Privacy preserving, crowd sourced crime Hawkes processes

George Mohler
IUPUI
gmohler@iupui.edu

P Jeffrey Brantingham
UCLA
pjb@anthro.ucla.edu

***Abstract*—Social sensing plays an important role in crime analytics and predictive policing. When humans play the role of sensor, several issues around privacy and trust emerge that must be carefully handled. We provide a framework for deploying predictive crime models based upon crowd-sourced information (crime reports, tips, Nextdoor posts, etc.) while protecting individual privacy and striving for a high level of algorithmic transparency. For this purpose we introduce a novel online Hawkes process estimation algorithm requiring no event history coupled with an online k-means type algorithm based upon the word movers distance. We illustrate the methodology using synthetic data, crime report data from Los Angeles, and public safety posts from Nextdoor in Indianapolis. In particular, we show that privacy and transparency can be maintained without sacrificing accuracy in space-time models of criminal incidents. Furthermore, our methodology provides a framework for sharing of information between private companies collecting crime tips or public safety information, law enforcement agencies, and the general public.**

***Index Terms*—Social Sensing, Predictive Policing, Online Hawkes Process, Topic Model, Word Movers Distance, Privacy**

I. Introduction

Social sensing plays an important role in crime analytics and predictive policing. For example, geolocated tweets are predictive of future hit-and-run incidents [1], mobile phone location data improves predictive models of crime hotspots [2], and neural networks trained on Google street view images can rank neighborhood crime levels based on the street level image better than humans [3].

While these secondary sources (e.g. social media, IOT sensors, etc.) may help to improve models of crime, human reporting remains the best source of information on crime and social disorder. Crime forecasts based upon calls-for-service alone were top performing solutions in the 2017 NIJ crime forecasting competition [4] and verified crime reports are the main input to a majority of crime hotspot models used for directing police patrols and interventions [5] [6]. Crime tips are a valuable tool that police departments employ [7] and more recently social networking platforms such as Nextdoor have started to provide mechanisms for community members to share information with law enforcement and each other.

When humans play the role of sensor, several issues around privacy and trust emerge that must be carefully handled. Communities have varying levels of trust in law enforcement and under-reporting of crime can occur if community members do not trust police to respond fairly and effectively. While a small percentage of crime incident data is generated due to police observation and arrests, the majority of data is collected from citizen initiated reports where the reporter is a victim or witness to a crime [8]. Mechanisms are put in place to encourage reporting while protecting reporter privacy. For example, crime tips are almost always anonymized and are often deleted from databases after several weeks or months. These issues are now arising in the private domain as well, for example Nextdoor will forward public safety posts to law enforcement only with the user's permission and removes social interactions and posts related to the original post.

The goal of this paper is to provide a framework for deploying predictive crime models based upon crowd-sourced information (crime reports, tips, Nextdoor posts, etc.) while protecting individual privacy and striving for a high level of algorithmic transparency. The latter condition is in response to recent criticisms of algorithmic bias in predictive policing models [9] and attempts to make predictive policing source code open [10]. We provide a schematic for a privacy preserving, crowd sourced crime model in Figure 1 similar to the privacy preserving framework outlined in [11]. A real-time crime model is maintained on a server, consisting of the current spatial risk of each of several crime "topics" across the city along with model parameters. When a user wishes to make a crime report, the client side application checks out the model from the server, the user's post is categorized on the client side, and the risk and model parameters are updated and pushed back to the server. The risk model can also be checked out by law enforcement for preventative patrols. Because the model is updated on the client side, no direct user information is pushed to the server and minimal user information (save for the neighborhood and topic) may be inferred from the model server.

