Spatio-Temporal Point Processes With Attention for Traffic Congestion Event Modeling

Shixiang Zhu, Ruyi Ding, Minghe Zhang, Pascal Van Hentenryck, and Yao Xie, *Member, IEEE*

***Abstract*— We present a novel framework for modeling traffic congestion events over road networks. Using multi-modal data by combining count data from traffic sensors with police reports that report traffic incidents, we aim to capture two types of triggering effect for congestion events. Current traffic congestion at one location may cause future congestion over the road network, and traffic incidents may cause spread traffic congestion. To model the non-homogeneous temporal dependence of the event on the past, we use a novel attention-based mechanism based on neural networks embedding for point processes. To incorporate the directional spatial dependence induced by the road network, we adapt the "tail-up" model from the context of spatial statistics to the traffic network setting. We demonstrate our approach's superior performance compared to the state-of-the-art methods for both synthetic and real data.**

***Index Terms*— Traffic congestion, attention, point process.**

I. Introduction

TRAFFIC congestion modeling is critical to modern transportation applications, such as route guidance or traffic network planning. For example, in Atlanta, which has over half a million daily commuters, reducing congestion is a top priority. The city spends millions of dollars on traffic-reducing measures, including toll lanes and high-capacity transport [1]. However, modeling the complex traffic dynamics and predicting traffic congestion events in real-time is vital but has remained extremely challenging. Indeed, traffic modeling is inherently intricate because of the complex spatio-temporal dynamics and the fact that congestion also stems from responses to real-time events such as traffic accidents. As a result, understanding and predicting congestion events can help cities cope with traffic more efficiently and plan future urban development.

Traffic sensors distributed over highway and road networks are widely deployed. They are key technology enablers that provide a unique opportunity to understand the traffic dynamic and congestion. Traffic sensor data reports traffic counts, i.e., the number of cars passing through per unit of time. These traffic counts are exploited by most existing works (reviewed in the section below), but traditional approaches do not model traffic events and incidents, which are fundamentally different in nature. An essential feature of traffic congestion modeling is the ability to capture *triggering effects*. For example, when congestion occurs, the effect will propagate, and subsequent congestion is typically more likely to happen along the affected road or highway. Moreover, other types of events with police intervention, such as response to traffic accidents, may also be related to traffic congestion. Such events logged into the police reports provide an additional data source useful for modeling and predicting traffic congestion.

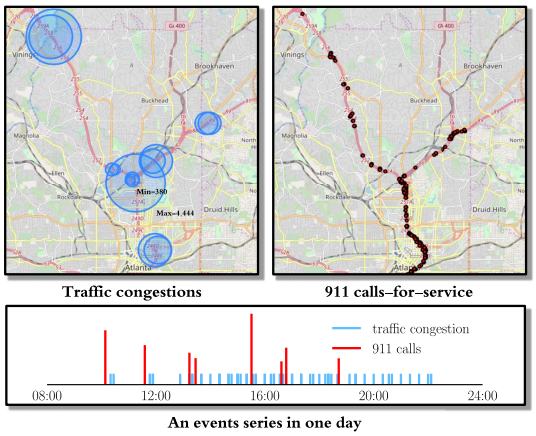


Fig. 1. An illustration of the Atlanta traffic dataset. The left panel shows the numbers of traffic congestions recorded by 14 traffic sensors. The size of the blue circle represents the number of traffic congestion events recorded by the traffic sensor. The right panel shows the spatial distribution of traffic incidents reported by 911 calls on the highway. Black dots represent the locations of traffic incidents. The panel in the bottom shows an event series in a single day. The height of the red bar indicates the length of the processing time.

In this paper, we aim to capture these two types of events and their triggering effects. Hawkes processes (also called self-exciting point processes) are a popular model for modeling such a triggering effect, and they have been successfully used in many different applications (see [2] for a review). A Hawkes process models the dependence between events using mutually dependent point processes, with intensities depending 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

Manuscript received 15 May 2020; revised 11 August 2020, 30 November 2020, and 17 January 2021; accepted 11 March 2021. Date of publication 30 March 2021; date of current version 8 July 2022. This work was supported in part by NSF CAREER Award under Grant CCF-1650913 and in part by NSF under Grant DMS-1938106 and Grant DMS-1830210. The Associate Editor for this article was Y.-S. Huang. *(Corresponding author: Yao Xie.)*

The authors are with Georgia Institute of Technology, 30332 Atlanta, Georgia (e-mail: yao.xie@isye.gatech.edu).

Digital Object Identifier 10.1109/TITS.2021.3068139

methods based on the popular Recurrent Neural Networks (RNNs) [3]–[9], which can capture some complex temporal dependence with particular assumptions, e.g., [3] assumes that the influence of an event decreases 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 major car accident occurs on the highway. The police are called to the scene, and they may need to wait for a specialized unit to move the wreckage. The process can take several hours during which the highway is blocked. In such a situation, the traffic event's influence may not decrease over a while and only start to relieve after the traffic blockage has been cleared. (2) We need to consider the special spatial correlation structure induced by road networks. 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. Existing Hawkes process models tend to discretize space and treat it as a multi-dimensional Hawkes process without considering such a special structure.

In this paper, we develop a novel spatio-temporal attention point processes (APP) model for traffic congestion events while capturing the relationship between exogenous and endogenous factors for traffic congestion. Specifically, we consider the police response to traffic incidents as *exogenous promotion* [10] on traffic congestion (since such events indicate traffic accidents) and consider the influence between congestion events as *endogenous self-excitation*. To model the dynamics of the endogenous self-excitation, we borrow the idea of the so-called *attention* mechanism [11], [12], which is originally proposed to capture the non-linear dependence between words in natural language processing. This approach allows us to 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 scores on the current event. Moreover, we also introduce a new element in the attention mechanism, the adaptive *score function* to measure the importance between past events and the current event, which is highly interpretable and adapts the conventional dot-product score [12] to measure non-homogeneous spatio-temporal distance in traffic networks. To tackle the directional spatial correlation induced by road networks, we also adopt the idea of the "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 efficiently implement our APP model's attention component, where only the most informative events in the past are retained for computation. We show that our proposed method outperforms the state-of-the-art in achieving both a higher likelihood function of the point process model and higher prediction accuracy, using real-data traffic and police data set from Atlanta, USA. The novelty of our paper can be highlighted as follows. Our APP model is the first attempt to combine traffic sensor count data with police reports to model traffic events, to the best of our knowledge. Our spatio-temporal model is tailored for our application by including a novel attention-based mechanism to capture the non-homogeneous spatio-temporal dependence of the historical events, and a "tail-up" model to capture the directional spatial dependence.

A. Related Work

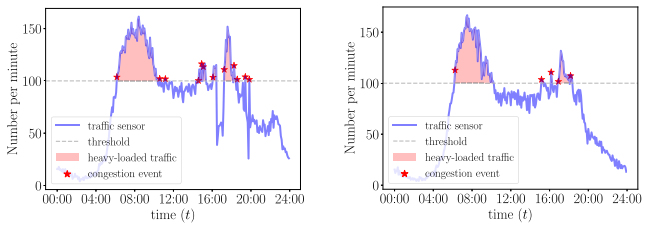
Earlier works [13]–[22] on traffic modeling focusing on predicting speed, volume, density, and travel time have achieved much success in this field. Other works [23]–[25] targeting at modeling traffic congestion based on the speed and density of vehicle stream have resulted in good mathematical descriptions for traffic flow. However, *traffic events modeling* is much less studied and in the nascent stage. Existing work in discrete event modeling using point processes have been used for modeling earthquake events, crime events, social networks, such as [26]–[29]. Such works often make strong assumptions and specify a parametric form of the intensity functions. Although they enjoy good interpretability and are efficient, parametric models may not be expressive enough to capture the event dynamics in some applications.

Recently there has been work on improving the expressive power of point process models. One approach is to develop RNNs based point process models [3], [4], [7], [9], which develop an RNN structure to mimic the conditional intensity function; however, the conditional intensity still discounts the influence of historical events. Another approach uses neural networks to directly model temporal dependence of sequence data. This includes [5], [8], which use RNNs as a generative model without specifying the conditional intensity function explicitly, and [30], which uses a neural network to parameterize the hazard function, and the conditional intensity can be derived by taking the derivative of the hazard function. However, the above approach focuses on temporal dependence rather than accounting for special structures in spatio-temporal dependence and is not directly applicable to our setting.

