu_1(t, k|\mathcal{Y}) = \sum_{y_j \in \mathcal{Y}} \mathbb{1}\{t \in [t_j, t_j + z_j)\} \cdot C\alpha_t(r_j, r_k) \tag{2.2}$$

where $\mathbb{1}\{F\}$ is the indicator function, i.e., it will take the value of 1 is F is true and 0 otherwise. The C is a constant to capture the influence of spatial correlation $\alpha_t(r_j, r_k)$.

2.3 Attention-based Point Process

The attention-based point process is used to model the nonlinear dependency between the current and past event with the attention mechanism [30, 31]. Usually, point process mod-

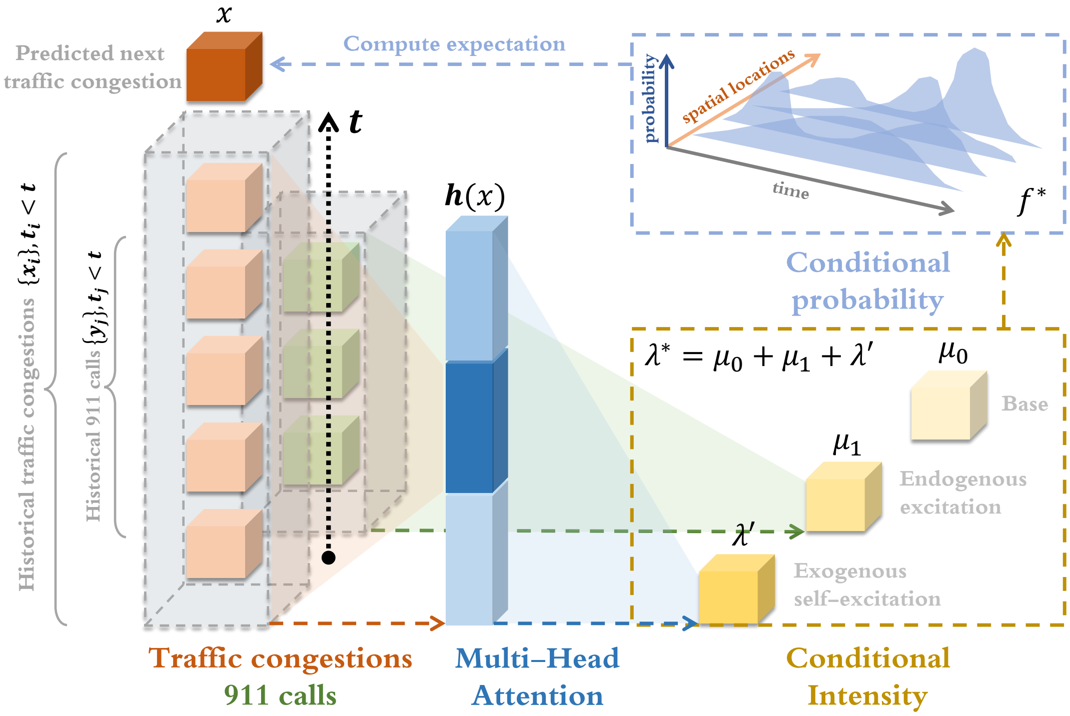


Figure 2.1: The architecture for traffic congestion modeling.

els assume the monotone dependency between events. We model the intensity with an attention neural network so that the dependency on history can be more flexible. Specifically, the endogenous term in the intensity function $\lambda'(t, k|\mathcal{H}_t)$ is based on the attention network. Moreover, we also introduce 'multi-head' mechanisms [31], which captures the representation in different subspace of event sequence.

As shown in Figure2.2, we assume $x_n := (t_n, s_n)$ represent the data point of current congestion happening at time t and located at s. The past congestion events are denoted as $z_i := (t_i, s_i) \in \mathcal{X}_{t_n}$. We introduce multiple heads in the model shown as $h_{m-1}(x_n), h_m(x_n), \dots$ in Figure 2.2. For each head, we use score $v_l(x_n, z_i)$ to evaluate the similarity between one past event and current event, which determines how much attention we should pay on this past event. We will discuss the score function in detail in the next section. To ensure the same weight for each event to analyze, we normalized the score for each $h(x_n)$, denoted as $w_l(x_n, z_i) \in [0, 1]$.

$$w_l(x_n, z_i) = \frac{v_l(x_n, z_i)}{\sum_{z_j \in \mathcal{X}_t} v_l(x_n, z_j)} \tag{2.3}$$

Then we are able to obtain the attention for each head on event x_n by multiplying the score with corresponding embedded past event vector $\phi_l(z_i)$ and adding them up. Formally, it can be written as

$$h_m(x_n) = \sum_{z_i \in \mathcal{X}_t} w_l(x_n, z_i) \phi_l(z_i) \tag{2.4}$$

where $\phi_l(z_i)$ is the embedded value of past event z_i, which is defined as $\phi_l(z) := z^T W$. Here $W \in \mathbb{R}^{\times\shortmid}$ is a weight matrix, where d is the dimension of event z and p is a higher dimension. To obtain the final attention, We concatenate the attentions from each head into the final attention vector $h(x) \in \mathbb{R}^{\mathbb{M}\shortmid}$, where M is the number of heads used in the model.

$$h(x) = \text{concat}[h_1(x), h_2(x), \dots, h_M(x)] \tag{2.5}$$

A nonlinear transformation from attention to the target endogenous intensity $\lambda'(t, k|\mathcal{H}_t)$ is deployed by a neural network with the weight matrix $W \in \mathbb{R}^{Mp}$.

$$\lambda'(x|\mathcal{H}_t) = \text{softplus}(h(x)^T W + b) \tag{2.6}$$

where the function softplus$(x) = log(1 + e^x)$ ensures the strictly positive output and the non-linearity of the model.

Now, we combine the result from 2.1, 2.2, 2.3 and obtain the final expression of the attention-based spatial temporal point process model.

$$\lambda(t, k|\mathcal{H}_t) = \underbrace{\mu_0(t, k)}_{\textbf{background intensity}} + \underbrace{\mu_1(t, k|\mathcal{Y})}_{\textbf{exogenous intensity}} + \underbrace{\lambda'(t, k|\mathcal{H}_t)}_{\textbf{endogenous intensity}} \tag{2.7}$$