The outline of the paper is as follows. In Section II, we provide the details of our algorithm. In Section IIa, we describe an online learning framework for space-time Hawkes processes where no event history is required for updating the intensity or its parameters. In Section IIb, we describe an online k-means type algorithm utilizing the word movers distance and word2vec to map crime report text to a topic. To our knowledge this is the first online Hawkes process estimation algorithm requiring no event history and the first online algorithm for topic modeling with word movers distance. In Section III, we exhibit several experiments using synthetic

0-7695-6411-9/18/$31.00 ©2018 IEEE
DOI 10.1109/SocialSens.2018.00016

data, crime report data from Los Angeles, and public safety posts from Nextdoor. In particular, we show that privacy and transparency can be maintained without sacrificing accuracy in space-time models of criminal incidents. Furthermore, our methodology provides a framework for sharing of information between private companies collecting crime tips or public safety information, law enforcement agencies, and the general public. We end with a discussion of our findings and suggestions for future directions in Section IV.

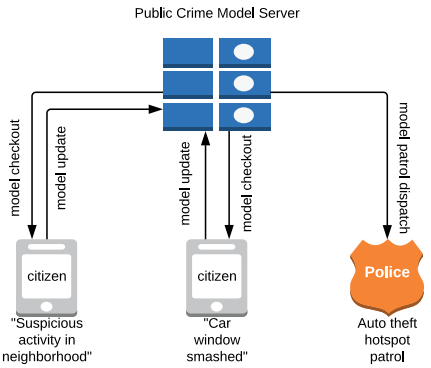

Fig. 1: Schematic for a crowd-sourced crime model with privacy.

II. Privacy preserving, crowd sourced Hawkes process

A. Online learning of Hawkes processes without histories

We consider a Hawkes process defined on a spatial discretization G with spatial units $g \in G$,

$$\lambda_g(t) = \mu_g + \sum_{\substack{t>t_j \\ g_j=g}} f(t-t_j). \tag{1}$$

In Equation 1, $\lambda_g(t)$ is the intensity (rate) of events in spatial unit g. The overall intensity is the superposition of a "background" rate μ_g, a Poisson process modeling spontaneous events, and a "triggering" kernel $f(t)$ that allows the overall intensity to increase following recent events in the history of the process in spatial unit g. The event times t_j comprise the history of the process, where g_j is the grid cell in which event j occurs.

For now we consider a single Hawkes process, but in subsequent sections we will allow for multiple independent Hawkes processes with intensities $\lambda_g^l(t)$ corresponding to different topics l. The Hawkes process is used to model crime patterns such as burglary [12] and retaliatory violence [13] where offenders repeat criminal activity shortly following recent crimes.

Model parameters can be learned from data by maximizing the log-likelihood function,

$$L = \sum_{g\in G}[\sum_{\substack{i \\ g_i=g}} \log(\lambda_g(t_i)) - \int_0^T \lambda_g(t)dt]. \tag{2}$$

Because inter-event times $t_i - t_j$ appear in the likelihood function, typically it is necessary to store the history of event times and locations for model training. Recently introduced online methods for Hawkes processes [14] require only a sliding history in a moving window, however we would like to develop an online learning algorithm that runs on the client side without any history storage on the server.

For this purpose we assume the triggering kernel $f(t)$ can be approximated by a fixed basis of exponential kernels,

$$f(t) = \sum_{k=1}^{K} \theta_k \omega_k e^{-\omega_k t}. \tag{3}$$

Here the parameters ω_k set the decay timescale of each individual exponential kernel and the θ_k are mixture parameters setting the relative weight of each kernel. Because $f(t)$ is a Poisson process the θ_k need not sum to 1.

This choice of kernel in Equation 3 has several advantages. By defining,

$$F_k^g(t) = \sum_{\substack{t>t_j \\ g_j=g}} \theta_k \omega_k e^{-\omega_k(t-t_j)}, \tag{4}$$

we can then write the overall intensity as,

$$\lambda_g(t) = \mu_g + \sum_{k=1}^{K} F_k^g(t). \tag{5}$$

This allows a sequential update for the intensity,

$$\lambda_g(t_i^-) = \mu_g + \sum_{k=1}^{K} e^{-\omega_k(t_i-t_{i-1})} F_k^g(t_{i-1}^+), \tag{6}$$

where t^+ indicates the right-sided limit of F_k^g. We therefore only have to store μ_g, $F_k^g(t_{i-1}^+)$ and the time of the last event t_{i-1} on the server and the user can perform the intensity update on the client side.