A recent work [31] also uses attention to model the historical information in point processes. However, they still assume the conditional intensity function follows a parametric exponential decaying model, which may not capture distant events, although they can be 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, [31] focuses on temporal point processes, while we consider spatio-temporal point processes; they use the dot-product score function to measure the similarity of two events, while we develop a more flexible score function based on neural networks. Another related work [21] models traffic counts using two individual attention structures to embed spatial and temporal information; however, traffic count modeling is completely different from traffic event modeling considered in our paper.

II. Datasets

In this section, we introduce the main datasets for our study, which consists of (1) a traffic congestion events dataset; (2) the 911 calls-for-service dataset; and (3) the Atlanta traffic network dataset.



(a) Wednesday, April 4th, 2018 (b) Thursday, April 5th, 2018

Fig. 2. Two examples of extracted traffic congestion recorded on the same traffic sensor labeled as "L1N" in Fig. 3 for two consecutive days. The rise of the traffic counts above the normal traffic level is followed by a drop below the level.

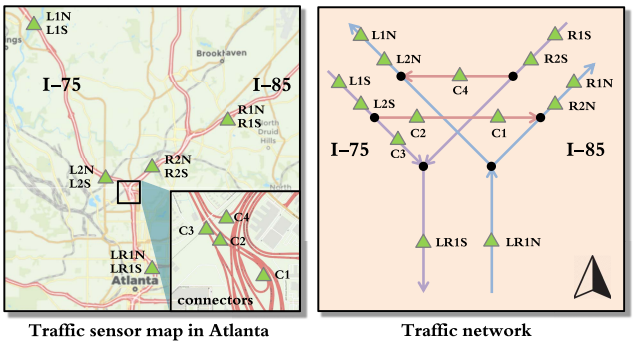


Fig. 3. The traffic network for major highways in Atlanta. The left panel shows the spatial distribution of traffic sensors, where green triangles represent locations of traffic sensors. Traffic sensors on the highway are bi-directional, i.e., two directions of the same location have separate traffic sensors to monitor the traffic condition. The right panel 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. Traffic Congestion

The traffic congestion dataset is collected from the Georgia Department of Transportation (GDOT) [32], which records the real-time traffic condition throughout the state of Georgia. These data are collected by traffic sensors installed on highways' main traffic points, which records the number of vehicles pass by the sensor every 5 minutes. The dataset also includes lane information (the number of lanes, the maximum speed of the lane, and the direction) at locations where the sensors are installed. The number of lanes can be used to estimate the maximum number of vehicles on the highway during that time. We assume the highway capacity – the maximum number of vehicles that a highway can support in normal conditions – is a linear function of the number of lanes.

As recognized in the literature [33], when traffic congestion initiates, the traffic count usually starts to increase and exceeds the normal volume. Then vehicles begin to slow down. As the congestion sets in, the throughput of the highway drops, and the traffic count falls below the normal level. We illustrate two examples in Fig. 2, which are extracted from the Atlanta traffic data. We find such patterns prevalent in traffic congestion events – the rise of the traffic counts above the normal traffic level is followed by a drop below the level. Thus, we detect a congestion event when the traffic counts exceed a threshold. Moreover, we found the traffic congestion events tend to cluster in time – in Fig. 2, the red stars indicate when the traffic counts exceed the threshold and the "stars" are clustered in times; this is a motivation for us to consider the self-exciting process model in the paper.

For our study, we consider $K = 14$ traffic sensors on two major highways in metro Atlanta, I-75 and I-85, shown in Fig. 3. We denote their geo-locations (latitude and longitude) on the traffic network by $r_k \in \mathcal{S} \subset \mathbb{R}^2$, $k = 1, \ldots, K$, where $\mathcal{S}$ denote the location space of traffic networks, which will be discussed in Section II-C. Let $\{x_i\}_{i=1}^{N_x}$ represent a sequence of traffic congestion events in a single day, where N_x is the number of the congestion events generated in one-day $[0, T)$. The i-th congestion event $x_i = (t_i, s_i)$ is a data tuple consisting of the occurrence time $t_i \in [0, T)$, and the sensor location index $s_i \in \{1, \ldots, K\}$. We extract 18,618 traffic congestion events from 174 days between April 2018 and December 2018. The maximum and the minimum number of events in a single day are 168 and 19, respectively.

B. 911 Calls-for-Service

As mentioned in Section I, traffic incidents may trigger unexpected congestion over traffic networks. Since the police respond to many traffic incidents, the 911 calls-to-service contains useful traffic incidents records. Thus, we use another data set, the 911 calls-for-service reports provided by the Atlanta Police Department (also used in [29], [34]) and extract all traffic incidents. When a 911 call reporting a traffic incident is received, the operator will start a record ID and dispatch police offices to the scene. The record includes the *call time* and occurrence location. The police officers arrive at the scene and start the investigation. Once the police complete the investigation and clean the scene, a *clear time* will be added to the record. The time interval between the call time and the clear time is called *processing time*. A long processing time is usually a strong indicator of a severe traffic incident that significantly impacts the highway traffic condition.

Let $\{y_j\}_{j=1}^{N_y}$ represent a sequence of traffic incidents extracted from 911 calls in a single day, where N_y is the total number of 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 $z_j > 0$ indicating the police processing time. We extract 19,805 such 911-call records on two major highways between April 2018 and December 2018 with processing time larger than 15 minutes. The incidents include many different categories ranging from speeding tickets to massive car pile-ups.

C. Traffic Network

Due to the nature of the traffic flow, there is a strong spatial dependence among the traffic data collected at different locations on the network. Moreover, the network topology and the direction of the flow will impact spatial correlations. For instance, there should not be a correlation for data collected at two locations that do not share a common traffic flow. In metro Atlanta, there are two major highways I-75 and I-85, passing through the city center. These two highways

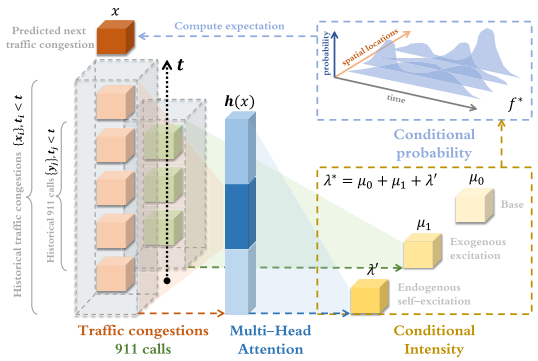


Fig. 4. The architecture for traffic congestion events modeling. The historical data, consisting of traffic congestion events and the 911 incidents, is fed into the multi-head attention mechanism to estimate the conditional intensity function for the point process model. The conditional intensity consists of three key components: (1) The background intensity μ_0 (which is estimated from the traffic data); (2) The exogenous promotion μ_1 (representing the influence of traffic events from police reports); (3) Endogenous self-excitation λ' (capturing the spatio-temporal dependence between congestions events).

start from the northwest and northeast side of the city, run due south, and intersect in Midtown, as shown in the left of Fig. 3. Between I-75 and I-85, two connectors bridge two highways via single-direction ramps. We extracted the network information of I-75, I-85 and their connectors in Atlanta from OpenStreetMap [35], 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 Fig. 3. Let $\mathscr{S} \subset \mathbb{R}^2$ represent the set of all geo-locations on the network. Suppose there are L road segments, and the set of locations on each segment is denoted as $\mathscr{S}_l \subset \{1, \ldots, L\}$, $l = 1, \ldots, L$. For any location $s \in \mathscr{S}$ on the network, we define the upstream portion $\vee_s \subseteq \{1, \ldots, L\}$ of the network to include s itself and all locations upstream from s. Define the downstream portion $\wedge_s \subseteq \mathscr{S}$ to include s itself and all locations downstream from s. For two locations $u, v \in \mathscr{S}$, 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 also 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 $u \nrightarrow v$.

III. Methodology

In this section, we propose an attention-based point process model for modeling traffic congestion events while consider the police 911-call data. The architecture of our model is shown in Fig. 4.

