PLOS ONE

RESEARCH ARTICLE

A LSTM-Hawkes hybrid model for posterior click distribution forecast in the advertising network environment

Sangwon Hwang, Inwhee Joe*

Department of Computer Science and Engineering, Hanyang University, Seoul, South Korea

* iwjoe@hanyang.ac.kr

Citation: Hwang S, Joe I (2020) A LSTM-Hawkes hybrid model for posterior click distribution forecast in the advertising network environment. PLoS ONE 15(6): e0232887. https://doi.org/10.1371/journal.pone.0232887

Editor: Xiaodi Huang, Charles Sturt University, AUSTRALIA

Received: September 29, 2019

Accepted: April 23, 2020

Published: June 5, 2020

Data Availability Statement: All relevant data are within the manuscript and its Supporting Information files.

Funding: This work was supported by Institute for Information & Communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (No.2020-0-00107, Development of the technology to automate the recommendations for big data analytic models that define data characteristics and problems).

Competing interests: The authors have declared that no competing interests exist.

Abstract

In the field of advertising technology, it is a key task to forecast posterior click distribution since 66% of advertising transactions depend on cost per click model. However, due to the General Data Protection Regulation, machine learning techniques to forecast posterior click distribution based on the sequences of an identified user's actions are restricted in European countries. To overcome this barrier, we introduce a contextual behavior concept for the advertising network environment and propose a new hybrid model, which we call the Long Short Term Memory—Hawkes model by combining a stochastic-based generative model and a machine learning-based predictive model. Also, to meet the computational efficiency for the heavy demand in mobile advertisement market, we define gradient exponential kernel with just three hyper parameters to minimize residuals. We have carefully tested our proposed model with production data and found that the LSTM-Hawkes model reduces the Mean Squared Error by at least 27.1% and up to 83.8% on average in comparison to the existing Hawkes Process based algorithm, Hawkes Intensity Process, as well as 39.77% on average in comparison to Multivariate Linear Regression. We have also found that our proposed model improves the forecast accuracy by about 21.2% on average.

Introduction

For the first time in 2017, global digital advertisement expenditure was 41% of the global advertising market, which exceeded the TV advertisement expenditure by 6%. Especially the mobile advertisement in digital marketing recorded 37.6% of the global growth rate, which is an impressive increase, enough to draw attention in the worldwide advertisement market [1]. These reports [2–4] substantiate that the mobile advertising market is leading the growth in the global advertising market. However, despite the rapid market growth, there is a controversial issue that Ad Tech faces. General Data Protection Regulation (GDPR) [5] has been actively enforced by EU since 25 May 2018.

Related work

The typical strategy of user targeting is based on analyzing the behavior of users across the internet or across the ad network individuals [6, 7, 8]. Another strategy which predicts if

someone is going to click an impression is matrix factorization inferring the relationship between user vector and context vector where the output is click through ratio [9, 10, 11]. However, the both traditional methods require user information or inferred user identification as the essential asset to forecast the click popularity or probability. Tracking user behavior between different ad tech individuals has been forbidden in EU as well as personal information transfer is also banned. Thus, technically both types of traditional methods are not feasible to be an real world application under GDPR. To overcome the barrier, we focus on an advertising network's posterior click distribution; not a user's CTR, nor an advertisement's CTR since 1) an advertising network or a publisher is still an identical user group remaining individual users anonymous, and 2)CTR only reflects short-term user engagement. We developed a model that can learn a general and efficient representation of the underlying dynamics from the event history with a hyper-parametric form.

Advertising network environment

An advertising network's environment consists of three main parts. First there is the advertiser who wants to advertise, as well as provide the actual advertisement contents. Next, there is the publisher who provides the landing pages for the advertisement contents. Lastly there is the advertising network that connects the advertiser and publisher together. Advertisers are grouped and managed in a system called the 'Demand Side Platform' and publishers are grouped together in a system called the 'Supply Side Platform'. Fig 1 simply describes the Ad Tech environment and its layers. ***Between layers***, due to the GDPR, user information including advertising IDs, IP addresses or any other unique device IDs from a mobile device cannot be transferred.

Materials and methods

Formulation and Preliminary Theory

When a random variable following a Bernoulli process which is the time of occurrence (or success) in Bernoulli trials approaches to positive infinity, it is infeasible to calculate the probability of a specific event occurrence due to the computational complexity. Therefore, to solve this problem, in modeling a posterior distribution of either prediction or simulation, most conventional statistical approaches have adopted 'Binomial Approximation to Poisson' which helps find probability of an independent trial in various fields [13–15]. However, the 'Binomial Approximation to Poisson' has Memoryless Property [16] because each trial is independent (Independent Increment) and the probability does not change (Stationary Increment). Thus, only event process with non-overlapping intervals can be used in the Binomial Approximation to Poisson in modeling a distribution.

In a click event distribution, where the event occurence time is t_i of set $T_j = \{t_1, t_2, \ldots, t_n\}$, inter-arrival time l_i of set $L_i = \{l_1, l_2, \ldots, l_n\}$ can be written as follows

$$t_{n-1} \leq l_n < t_n \quad \Rightarrow l_n = [t_{n-1}, t_n).$$

We then can easily find the concurrent occurrences in any of real world's advertising click distribution. To solve the memoryless property and handle the overlapping intervals, we propose a generative model based on a self-exciting process, a type of non-homogenous process, called Hawkes [17]. In this chapter, we provide mathematical induction from binomial distribution to Hawkes process to derive memory kernel of Hawkes.

Binomial approximation to poisson. Suppose that a random variable X follows binomial distribution $B(n, p)$, and that the expected value of X is λ. When n is close to positive infinity, λ

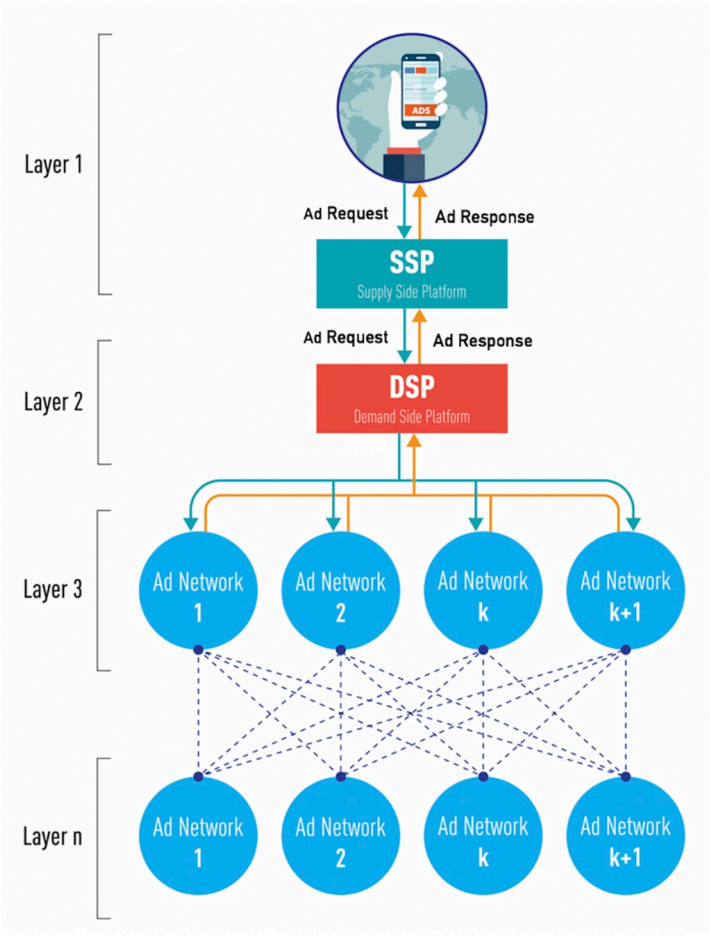


Fig 1. Advertising network architecture. Using Charles, a debugging proxy server application [12], we captured and analyzed packets from an advertising network and found out that the advertising network environment has a multilayer structure with the cycle flow as shown above. Advertising network environment is composed of advertisers (DSP), suppliers (SSP), and advertising networks (Ad Networks).