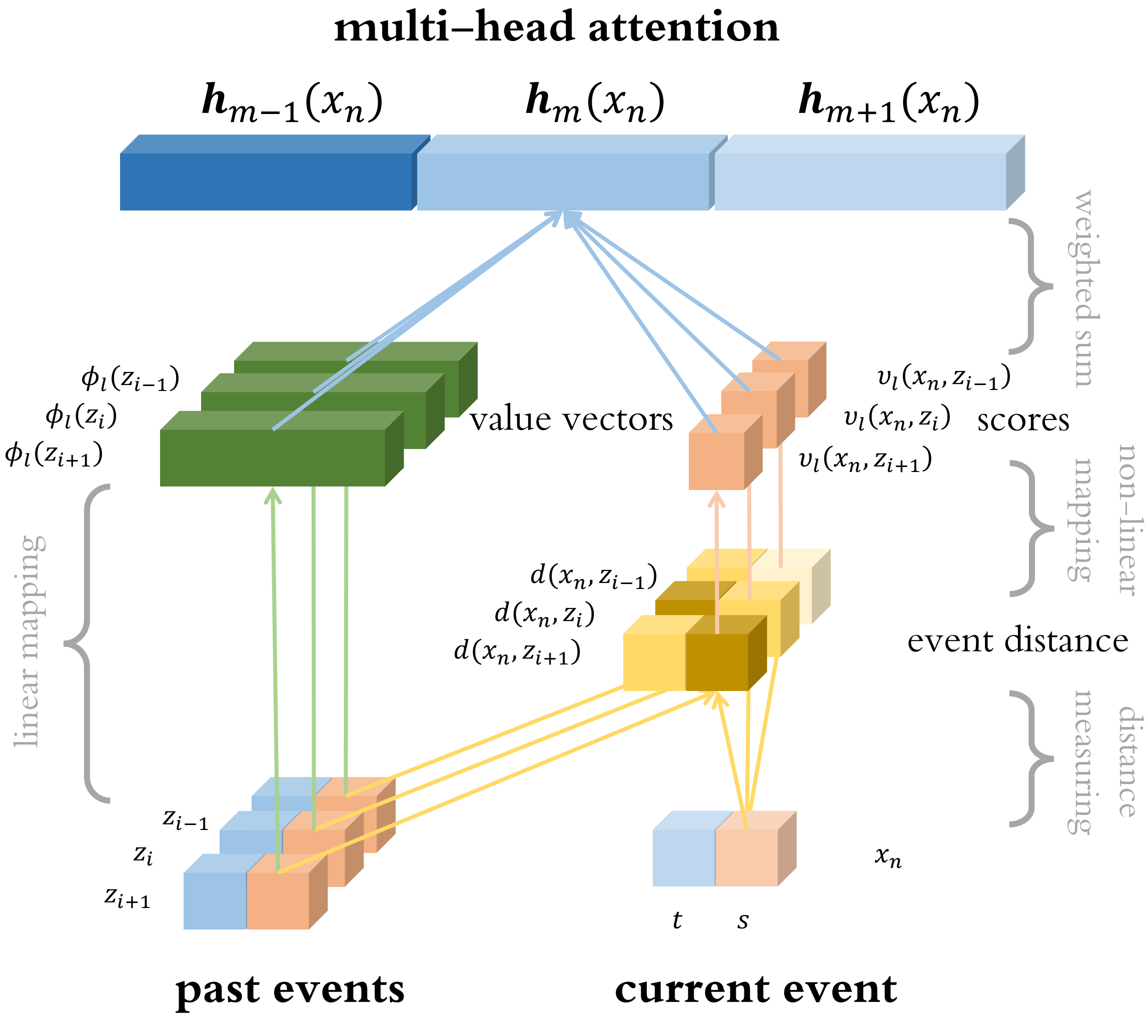


Figure 2.2: The multi-head attention architecture

2.4 Score Function

The score function determines how likely the current event be triggered by another event in the history. In most of attention models, dot-product is used in the score function, which is the Euclidean distance of two events in an embedded space. However, the spatial influence of past event might not follow the rule of dot product especially in the setting of traffic. The effect of events spread along the traffic road with directions and will vary over time. Therefore, we adopt the spatial correlation $\alpha_t(s_n, s_i)$ at time t, which will be discussed in

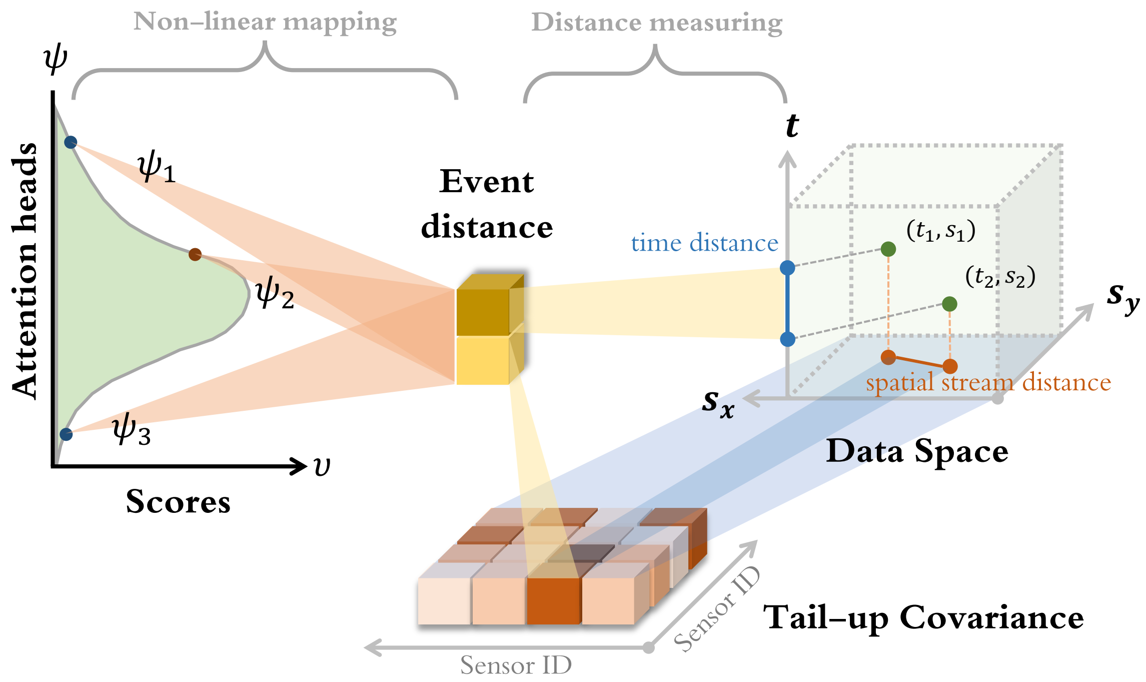


Figure 2.3: An illustration of score function

the next section.

As shown in Figure 2.3, the score $v_m(x_n, x_i)$ for the m-th attention head can be expressed as:

$$v_m(x_n, x_i) = \psi_{\theta_m}(t_n - t_i, \alpha_{t_n}(s_n, s_i)) \tag{2.8}$$

where ψ_{θ_m} is a multi-layer neural network parameters by a set of parameters denoted as θ_m. The input of the neural network is the time difference $t_n - t_i$ and the spatial correlation $\alpha_{t_n}(s_n, s_i)$. The output is a non-negative score which can be interpreted as a weighted spatial-temporal distance. As variant initialization, the output of each head in multi-head structure will be different. Therefore, such structure can capture more information and obtain a higher non-linearity.

2.5 Tail-up Spatial Model in Score Function

We adopt the tail-up spatial model [32, 33, 34] to capture the spatial correlation between two locations $u, v \in \mathcal{K}$ at time t denoted as $\alpha_t(u, v)$. Tail-up model utilizes moving average [32] to construct the spatial correlation on the stream network, which is always used in the analysis on the river system[35, 36, 37]. There are three advantages of tail-up model against other ones: (1) the tail-up model using stream distance rather than euclidean distance, which defined as the shortest distance between two locations along the roads. This ensures the influence of a traffic congestion only spread to the flow connected segments. (2) The model assumes the statistical independence between the unconnected segments. (3) Proper weighting let the sum of output variance of traffic flow from an intersection is equal to the inputs, which ensures the stationary of the stream system. A traffic congestion event may only cause congestion on the spots upstream. Therefore, if there is traffic come from location u to v, we denote u is flow-connected to v.