A. Spatio-Temporal Point Processes

The spatio-temporal point processes (STPPs) [2] consist of a sequence of events, represented by their times, locations, and marks. In our setting, the events are traffic congestion events. Let $\{x_1, x_2, \ldots, x_{N_x}\}$ represent a sequence of congestion events sampled from a STPP and $\{y_1, y_2, \ldots, y_{N_y}\}$ represent a sequence of 911-call incidents recorded in the same time period. Recall that N_x and N_y are the number of congestion events (extracted from traffic counts) and the number of traffic incidents (extracted from 911-calls), respectively, in the time horizon $[0, T)$. Let $\mathcal{H}_t = \mathscr{X}_t \cup \mathscr{Y}_t$ denote the σ-algebra generated by all historical events before time t, where $\mathscr{X}_t = \{x_i\}_{t_i < t}$ and $\mathscr{Y}_t = \{y_j\}_{t_j < t}$. The distribution in STPPs are characterized via a conditional intensity function $\lambda(t, k|\mathcal{H}_t)$, which is the conditional probability of observing an event at $(t, k) \in [0, T) \times \{1, \ldots, K\}$, given the events' history $\mathcal{H}_t$. Formally, $\lambda(t, k|\mathcal{H}_t) = \mathbb{E}\left[N_k([t, t + dt))|\mathcal{H}_t\right]/dt$, where $N_k(A)$ is the counting measure of events for sensor k over any set $A \subseteq [0, T)$.

To capture the dynamic between traffic congestion and traffic incidents, we consider the following form for the conditional intensity function in our model:

$$\lambda(t, k|\mathcal{H}_t) = \underbrace{\mu_0(t, k)}_{\text{background intensity}} + \underbrace{\mu_1(t, k|\mathscr{Y}_t)}_{\text{exogenous promotion}} + \underbrace{\lambda'(t, k|\mathscr{X}_t)}_{\text{endogenous self-excitation}}, \quad (1)$$

where $\mu_0(t, k)$ is the background intensity at (t, k), which can be estimated from data. The exogenous promotion $\mu_1(t, k|\mathscr{Y}_t)$ captures the influence of past 911-call incidents reported by the police before time t. The endogenous self-excitation $\lambda'(t, k|\mathscr{X}_t)$ captures the influence of past traffic congestion before time t. In the remainder of the section, we will discuss how to model these two types of triggering effects in Section III-B and Section III-C, respectively.

B. Police Intervention as Exogenous Promotion

As we discussed in Section II, police response to traffic incidents is often correlated with an increase in the strain on urban traffic. Such strain only spreads along the traffic direction and decays as the distance increases from where the traffic incident originated. Due to this consideration, the spatial correlation between two locations $u, v \in \mathscr{S}$ on the highway is determined by the traffic network structure and their "stream distance" $d(u, v)$, which is the distance to reach from u to v in the directed graph. Also, the spatial correlation may vary from time to time since the traffic intensity is always changing throughout the day. Here, we denote such spatial correlation between two arbitrary locations $u, v \in \mathscr{S}$ on the traffic network at time $t \in [0, T)$ as $\alpha_t(u, v) \in [0, 1]$. We will discuss the estimation of the spatial correlation in Section III-E.

Now we consider the traffic incidents extracted from the police reports, $y_j = (t_j, r_j, z_j)$, at (t, k) as an additive exogenous promotion when t is in the middle of process of the event y_j, i.e., $t \in [t_j, t_j + z_j)$. Formally, the exogenous promotion in (1) can be defined as

$$\mu_1(t, k|\mathscr{Y}_t) = \sum_{y_j \in \mathscr{Y}_t} \gamma\, \alpha_t(r_k, r_j) \cdot \delta_t\big([t_j, t_j + z_j)\big) \quad (2)$$

where $\delta_t(A)$ is the Dirac measure, i.e., taking the value 1 when $t \in A$ and 0 otherwise. The parameter $\gamma > 0$ captures the decay rate of the influence.

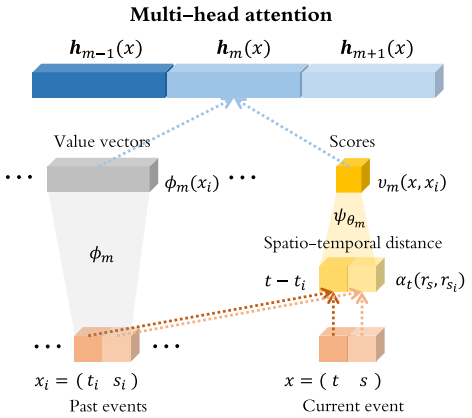


Fig. 5. The multi-head attention architecture.

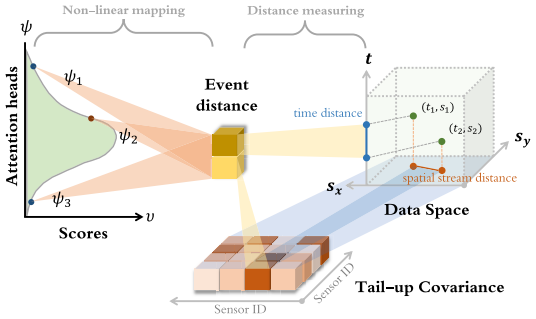


Fig. 6. The illustration of the score calculation. First, we measure the distance between two events, in which their spatial correlation is represented by our tail-up model; Then we calculate the score by transforming their spatio-temporal distance via a non-linear mapping. Note that we use different non-linear mapping for different attention head.

C. Attention-Based Self-Excitation Modeling

The idea of Attention Point Processes (APP) is to model the non-linear and long-term dependence of the current traffic congestion event from past congestion events using the attention mechanism [11], [12]. The attention model has recently become popular in natural language processing and computer vision to tackle non-linear and long-range dependencies. Still, it has been rarely used outside of these domains. We adopt the attention mechanism for point process models by redefining "attention" in our context and adopt the "multi-heads" mechanism [12] to represent more complex non-linear triggering effects. Specifically, we model the endogenous self-excitation $\lambda'(t, k|\mathscr{X}_t)$ in (1) using the output of the attention structure. The exact definition of the conditional intensity is as follows.

For the notational simplicity, let $x := (t, s)$ represent the data tuple of the current congestion event and $x' := (t', s') \in \mathscr{X}_t, t' < t$ represent the data tuple of another event in the past. As shown in Fig. 5, for the m-th attention head, we score the current congestion event against its past event, denoted as $\upsilon_m(x, x') \in \mathbb{R}^+$. For the event x, the score $\upsilon_m(x, x')$ determines how much *attention* to place on the past event x' as we encode the history information, which will be further discussed in Section III-D. The normalized score $\widetilde{\upsilon}_m(x, x') \in [0, 1]$ for events x and x' is obtained by employing the softmax function as:

$$\widetilde{\upsilon}_m(x, x') = \frac{\upsilon_m(x, x')}{\sum_{x_i \in \mathscr{X}_t} \upsilon_m(x, x_i)}, \tag{3}$$

The m-th *attention head* $\boldsymbol{h}_m(x) \in \mathbb{R}^p$ for the event x is defined as

$$\boldsymbol{h}_m(x) = \sum_{x_i \in \mathscr{X}_t} \widetilde{\upsilon}_m(x, x_i)\phi_m(x_i), \tag{4}$$

where the value embedding $\phi_m(x) = x^\intercal W_m^v$ and $W_m^v \in \mathbb{R}^{d \times p}$ is a weight matrix, d is the data dimension (here $d = 3$), and p is the dimension of value embedding. In comparison with the *self-attention* mechanism [12], the current event x and past event x_i are analogous to *query* and *key*, respectively, and the value embedding for the past event $\phi_m(x_i)$ is analogous to *value*. The multi-head attention $\boldsymbol{h}(x) \in \mathbb{R}^{Mp}$ for event x is the concatenation of M single attention heads:

$$\boldsymbol{h}(x) = \text{concat}\left(\boldsymbol{h}_1(x), \ldots, \boldsymbol{h}_M(x)\right).$$

Denote $\lambda'(t, k|\mathscr{X}_t)$ in (1) as $\lambda'(x|\mathscr{X}_t)$. The historical information in $\mathscr{X}_t$ can be modeled as a non-linear transformation of the multi-head attention $\boldsymbol{h}(x)$. Hence, the endogenous self-excitation λ' can be specified as:

$$\lambda'(x|\mathscr{X}_t) = \lambda'(x|\boldsymbol{h}(x)) = \text{softplus}\left(\boldsymbol{h}(x)^\intercal W + b\right), \tag{5}$$

where $W \in \mathbb{R}^{Lp}$ is the weight matrix and $b \in \mathbb{R}$ is the bias term. The function $\text{softplus}(x) = \log(1 + e^x) > 0$ is a smooth approximation of the ReLU function, which ensures that the intensity strictly is positive at all times.