https://doi.org/10.1371/journal.pone.0232887.g001

approximates to np and by binomial approximation to the Poisson distribution, the probability of X approximates to the probability mass function of a Poisson distribution.

$$f(x) = (n!/x!(n-x)!)(\lambda/n)^x((1-\lambda)/n)^{n-x},$$
$$\lim_{n\to\infty} f(x) = (\exp^{\lambda}\lambda^x)/x! \quad where \quad X \sim B(n,p).$$

Process. When time unit expands or reduces to by t ($t > 0$), the average event occurrences becomes λt during the updated time unit where the average occurrences per default time unit is $\lambda = np$. Distribution with expanded (or reduced) time unit by t is called Process. And, correspondingly, PDF of the process becomes

$$f(x,\lambda t) = (\exp^{-\lambda t}(\lambda t)^x)/x!. \quad (1)$$

Poisson distribution and exponential distribution. Suppose that time period for a specific event to take place is a probability variable t and that T is the time when a specific event X_T takes place. Under this circumstance, the probability of that event X_T occurs after t is equal to that the event X_T does not take place within t. The PDF of Poisson distribution becomes

following equation

$$P(t < T) = f(X_T = 0, \lambda t) = (\exp^{-\lambda t}(\lambda t)^0)/0! = \exp^{-\lambda t}. \quad (2)$$

Set Eq (2) as $S(t)$ then, the probability of that event X_T occurs is $1 - S(t)$ which is cumulative distribution function (CDF) for the probability variable t. Set this function as $F(t)$ as following Eq (3)

$$F(t) = P(0 \leq T \leq t) = 1 - \exp^{-\lambda t}. \quad (3)$$

Since derivative of CDF is PDF, PDF for the random variable t is

$$f(t) = \frac{d}{dt}F(t) = \lambda \exp^{-\lambda t}. \quad (4)$$

Finally, it is concluded that the probability variable t follows exponential distribution with $f(t)$ and $F(t)$, PDF and CDF respectively.

Non-homogeneous process. Lambda intensity function $\lambda(t)$ determines non-homogeneity. If the event dynamics has positive covariance, intensity function becomes non-homogeneous, known as self exciting, otherwise it becomes homogenous.

$$COV(N(s,t), N(t,u)) > 0 \quad (5)$$

where N is point process and distance (s, t), (t, u) satisfy $s < x < t$ and $s < t < u$.

Mathmatically the intensity function of non-homogeneous processes is defined as the instantaneous rate at which events occur [18]. However, in the real world implementation for Hawkes, technically it has to be the conditional probability of a specific event occurrence during Δt, which can be interpreted as time unit, at t where the event has not taken place by t. Thus, we suppose that limitation of time unit does not go to zero. We specify each event with the index variable i and suppose that the set of event times is history H_t. Therefor, we can achieve following equations Eq (5) from Eqs (2), (3) and (4).

$$\begin{aligned} \lambda^*(t) &= f(t|H_t)/S(t|H_t) \\ &= f(t|H_t)/(1 - F(t|H_t)) \\ &= \mathbb{P}(t_{i+1} \in [t, t + \Delta t], t_{i+1} \notin \{(t_i, t), H_t\})/\Delta t \\ &= E(N(t + \Delta t) - N(t)|H_t)/\Delta t \end{aligned} \quad (6)$$

where $H_t = \{t_1, t_2, \ldots, t_i\}$ and $N(T)$, a counting process, is number of event occurrences during T.

LSTM-Hawkes hybrid model

The maximum likelihood estimation using exponential kernel has a problem with residuals because they become multiplied by real numbers depending on the batch size. We solved this problem by scaling cumulative intensity value of the kernel with differential coefficient γ that minimizes the residuals, defined Gradient Exponential Kernel Eq (9).

Also, to forecast posterior event time t_i, we used LSTM [19] instead of Thinning algorithm [20] which has been dominantly used with Hawkes, for accuracy enhancement.

In this chapter, we first list all variables and parameters used in the proposed model and define the memory kernel that we suggest. Second, in the subsection 'Sampling Method', we describe LSTM architecture and show the test result that empirically proves LSTM is more accurate than Thinning algorithm. Lastly, we propose the LSTM-Hawkes Forecast algorithm [Flow Diagram 1] which combines the generative model with LSTM prediction to draw the posterior process of advertising click event.

Hawkes process and its memory kernel. Hawkes process is self-exiting which satisfies Eq (5) in which the intensity rate Eq (6) explicitly and proportionally increases depending on past event occurrences. Hawkes process is formed as Eq (7) where $\mu(t)$ is the background intensity based on the observed activities which can be represented in two ways 1) expected intensity at t based on the observed endogenous events (in our case click) or 2) background intensity describing arrivals of events triggered by exogonous event (in our case conversion). Also hyper parameter α increases intensity rate by α at each t_i then decreases exponentially back toward $\mu(t)$. Both parameters are estimated by maximum likelihood and the initial value of $\mu(0)$ is set to expected CTR per time unit in this research.

$$\begin{aligned}\lambda(t) &= \mu(t) + \sum_{t_i<t}\phi(t-t_i),\\ \lambda(t) &= \mu(t) + \int_{-\infty}^{t}\phi(t-t_i)dN(t_i).\end{aligned} \tag{7}$$

Commonly used memory kernels with Hawkes process are Exponential kernel, original model proposed by Hawkes, and Power Law kernel, frequently used in social network model [21–23], proposed by Ozaki [24]. In the paper, we have chosen the exponential kernel Eq (8) to achieve better-case time complexity since the number of elementary operations performed increase depending on the umber of variables to approximate. The compared algorithm Eq (14) in our experiments adopts power law kernel Eq (9).

$$\phi(t-t_i) = \alpha e^{-\delta(t-t_i)}, \tag{8}$$

$$\phi(t-t_i) = km^{\beta}(t-t_i+c)^{-(1+\theta)}. \tag{9}$$

Gradient exponential kernel Gradient exponential kernel is defined as Eq (10), and the code implemented with consecutive loops of while, notated in a pseudo code of Algorithm 1, 2, 3, and 4 where the domain for the memory kernel satisfies $t \in eP$ as following

$$\phi(t-t_i) = \gamma\alpha e^{-\delta(t-t_i)} \quad where \quad t \in eP. \tag{10}$$

Hyper parameter estimation. Definition of the lambda intensity function $\lambda(t)$ in non-homogeneous processes was presented in the subchapter 'Formulation and Preliminary Theory' as Eq (6) as well as we derive our own memory kernel as Eq (10). By minimizing negative log likelihood (NLL) over the observed data, hyper parameters for the memory kernel can be approximated.