The traffic at location $u \in \mathcal{K}$ can be viewed as a white-noise random process Z_u. Thus, the random variable of the other observable location can be developed as the integration of moving average function of this white noise process along the road network [33].

$$Z_u = \mu_u + \int_{\vee_u} g(r-u)\sqrt{\frac{w(r)}{w(u)}}dB(r) \tag{2.9}$$

where, μ_u is the mean process at location u, $\vee_u$ denotes the upstream of location u. $w(r) = w^l$ which denotes the weight on segment l and $r \in l$. The weights of segments are pre-calculated to ensure the stationary of variance [36]. In the traffic case, the weights can be estimated by the average volume of vehicles on each road segment. Moving average function $g(\cdot)$ should be square-integrable and defined on $\mathbb{R}$ [38]. $B(r)$ is a Brownian process processing towards the end of traffic network. The spatial correlation $\alpha(u, v)$ is obtained via the covariance between the moving average random variables Z_u, Z_v by

$\mathrm{cov}(Z_u, Z_v) = \mathbb{E}(Z_u Z_v) - \mathbb{E}(Z_u)\mathbb{E}(Z_v)$.

$$\alpha(u, v) = \int_{\vee_u \cap \vee_v} g(s-u)g(s-v)\frac{w(s)}{\sqrt{w(u)w(v)}}ds \tag{2.10}$$

Denote $\Delta(r)$ as the stream distance between spots u and v. We can deploy assumptions on the integration term $C(\Delta(r)) = \int_{\mathbb{R}} g(r)g(r - \Delta(r))dr$. By choosing a proper moving average function, we can re-parametrize the integration $C(\cdot)$. We use exponential tail-up model here [33].

$$C(\Delta(r)) = \beta exp(-\Delta(r)/\sigma) \tag{2.11}$$

where, β, σ are parameters of the model. Let $d(u, v)$ be the stream distance between location u, v, the above covariance can be simplified as,

$$\alpha(u, v) = \begin{cases} C(d(u, v))\sqrt{\frac{w(u)}{w(v)}} & u, v \text{ flow-connected} \\ 0 & u, v \text{ flow-unconnected} \end{cases} \tag{2.12}$$

2.6 Online Attention Model for streaming data

For streaming data, the attention calculation may have increasingly computational complexity as the growth of the number of past events. Here, we propose an adaptive online attention algorithm to address this issue, where only a fixed number of ’important’ history events are taken into the calculation. We only consider the events with a higher average scores, which indicates a higher similarity to the current event. In both simulation and real data experiment, we show that this is a good estimation to the original attention algorithm as a small part of events could impose dominant influence on their future events. This online estimation can make the APP model more efficient without too much expense of accuracy.

The procedure for collecting such ’important’ events can be demonstrated as following: Firstly, for the j-th past event x_j, we calculate the set of its scores against all future events

x_n as $\mathcal{C}_{j,m} \coloneqq \{w_m(x_j, x_n)\}_{t_j<t_n}$ for attention head m. Then, the average score for event x_j can be computed by $\mu_{j,m} = (\sum_{s\in\mathcal{C}_{j,m}} s)/|\mathcal{C}_{j,m}|$, where $|A|$ denotes the number of elements in set A. We define the set of events to be used in the online procedure $\mathcal{A}_{n,m}$ recursively. Denote η as the max number of event to retain in the process.

$$\begin{cases} \mathcal{A}_{n,m} = \mathcal{X}_{t_n+1}, & \forall n \leq \eta \\ \mathcal{A}_{n,m} = \mathcal{A}_{n-1,m} \cup \underset{z_j:t_j<t_n}{\text{argmax}}(\mu_{j,m}) \cap \underset{x_j:t_j<t_n}{\text{argmin}}(\mu_{j,m}), & \forall n > \eta \end{cases} \tag{2.13}$$

To perform the online attention, we use the event set $\mathcal{A}_{n,m}$ in place of $\mathcal{X}_{t_n}$.

2.7 Learning and Inference

Hence, the model is fully defined with a set of unknown parameters $\{W, b, \gamma, \beta, \sigma, \{\theta_m, W_m\}_{m=1}^M\}$. We fit the model by maximum likelihood, which can be solved by the stochastic gradient descent. The close form of likelihood can be written from the conditional intensity in 2.7. Suppose there a total of n samples before the time horizon T denoted as $\boldsymbol{x} = \{(t_i, s_i)\}_{i=1}^{N_x(T)}$. Let $F^*(t,k) = \mathbb{P}\{t_{n+1} < t, k|\mathcal{H}_t\}$ be the conditional probability that the next congestion event(t_{n+1}, k) happens before t given the history of the previous events and let $f^*(t,k)$ be the corresponding density probability. We use $\lambda^*(t,k)$ denotes the conditional intensity function $\lambda(t,k|\mathcal{H})$ for convenience, which is defined as $\lambda^*(t,k) = f^*(t,k)/(1-F^*(t,k))$. By definition, we can show that $\lambda^*(t,k) = d\text{log}(1-F^*(t,k))/dt$. Therefore, $\int_{t_n}^t \lambda^*(\tau,k)d\tau = -log(1-F^*(t,k))$. If the $(n+1)$-th event does not exist at the time of t_n, $F*(t,k) = 0$. Thus, $F*(t,k) = 1 - \text{exp}\{-\int_{t_n}^t \lambda^*(\tau,k)d\tau\}$ and

$$f^*(t,k) = \lambda^*(t,k) \cdot \text{exp}\{-\int_{t_n}^t \lambda^*(\tau,k)d\tau\} \tag{2.14}$$

Then the log-likelihood of observing the sequence $\boldsymbol{x}$ is:

$$\ell(\boldsymbol{x}) = \sum_{i=1}^{N_x(T)} \log\lambda^*(t_i, s_i) - \sum_{k\in\mathcal{K}} \int_0^T \lambda^*(t, k)dt \tag{2.15}$$

The second integration term cannot be computed analytically. Thus, We use numerical integration as an estimation here. Given a sequence of events $\{x_i\}_{i=1,\ldots,n}$, we can estimate the next events $(\hat{t}_{n+1}, \hat{s}_{n+1})$ by calculating the expectation of conditional probability in 2.14:

$$\begin{bmatrix} \hat{t}_{n+1} \\ \hat{s}_{n+1} \end{bmatrix} = \begin{bmatrix} \int_{t_n}^T \tau \sum_{k\in\mathcal{K}} f^*(\tau, k)d\tau \\ \underset{k\in\mathcal{K}}{\text{argmax}} \int_{t_n}^T f^*(\tau, k)d\tau \end{bmatrix} \tag{2.16}$$

CHAPTER 3

EXPERIMENT RESULTS

In this section, we conduct experiments on four synthetic datasets and three public real datasets to illustrate the effectiveness of our model in capturing the temporal patterns. Then, we use the model on the Atlanta traffic dataset to test its performance on the real spatio-temporal case. We evaluate our model with/without online attention(*APP/OAPP*) and other baseline methods by comparing their log-likelihood and visually show their conditional intensity function in both temporal and spatial scenarios. There are five baseline models that we test as following.

Long-Short Term Memory(*LSTM*) is a specialized recurrent neural network to deal with sequential data modeling. The historical events are feed as a high-dimensional embedding and then we can generate the hidden states. Given the last hidden state, we can generate the next event.

Recurrent Marked Temporal Point Process (*RMTPP*) [39] utilizes the following structured conditional intensity function λ^*. Formally, it is defined as $\lambda^*(t) = \exp(\boldsymbol{v}^T \boldsymbol{h}_j + w(t - t_j) + b)$, where $\boldsymbol{h}_j$ represent the j-th hidden space treated as the historical influence up to the j-th event. $w(t - t_j)$ denotes the time influence on the intensity function. $\boldsymbol{v}, w, b$ are trainable parameters.

Neural Hawkes Process(*NHP*)[40] defines the conditional intensity function λ^* with a continuous-time long-short term memory structure, denoted as $\lambda^* = f(\boldsymbol{w}^T \boldsymbol{h}_t)$, where the hidden state of time t represents the historical influence. Here, w is trainable parameter and $f(\cdot)$ is a softmax function to ensure the output is always positive.

Self-Attentive Hawkes Process(*SAHP*)[41] also uses attention neural network to modeling point process. The conditional intensity function λ^* is defined as $\lambda^*(t) = \text{softmax}(\mu + \alpha\text{exp}(w(t - t_j)))$, where μ, α, w are computed via a nonlinear layer of neural network

$\mu = \text{softplus}(\boldsymbol{h}W_\mu)$, $\alpha = \tanh(\boldsymbol{h}W_\alpha)$, $w = \text{softplus}(\boldsymbol{h}W_w)$. Here, $\boldsymbol{h}$ is the historical embedding compute from an attention neural network via computing the scores between history and current event.