D. Score Function

The score function quantifies how likely a past event triggers one event. The most commonly used score function in attention models is the dot-product score function. Typically, the score is obtained by computing the inner product between the query x and the key x' given two mapping matrices W, W' (the so-called key/query embedding), i.e., $\upsilon(x, x') = x^\intercal W^\intercal W' x'$. This can be viewed as their inner product in the embedding space. However, as discussed in Section III-B, for our setting, the correlation between two locations on the highway may not depend on their Euclidean distance, and this correlation may also vary over time. We adopt the spatial correlation $\alpha_t(r_{s_n}, r_{s_i})$ at time t between locations of two events $r_{s_n}, r_{s_i} \in \mathscr{S}$ rather than their Euclidean distance.

As illustrated in Fig. 6, the score function $\upsilon_m(x, x')$ for the m-th attention head can be expressed as:

$$\upsilon_m(x, x') = \psi_{\theta_m}\Big(t - t', \alpha_t(r_s, r_{s'})\Big), \tag{6}$$

where the $\psi_{\theta_m}(\cdot, \cdot) \in \mathbb{R}^+$ is a multi-layer neural network parameterized by a set of weights and biases denoted by θ_m. The neural network takes the spatial correlation $\alpha_t(r_s, r_{s'})$ and the temporal distance $t - t'$ as input and yields a non-negative score. The score function can be interpreted as a weighted spatio-temporal distance, where the weights of time and space

are learned from data. Note that the score function for each attention head may be different.

E. Tail-Up Spatial Model in Score Function

To capture the spatial correlation $\alpha_t(u, v)$ between two locations $u, v \in \mathscr{S}$ on the traffic network at time $t \in [0, T)$, we adopt the idea of tail-up spatial model, which is originally based on moving averages [36]–[39] and widely used in river systems modeling [40]–[42]. For the notational simplicity, we omit t in the following. There are three major advantages of tail-up models against other methods: (1) The tail-up models use stream distance rather than the Euclidean distance, which is defined as the shortest distance along with the traffic network between two locations. (2) The statistical independence is imposed on the observations located on stream segments that do not share the same traffic flow. (3) Proper weighting is incorporated on the covariance matrix entries when the road segments in the network are splitting into multiple segments to ensure that the resulting covariance is stationary. The tail-up models are appropriate when we want to enforce zero covariance when locations are flow-unconnected. This situation can occur when the flow dominates a variable (e.g., when traffic congestion enters a stream and can influence the upstream, it induces correlation in sensor measurements only when they are flow-connected).

Assume the traffic at location $u \in \mathscr{S}$ can be viewed as a white-noise random process Z_u, where observable locations on the traffic network can be developed by creating random variables as the integration of a moving average function over the process on the upstream of the network [41],

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

where μ_u is a deterministic mean process at location u, $\vee_u$ denotes the upstream of location u and $w(r) = w^l$ for all location $r \in \mathscr{S}_l$ on the segment l, which is the weight that satisfies the additivity constraint to ensure the stationarity in variance. The weights can be estimated using normalized average traffic counts for each segment on the traffic network. The moving average function $g(\cdot)$ is square-integrable and defined on $\mathbb{R}$. The $B(r), r \in \mathscr{S}$ is a Brownian process starting from sources of the traffic network, progressing toward the outlets, separating or merging themselves at traffic forks. The spatial correlation $\alpha(u, v)$ is obtained by $\text{cov}(Z_u, Z_v) = \mathbb{E}[Z_u Z_v] - \mathbb{E}[Z_u]\mathbb{E}[Z_v]$, i.e.,

$$\alpha(u, v) = \int_{\vee_u \cap \vee_v} g(r-u)g(r-v)\frac{w(r)}{\sqrt{w(u)w(v)}}dr.$$

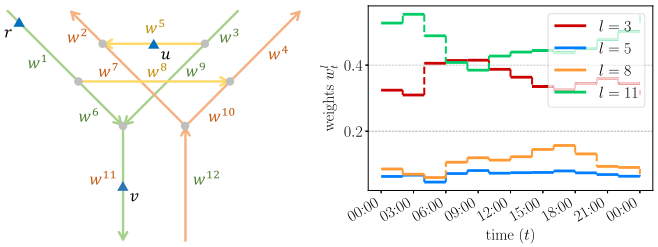
Let Δr be a stream distance on $\mathbb{R}^+$ and define $C(\Delta r) := \int_{\mathbb{R}} g(r)g(r-\Delta r)dr$. By choosing a particular moving average function, we can reparameterize $C(\cdot)$ in forms typically seen in the spatial statistical literature. We adopt the tail-up exponential model here [39], i.e.,

$$C(\Delta r) := \beta \exp(-\Delta r/\sigma),$$

where β, σ are parameters of the tail-up model. Let $d(u, v) \in \mathbb{R}^+$ be the stream distance between locations $u, v \in \mathscr{S}$ on the traffic network. The above covariance can be simplified as:

$$\alpha(u, v) = \begin{cases} C\left(d(u, v)\right)\sqrt{\dfrac{w(u)}{w(v)}}, & u \to v \\ 0, & u \not\to v. \end{cases} \tag{7}$$

Note that the tail-up model simplifies the parameterization of spatial correlation in the score function by replacing two complete weight matrices to be only with two trainable parameters β, σ.



(a) Weight example (b) Estimated weights w_t^l

Fig. 7. Weighting example for the tail-up model: arrows indicate flow directions of segments on the traffic network; gray dots represent confluences; w^l is the weight of the segment l; triangle u, v, and r are observable locations.

For example in Fig. 7, we first define all weights for each segment, i.e., $\{w^l\}_{l=1}^L$. The structure of the traffic network ensures that the sum of weights that flow into a confluence equals to the sum of weights that flow out of the same confluence (e.g. $w^1 = w^6 + w^7$ and $w^6 + w^9 = w^{11}$). Then we can obtain

$$\alpha(u, r) = C(d(u, v))\sqrt{w(u)/w(v)} = C(d(u, v))\sqrt{w^5/w^{11}},$$

and $\alpha(u, r) = 0$ because locations u and r are flow-unconnected.

In the experiment, we consider the tail-up model to be time-variant, i.e., for each time t, we can obtain a set of weights $\{w_t^l\}_{l \in \mathscr{L}}$ and spatial correlation $\alpha_t(u, v)$ by estimating the traffic counts at time t, since the distribution of the traffic flow may vary over time. For example, as shown in Fig. 7 (b), the time-variant weights of four traffic sensors are estimated based on the normalized traffic counts for every two hours.

F. Online Attention for Streaming Data

For streaming data, calculating attention may be computationally intractable since the past events may accumulate as time goes on. Here, we propose an adaptive online attention algorithm to address this issue. The main idea is only to keep a fixed number of "important" historical events with high average scores when calculating the attention head. Using both synthetic and real data, we will show that, in many cases, only a few events have a dominant influence on future events. The online APP model can be performed efficiently using a small part of historical data.

The procedure for collecting "important" events in each attention head is as follows. When the i-th event occurs, for a past event $x_j, t_j < t_i$ in the m-th attention head, we denote the set of its score against the future congestion

Algorithm 1 Event Selection for Online Attention

Input: data $\boldsymbol{x} = \{x_i\}_{i \geq 1}$ and threshold η.
Initialize $\mathscr{A}_{m,0} = \emptyset$.
for $i = 1$ **to** $+\infty$. **do**
  for $m = 1$ **to** M. **do**
    $\mathscr{A}_{m,i} \leftarrow \mathscr{A}_{m,i-1} \cup \{x_i\}$.
    Initialize $\mathscr{C}_{m,i} = \emptyset$.
    for $j = 1$ **to** $i - 1$ **do**
      $\mathscr{C}_{m,j} \leftarrow \mathscr{C}_{m,j} \cup \widetilde{v}_m(x_i, x_j)$.
      $\bar{v}_{m,j} \leftarrow (\sum_{v \in \mathscr{C}_{m,j}} v)/|\mathscr{C}_{m,j}|$.
    end for
    if $i > \eta$ **then**
      $\mathscr{A}_{m,i} \leftarrow \mathscr{A}_{m,i-1} \setminus \underset{j:t_j < t_i}{\arg\min} \{\bar{v}_{m,j}\}$.
    end if
  end for
end for

events $\{\widetilde{v}_m(x_i, x_j)\}_{i:t_j \leq t_i}$ as $\mathscr{C}_{m,j}$. Then the average score of the event x_j can be computed by $\bar{v}_{m,j} = (\sum_{v \in \mathscr{C}_{m,j}} v)/|\mathscr{C}_{m,j}|$, where $|\mathscr{C}|$ denotes the number of elements in a set $\mathscr{C}$. Hence, a recursive definition of the set $\mathscr{A}_{m,n}$ for selected events in the m-th attention head up until the occurrence of the event x_i is written as:

$$\mathscr{A}_{m,i} = \mathscr{X}_{t_{i+1}}, \quad \forall i \leq \eta,$$
$$\mathscr{A}_{m,i} = \mathscr{A}_{m,i-1} \cup \{x_i\} \setminus \underset{j:t_j < t_i}{\arg\min} \{\bar{v}_{m,j}\}, \quad \forall i > \eta,$$

where η is the maximum number of events we will retain. The complete event selection procedure is explained Algorithm 1. To perform the online attention, we substitute $\mathcal{H}_t$ in (3) and (4) with $\mathscr{A}_{m,i}$ for all attention heads, where i is the number of occurred events before time t.