$f(t)$ Eq (4) is PDF of homogeneous Poisson process. Let $f'(t)$ is conditional probability on associated history up to t_{i-1} which is $f'(t) = \prod_{i=1}^{T} f(t_i|H_{i-1})$ then Eqs (2) and (3) can also be re-defined by $f'(t)$ as $S'(t)$ and $f'(t)$ respectively. Correspondingly, likelihood function for Hawkes can be derived as Eq (11) and its log likelihood function as Eq (12), driven by Rubin [25, 26]

$$L(t_i|\theta) = \prod_{i=1}^{T} f'(t_i|\theta) = \prod_{i=1}^{T}\lambda(t_i|\theta)(1-F'(t_i|\theta)), \tag{11}$$

$$l(t_i|\theta) = -\int_0^T \lambda(t_i|\theta)dt_i + \int_0^T log\lambda(t_i|\theta)dt_i. \tag{12}$$

Therefore, when eP_j is the domain elements for the random variable t with PDF $f'(t)$ and the memory kernel is Eq (10), we can derive NLL function for Gradient Exponential Kernel as

follows

$$-l(\tau_1, \ldots, \tau_n|\theta) = \mu \cdot \tau_n - \sum_{m=1}^{n}(\gamma\alpha/\delta)(e^{-\delta\cdot\tau_m} - 1) - \sum_{m=1}^{n} log(\mu + \gamma\alpha A(m)), \quad (13)$$

where $A(m) = \sum_{t_i<eP_m} e^{-\delta(eP_m - t_i)}$ for $i \geq 2$, t_i denotes the event time and A(1) is equal to 0. In our proposed model, optimization is proceeded by minimizing the negative log likelihood Eq (12).

Contextual behavior. To define behavior model, we introduce the behavior concept from existing models [27, 28]. User group behavior in advertising network environments is defined based on a set of activities. Also, action sequence, a set of endogenous and exogenous evnets, comprises each activity as element. The behavior model describes how specific ad-networks or publishers perform activities and how often the actions occur at different time.

LSTM We derived the definition of Hawkes and the proposed kernel in a closed form in the previous section. By introducing required parameters qualified to sets, variables and functions for Hawkes in Table 1, we were able to set up the NLL of our generative model. Now, activity sequences defined in Fig (2) can be set as input data for a RNN to learn and forecast T_i'. We have chosen one layer LSTM and also notated required parameters as *lw*, lookback window, $H_{(i)}$, history of activity sequences sampled *lw* in order, and $g(t_{(i)}|H_{(i)})$, function in the Neural Network. In the experiments with the real data sets, we configured conversion data as exogenous event. Fig 3 shows the architecture of LSTM configured in our method.

Table 1. Definitions.

Parameter	Interpretation
α	Increases intensity rate ($\alpha \geq 1$)
δ	Decreases intensity rate ($\delta \geq 0$)
eP_j	Evaluation Point, as argument of $\lambda(t)$
eP	Set of eP_j, equal to domain t *for* $\lambda(t)$
J	The index of the last element of the set eP
t_i	Observed event time
T_i	Set of t_i up to i
I	The index of the last element of the set T
eP_j'	Evaluation Point, as argument of $\lambda(t)$ for the predictive period
eP'	Set of eP_j, equal to domain of $t\lambda(t)$ for the predictive period
j'	The index of the last element of the set eP'
t_i'	Forecasted event time
T_i'	Set of t_i' up to i
I'	The index of the last element of the set T_i'
τ_n	Distance $[t_i, eP_j)$ such that $t_i \leq eP_j$
γ	Differential coefficient earned by gradient descent
	using Mean Squared Error (MSE) as objective function
$\mu(t)$	Background intensity based on the observed activities,
	sets of view and click, during the configured look-back window at t
θ	Set of hyper parameters
$L(\theta)$	Likelihood with parameters θ
$l(\theta)$	Log likelihood function with parameters θ
CIF	Cumulative intensity function

https://doi.org/10.1371/journal.pone.0232887.t001

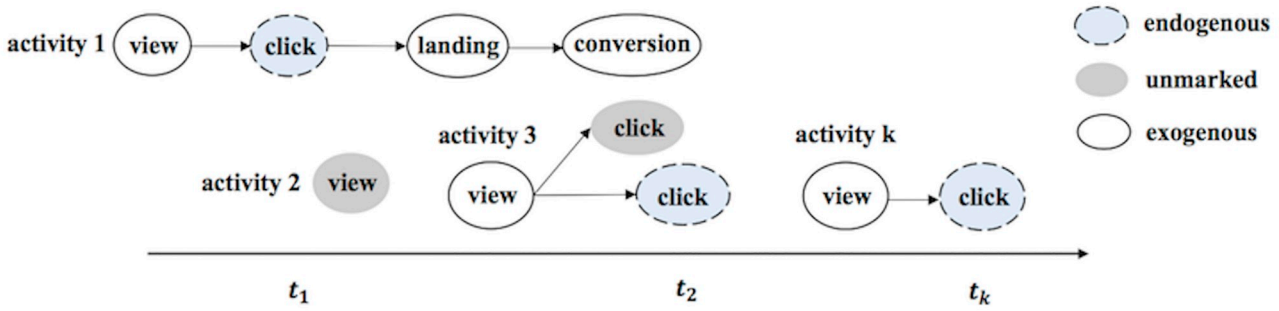


Fig 2. Behavior model in an Ad network. In the case where a user just view an advertisement (no click occurrence), user group's activity is not temporally marked up since view is not the main interest event in our experiment.

https://doi.org/10.1371/journal.pone.0232887.g002

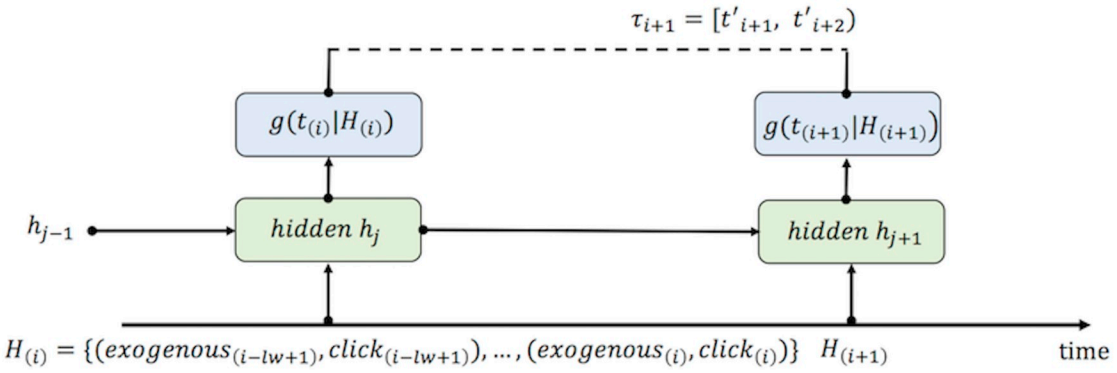


Fig 3. LSTM architecture. LSTM is good fit to both cases 1) where eP_j is equal to t_i, $(i = j)$, mathematical definition, and 2) where eP_j is equal to t_i, $(i \neq j)$, technical implementation.

https://doi.org/10.1371/journal.pone.0232887.g003

LSTM-Hawkes model. LSTM-Hawkes is made up of four consecutive steps, A) model configuration, B) parameter optimization by minimizing Eq (12), C) earning differential coefficient γ which minimizes residual error, and D) forecasting posterior event time during the prediction period and drawing $\lambda(t)$. Those steps and each algorithm are described in Algorithm 1, 2, 3, and 4 as pseudo codes below. In the Algorithm 1, we chose evaluation points by dividing the distance $[t_1, t_n)$ with the size of eP which means at the implementation level, Δt which is $t - t_i$ in the Eq (8), is equal to τ_n. After that, the hyper parameters are defined with the initial values and optimization is proceeded. In the Algorithm 4, by calculating lambda intensity with the results from LSTM prediction T'_i, our model finally draws cumulative intensity values for $[t_1, t'_n)$.