Hawkes Process (*HP*) [42] is the original and the state of art method on modeling the temporal point process. The conditional intensity function is defined as $\lambda^*(t) = \mu + \alpha \sum_{t_j<t} \beta\text{exp}(-\beta(t - t_j))$, where the parameters μ, α, β are trainable. We can derive the log-likelihood formula via this definition and train the model by maximizing the likelihood.

In the experiment, we use 3-layer neural network for the structure in 2.4 and utilize 3 attention heads. Each dataset are split with 80% training set and 20% for testing. First, we estimate the unknown parameters in the model by maximizing log-likelihood via training samples. Here, we employ Adam optimizer to minimizing the negative log-likelihood, which is defined in the equation 2.15. We use a learning rate with initial value 10^{-3} and decreasing exponentially every epoch. The batch size in the experiment is 64. For the testing part, we plug in the parameters and evaluate both conditional intensity and log-likelihood of testing data. Plus, for online learning model (*OAPP*), the max number of events to be retained is 10. i.e. $\eta = 10$ in the equation 2.13.

3.1 Synthetic Data

To evaluate the performance of our model in evaluating the conditional intensity function, we first conduct the simulation on time series analysis. In the experiment, the ground truth of intensity function is given and the events are generated via thinning algorithm. To adapt our *APP* model to time series modeling, we only consider the temporal distance when calculating the score function in the attention network. Therefore, the equation 2.8 is modified to $v_m(x_n, x_i) = \psi_{\theta_m}(t_n - t_i)$.

We perform the simulation on four dataset listed in figure 3.1 and 3.2, including Hawkes Process, Self-correcting process and two non-homogeneous process. (1) **Hawkes Process**: the intensity function is given by $\lambda^*(t) = \mu + \alpha \sum_{t_j<t} \beta\text{exp}(-\beta(t - t_j))$, where $\mu = 10$,

Table 3.1: Average maximum log-likelihood on synthetic data.

Data set	SAHP	NHP	RMTPP	APP	OAPP
Hawkes	20.8	20.0	19.7	**21.2**	21.1
self-correction	3.5	5.4	6.9	7.1	**7.1**
non-homo 1	432.4	445.6	443.1	442.3	**457.0**
non-homo 2	364.3	410.1	405.1	**428.3**	420.1

$\alpha = 1$ and $\beta = 1$ are used in our experiment;(2) **Self-correcting process**: The conditional intensity function is given by $\lambda^*(t) = \exp(\mu t - \sum_{t<t_j} \alpha)$, which means the conditional probility is increasing exponentially and will drop once a correction event happens. In the experiment, we use $\mu = 10, \alpha = 1$; (3) **non-homogeneous Poisson** 1: THe intensity funciton is given by $\lambda^*(t) = c \cdot \Phi(t - 0.5) \cdot U[0, 1]$, where $c = 100$ is the total number of events observed and $\Phi(t)$ is the PDF of a standard normal distribution;(4) **non-homogeneous Poisson** 2: The intensity function has two peaks which is a composition of two normal distributions centered at different t. Formally, the intensity function can be written as $\lambda^*(t) = c_1 \cdot \Phi(6(t - 0.35)) \cdot U[0, 1] + c_2 \cdot \Phi(6(t - 0.75)) \cdot U[0, 1]$, where $c_1 = 50$ and $c_2 = 50$ which indicates the number of events generated from each distribution. For every simulated conditional intensity function, we generate the set of training sequence with a size of 5000, where the data point only contains the time information. i.e. the time when the event occurs.

Figure 3.1 shows the results of average log-likelihood and variance versus training epochs for each simulation dataset. The higher log-likelihood, the better performance of the model. The red dash lines represent the log-likelihood of using *APP* model. For figure 3.1 we can see our *APP* model outperform the other baseline models. Moreover, *OAPP* model only choose the most significant historical event to calculate the attention score, which has very little loss in term of final log-likelihood compared with the *APP* model.

Figure3.2 shows the intensity function of each model, which is more intuitive and can be compared with the baseline(the gray line) directly. For homogeneous cases such as

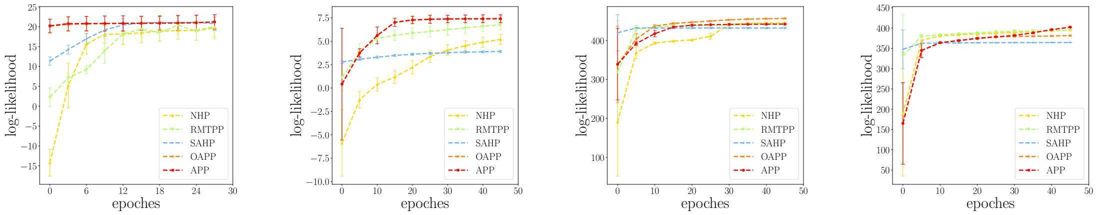
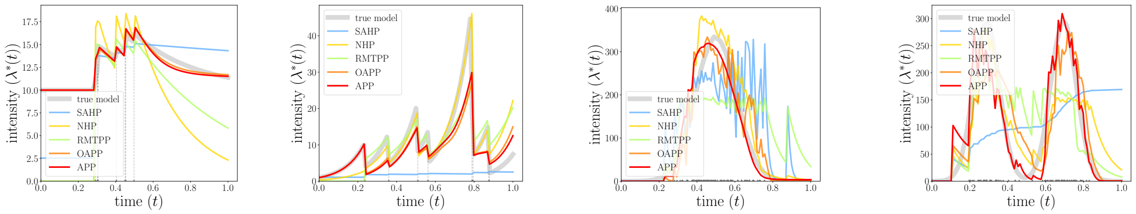


Figure 3.1: The average log-likelihood of synthetic data versus the number of epochs. For each data set, we maximize its log-likelihood to evaluate the performance of different models.



(a) Hawkes (b)Self-correcting (c)Non-homogeneous (d)Non-homogeneous

Figure 3.2: The conditional intensity function estimated from the synthetic data. The dash line and triangle on the x-axis denotes the time when events happen. The gray line indicates the ground truth generate by thinning algorithm.

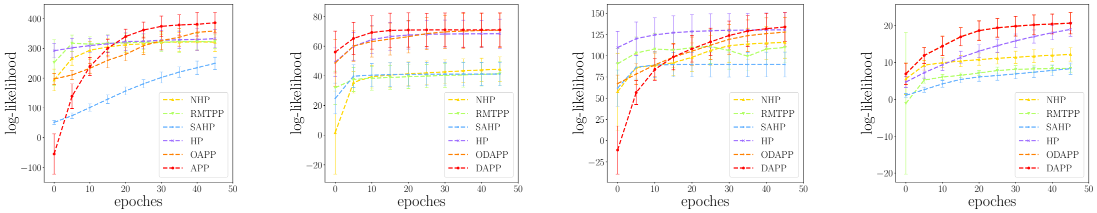
Hawkes process and self-correcting process, our *APP* model can capture the amplitude and the decaying rate precisely. As shown in previous section, our model do not provide the exponential term in the intensity function, but we can capture the exponential shape decay process such as Hawkes process. For non-homogeneous cases, *APP* model can output a much smoother intensity curve compared with the baselines. Besides, the *OAPP* model can capture the main shape of intensity curve in both homogeneous and non-homogeneous cases.

3.2 Experiment on Real Temporal Dataset

We test our model on some public temporal point process datasets as well as on the Atlanta traffic data set. As there is no accurate intensity function for a real data set, we compare the log-likelihood of each model to figure out which one fit the real event sequence best.