G. Learning and Inference

Note that our model is jointly parameterized by a set of parameters $\{W, b, \gamma, \beta, \sigma, \{\theta_m, W_m^v\}_{m=1}^M\}$. We fit the model by the standard maximizing log-likelihood approach, which can be solved conveniently via the stochastic gradient descent. Equipped with the definition of conditional intensity in (1), we can write down the likelihood function of the data explicitly. Suppose there are n observed events before T, denoted as $\boldsymbol{x} = \{(t_i, s_i)\}_{i=1}^n$. Let $F^*(t,k) = \mathbb{P}\{t > t_i, k|\mathcal{H}_t\}$ be the conditional cumulative function at (t,k), where t_i is the occurrence time of the last event before time t. Let $f^*(t,k)$ be the corresponding conditional density probability. For simplicity, denote the conditional intensity function $\lambda(t,k|\mathcal{H}_t)$ as $\lambda^*(t,k)$. The conditional intensity function for an arbitrary sensor k is defined by $\lambda^*(t,k) = f^*(t,k)/(1 - F^*(t,k))$. From the definition above, we can show $\lambda^*(t,k) = -d\log(1 - F^*(t,k))/dt$ and hence, $\int_{t_n}^t \lambda^*(\tau,k)d\tau = -\log(1 - F^*(t,k))$, where $F^*(t,k) = 0$, since the $(n+1)$-th event does not exist at time t_n. Therefore, $F^*(t,k) = 1 - \exp\{-\int_{t_n}^t \lambda^*(\tau,k)d\tau\}$ and

$$f^*(t,k) = \lambda^*(t,k) \cdot \exp\{-\int_{t_n}^t \lambda^*(\tau,k)d\tau\}. \tag{8}$$

Then the log-likelihood of observing a sequence $\boldsymbol{x}$ can be written as:

$$\ell(\boldsymbol{x}) = \sum_{i=1}^n \log \lambda^*(t_i, s_i) - \sum_{k=1}^K \int_0^T \lambda^*(t,k)dt. \tag{9}$$

Note that we can further predict a future event $(\hat{t}_{n+1}, \hat{s}_{n+1})$ given the past observations $\{x_i\}_{i=1,\ldots,n}$ by using the conditional probability defined in (8) as follows:

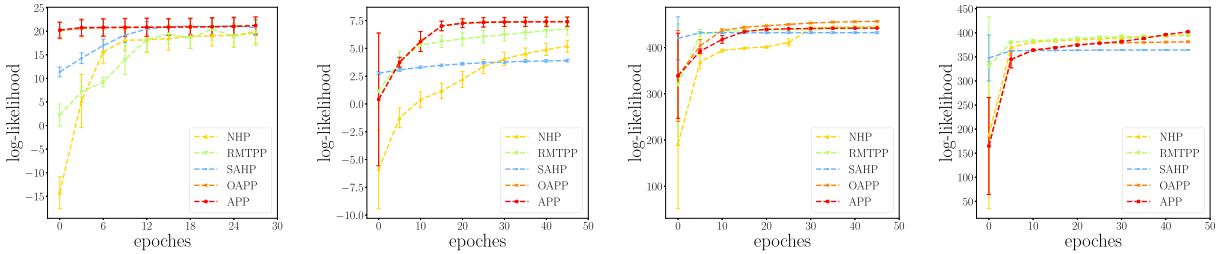
$$\begin{bmatrix} \hat{t}_{n+1} \\ \hat{s}_{n+1} \end{bmatrix} = \begin{bmatrix} \int_{t_n}^T \tau \sum_{k=1}^K f^*(\tau,k)d\tau \\ \underset{k}{\text{argmax}} \int_{t_n}^T f^*(\tau,k)d\tau \end{bmatrix}. \tag{10}$$

In general, the integration above cannot be obtained analytically. Therefore, we use standard numerical integration techniques to calculate the expectation.

IV. Experimental Results

In this section, we first conduct experiments on four synthetic datasets to illustrate our attention-based point process model's effectiveness. Then we test our model on the large-scale Atlanta traffic dataset. We evaluate our model with and without the online attention, and other alternative methods by comparing their log-likelihood and inspecting the recovery of corresponding conditional intensities in both temporal and spatial scenarios. Below, we refer to the attention-based point process model as `APP` and refer to its online version as `OAPP`. There are five alternative methods that we are considering in the following experiments:

(a) Long-Short Term Memory (`LSTM`) is a specialized recurrent neural network (RNN) commonly used for sequential data modeling. Here, we feed the event series into the LSTM and obtain the hidden state to summarize the historical information. Then the LSTM can generate the next event via an output layer given the last hidden state [43].
(b) Recurrent Marked Temporal Point Process (`RMTPP`) assumes the following form for the conditional intensity function λ^* in point processes, denoted as $\lambda^*(t) = \exp\left(\boldsymbol{v}^\top \boldsymbol{h}_i + \omega(t - t_i) + b\right)$, where the i-th hidden state in the RNN $\boldsymbol{h}_i$ is used to represent the influence of historical data up to the most recent event i, and $w(t - t_i)$ represents the current influence. The $\boldsymbol{v}, \omega, b$ are trainable parameters [3].
(c) Neural Hawkes Process (`NHP`) specifies the conditional intensity function in the temporal point processes using a continuous-time long-short term memory (LSTM), denoted as $\lambda^*(t) = f(\boldsymbol{\omega}^\top \boldsymbol{h}_i)$, where the hidden state of the LSTM up to time t_i that represents the influence of historical data, the $f(\cdot)$ is a softplus function which ensures the positive output given any input [4].
(d) Self-Attentive Hawkes Process (`SAHP`) adopts the self-attention mechanism to model the historical information in the conditional intensity function specified as $\lambda^*(t) = \text{softmax}\big(\mu + \alpha \exp\{\omega(t - t_j)\}\big)$, where μ, α, w are computed via three non-linear mappings: $\mu = \text{softplus}(\boldsymbol{h}W_\mu)$, $\alpha = \tanh(\boldsymbol{h}W_\alpha)$, $\omega = \text{softplus}(\boldsymbol{h}W_\omega)$. Here $W_\mu, W_\alpha, W_\omega$ are trainable parameters [31].



(a) Hawkes (b) self-correcting (c) non-homogeneous (d) non-homogeneous

Fig. 8. The average log-likelihood of synthetic datasets versus training epochs. For each synthetic dataset, we evaluate the performance of the five methods based on the maximum log-likelihood averaged per series calculated for the test data.

(e) Hawkes Process (HP) is one of the most common model for temporal events data. The conditional intensity function of the Hawkes process is given by $\lambda^*(t) = \mu + \alpha \sum_{t_j<t} \beta \exp\{-\beta(t-t_j)\}$, where parameters μ, α, β can be estimated via maximizing likelihood [26].

The experiment setup is as follows. A three-layer neural network in (6) and three attention heads in (5) are employed in our model. Note that using multiple layers in neural networks and multiple heads are critical to achieving our approach's good performance. Each dataset is divided into 80% training and 20% testing data. In particular, we use 50% events in the training data to initialize OAPP; this corresponds to $\eta = 0.5\, n$, n is the maximum length of sequences in the dataset. It takes 20 epochs for our model to converge (on a personal laptop, this takes about 12 minutes) for synthetic data. For real traffic data, these numbers increase to 50 epochs and about 45 minutes. To optimize our model, we employ Adam optimizer [44] with a learning rate $= 10^{-3}$ while the batch size is 64. The objective is to minimize the negative log-likelihood function derived in (9). Both our method and alternative approaches are implemented using Python and TensorFlow.