Algorithm 1: Model Configuration

```
Data: Observed data set t_i
Result: λ(eP_j)
  d ← (t_n − t1)/K
  ▷ Set an equal interval. K is size of set eP
  eP_1 ← t_1
while (2 ≤ j ≤ K) do
  eP_j ← t_1 + (j − 1)d
  j = +1
  α, δ ← α_1, δ_1
  ▷ Hyper parameter setup
```

Algorithm 2: Parameter Optimization

Date: $\lambda(eP_j)$
Result: α, δ, γ
  $l(\theta) \leftarrow$ Eq (13)
  ▷ *Set log likelihood.* $l(\theta)$
while (*Not converged*) **do**
  Run optimization function minimizing $-l(\theta)$
Return Parameters of gradient exponential kernel α, δ

Algorithm 3: Earning Differential Coefficient γ

Date: Observed data t_i
Result: γ
for $j = 1; j \leq J; j++$ **do**
  for $i = 1; t_i \leq eP_j; i++$ **do**
    $CIF.Append(\lambda(ep_j - t_i))$
$CIF \leftarrow$ *Sum of CIF values by minute*
$OriginalIntensity \leftarrow$ *Sum of observed event number by minute*
$n \leftarrow$ *size of CIF*
while $(1 \leq j \leq n)$ **do**
  Residuals = $Original_I ntensity_i - CIF_i$
  Optimization of Gradient Descent using MSE as loss function
  Return γ

Algorithm 4: Forecasting posterior event time $t_i^{'}$, $\lambda(t_i^{'})$

Date: *CIF of gradient hawkes*
Result: *CIF for* $[t_1, t_n^{'})$
  T' *LSTM Sampling*$(T,$ *predictionPeriod*$)$
    $d \leftarrow (t_n^{'} - t_1^{'})/K^{'}$
  ▷ *Set an equal interval.* K' *is size of set* eP'
    $eP_1^{'} \leftarrow t_1^{'}$
while $(2 \leq j \leq K')$ **do**
  $eP_j^{'} \leftarrow t_1^{'} + (j-1)d$
  $j = +1$
for $j = 1; j \leq J'$ $j ++$ **do**
  for $i = 1; t_i^{'} \leq eP_j^{'}; i++$ **do**
    $CIF.Append(\lambda(ep_j^{'} - t_i^{'}))$
  Return CIF
  ▷*CIF for* $[t_1, t_n^{'})$

Results and analysis

Data

Criteo, an Ad Tech company, possesses cutting edge user re-targeting technology and has led the development of prediction methods based on machine learning algorithms. Being a leader in this field, Criteo has also notably hosted the Kaggle's CTR forecast competition in 2014 and 2015. We applied the click and conversion data sets [29] provided by the Criteo Lab and compared the results between the the model/algorithm proposed and the Hawkes Intensity Process algorithm (HIP) [21] algorithm. We evaluated the forecasting accuracy of the posterior distribution of a click event from a single advertising network.

Goodness of fit

Residual analysis [30] is a reliable measurement for the Hawkes model and has been widely used to evaluate the precision of a fit. Let t_i be a point process with intensity $\lambda(t_i)$ whose PDF is Eq (10) and s_i be equal to $\lambda(t_i)$ of Eq (5), then s_i becomes a unit rate Poisson process transformed by a Hawkes model. Thus, if the model fits well, the transformed process should

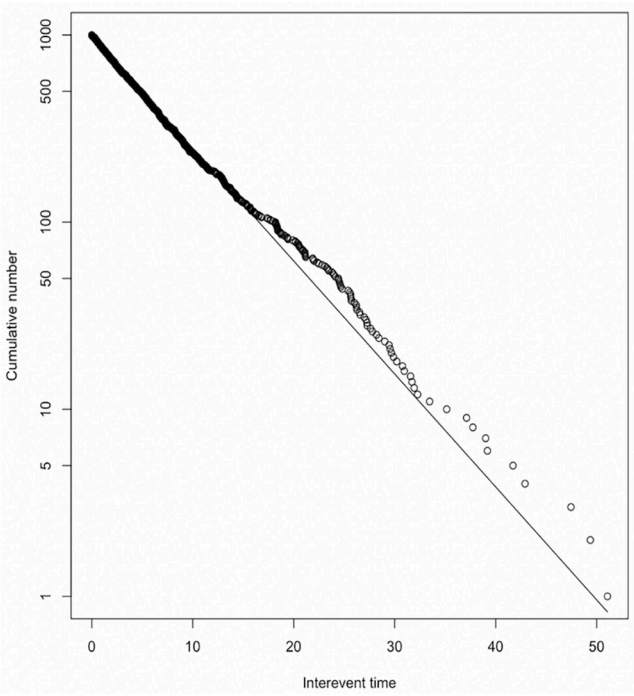


Fig 4. log-log-plot of residual's interevent times. A thousand events are sampled.

https://doi.org/10.1371/journal.pone.0232887.g004

resemble a unit Poisson process. Also, the residual's inter-event time is supposed to be an independent exponential variable. Therefore, log-log-plot of the residual's inter-event time should be close to the linear line. Fig 4 shows the log-log-plot of the residual inter-event time from the LSTM-Hawkes model and it proves that the shape of the distribution is very close to the linear line. Also, the first quintile of residual's inter-event times are distributed the most. Thus, we can conclude that the LSTM-Hawkes model is an excellent way to determine a precise fit.

Compared algorithm HIP

The HIP model is a Hawkes-based model that uses Power-Law-Kernel Eq (9). Since HIP mathematically induces the expectation function $\xi(t)$ of $\lambda(t)$ referring to exogenous events to predict endogenous event, it is great to compare with our model having the same input sequence. The only difference between two models is that HIP supposes that exogenous events are given even for the predictive period while LSTM-Hawkes does not.

$$\xi(t) = Expectation\ \ of\ \ \lambda(t)\ \ where\ \ \lambda(t) = \mu s(t) + \sum_{t_i<t} km^{\beta}(\tau + c)^{-(l+\theta)}. \tag{14}$$

Compared algorithm Multivariate Linear Regression (MLR)

MLR has been widely adopted as an compared algorithm in population prediction in social media [31]. With production data we used for the experiment, next predicted intensity value based on the history $H_{(}\ i)$ is calculated as below.