The second advantage of the exponential kernel basis is that μ_g and θ_k can be updated via an online stochastic gradient descent step on the client side, with no event history required. Following [15], the sgd update is given by,

$$\mu_g \leftarrow \mu_g + dt\big[\mu_g/\lambda_g(t_i^-) - \mu_g(t_i - t_{prev}^g)\big] \tag{7}$$

$$\theta_k \leftarrow \theta_k + dt\big[F_k^g(t_i^-)/\lambda_g(t_i^-) - \theta_k\big], \tag{8}$$

where g indexes the grid cell containing the new event with time t_i, and t_{prev}^g is the time of the last event in grid cell g (which may not be the same as the time of the most recent event overall t_{i-1}). The update in Equation 7 is an EM-type step [15] with the gradient multiplied by μ_g and the update in Equation 8 has the gradient multiplied by θ_k. Empirically these modifications allow for larger step sizes dt and quicker convergence.

B. Online word mover distance topic model

In the previous section we developed an online, client-side estimation strategy for a space-time intensity of events of a single category. In this section we develop an online, client-side topic model for text reports accompanying crime and social disorder events. The idea is that the event is first mapped to a topic, and then the intensity and model parameters for the topic are updated as in Equations 6, 7 and 8.

For this purpose we consider $l = 1, ..., L$ topics, each represented by $k_l = 1, ..., C_l$ centers. Each topic center w_{k_l} may be viewed as a word2vec representation of a word, though there may be no word in the corpus with that exact representation. Next consider the word2vec transformation of the bag of words $V = \{v_i\}_{i=1}^m$ (with stop words removed) of a given crime report. The word movers distance [16] of the bag of words V to the centers W_l of a particular topic is given by,

$$d(V, W_l) = \frac{1}{m} \sum_{i=1}^{m} \min_{k_l \in W_l} \|v_i - w_{k_l}\|_2. \tag{9}$$

Thus the topic assigned to each report is the topic with the closest set of centers according to the word movers distance.

Because topics may change over time, but also for cold starts, we would like to update the topic centers as new reports are generated. For this purpose we use a K-means like heuristic,

$$w_{k_l} \leftarrow (1 - dt) w_{k_l} + dt \; v_i \tag{10}$$

where W_l is topic of V (closest word movers distance) and v_i is the closest word in V to w_{k_l}. Equation 10 is analogous to Euclidean based online kmeans [17] where the learning rate dt needs to go to zero for convergence. We do not have a theoretical justification for this algorithm, but we show in the next section that the algorithm works reasonably well for our desired application.

Several practical issues remain for the implementation of the online topic model. First, we need a word2vec model for performing the transformations. In light of the privacy concerns we are attempting to address, and also because we may have limited data to start, we advocate for the use of a pre-trained word2vec model. Here we use an open source word2vec model trained on Google News [18] that could be embedded within the application on the client. Second, we need initial center words for each topic at the outset. Here we propose using a set of seed words for each topic that are either determined manually by a human expert or that are chosen from a limited training data set where privacy need not be maintained during training. We give specific examples in the next section and henceforth refer to our topic model as the Seeded Online Word Movers Distance (SOWMD) model.

Our overall algorithm for the crowd-sourced online Hawkes process is given in Algorithm 1. The client receives the intensity, model parameters, and topic centers from the server. The user inputs text, the word2vec representation of the text is categorized into a topic, and then the topic center is updated. Next the Hawkes parameters and intensities of that topic are updated and finally sent back to the server along with the updated topic centers.