A. Synthetic Data

For the ease of presentation, we compare conditional intensities between approaches using time series data, where each data point is an one-dimensional timestamp. To adapt our APP model to time-only scenarios (in order to compare with the existing methods), we only consider the temporal distance in the score function of APP. i.e., substitute (3) with $v_m(x_n, x_i) = \psi_{\theta_m}(t_n - t_i)$.

The following experiments with synthetic data validate that our APP can capture the temporal pattern of synthetic data generated from conventional generative processes. To quantitatively evaluate each method's performance, we measure the average maximum likelihood on each synthetic dataset. Since the true intensities of these generating processes are known in the synthetic examples, we plot the conditional intensity over time (for one given series of events) to examine if the method can recover the true intensities.

The synthetic data are obtained by the following four generative processes: (1) *Hawkes process*: the conditional intensity function is given by $\lambda^*(t) = \mu + \alpha \sum_{t_j<t} \beta \exp -\beta((t-t_j))$, where $\mu = 10$, $\alpha = 1$, and $\beta = 1$; (2) *self-correction point process*: the conditional intensity function is given by $\lambda^*(t) = \exp(\mu t - \sum_{t_i<t} \alpha)$, where $\mu = 10$, $\alpha = 1$; (3) *non-homogeneous Poisson* 1: The intensity function is given by $\lambda^*(t) = c \cdot \Phi(t - 0.5) \cdot U[0, 1]$ where $c = 100$ is the sample size, the $\Phi(\cdot)$ is the PDF of standard normal distribution, and $U[a, b]$ is uniform distribution between a and b; (4) *non-homogeneous Poisson* 2: The intensity function is a composition of two normal functions, where $\lambda^*(t) = c_1 \cdot \Phi(6(t - 0.35)) \cdot U[0, 1] + c_2 \cdot \Phi(6(t - 0.75)) \cdot U[0, 1]$, where $c_1 = 50$, $c_2 = 50$. Each synthetic dataset contains 5,000 sequences with an average length of 30, where each data point in the sequence only contains the occurrence time of the event.

TABLE I
AVERAGE MAXIMUM LOG-LIKELIHOOD ON SYNTHETIC DATA

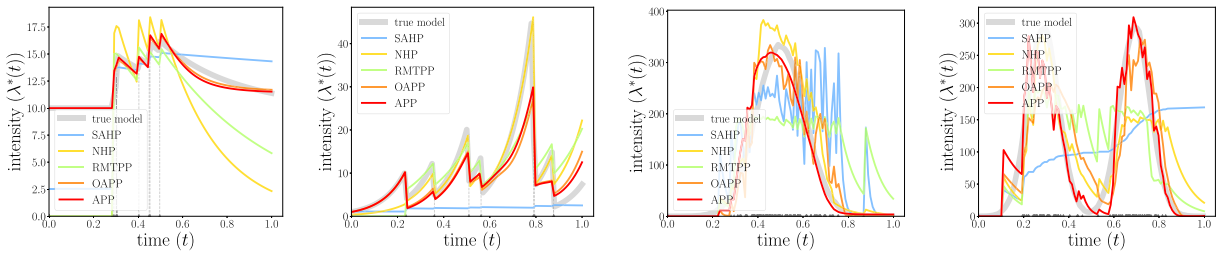
DATA SET	SAHP	NHP	RMTPP	APP	OAPP
HAWKES	20.8	20.0	19.7	**21.2**	21.1
SELF-CORRECTION	3.5	5.4	6.9	7.1	**7.1**
NON-HOMO 1	432.4	445.6	443.1	442.3	**457.0**
NON-HOMO 2	364.3	410.1	405.1	**428.3**	420.1

Fig. 8 summarizes the log-likelihood value of each model versus the training epochs, where each epoch includes 125 batches, and each batch randomly takes 40 sequences as training data. A higher log-likelihood value indicates a better performance of the model. As Fig. 8 and Table I show, our APP outperforms other four baseline methods on all four synthetic datasets by converging to larger log-likelihood values. Besides, our OAPP also shows competitive performances, when only 50% of events are used in online attention calculation.

Fig. 9 shows the estimated intensities using different methods in contrast to the true latent intensities indicated by the gray lines. We compare the predictive performance of the proposed model fitted to three types of time series models. Our APP can better capture the true conditional intensity function for all four synthetic datasets compared to the other four baseline methods.

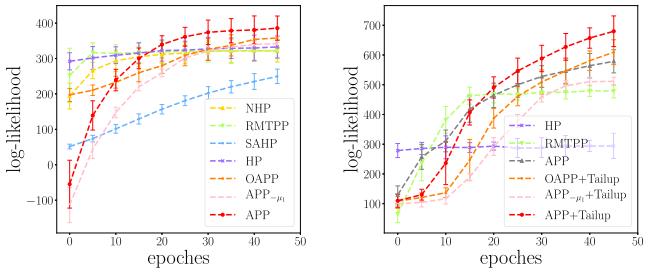
B. Traffic Data

This section further evaluates our model on the real Atlanta traffic dataset. As a sanity check, we also add two more



(a) Hawkes (b) self-correcting (c) non-homogeneous (d) non-homogeneous

Fig. 9. The conditional intensity function estimated from synthetic datasets. Triangles at the bottom of each panel represent events. The ground truth of conditional intensities is indicated by the grayline.



(a) time only (b) space and time

Fig. 10. Log-likelihood of Atlanta traffic data versus training epochs with and without considering spatial information.

TABLE II

MAXIMUM LOG-LIKELIHOOD, LOCATION-PREDICTION ACCURACY, AND TIME-PREDICTION ERROR FOR ATLANTA TRAFFIC DATASET

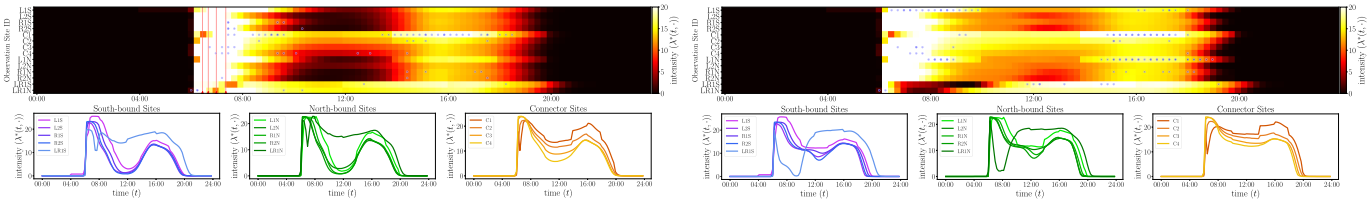
MODELS	max ℓ (TIME ONLY)	max ℓ (TIME & SPACE)	LOCATION ACCURACY	TIME MAE
LSTM	N/A	N/A	18.5%	44.0
HP	339.9	307.5	8.82%	39.1
RMTPP	339.2	490.1	22.0%	27.6
NHP	324.4	N/A	N/A	24.1
SAHP	326.7	N/A	N/A	49.4
APP$_{-\mu_1}$+TAILUP	378.4	512.9	28.8%	17.6
APP + EUCLIDEAN	392.3	570.7	30.9%	5.5
APP + TAILUP	**458.5**	**636.2**	**37.6%**	**3.7**
OAPP + TAILUP	437.5	615.9	36.9%	3.7

baselines in the comparison: (i) replacing the tail-up spatial correlation defined in (7) with the Euclidean distance (`APP+Euclidean`), (ii) removing the exogenous promotion μ_1 defined in (1) (`APP`$_{-\mu_1}$`+Tailup`). Below, we refer to our proposed approach as `APP+Tailup`).

As shown in Fig. 10, we report the average log-likelihood per sequence for each method over training epochs. The results show that our method outperforms the competitors by attaining the maximum log-likelihood for both time-only and space-time traffic datasets when the algorithm convergence. Apart from reporting the maximum log-likelihood of each method, in Table II, we also evaluate the average accuracy for predicting the location of the next event and the mean absolute error (MSE) for predicting the time of the next event, respectively. We use equation (10) for making the prediction and deem it correct if the prediction location is the same as the actual location of the event (recall that the event location is discrete). Then the prediction accuracy is the number of correctly predicted events over the total number of events. We remark that the prediction accuracy, if using random guess, is 7.1% since there are 14 sensors. Similarly, given observed past events, we can use equation (10) to predict the next event's time. The result shows that our method obtains the highest location accuracy and the lowest time error comparing to other baselines. The experiments also validate that considering the tail-up spatial correlation and the police intervention in our model effectively improves the performance.