$$f(H_{(}i)) = \beta_0 + exogenous_i * \beta_1 + endogenous_i * \beta_2$$

Error score

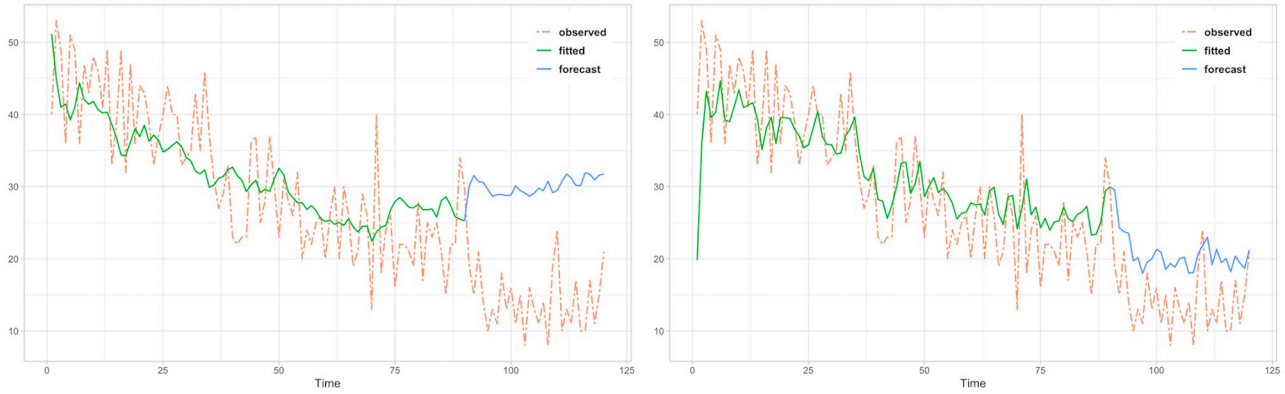
$$s = \begin{cases} \sum_{i=1}^{n} e^{-(d/a_1)} - 1 & for\ d < 0, \\ \sum_{i=1}^{n} e^{(d/a_2)} - 1 & for\ d \geq 0. \end{cases} \tag{15}$$

Commonly used error measurements such as the MSE, Mean Absolute Error (MAE), and the Root Mean Square Error (RMSE) are only designed to measure the raw residuals. However, different from these methods, scores metric [32] is devised to take the negative error into account, which reduces the error score when the residuals are less than a certain point. Not only does the score metric method discover how many residuals the model produces, but it is also able to discover how well the model fits and predicts. In the experiment, we set up this certain point using the standard deviation of the observed data during the time period for the prediction. In the 1st test section, both a_1 and a_2 are 4.057 and in the 2nd test section both a_1 and a_2 are 7.495. The final score is the sum of each function conditioned on the distance mark.

Experimental results

To assess the comparison between the HIP, MLR and LSTM Hawkes models, we selected two different sections where the moving average trend of the click is monotonically decreasing in the first section and increasing in the second section. Test results of the first section are presented in Figs 5, 6 and 7 and Table 2. Also, test results of the second section are presented in Figs 8, 9 and 10 and Table 3. For an accurate performance assessment of the HIP model and MRL, we chose several correlation coefficients so that those models could show its accuracy in prediction, precision in fitting, as well as it's limitation in both the fitting and prediction. For the accuracy of our assessment we chose four equally dispersed correlation coefficients of 0.2, 0.4, 0.6, and 0.8. The observed click is the endogenous event, and the exogenous event is the observed conversion.

First, in the forecasting test in the first section, it shows that the forecast made using the LSTM-Hawkes model reduces 73.9% of MSE on average and at least 27.1% compared to the HIP model where the correlation coefficient is 0.8, as well as 30.16% of MSE on average

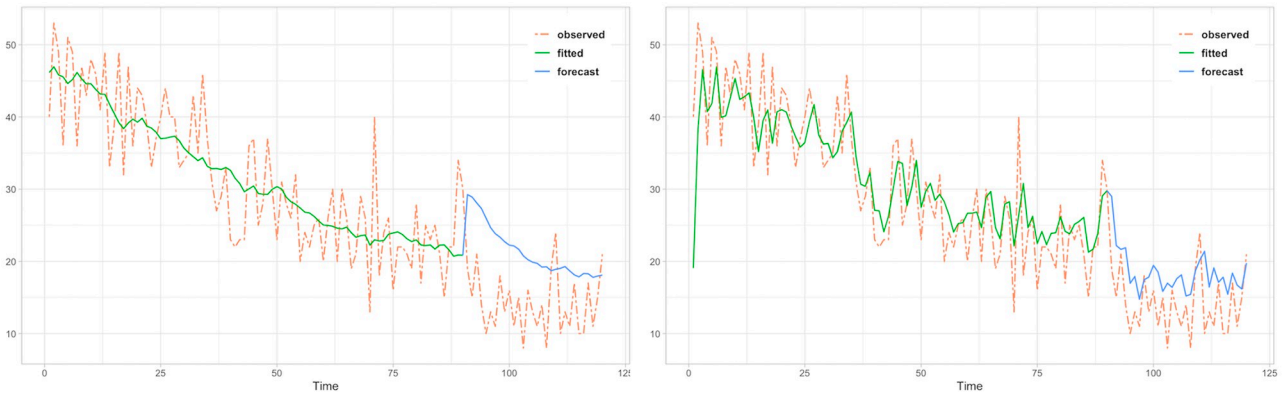


(a) HIP where the coefficient of exogenous event is 0.2

(b) LSTM-Hawkes with Nelder-Mead

Fig 5. HIP with coefficient 0.2 shows the most MSE in HIP test in the 1st section as same as LSTM-Hawkes with SANN does in LSTM-Hawkes Test.

https://doi.org/10.1371/journal.pone.0232887.g005

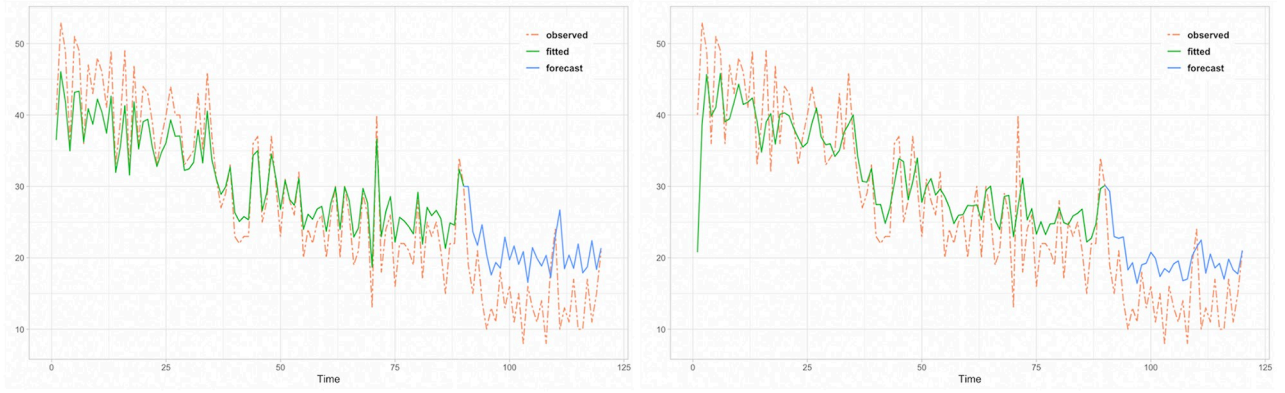


(a) HIP where the coefficient of exogenous event is 0.8

(b) LSTM-Hawkes with Conjugate Gradient Descent

Fig 6. As correlation coefficient gets higher, HIP detects the sudden drop better. However since HIP $\xi(t)$ is the expectation value of $\lambda(t)$ with power-law decay, it tends to follow the trend a little more quadric rather than predict actual intensity value each time unit.