Atlanta Traffic Data: Our Atlanta traffic dataset is a sequence of time and locations where a traffic congestion happens on two main highways at Atlanta. Here, we only consider the temporal information. **Stock Data**: [43] Stock data includes the $11k$ stock trading information. **Citation Data**: [44] Patent data includes the $100k$ citation sequences. **Tweet Data**: [45] Tweet collect the tweet and retweet timestamp for $22k$ tweets. We test our *APP* and *OAPP* model on these four real datasets and compare the result with baselines, shown in Figure 3.3

As shown in figure 3.3, for all the dataset, the *APP* and *OAPP* have a higher log-likelihood after convergence, which shows our model fits the real data set very well.



(a)Traffic (b)Stock (c)Citation (d)Tweet

Figure 3.3: The log-likelihood of *APP* and baseline models on the real dataset. The higher log-likelihood, the better performance.

3.3 Spatial-Temporal Analysis on Traffic Dataset

In this section, we further consider the spatial temporal cases to illustrate the effectiveness of our *APP* model. In Atlanta traffic dataset, the locations are categorized into 14 with latitudes and longitudes, which are the location of traffic sensors. In addition, we conduct the tail-up spatial correlation model in this section. Therefore, our comparison in this experiment part will not only on the baseline models but also on whether tail-up spatial correlation model improve the performance of *APP* model. We first use Euclidean distance in the score function denoted as (*APP+Euclidean*), which is compared with the one with Tail-up model denoted as (*APP+Tailup*). In this section, we evaluate the model with the final log-likelihood and the prediction accuracy. Our model is trying to predict the future locations of events. Plus, we visualize the scores of APP model and the conditional

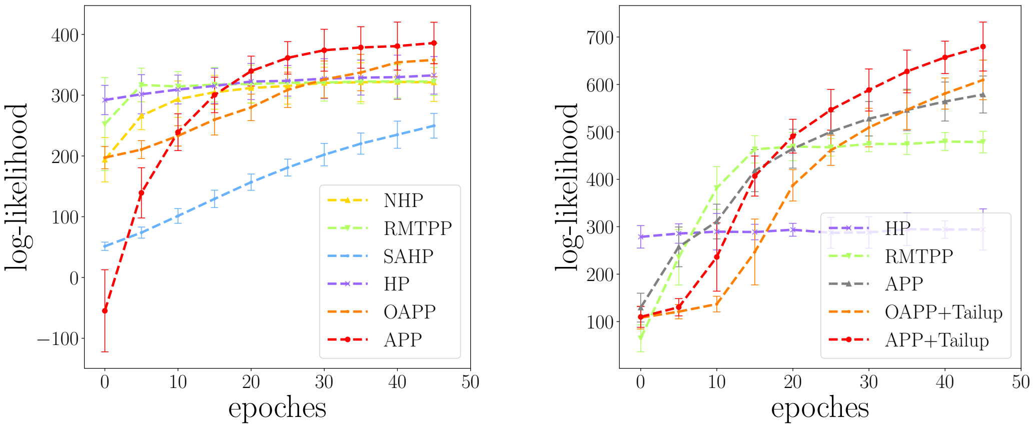
Table 3.2: Average maximum log-likelihood and prediction accuracy using Atlanta traffic dataset.

Models	$\max \ell$ (time only)	$\max \ell$ (time & space)	prediction accuracy
LSTM	N/A	N/A	18.5%
HP	339.9	307.5	8.82%
RMTPP	339.2	490.1	22.0%
NHP	324.4	N/A	N/A
SAHP	326.7	N/A	N/A
APP + Euclidean	392.3	570.7	30.9%
APP + Tailup	**458.5**	**636.2**	**37.6%**
OAPP + Tailup	437.5	615.9	36.9%

intensity function to provide some intuitive illustration of the fitting results.

Figure 3.4 shows the average log-likelihood for each method over the number if training epochs. Our *APP* model (red dash line) outperforms the other baseline methods in both temporal and spatial& temporal scenarios. Besides, by adding up space information to the model, the log-likelihood improves a lot compared with the time-only experiments. Also, we show that the *APP+Tailup* (red dash line in the right figure) has the bests performance, which shows the effectiveness of the *APP* model and the tail-up spatial correlation assumption to modeling the traffic congestion problem.

Besides, we also evaluate the conditional intensity of each traffic sensor using *APP* model. There are 14 sensors in total along I-75 and I-85 in Atlanta, which are visualized as heatmaps on figure 3.5. We select two representative days May 8th, 2018 and April 24th, 2018, which show different patterns of the temporal and spatial information of traffic congestion. On May 8th, 2018, the conditional intensity shows two peaks at 7am and 16pm, which is the normal rush hour for a workday. In the heatmap, the sensors(rows) are arranged by its location on the highway. i.e. the adjunct sensors are grouped together. Thus, the slight propagation of conditional intensity indicates the congestion events travel along the road between sensors, which refers to a traffic congestion phenomenon 'phantom



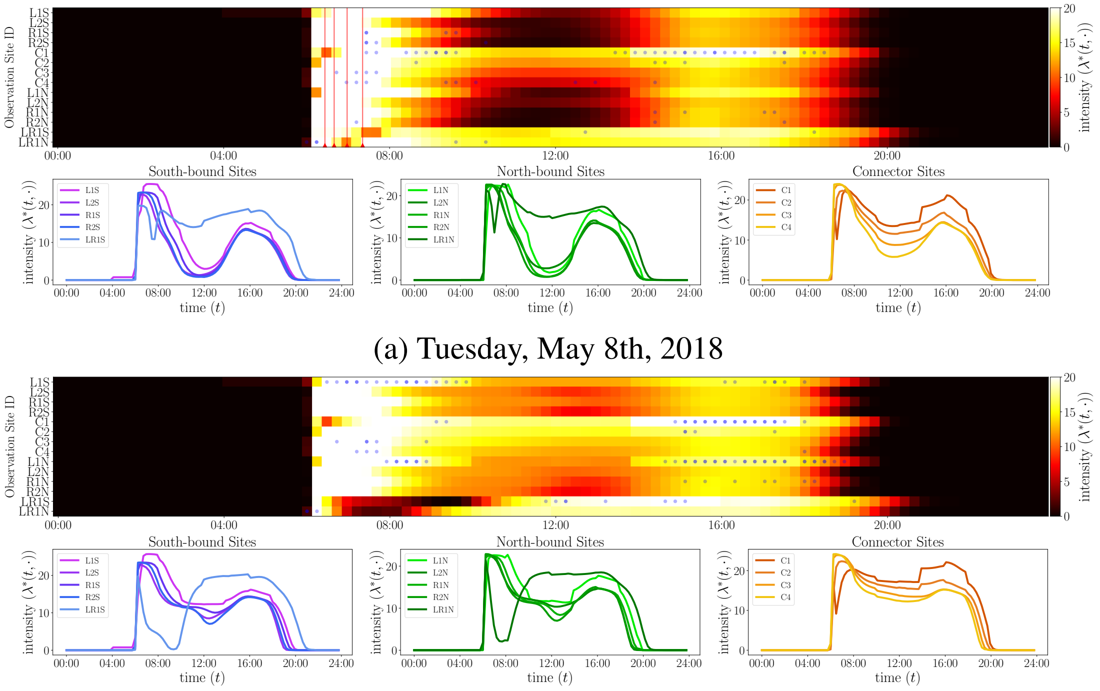
Time only

Space & time

Figure 3.4: The average log-likelihood versus the number of training epochs with or without considering the spatial information

traffic jam' [46]. Due to the response time, when the front vehicles restart on the road, there will be delay for the following vehicles to catch up. Thus, this will cause the propagation of slowing down of traffic flow. Eventually, it will cause congestion even without traffic accidents. Referring to the red vertical lines, when a 911-call incident happen on the traffic network, there always be congestion events(blue dots). On contrast, we select another day, April 24th, 2018, as an example of atypical conditional intensity function, when special events affect the road system. Compared with what on May 8th, the intensity function on April 24th are more flatten and doesn't show clear peaks. As AJC news shown[47], there was a record-break rain in that day, which cause a strong impact on the traffic system.