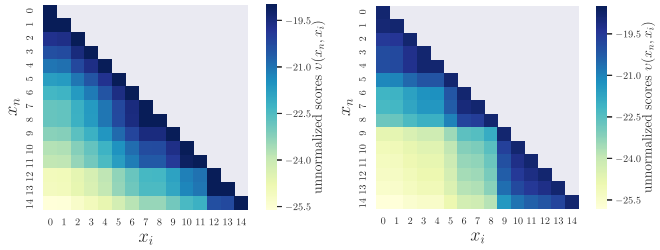
Moreover, we examine each traffic sensor's conditional intensity computed by the fitted APP on the traffic dataset. As discussed in Section I, there are 14 traffic sensors installed along two major highways (I-75 and I-85) in Atlanta, as shown in the left of Fig. 3. For better presenting the spatial pattern captured by the model, we select two typical days in Atlanta, which are May 8th, 2018, and April 24th, 2018, respectively. Fig. 11 (a) shows a clear temporal pattern on a regular weekday (Tuesday, May 8th, 2018), where the intensity of each sensor reaches its pinnacle in both morning (around 7:00) and evening (around 16:00) rush hours; Fig. 11 (b) shows the intensities on another weekday (Tuesday, April 24th, 2018). On this day, Atlanta broke a 135-year-old rainfall record when it got 4.16 inches of rain [45]. The previous record, set in 1883, was 2.4 inches. As we can see from the figure, the heavy rain and subsequent flood in the city led to an unusual traffic congestion level. Different from the results shown in Fig. 11 (a), the traffic congestion level remains at a relatively high level throughout the entire day.

We categorize the traffic sensors into three sub-groups based on their locations and then plot their conditional intensities individually. Note that similar temporal patterns can be found among the traffic sensors in the same sub-group. This can be explained since these sensors are in the same traffic direction and sharing the same traffic flow. The delay can be explained by a so-called "phantom traffic jam" phenomenon [46]. This situation usually begins when a part of traffic flow slows down slightly, causing the flow behind that part to slow down even more. The effect then spreads out through the traffic network and becomes worse further up the stream. For example, as the sensor *L1S*, *L2S*, *LR1S* show (which are along the southbound of I-75), the peak intensity of the first sensor has an about half an hour delay against the following sensors. Similar



(a) Tuesday, May 8th, 2018 (b) Tuesday, April 24th, 2018

Fig. 11. (a) The heatmap shows the conditional intensities of 14 traffic sensors in a single day. Each row represents a traffic sensor (associated with a unique site ID), each column represents a particular time slot, blue dots correspond to the occurrence of events, and red vertical lines to traffic incidents extracted from 911 calls. The color depth in the heatmap represents the level of intensity. (b) We categorize the conditional intensity into three subplots; the three plots from left to right represent the intensity of five sites on northbound highways, five sites on southbound highways, and four sites on connectors, respectively.



(a) synthetic Hawkes (b) real traffic

Fig. 12. Visualization of scores (discussed in Section III-D) between pairs of events in the sequence for (a) synthetic data generated from a Hawkes process model and (b) traffic data collected from the GDOT. Here x_n represents the current event and x_i represents past events, where $t_i < t_n$. The color of the entry at the n-th row and the i-th column represents the score $v(x_n, x_i)$ learned from data using our DAPP model.

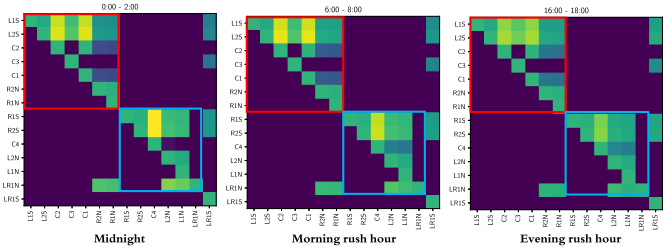


Fig. 13. Comparison of the spatial correlation between 14 traffic sensors generated by the tail-up model. Each row or column corresponds to one specific traffic sensor. Brighter color indicates a higher correlation. The red and blue boxes include the correlations between traffic sensors located in southbound and northbound of highways, respectively.

phenomenons can also be found among the sensor *L1N*, *L2N*, *LR1N*.

C. Score Function Interpretation

We now interpret the score function by visualizing the scores of a sequence of 15 congestion events extracted from real data. As shown in Fig. 12, each entry of the heatmap shows the score for one of the event x_n against its historical events $\{x_i\}_{i=1}^{n-1}$. The entry at the n-th row and the i-th column of the lower triangular matrix represents scores of the event x_n against its past event x_i, i.e., $v(x_n, x_i)$.

As demonstrated in Fig. 12, our APP can capture complex dependence between events accurately. In particular, Fig. 12 (a) shows the scores of events generated from a Hawkes process defined in Section IV-A. Note that the scores for events against their past resemble an exponential decay, reflecting the reality (in this case, the kernel function is exponentially decaying.) We also conduct a similar experiment on the traffic dataset, as shown in Fig. 12 (b). Note that there is a "network community" structure in that the first nine events pose a much weaker impact on their subsequent events than others, i.e., the first nine scores in the last row are remarkably lower than the other scores. By investigating the data further, we realize that the traffic sensor observes the first nine events on the highway northbound. In contrast, all the other events are observed by sensors installed in the opposite direction. These two sets of traffic sensors are not flow-connected, which explains the score matrix we learned in Fig. 12 (b).

D. Tail-Up Spatial Correlation Interpretation

The spatial correlation between 14 traffic sensors learned by our tail-up model is highly interpretable. As shown in Fig. 13, we visualize three covariance matrices of 14 traffic sensors corresponding to three different periods: the morning rush hour, the evening rush hour, and the midnight. The spatial structure of these covariance matrices reveals that: (1) There are two "network communities" among two sets of traffic sensors, which correspond to the sensors along the southbound and northbound highways. (2) The covariance between the highway connectors (*C1*-*C4*) and the southbound of highways (*L1S*, *L2S*, *R1S*, *R2S*) are different depending on whether it is rush hour or not. These observations confirm the idea that the triggering effect between congestion events can be spatially structured and directional. The highway connectors, in particular, play a vital role in defining such a structure.

V. Conclusion

We developed a novel attention-based point process model for modeling the dynamics of traffic congestion events considering the influence of traffic incidents. In building the model, we combine two data sources: the traffic sensor counts and the police reports (which contain comprehensive records of traffic incidents since police typically are responding to traffic incidents as part of their duty). Our main goal is to model traffic congestion events' self and mutual triggering effect while considering the traffic networks' structures. We adopt the state-of-the-art spatio-temporal point process model, together with the attention models, to achieve

our goals. As demonstrated by our experiments, our method achieves a good performance in achieving a higher likelihood of a point process and a higher prediction accuracy compared with previous approaches. By testing on various kinds of point process models, we demonstrate the flexibility of our proposed method.

REFERENCES

[1] D. Wickert. *Is There a fix for I-85 Traffic?*. (Feb. 2020). [Online]. Available: https://www.ajc.com/news/transportation/there-fix-for-traffic/FBjOFLgKW7kGphc3itpvnJ/

[2] A. Reinhart, "A review of self-exciting spatio-temporal point processes and their applications," *Stat. Sci.*, vol. 33, no. 3, pp. 299–318, Aug. 2018.

[3] 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 *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, New York, NY, USA, Aug. 2016, pp. 1555–1564.

[4] H. Mei and J. M. Eisner, "The neural Hawkes process: A neurally self-modulating multivariate point process," in *Advances in Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates, 2017, pp. 6754–6764.

[5] S. Li, S. Xiao, S. Zhu, N. Du, Y. Xie, and L. Song, "Learning temporal point processes via reinforcement learning," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst.* Red Hook, NY, USA: Curran Associates, 2018, pp. 10804–10814.

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

[7] S. Xiao, J. Yan, X. Yang, H. Zha, and S. M. Chu, "Modeling the intensity function of point process via recurrent neural networks," in *Proc. 31st AAAI Conf. Artif. Intell.* Palo Alto, CA, USA: AAAI Press, 2017, pp. 1597–1603.

[8] S. Xiao, M. Farajtabar, X. Ye, J. Yan, L. Song, and H. Zha, "Wasserstein learning of deep generative point process models," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.* Red Hook, NY, USA: Curran Associates, 2017, pp. 3250–3259.