https://doi.org/10.1371/journal.pone.0232887.g006



(a) MLR where the coefficient of exogenous event is 0.2

(b) LSTM-Hawkes of BFGS with Hessian Matrix

Fig 7. MLR shows overfitting issue and it still retains it in the period of prediction.

https://doi.org/10.1371/journal.pone.0232887.g007

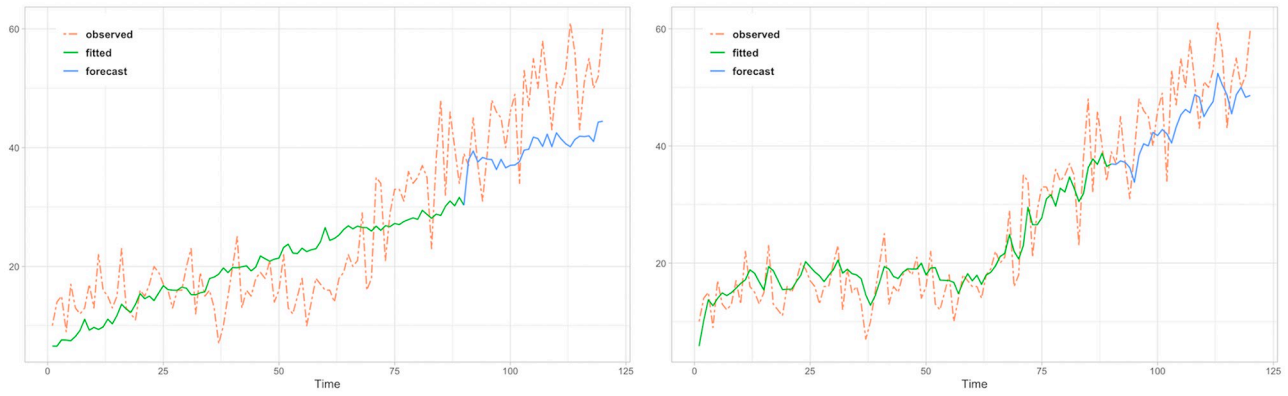
compared to the MLR. Fig 5(a) and 5(b) also shows the difference between the HIP and Hawkes models where a sudden drop occurs at the beginning part of prediction period. In contrast to HIP that does not follow the moving average due to the low correlation coefficient, LSTM-Hawkes follows the moving average closely, greatly reducing residuals. In Fig 6(a), HIP

Table 2. MSE and accuracy.

HIP		MLR	LSTM-Hawkes	
Correlation Coefficient	**MSE**	**MSE**	**Method**	**MSE**
0.20045197016606	272.86493629506	66.489085	Nelder-Mead	55.3646851328
0.40864585590557	167.08929263969	63.931454	BFGS with Hessian Matrix	31.8141777151
0.60923578698778	166.47562141766	62.859002	Conjugate Gradient Descent	44.5680031396
0.80069927612292	75.905022392	62.859002	SANN	47.1453170336
Avg Accuracy: 0.247342286175		Avg Accuracy: 0.4781933	Avg Accuracy: 0.531401427812	

MSE and Accuracy where the trend of moving average is monotonically decreasing.

https://doi.org/10.1371/journal.pone.0232887.t002



(a) HIP where the coefficient of exogenous event is 0.8

(b) LSTM-Hawkes with BFGS

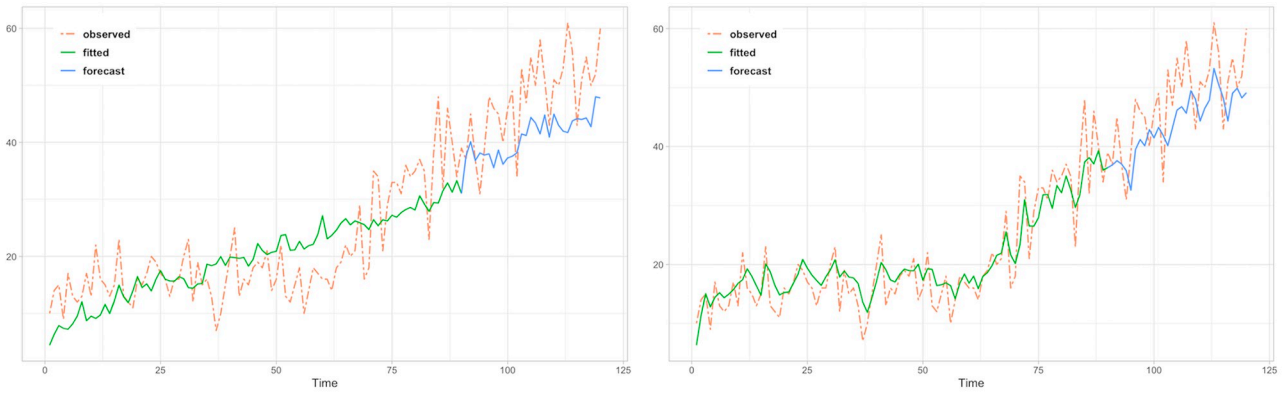
Fig 8. Rather than predictive period, it needs to compare the observed time period more. Since an under-fitted part is found with HIP result, prediction performance is quite noticeable.

https://doi.org/10.1371/journal.pone.0232887.g008

follows the moving average closely with a high coefficient (0.8) of exogenous data but still it's MSE is at least 20 points higher than that of LSTM-Hawkes. MLR also shows great performance in fitting and learning. However this fitting capability eventually causes the overfitting issue and it is proved that LSTM-Hawkes reduces residuals more than MLR in Fig 10. The higher variance of observed data has, the more residuals the feature of MLR produces. Thus, in any case where the dynamics has bigger amplitude or higher frequency, measured MSE of MRL should be greater than the MSE of LSTM-Hawkes.

The forecasting test for the second section shows that LSTM-Hawkes reduces 83.8% of MSE on average and at least 37.5% compared to the HIP where the correlation coefficient is 0.6 as well as 49.39% of MSE on average compared to the MLR. Although both HIP (coefficient 0.6) and LSTM-Hawkes show great performance in prediction as can be seen in Fig 8, between 50 and 70 less-fitted parts are found with HIP while LSTM-Hawkes proves the goodness of fit. This gap is also shown in Fig 7(a) as well. From the tests, we were able to verify that our proposed model of the LSTM-Hawkes significantly outperforms the HIP model in both the fitting and forecasting criteria. As MLR showed the overfitting issue in the first experiment, at this time the MLR produces greater MSE again. Due to higher variance of the observed data of second section, MLR achieves an overfitted model with higher variance. While the compared algorithms show the similar performance in prediction (MLR's fitting capability outperforms HIP), the LSTM-Hawkes outperforms the both compared methods in every experimental case.

Pertaining to the compatibility test with various maximum likelihood estimation algorithms based on both gradient descent method and Newtonian method, we could prove that LSTM-Hawkes has great compatibility with all of the tests. We set up our model for the compatibility test with Nelder-Mead [33], BFGS with Hessian Matrix [34], Conjugate Gradient Descent [35], and Simulated Annealing [36] and have our model optimized by estimating hyper parameters with the maximum likelihood estimation functions mentioned above. Fig 9 shows learning results based on these optimization functions and we could discover that the full interquartile range (IRQ) of MAE is a lot smaller than that of HIP (Fig 11). Also, since our model is not fully dependent with exogenous data to refer, the outlier-points cannot be found. It is expected that compatibility with different optimization algorithms will always be guaranteed.