Algorithm 1: Crowd-sourced Online Hawkes

Server

Send: μ_g^l, $F_k^{g,l}(t_{i-1}^+)$, ω_k^l, $t_{prev}^{g,l}$, θ_k^l, w_{k_l}, dt

Receive: μ_g^l, $F_k^{g,l}(t_i^+)$, $t_{prev}^{g,l}$, θ_k^l, w_{k_l}

Client

Receive: μ_g^l, $F_k^{g,l}(t_{i-1}^+)$, ω_k^l, $t_{prev}^{g,l}$, θ_k^l, w_{k_l}, dt

UserInput: t_i, x_i, text

1. $g \leftarrow$ Grid(x_i) \\ Get grid cell of event location
2. $V \leftarrow$ word2vec(text) \\ Get bag of word vectors
3. $l = \arg\min_s d(V, W_s)$ \\ Get topic of event
4. $w_{k_l} \leftarrow (1 - dt) w_{k_l} + dt \; v_i$ \\ Update centers

\\ Then update Hawkes parameters and intensity

5. $\mu_g^l \leftarrow \mu_g^l + dt\big[\mu_g^l / \lambda_g^l(t_i^-) - \mu_g^l (t_i - t_{prev}^{g,l})\big]$
6. For each k: $\theta_k^l \leftarrow \theta_k^l + dt\big[F_k^{g,l}(t_i^-)/\lambda_g^l(t_i^-) - \theta_k^l\big]$
7. For each s, m, k:
   $F_k^{s,m}(t_i^+) \leftarrow e^{-\omega_k (t_i - t_{i-1})} F_k^{s,m}(t_{i-1}^+) + \theta_k^m \omega_k^m 1_{\substack{s=g \\ m=l}}$
8. $t_{prev}^{g,l} \leftarrow t_i$

Send: μ_g^l, $F_k^{g,l}(t_i^+)$, $t_{prev}^{g,l}$, θ_k^l, w_{k_l}

III. Experimental results

A. Synthetic Hawkes process

We first test the online estimation algorithm developed in Section IIA on a simulated Hawkes process. We simulate a Hawkes process with $\mu = .1$ and exponential triggering kernel $\theta\omega \exp(-\omega t)$ with parameters $\theta = .5$ and $\omega = .3$. For the kernel basis we use $\omega_1 = .5$, $\omega_2 = .1$ and $\omega_3 = .05$ and we use an online learning rate of $dt = .01$. In Figure 2 we track the parameter estimates and error over the course of successive online gradient descent iterations. Quickly the background rate μ converges to the true value. Here we fix $dt = .01$ to track non-stationary trends in the background rate, whereas online gradient descent will only converge when $dt \to 0$. The estimated (effective) parameter $\theta_1 + \theta_2 + \theta_3$ also converges to the true value of $\theta = .5$. We note that the l_2 error of the triggering kernel decreases but does not go to zero. This is because of the choice of using only 3 basis functions; our primary goal is to obtain reasonably accurate privacy preserving models rather than solutions with the highest accuracy.

B. Topic modeling of Los Angeles crime reports

Next we test the SOWMD topic model on crime report data from 2009-2014 in Los Angeles. The dataset consists of 805523 events containing an incident category, date, text narrative (description of the event), and spatial coordinates of where the event occurred. To obtain seeds for the SOWMD model, we use the first 1000 events of the dataset and take the 10 most frequent words in each of the 10 crime types found in the 1000 events. We use a pre-trained word2vec model trained on Google News [18] to obtain word vectors for each word (the same model is used across all of our experiments). These

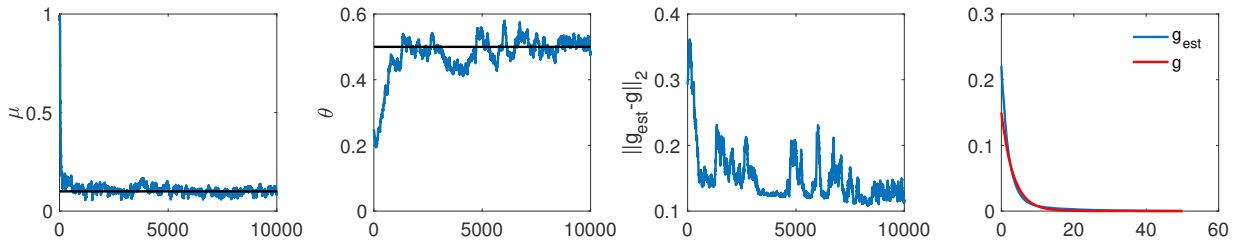

Fig. 2: Estimated μ vs online gradient descent iterations (far left), true θ vs estimated $\theta_1 + \theta_2 + \theta_3$ (middle left), l_2 error of triggering kernel vs iterations (middle right) and triggering kernel estimate at final iteration (far right).