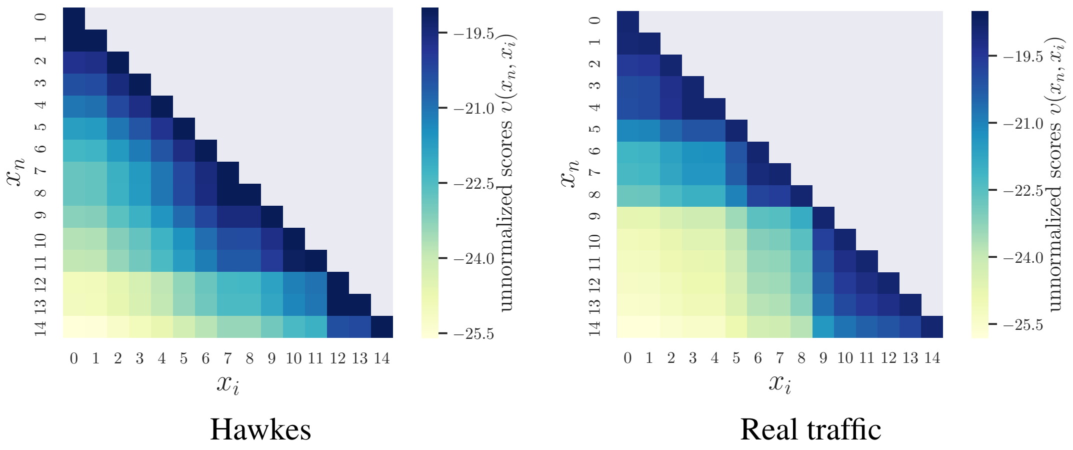
Score function: In the real data experiment, we retrieve the score function of *APP* model shown as the right figure in Figure 3.6. In previous chapter, we describe the score function is computed via the distance between current event and historical events. Compared with the score function of synthetic Hawkes process (left figure), the real data score has community structure and shows higher non-linearity. The brighter the cell is, the lower the score between current events (y-axis) against historical events (rows). Events in Hawkes process has a monotonously decreasing influence on current event which decreases exponentially with time. Correspondingly, there shows block structure in the traffic structure,



(a) Tuesday, May 8th, 2018

(b) Tuesday, April 24th, 2018

Figure 3.5: Heatmaps of the conditional intensities of 14 traffic sensors on two typical days, where the rows represent a traffic sensor, each column represent a 5-minute time slot. The color depth represent the level of intensity, the blue dots are the time of traffic congestion events happen, and the red vertical line is the time of 911-call. These sensors are categorized into three groups southbound, northbound and traffic connectors. The bottom line charts shows the conditional intensities for different groups.



Hawkes

Real traffic

Figure 3.6: Visualization of scores between pairwise events in the sequence, learning from synthetic Hawkes process and real Atlanta traffic data

which indicates part of historical events has similar influence on the current traffic congestion. Digging into the events, we find the sensors in the same bound of highways will has similar scores. For instance, in the right figure 3.6, first 3 incidents happened on the northbound of I-75, which had less effect on the last 6 event happened on I-85 southbound and had less scores because they are flow unconnected.

CHAPTER 4
DISCUSSION

In this chapter, we will discuss the disadvantages of the attention point process model. Our model has high flexibility and capability to capture the dependency between historical events and future events, which require a precise calculation on the score between events. Thus, the calculation will be complicated so that we put forward the online version to estimate the model. However, one of the disadvantages is it is hard to balance the computational complexity and accuracy. i.e., for different datasets, the set of threshold η always vary from each other. Sampling and testing on the original dataset should be down every time before online attention model to find a proper number of historical events. Moreover, the attention structure itself is more complicated compared with the baseline models, so it requires more feeding data and time to fit.

CHAPTER 5
CONCLUSION

We develop a novel attention-based point process model for modeling the dynamics of traffic congestion with consideration of the influence of 911-call incidents reported by the police. The goal is to model traffic congestion events and the triggering effect while taking advantage of the structure knowledge of the traffic network.

As demonstrated by our experiments, our method achieves the best performance in maximizing the likelihood function of a point process compared with previous approaches as well as prediction accuracy on the traffic dataset. Besides, by implementing various kinds of point process models, we show that our model exceeds the others in terms of robustness and flexibility. Furthermore, based on the structural information of dynamic networks, our model can be generalized in such a the way that the prediction of the current event of a particular type might depend more on some specific kinds of events by exploring the structure of the score matrices. This gives us a a new method for implementing causality inference in networks.

Appendices

APPENDIX A
ATLANTA TRAFFIC DATA

In this section, we introduce a unique large-scale traffic dataset, which consists of three sub-datasets: (1) traffic congestion sub-dataset; (2) 911 call-for-service sub-dataset; and (3) traffic network sub-dataset.

A.1 Traffic congestion

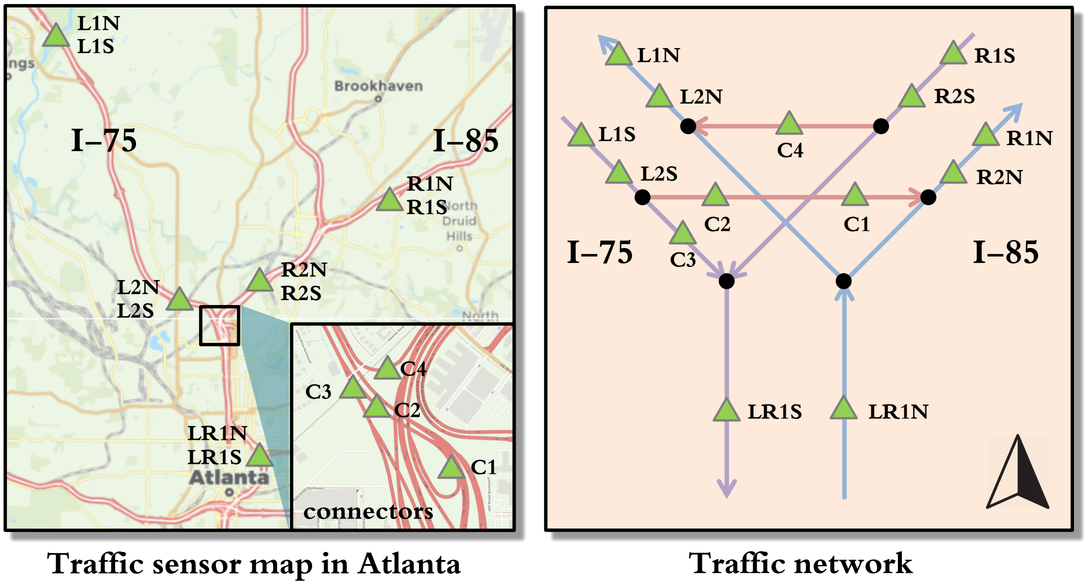
The traffic congestion data is a sub-dataset collected from Georgia Department of Transportation (GDOT) [48], which records the real-time traffic condition on roads throughout the state of Georgia. These traffic data are recorded by traffic sensors installed on main traffic points in the highway system, where the data of each sensor is organized as a series of numbers that indicate how many vehicles pass through the sensor every 5 minutes. The dataset also provides lane information at the locations where the sensors are installed. The number of lanes at the specific location of the highway allows us to estimate the maximum number of vehicles that the highway is able to process. We assume that the maximum number of vehicles that a highway can process is a linear function of the number of lanes.