[9] S. Zhu, H. S. Yuchi, and Y. Xie, "Adversarial anomaly detection for marked spatio-temporal streaming data," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 8921–8925.

[10] M.-A. Rizoiu, L. Xie, S. Sanner, M. Cebrian, H. Yu, and P. Van Hentenryck, "Expecting to be hip: Hawkes intensity processes for social media popularity," in *Proc. 26th Int. Conf. World Wide Web (WWW)*, Apr. 2017, pp. 735–744.

[11] T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Lisbon, Portugal, 2015, pp. 1412–1421.

[12] A. Vaswani *et al.*, "Attention is all you need," in *Advances in Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates, 2017, pp. 5998–6008.

[13] A. Abadi, T. Rajabioun, and P. A. Ioannou, "Traffic flow prediction for road transportation networks with limited traffic data," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 653–662, Apr. 2015.

[14] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: A deep learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 865–873, Apr. 2015.

[15] 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, Apr. 2017.

[16] Z. Cui, R. Ke, Z. Pu, and Y. Wang, "Deep bidirectional and unidirectional LSTM recurrent neural network for network-wide traffic speed prediction," 2018, *arXiv:1801.02143*. [Online]. Available: https://arxiv.org/abs/1801.02143

[17] B. Liao *et al.*, "Deep sequence learning with auxiliary information for traffic prediction," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, New York, NY, USA, Jul. 2018, pp. 537–546.

[18] Z. Yuan, X. Zhou, and T. Yang, "Hetero-ConvLSTM: A deep learning approach to traffic accident prediction on heterogeneous spatio-temporal data," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, New York, NY, USA, Jul. 2018, pp. 984–992.

[19] 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 Trans. Intell. Transp. Syst.*, vol. 21, no. 3, pp. 1332–1342, Mar. 2020.

[20] 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 *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, New York, NY, USA, Jul. 2019, pp. 1720–1730.

[21] C. Zheng, X. Fan, C. Wang, and J. Qi, "GMAN: A graph multi-attention network for traffic prediction," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 1, Apr. 2020, pp. 1234–1241.

[22] L. Zhu, F. R. Yu, Y. Wang, B. Ning, and T. Tang, "Big data analytics in intelligent transportation systems: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 1, pp. 383–398, Jan. 2019.

[23] R. E. Wilson, "An analysis of Gipps's car-following model of highway traffic," *IMA J. Appl. Math.*, vol. 66, no. 5, pp. 509–537, Oct. 2001.

[24] A. Zeroual, F. Harrou, Y. Sun, and N. Messai, "Monitoring road traffic congestion using a macroscopic traffic model and a statistical monitoring scheme," *Sustain. Cities Soc.*, vol. 35, pp. 494–510, Nov. 2017.

[25] A. Solé-Ribalta, S. Gómez, and A. Arenas, "A model to identify urban traffic congestion hotspots in complex networks," *Roy. Soc. Open Sci.*, vol. 3, no. 10, Oct. 2016, Art. no. 160098.

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

[27] M. G. Rodriguez, J. Leskovec, and A. Krause, "Inferring networks of diffusion and influence," in *Proc. 16th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, New York, NY, USA, Feb. 2010, pp. 1019–1028.

[28] B. Yuan, H. Li, A. L. Bertozzi, P. J. Brantingham, and M. A. Porter, "Multivariate spatiotemporal Hawkes processes and network reconstruction," *SIAM J. Math. Data Sci.*, vol. 1, no. 2, pp. 356–382, Jan. 2019.

[29] S. Zhu and Y. Xie, "Spatial-temporal-textual point processes with applications in crime linkage detection," 2019, *arXiv:1902.00440*. [Online]. Available: https://arxiv.org/abs/1902.00440

[30] T. Omi, N. Ueda, and K. Aihara, "Fully neural network based model for general temporal point processes," in *Advances in Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates, 2019, pp. 2120–2129.

[31] Q. Zhang, A. Lipani, O. Kirnap, and E. Yilmaz, "Self-attentive Hawkes process," in *Proc. 37th Int. Conf. Mach. Learn.*, vol. 119, Jul. 2020, pp. 11183–11193.

[32] Georgia Department of Transportation. *Traffic Analysis and Data Application (TADA)*. Accessed: Feb. 2020. [Online]. Available: https://www.dot.ga.gov/DS/Data

[33] M. Treiber and A. Kesting, "Traffic flow dynamics," *Flow Dynamics: Data, Models and Simulation*. Berlin, Germany: Springer-Verlag, 2013.

[34] S. Zhu and Y. Xie, "Crime event embedding with unsupervised feature selection," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 3922–3926.

[35] OpenStreetMap contributors. (2017). *Openstreetmap*. [Online]. Available: https://www.openstreetmap.org

[36] R. P. Barry and J. M. V. Hoef, "Blackbox Kriging: Spatial prediction without specifying variogram models," *J. Agricult., Biol., Environ. Statist.*, vol. 1, no. 3, pp. 297–322, 1996.

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

[38] J. M. V. Hoef, E. Peterson, and D. Theobald, "Spatial statistical models that use flow and stream distance," *Environ. Ecol. Statist.*, vol. 13, no. 4, pp. 449–464, Dec. 2006.

[39] J. M. V. Hoef and E. E. Peterson, "A moving average approach for spatial statistical models of stream networks," *J. Amer. Stat. Assoc.*, vol. 105, no. 489, pp. 6–18, Mar. 2010.

[40] J. Chen, S.-H. Kim, and Y. Xie, "S^3t: An efficient score-statistic for spatio-temporal surveillance," *Forthcoming Sequential Anal., Des. Methods Appl.*, vol. 39, no. 4, pp. 563–592, Jan. 2021.

[41] N. Cressie, J. Frey, B. Harch, and M. Smith, "Spatial prediction on a river network," *J. Agricult., Biol., Environ. Statist.*, vol. 11, no. 2, pp. 127–150, Jun. 2006.

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

[43] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[44] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2015. [Online]. Available: http://arxiv.org/abs/1412.6980
[45] B. Nitz. (Apr. 2018). *Atlanta Breaks 135-Year-old Rainfall Record*. [Online]. Available: https://www.wsbtv.com/news/local/wet-roads-could-make-for-messy-commute/737464010/
[46] D. C. Gazis and R. Herman, "The moving and 'phantom' bottlenecks," *Transp. Sci.*, vol. 26, no. 3, pp. 223–229, Aug. 1992.

Shixiang Zhu received the B.S. and M.S. degrees in computer science from the Beijing University of Posts and Telecommunications, in 2014 and 2017, respectively. He is currently pursuing the Ph.D. degree in machine learning with the H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology. His research interests include machine learning in spatio-temporal analysis, optimization in urban planning, and related applications.

Ruyi Ding received the B.S. degree in information engineering from Zhejiang University, China, in 2018, and the M.S. degree in electrical and computer engineering from the Georgia Institute of Technology in 2020. His research focuses on hardware-oriented security and side-channel analysis on deep neural networks.

Minghe Zhang received the B.S. degree in electronic engineering from Tsinghua University in 2017 and the M.S. dergee in electrical and computer engineering in 2019. He is currently pursuing the Ph.D. degree in machine learning with the H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology. His research interests include machine learning, network change detection, and online learning.

Pascal Van Hentenryck is currently the A. Russell Chandler III Chair and a Professor of industrial and systems engineering with Georgia Tech. He is credited with pioneering advances in constraint programming and stochastic optimization, bridging theory, and practice to solve real-world problems across a range of domains, including sports scheduling, protein folding, kidney matching, disaster relief, power systems, recommender systems, and transportation. He has developed several optimization technologies, including CHIP, Numerica, the optimization programming language, and comet. He has also published several books, including *Online Stochastic Combinatorial Optimization*, *Hybrid Optimization*, and *Constraint-Based Local Search*.

Yao Xie (Member, IEEE) received the Ph.D. degree in electrical engineering from Stanford University, with a focus on mathematics. She is currently an Associate Professor and the Harold R. and Mary Anne Nash Early Career Professor with the H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology and also an Associate Director of the Machine Learning Center. Her research areas are statistics, sequential analysis and sequential change-point detection, machine learning, and signal processing. She received the National Science Foundation (NSF) CAREER Award in 2017. She is also an Associate Editor for IEEE TRANSACTIONS ON SIGNAL PROCESSING.