10 word vectors form the centers of each of the 10 clusters we use to seed the SOWMD algorithm. We then apply the online algorithm to the next 50000 events, as well as the LDA algorithm provided with gensim [19]. In Table I we compare SOWMD vs LDA using the UCI coherence measure [20], which is based upon the frequency of pairs of topic words relative to their individual frequencies and has good correlation with human measures of topic cohesiveness [21]. On average the SOWMD model has a higher UCI coherence level, as has been reported for offline word movers distance models. Both models have some coherent topics, for example topic 1 for LDA refers to fraud whereas topic 1 for SOWMD refers to assault. Several low coherence topics also emerge, topic 3 and 8 in the case of LDA and topic 4 in the case of SOWMD.

We note that, while the SOWMD algorithm was seeded with crime type categories, as the algorithm moves forward in time the topics evolve and are not meant to simply classify crime type. We explore this effect more in the next section in terms of the accuracy of the coupled SOWMD-online Hawkes model when used for predicting space-time crime patterns.

C. Space-time crime prediction in Los Angeles

Building off of the experiment in the previous section, we next test the full Algorithm 1 for simultaneous topic and Hawkes intensity estimation. We again use the LAPD crime data set, however this time we estimate an intensity $\lambda_g^l(t)$ for each of the 10 topics. We then use the estimated intensity to rank all grid cells g in the city each day according to their risk of incidents of type l. One popular choice for measuring the accuracy of a crime ranking algorithm is the Predictive Accuracy Index (PAI) [4], which measures the percentage of incidents captured in the top k grid cells flagged for patrol. The PAI is area normalized (by the area of the k cells) so that a PAI of 1 corresponds to random predictions.

In Figure 3 we display the PAI for the online, crowd-sourced Hawkes process for several topics learned by the algorithm. For comparison, we use an offline Hawkes process [5] trained on the given crime categories and display those results as a baseline. The highest PAI corresponds to aggrevated assault and the offline model. Violent crimes occurring on the street network have a higher PAI because the incidents are more highly concentrated compared to property crime. The online model learns a similar category where the most frequent words are attempted, stab, and knife. The PAI is not as high as the offline assault model, but the online intensity is still reasonably accurate. Additionally, for the next two topics related to property and vehicle crime, the online model outperforms the offline model in terms of vehicle theft and burglary. One explanation for the improvement in accuracy is that the SOWMD-Hawkes model may learn more coherent topics compared to crime categories in some cases, therefore leading to more concentrated Hawkes intensities and higher PAI values.

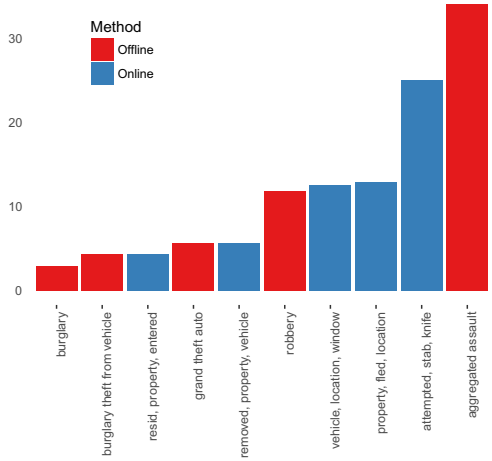

Fig. 3: PAI comparison of offline Hawkes process trained using crime type categories and online Hawkes process using SOWMD topic categories.

D. Modeling Nextdoor public safety posts in Indianapolis

In our final experiment, we apply the online joint SOWMD-Hawkes model in Algorithm 1 to public safety posts from Nextdoor. The dataset consists of 115 posts tagged as public safety between July 1st and December 7th 2017 from the Meridian-Kessler area of Indianapolis. We use a time only Hawkes process and use the text of each post to extract the topic. Because the dataset is limited in size, we seed the topic model with simply the first 10 posts (comprising each of the 10 topics).