Here, we consider 14 traffic sensors installed on two major highways (I-75 and I-85) in Atlanta shown in Figure A.1, indexed by $\mathcal{K} = \{1, 2, \dots, K\}, K = 14$ and we denote their geo-locations (latitude and longitude) on the traffic network by $r_k \in \mathcal{S} \subset \mathbb{R}^2, \forall k \in \mathcal{K}$, where $\mathcal{S}$ is the location space of the traffic network, which will be discussed in Section A.3. A traffic congestion event can be detected at certain time by a traffic sensor when the real-time traffic count exceeds the maximum number of vehicles that are allowed to pass through. Let $\{x_i\}_{i=1}^{N_x(T)}$ represents a sequence of traffic congestion events in a single day, where $N_x(T)$ is the number of the congestion events generated in the one-day horizon $[0, T)$. The i-th congestion event $x_i = (t_i, s_i)$ is a data tuple consisting of the occurrence time $t_i \in [0, T)$, the sensor index $s_i \in \mathcal{K}$. We extracted 18,618 traffic congestion events for 174 days between April 2018 and December 2018 from the sub-dataset. The maximum and minimum number of events in a single day is 168 and 19, respectively.

A.2 911 calls-for-service

As mentioned in introduction, traffic incidents may trigger unexpected congestion on the traffic network. We collected another sub-dataset from 911 calls-for-service reports for the traffic incidents provided by Atlanta Police Department (APD) [26, 49]. Such reports are generated by mobile patrol operations in the city, which handle 911 calls twenty-four hours a day. When a 911 call about a traffic incident comes in, a new incident record, including the *call time* and occurrence location, will be created at the dispatch center. Typically, after the new call arrives, the operator will assign an officer to handle the call. The unit arrives at the scene and starts the investigation. Once the police complete the investigation and clean the scene, the police report will be closed and record the *clear time*. The time interval that takes police to process the call between the call time and the clear time is called *processing time*. A 911 call with long processing time usually imposes a significant impact on the traffic condition of the highway where the 911 call is initiated.

Let $\{y_j\}_{j=1}^{N_y}$ represent a sequence of traffic incidents reported by 911 calls in a single day, where N_y is the total number of the recorded 911-call incidents in one day. The j-th 911-call incident $y_j = (t_j, r_j, z_j)$ is a data tuple consisting of the call time $t_j \in [0, T)$, the occurrence location $r_j \in \mathcal{S}$ on the traffic network, and the processing time $z_j \in \mathbb{R}_+$ indicating the length of time that the police takes to resolve the case. We select 19,805 such 911-call incidents that occurred on two major highways from the same period (between April 2018 and December 2018) with processing time larger than 15 minutes. Recorded 911-call incidents span over ten different categories, ranging from speeding tickets to massive car pileups.



Traffic sensor map in Atlanta **Traffic network**

Figure A.1: The traffic network for major highways in Atlanta. *Left* shows the spatial distribution of traffic sensors, where green triangles represent locations of traffic sensors. Traffic sensors on the highway are bi-directional, i.e., two directions of the same location have separate traffic sensors to monitor the traffic condition. *Right* shows the traffic network and where traffic sensors located on the network. Each line segment represents one specific road segment and black dots represent the confluence of two roads.

A.3 Traffic network

Due to the nature of traffic flow, there are strong spatial dependence among the traffic data collected at different locations on the network. The network topology and the direction of the flow impose constraints on modeling such spatial correlations. For example, there should not be correlation for data collected at two locations that do not share common traffic flow. In Downtown Atlanta, there are two major highways I-75/85 through the core of the city. Beginning at northwest/northeast of the city, two highways generally run due south, meeting each other in the Midtown as shown in the left of Figure A.1. Between I-75 and I-85, there are also two connectors that bridge two highways via single-direction ramps.

We extracted the network information of I-75 / I-85 and their connectors in Atlanta from OpenStreetMap [50], which is an editable map database and allows us to construct, visualize, and analyze complex traffic networks. The traffic network of a city is represented by a set of road segments defined in the OpenStreetMap dataset as shown in the right of Figure A.1. Let $\mathcal{S} \subset \mathbb{R}^2$ represents the set of all geo-locations on the network. We index road segments on the network by $\mathcal{L} = \{1, \ldots, L\}$, where the set of locations on each segment is denoted as $\mathcal{S}_l \subset \mathcal{S}, \forall l \in \mathcal{L}$. For any location $s \in \mathcal{S}$ on the network, we define the upstream portion $\vee_s \subseteq \mathcal{S}$ of the network to include s itself and all locations upstream from s. We define the downstream portion $\wedge_s \subseteq \mathcal{S}$ to include s itself and all locations downstream from s. For two locations $u, v \in \mathcal{S}$, the distance $d(u, v) \in \mathbb{R}^+$ is defined as the stream distance along the highway if one of the two locations belongs to the downstream of the other. We denote $u \to v$ when v belongs to $\vee_u$ and the two points are said to be *flow-connected*. When two points are *flow-unconnected*, neither u belongs to $\wedge_v$ nor v belongs to $\wedge_u$, and the relationship between u and v is denoted $u \not\to v$.

REFERENCES

[1] N. Du, H. Dai, R. Trivedi, U. Upadhyay, M. Gomez-Rodriguez, and L. Song, “Recurrent marked temporal point processes: Embedding event history to vector,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD ’16, San Francisco, California, USA: Association for Computing Machinery, 2016, pp. 1555–1564, ISBN: 9781450342322.

[2] H. Mei and J. M. Eisner, “The neural hawkes process: A neurally self-modulating multivariate point process,” in *Advances in Neural Information Processing Systems 30*, Curran Associates, Inc., 2017, pp. 6754–6764.

[3] S. Li, S. Xiao, S. Zhu, N. Du, Y. Xie, and L. Song, “Learning temporal point processes via reinforcement learning,” in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, ser. NIPS ’18, Montréal, Canada: Curran Associates Inc., 2018, pp. 10 804–10 814.

[4] U. Upadhyay, A. De, and M. Gomez Rodriguez, “Deep reinforcement learning of marked temporal point processes,” in *Advances in Neural Information Processing Systems 31*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., Curran Associates, Inc., 2018, pp. 3168–3178.

[5] S. Xiao, J. Yan, X. Yang, H. Zha, and S. M. Chu, “Modeling the intensity function of point process via recurrent neural networks,” in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, ser. AAAI ’17, San Francisco, California, USA: AAAI Press, 2017, pp. 1597–1603.

[6] S. Xiao, M. Farajtabar, X. Ye, J. Yan, L. Song, and H. Zha, “Wasserstein learning of deep generative point process models,” in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS ’17, Long Beach, California, USA: Curran Associates Inc., 2017, pp. 3250–3259, ISBN: 9781510860964.

[7] S. Zhu, H. S. Yuchi, and Y. Xie, “Adversarial anomaly detection for marked spatio-temporal streaming data,” 2019. eprint: `arXiv:1910.09161`.

[8] T. Luong, H. Pham, and C. D. Manning, “Effective approaches to attention-based neural machine translation,” in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal: Association for Computational Linguistics, Sep. 2015, pp. 1412–1421.

[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in Neural Information Processing Systems 30*, Curran Associates, Inc., 2017, pp. 5998–6008.

[10] A. Abadi, T. Rajabioun, and P. A. Ioannou, “Traffic flow prediction for road transportation networks with limited traffic data,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 653–662, 2015.

[11] Y. Lv, Y. Duan, W. Kang, Z. Li, and F. Wang, “Traffic flow prediction with big data: A deep learning approach,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 865–873, 2015.

[12] X. Ma, Z. Dai, Z. He, J. Ma, Y. Wang, and Y. Wang, “Learning traffic as images: A deep convolutional neural network for large-scale transportation network speed prediction,” *Sensors*, vol. 17, no. 4, p. 818, 2017.

[13] Z. Cui, R. Ke, and Y. Wang, “Deep bidirectional and unidirectional LSTM recurrent neural network for network-wide traffic speed prediction,” *CoRR*, vol. abs/1801.02143, 2018. arXiv: `1801.02143`.