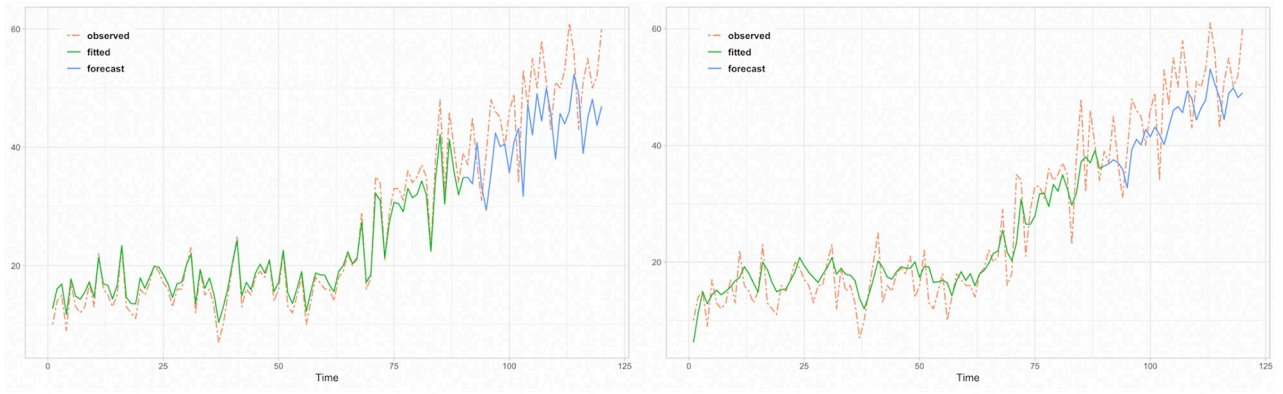


(a) HIP where the coefficient of exogenous event is 0.6

(b) LSTM-Hawkes with SANN

Fig 9. The under-fitted part is still found with HIP where the coefficient of exogenous event is 0.6 while both algorithm perform great in prediction resulting 76.9 and 40.7 MSE respectively.

https://doi.org/10.1371/journal.pone.0232887.g009



(a) MLR where the coefficient of exogenous event is 0.8

(b) LSTM-Hawkes with Conjugate Gradient Descent

Fig 10. MLR shows over fitting problem in the learning period showing higher MSE.

https://doi.org/10.1371/journal.pone.0232887.g010

Lastly, when we assessed the score metric, Eq (15), of the two models, we were able to see that the LSTM-Hawkes model always obtained a greater percentage of negative error score than those of the HIP model and MLR presented in Fig 12. Thus, this always resulted in a lower score metric for the LSTM-Hawkes model, proving it's greater accuracy of prediction and goodness of fit.

Table 3. MSE and accuracy.

HIP		MLR	LSTM-Hawkes	
Correlation Coefficient	**MSE**	**MSE**	**Method**	**MSE**
0.20299594548747	801.44661263499	86.743375	Nelder-Mead	48.0034938190
0.40473704703628	101.32908279598	85.012468	BFGS with Hessian Matrix	41.3346206943
0.60048848461137	76.966689384563	84.804654	Conjugate Gradient Descent	42.6831118624
0.79396502231167	84.072450398227	84.804654	SANN	40.7298560311
Avg Accuracy: 0.741641169307		Avg Accuracy: 0.845652	Avg Accuracy: 0.882077335865	

MSE and Accuracy where the moving average trend is monotonically increasing.

https://doi.org/10.1371/journal.pone.0232887.t003

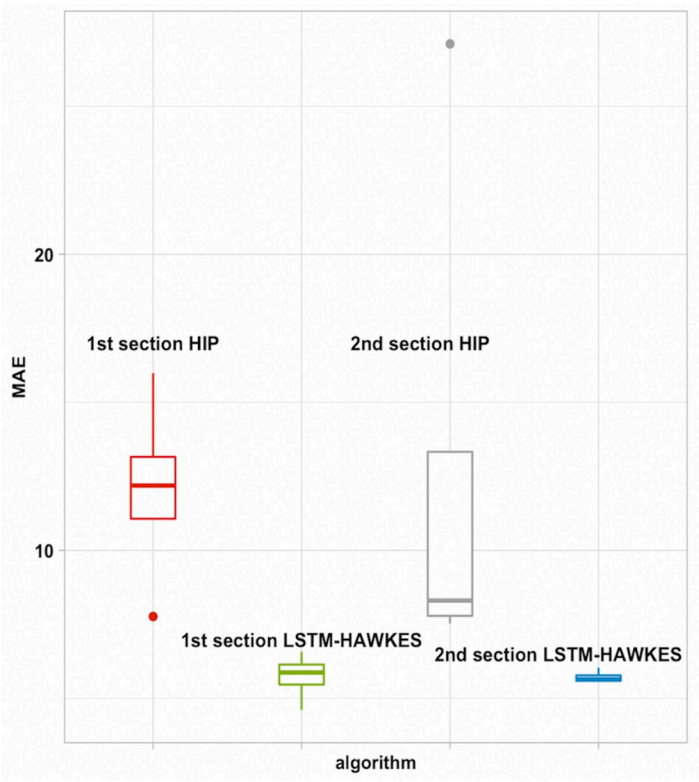
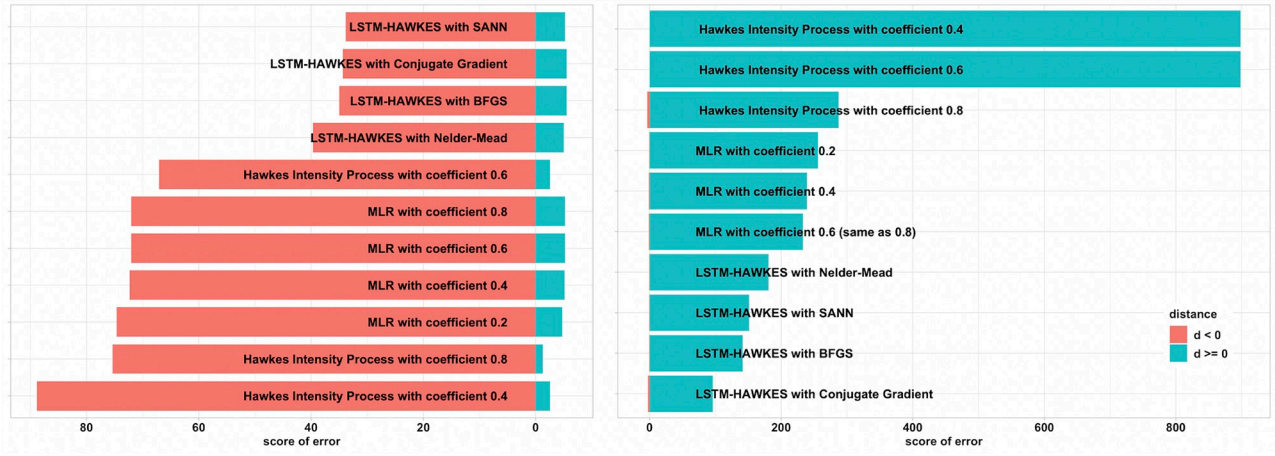