In Figure 4 we provide an example anonymized post and the corresponding topic that was learned in training. Many posts contain personal identifying information (names, address, etc.) and in some posts a victim and/or perpetrator is identified. For this reason, along with the fact that private companies may only want to share limited data with law enforcement,

TABLE I: Topic coherence comparison between LDA and Seeded Online Word Movers Distance (SOWMD) topic model.

LDA (Aggregate UCI=.13)	UCI
1. used, info, without, personal, permission, card, credit, open, obtain, number	1.442228399
2. kill, stated, verbal, dispute, fled, threatened, going, phone, location, knife	0.475918243
3. id, theft, check, took, location, use, property, cash, forged, checks	-1.901758853
4. fled, face, punched, approached, grabbed, head, struck, property, hit, left	0.371021747
5. fled, vehicle, property, window, door, location, side, rear, used, open	0.412661259
6. fled, vehicle, property, location, approached, demanded, took, s2, money, handgun	0.493246568
7. property, fled, removed, location, vehicle, entered, window, entry, direction, took	0.443388061
8. vehicle, order, return, court, failed, upon, location, child, mo, missing	-2.994888929
9. became, times, causing, face, argument, struck, punched, verbal, angry, involved	1.571898116
10. entered, store, without, paying, location, exited, property, items, removed, concealed	1.027530669
SOWMD (Aggregate UCI=.81)	**UCI**
1. pushed, grabbed, punched, dispute, approached, causing, face, verbal, kicked, hands	1.044113194
2. property, vehicle, removed, window, entered, smashed, location, fled, took, door	0.445021529
3. vehicle, vandalized, window, fled, smashed, tires, rear, approached, slashed, parked	0.329571273
4. vehicle, stated, became, used, verbal, face, kill, money, causing, approached	-0.126301107
5. used, info, personal, open, employment, obtain, id, identity, without, ssn	1.304224724
6. used, info, personal, obtain, employment, gain, ssn, identity, property, tax	0.930475982
7. threatened, kill, fear, called, shoot, harm, knife, life, gun, verbally	0.808682825
8. used, identity, info, personal, gain, purchases, permission, card, purchase, name	1.456026639
9. fled, location, property, vehicle, removed, entered, door, window, took, dir	0.425668881
10. used, info, id, card, without, personal, permission, credit, obtain, open	1.447870606

we believe a privacy preserving Hawkes process could find application in partnerships between companies like Nextdoor, public agencies, and the general public.

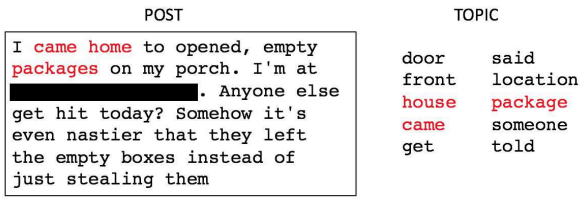

Fig. 4: Example anonymized Nextdoor post and corresponding topic.

In Figure 5 we plot the learned intensities for the top four topics. The topics have four general themes: car break-ins common in the area, packages stolen from houses, bikes stolen, and a fourth topic where people are posting that they see police in the area. These type of intensities could serve several functions. They could alert police to areas in the city where patrols or other interventions are needed. They also might give citizens valuable information on social disorder and crime in their city. For example, knowing that the risk of package theft increases several-fold in October 2017 is valuable information for community members to access. Currently users can only see posts from their neighborhood, presumably for privacy reasons, but privacy preserving intensities such as those in Figure 5 might be shared across neighborhoods without exposing sensitive information.