[14] B. Liao, J. Zhang, C. Wu, D. McIlwraith, T. Chen, S. Yang, Y. Guo, and F. Wu, “Deep sequence learning with auxiliary information for traffic prediction,” in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ser. KDD ’18, London, United Kingdom: Association for Computing Machinery, 2018, pp. 537–546, ISBN: 9781450355520.

[15] Z. Yuan, X. Zhou, and T. Yang, “Hetero-convlstm: A deep learning approach to traffic accident prediction on heterogeneous spatio-temporal data,” in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ser. KDD ’18, London, United Kingdom: Association for Computing Machinery, 2018, pp. 984–992, ISBN: 9781450355520.

[16] Y. Gu, W. Lu, X. Xu, L. Qin, Z. Shao, and H. Zhang, “An improved bayesian combination model for short-term traffic prediction with deep learning,” *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–11, 2019.

[17] Z. Pan, Y. Liang, W. Wang, Y. Yu, Y. Zheng, and J. Zhang, “Urban traffic prediction from spatio-temporal data using deep meta learning,” in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ser. KDD ’19, Anchorage, AK, USA: Association for Computing Machinery, 2019, pp. 1720–1730, ISBN: 9781450362016.

[18] C. Zheng, X. Fan, C. Wang, and J. Qi, *Gman: A graph multi-attention network for traffic prediction*, 2019. eprint: `arXiv:1911.08415`.

[19] L. Zhu, F. R. Yu, Y. Wang, B. Ning, and T. Tang, “Big data analytics in intelligent transportation systems: A survey,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 1, pp. 383–398, 2019.

[20] R. E. Wilson, “An analysis of gipps’s car-following model of highway traffic,” *IMA journal of applied mathematics*, vol. 66, no. 5, pp. 509–537, 2001.

[21] A. Zeroual, F. Harrou, Y. Sun, and N. Messai, “Monitoring road traffic congestion using a macroscopic traffic model and a statistical monitoring scheme,” *Sustainable cities and society*, vol. 35, pp. 494–510, 2017.

[22] A. Solé-Ribalta, S. Gómez, and A. Arenas, “A model to identify urban traffic congestion hotspots in complex networks,” *Royal Society open science*, vol. 3, no. 10, p. 160 098, 2016.

[23] A. G. Hawkes, “Spectra of some self-exciting and mutually exciting point processes,” *Biometrika*, vol. 58, no. 1, pp. 83–90, 1971.

[24] M. Gomez Rodriguez, J. Leskovec, and A. Krause, “Inferring networks of diffusion and influence,” in *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD ’10, Washington, DC, USA: Association for Computing Machinery, 2010, pp. 1019–1028, ISBN: 9781450300551.

[25] B. Yuan, H. Li, A. L. Bertozzi, P. J. Brantingham, and M. A. Porter, “Multivariate spatiotemporal hawkes processes and network reconstruction,” *SIAM Journal on Mathematics of Data Science*, vol. 1, no. 2, pp. 356–382, 2019. eprint: `https://doi.org/10.1137/18M1226993`.

[26] S. Zhu and Y. Xie, “Spatial-temporal-textual point processes with applications in crime linkage detection,” 2019. eprint: `arXiv:1902.00440`.

[27] T. Omi, n. ueda, and K. Aihara, “Fully neural network based model for general temporal point processes,” in *Advances in Neural Information Processing Systems 32*, Curran Associates, Inc., 2019, pp. 2120–2129.

[28] Q. Zhang, A. Lipani, O. Kirnap, and E. Yilmaz, *Self-attentive hawkes processes*, 2019. eprint: `arXiv:1907.07561`.

[29] F. P. Schoenberg, D. R. Brillinger, and P. Guttorp, “Point processes, spatial-temporal,” *Wiley StatsRef: Statistics Reference Online*, 2014.

[30] M.-T. Luong, H. Pham, and C. D. Manning, “Effective approaches to attention-based neural machine translation,” *arXiv preprint arXiv:1508.04025*, 2015.

[31] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[32] R. P. Barry, M Jay, and V. Hoef, "Blackbox kriging: Spatial prediction without specifying variogram models," *Journal of Agricultural, Biological, and Environmental Statistics*, pp. 297–322, 1996.

[33] J. M. Ver Hoef, E. Peterson, and D. Theobald, "Spatial statistical models that use flow and stream distance," *Environmental and Ecological statistics*, vol. 13, no. 4, pp. 449–464, 2006.

[34] V. Garreta, P. Monestiez, and J. M. Ver Hoef, "Spatial modelling and prediction on river networks: Up model, down model or hybrid?" *Environmetrics*, vol. 21, no. 5, pp. 439–456, 2010.

[35] E. E. Peterson, D. M. Theobald, and J. M. ver Hoef, "Geostatistical modelling on stream networks: Developing valid covariance matrices based on hydrologic distance and stream flow," *Freshwater biology*, vol. 52, no. 2, pp. 267–279, 2007.

[36] N. Cressie, J. Frey, B. Harch, and M. Smith, "Spatial prediction on a river network," *Journal of Agricultural, Biological, and Environmental Statistics*, vol. 11, no. 2, p. 127, 2006.

[37] J. Chen, S.-H. Kim, and Y. Xie, "An efficient score-statistic for spatio-temporal surveillance," *arXiv preprint arXiv:1706.05331*, 2017.

[38] J. M. Ver Hoef and E. E. Peterson, "A moving average approach for spatial statistical models of stream networks," *Journal of the American Statistical Association*, vol. 105, no. 489, pp. 6–18, 2010.

[39] N. Du, H. Dai, R. Trivedi, U. Upadhyay, M. Gomez-Rodriguez, and L. Song, "Recurrent marked temporal point processes: Embedding event history to vector," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 1555–1564.

[40] H. Mei and J. M. Eisner, "The neural hawkes process: A neurally self-modulating multivariate point process," in *Advances in Neural Information Processing Systems*, 2017, pp. 6754–6764.

[41] Q. Zhang, A. Lipani, O. Kirnap, and E. Yilmaz, "Self-attentive hawkes processes," *arXiv preprint arXiv:1907.07561*, 2019.

[42] A. G. Hawkes, "Spectra of some self-exciting and mutually exciting point processes," *Biometrika*, vol. 58, no. 1, pp. 83–90, 1971.

[43] E. Bacry, I. Mastromatteo, and J.-F. Muzy, “Hawkes processes in finance,” *Market Microstructure and Liquidity*, vol. 1, no. 01, p. 1 550 005, 2015.

[44] T. Ji, Z. Chen, N. Self, K. Fu, C.-T. Lu, and N. Ramakrishnan, “Patent citation dynamics modeling via multi-attention recurrent networks,” *arXiv preprint arXiv:1905.10022*, 2019.

[45] Q. Zhao, M. A. Erdogdu, H. Y. He, A. Rajaraman, and J. Leskovec, “Seismic: A self-exciting point process model for predicting tweet popularity,” in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015, pp. 1513–1522.

[46] D. C. Gazis and R. Herman, “The moving and “phantom” bottlenecks,” *Transportation Science*, vol. 26, no. 3, pp. 223–229, 1992.

[47] B. Nitz, *Atlanta breaks 135-year-old rainfall record – and more is on the way*, `https://www.ajc.com/news/local/atlanta-breaks-135-year-old-rainfall-record-and-more-the-way/ToXxI0475c7evyvMp2FrMP/`.

[48] G. D. of Transportation, *Traffic analysis and data application (tada)*, `http://www.dot.ga.gov/DS/Data`.

[49] S. Zhu and Y. Xie, “Crime event embedding with unsupervised feature selection,” in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 3922–3926.

[50] OpenStreetMap contributors, *Planet dump retrieved from https://planet.osm.org*, `https://www.openstreetmap.org`, 2017.