IV. Discussion

A. On the transparency of Algorithm 1

On the one hand, Algorithm 1 is transparent in the sense that the source code is open and lives on the client side of the application. That being said, a deep neural network might also satisfy the same conditions while being viewed as less transparent than a logistic regression. The most opaque step is step 2, where word2vec is used to represent word vectors in a low dimensional space. However, the overall SOWMD algorithm has an intuitive description: a topic is chosen for the text such that the words in the topic are semantically closest on average to the words of the text. The Hawkes parameter and intensity updates 5-7 are fairly simple and the main takeaway from a user perspective is that, by posting a crime report, the user is increasing the estimated risk of incidents (of the categorized topic) in their neighborhood g. In the scenario where police are patrolling based upon the estimated intensity, then filing a crime report through the application may be viewed as a probabilistic request for extra patrols in one's neighborhood.

B. On the privacy of Algorithm 1

In terms of privacy we have ignored formal definitions such as differential privacy [22]. Instead, we have focused on the goals of 1) storing minimal user information 2) reducing the risk of identification using geolocation information and 3) minimizing the risk of identification using the text from the report. In terms of 1), no historical data is stored save for the most recent event time in each grid cell. Some information still may be inferred in terms of 2) and 3). For example, if the grid cell size is small enough then the location of the reporter is known up to that distance scale. This is consistent with methods police agencies currently employ to reduce geolocation identification, namely rounding to the nearest block or some other larger geographical unit. Some information on the text of the report may also be inferred from the SOWMD word center update, in particular the words semantically closest to the centers of the selected topic. This risk can be reduced by keeping the number of centers small or only updating the center if the closest report word is sufficiently close in the word vector space (for example to prevent the person's last name from being exposed even when it is a far distance from all centers).

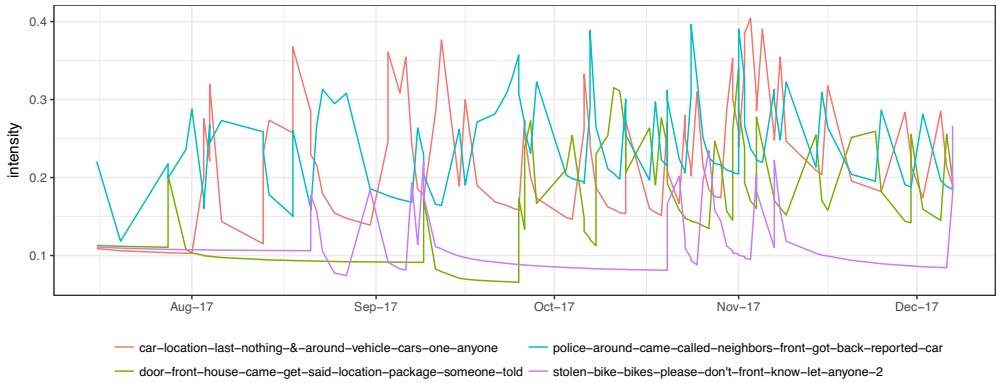

Fig. 5: Crowd-sourced Hawkes intensities for four topics (with top 10 keywords displayed) learned from Nextdoor public safety posts in the Meridian-Kessler neighborhood in Indianapolis from July-November 2017.

C. Future directions

The experiments here were conducted on historical data and the impact of incorporating crowd-sourced information from crime tips, Nextdoor, and similar sources has to date not been tested in the field. Future research may focus on the impact of this type of crowd-sourcing, in terms of the impact on crime rates as well as perceptions of predictive policing. Similar Hawkes process models also arise in other social settings and recent applications include Twitter resharing [23], IPTV viewing behavior [24], and human mobility [25]. The methods here may provide computationally efficient, privacy preserving alternatives to recently introduced Hawkes-topic models based upon LDA [26].

V. Acknowledgements

This work was supported in part by NSF grants SCC-1737585, SES-1343123, ATD-1737996 and ATD-1737770.

References

[1] X. Wang, M. S. Gerber, and D. E. Brown, "Automatic crime prediction using events extracted from twitter posts," in *International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction*. Springer, 2012, pp. 231–238.

[2] A. Bogomolov, B. Lepri, J. Staiano, N. Oliver, F. Pianesi, and A. Pentland, "Once upon a crime: towards crime prediction from demographics and mobile data," in *Proceedings of the 16th international conference on multimodal interaction*. ACM, 2014, pp. 427–434.

[3] A. Khosla, B. An An, J. J. Lim, and A. Torralba, "Looking beyond the visible scene," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3710–3717.

[4] G. O. Mohler and M. Porter, "Rotational grid, pai-maximizing crime forecasts," *preprint*, 2017.

[5] G. O. Mohler, M. B. Short, S. Malinowski, M. Johnson, G. E. Tita, A. L. Bertozzi, and P. J. Brantingham, "Randomized controlled field trials of predictive policing," *Journal of the American Statistical Association*, vol. 110, no. 512, pp. 1399–1411, 2015.

[6] S. Chainey, L. Tompson, and S. Uhlig, "The utility of hotspot mapping for predicting spatial patterns of crime," *Security Journal*, vol. 21, no. 1, pp. 4–28, 2008.

[7] R. Kanable, "Talking to tipsters: Websites and text messages allow valuable, anonymous communication," *Law Enforcement Technology*, vol. 35, no. 11, pp. 10–12, 2008.

[8] P.-P. Paré, R. B. Felson, and M. Ouimet, "Community variation in crime clearance: A multilevel analysis with comments on assessing police performance," *Journal of Quantitative Criminology*, vol. 23, no. 3, pp. 243–258, 2007.

[9] K. Lum and W. Isaac, "To predict and serve?" *Significance*, vol. 13, no. 5, pp. 14–19, 2016.

[10] [Online]. Available: https://www.civicscape.com

[11] J. Hamm, J. Luken, and Y. Xie, "Crowd-ml: A library for privacy-preserving machine learning on smart devices."

[12] G. Mohler, M. Short, P. J. Brantingham, F. Schoenberg, and G. Tita, "Self-exciting point process modeling of crime," *Journal of the American Statistical Association*, vol. 106, no. 493, pp. 100–108, 2011.

[13] M. B. Short, G. O. Mohler, P. J. Brantingham, and G. E. Tita, "Gang rivalry dynamics via coupled point process networks," *DCDS B*, vol. 19, no. 5, pp. 1459–1477, 2014.

[14] Y. Yang, J. Etesami, N. He, and N. Kiyavash, "Online learning for multivariate hawkes processes," in *Advances in Neural Information Processing Systems*, 2017, pp. 4944–4953.

[15] E. Lewis and G. Mohler, "A nonparametric em algorithm for multiscale hawkes processes," *preprint*, pp. 1–16, 2011.

[16] M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger, "From word embeddings to document distances," in *International Conference on Machine Learning*, 2015, pp. 957–966.

[17] L. Bottou and Y. Bengio, "Convergence properties of the k-means algorithms," in *Advances in neural information processing systems*, 1995, pp. 585–592.

[18] [Online]. Available: https://code.google.com/archive/p/word2vec/

[19] [Online]. Available: https://radimrehurek.com/gensim/

[20] D. Newman, J. H. Lau, K. Grieser, and T. Baldwin, "Automatic evaluation of topic coherence," in *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 2010, pp. 100–108.

[21] K. Stevens, P. Kegelmeyer, D. Andrzejewski, and D. Buttler, "Exploring topic coherence over many models and many topics," in *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Association for Computational Linguistics, 2012, pp. 952–961.

[22] C. Dwork, "Differential privacy: A survey of results," in *International Conference on Theory and Applications of Models of Computation*. Springer, 2008, pp. 1–19.

[23] 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*. ACM, 2015, pp. 1513–1522.

[24] H. Xu and H. Zha, "A dirichlet mixture model of hawkes processes for event sequence clustering," *arXiv preprint arXiv:1701.09177*, 2017.

[25] P. Wang, Y. Fu, G. Liu, W. Hu, and C. Aggarwal, "Human mobility synchronization and trip purpose detection with mixture of hawkes processes," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2017, pp. 495–503.

[26] X. He, T. Rekatsinas, J. Foulds, L. Getoor, and Y. Liu, "Hawkestopic: A joint model for network inference and topic modeling from text-based cascades," in *Proc. 32nd Intl. Conf. on Machine Learning*, 2015.