Fig 11. Comparability test. Full quantile range of Mean Absolute Error result of both HIP and LSTM-Hawkes.

https://doi.org/10.1371/journal.pone.0232887.g011



(a) Test Section 2: Increasing Trend

(b) Test Section 1: Decreasing Trend

Fig 12. Under the circumstance that the moving average of ad clicks slopes downward, the HIP gets a higher error score where the distance mark *d* in Eq (15) is negative, and vice versa where the moving average of ad clicks slopes upward. HIP with the exogenous data with coefficient 0.2 is not included in the figure since it's value is outlier, 2348.205812 in the downward trend and -1779.626 in the upward trend.

https://doi.org/10.1371/journal.pone.0232887.g012

Conclusion

Conclusion

In the real production environment, an advertising network forecasting application does require the prediction results within milliseconds or even microseconds [37]. Therefore, the approach to the short interval prediction using Neural Network is not just scientifically important but also practically required since it does take great advantages in fast forecasting. Plus, because Algorithm (1), (2) are able to be performed in parallel or con-currently while Algorithm (3) is being conducted, the actual processing time in the production environment would expect to be even shorter. Thus, the LSTM-Hawkes using stochastic based generative model always guarantees the time efficiency.

Discussion

These days, to tract more consumers, advertisers strongly focus on conversion events which actually create sales. Thus, the percentage of mobile advertising bidding based on conversion events increases. However, since conversion events rarely occurs, return Conversion Ratio (CVR) predicted based on a historical dataset is a great challenge for existing models including the Hawkes and other Neural Network models. However, different from the original mathematical definition of the kernel, we did not suppose that limitation of Δt goes to zero. Therefore, we can easily expand the application of the model for the longer interval forecast based on bigger time unit. In future research, we plan to develop and implement a CVR prediction model, expanding to LSTM-Hawkes, which overcomes the data sparsity problem.

Supporting information

S1 Dataset. To help any of researchers to reproduce the result, we provide the codes of the model. Having the original data sets that can be found at http://labs.criteo.com/wp-content/uploads/2014/07/criteoconversionlogs.tar.gz, we first preprocess the data to conform behavior model (Fig 4) which can be found from the supporting information attachment. Then, we have processed Algorithm 1 to 4 in order to achieve the experimental results which also can be found from the supporting information attachment.
(ZIP)

S1 File.
(R)

Author Contributions

Conceptualization: Sangwon Hwang, Inwhee Joe.

Data curation: Sangwon Hwang.

Formal analysis: Sangwon Hwang.

Investigation: Sangwon Hwang, Inwhee Joe.

Methodology: Sangwon Hwang.

Project administration: Sangwon Hwang, Inwhee Joe.

Resources: Sangwon Hwang.

Software: Sangwon Hwang.

Supervision: Inwhee Joe.

Validation: Sangwon Hwang.

Visualization: Sangwon Hwang.

Writing – original draft: Sangwon Hwang.

Writing – review & editing: Sangwon Hwang, Inwhee Joe.

References

1. Advertising Expenditure Forecasts. Zenith Media report 2018.
2. Lee Heejun, and Chang-Hoan Cho. Digital advertising: present and future prospects. International Journal of Advertising (2019): 1–10.
3. Wong Choy-Har, et al. Mobile advertising: the changing landscape of the advertising industry. Telematics and Informatics 32.4 (2015): 720–734. https://doi.org/10.1016/j.tele.2015.03.003
4. Chowdhury, Humayun Kabir. Consumer attitude toward mobile advertising in an emerging market: An empirical study. International Journal of Mobile Marketing 1.2 (2006).
5. REGULATION (EU) 2016/679 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL.
6. Han, Min Ho. Retargeting advertising product recommending user device and service providing device, advertising product recommending system including the same, control method thereof, and non–transitory computer readable storage medium having computer program recorded thereon. U.S. Patent Application No. 15/320,632.
7. Cheng, Haibin, and Erick Cantú-Paz. "Personalized click prediction in sponsored search." Proceedings of the third ACM international conference on Web search and data mining. 2010.
8. Bilenko, Mikhail, and Matthew Richardson. "Predictive client-side profiles for personalized advertising." Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining. 2011.
9. Juan, Yuchin. Field-aware factorization machines for CTR prediction. Proceedings of the 10th ACM Conference on Recommender Systems. ACM, 2016.
10. Rendle, Steffen, et al. "Fast context-aware recommendations with factorization machines." Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval. 2011.
11. Pan, Junwei, et al. "Field-weighted factorization machines for click-through rate prediction in display advertising." Proceedings of the 2018 World Wide Web Conference. 2018.
12. Cross-platform HTTP debugging proxy server application. https://www.charlesproxy.com/overview/about-charles.
13. Bonnefoi, Rémi, Christophe Moy, and Jacques Palicot. "Improvement of the LPWAN AMI backhaul's latency thanks to reinforcement learning algorithms." EURASIP Journal on Wireless Communications and Networking 2018.1 (2018): 34.
14. Boubchir, Larbi, Somaya Al-Maadeed, and Ahmed Bouridane. "Undecimated wavelet-based Bayesian denoising in mixed Poisson-Gaussian noise with application on medical and biological images." 2014 4th International Conference on Image Processing Theory, Tools and Applications (IPTA). IEEE, 2014.
15. Pérez Patrick, Michel Gangnet, and Andrew Blake. "Poisson image editing." ACM Transactions on graphics (TOG) 22.3 (2003): 313'318. https://doi.org/10.1145/882262.882269
16. Feller, W. (1971) Introduction to Probability Theory and Its Applications, Vol II (2nd edition),Wiley. Section I.3 ISBN 0-471-25709-5.
17. Hawkes Alan G. Spectra of some self-exciting and mutually exciting point processes functions. Biometrika 58.1 (1971): 83–90.
18. Patrick J Laub, Thomas Taimre, and Philip K Pollett. "Hawkes processes." arXiv preprint arXiv:1507.02822, 2015.
19. Hochreiter Sepp, and Jürgen Schmidhuber. Long short-term memory. Neural computation 9.8 (1997): 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735 PMID: 9377276
20. Ogata Yosihiko. On Lewis' simulation method for point processes. IEEE Transactions on Information Theory 27.1 (1981): 23–31. https://doi.org/10.1109/TIT.1981.1056305
21. Rizoiu, Marian-Andrei. Expecting to be hip: Hawkes intensity processes for social media popularity. Proceedings of the 26th International Conference on World Wide Web. International World Wide Web Conferences Steering Committee, 2017.

22. Kwak, Haewoon. What is Twitter, a social network or a news media?. Proceedings of the 19th international conference on World wide web. AcM, 2010.

23. Ahn, Yong-Yeol. Analysis of topological characteristics of huge online social networking services. Proceedings of the 16th international conference on World Wide Web. ACM, 2007.

24. Ozaki, Tohru. Maximum likelihood estimation of Hawkes' self-exciting point processes. Annals of the Institute of Statistical Mathematics 31.1 (1979): 145–155.

25. Rasmussen, Jakob Gulddahl. Temporal point processes: the conditional intensity function. Lecture Notes, Jan (2011).

26. Rubin Izhak. Regular point processes and their detection. IEEE Transactions on Information Theory 18.5 (1972): 547–557. https://doi.org/10.1109/TIT.1972.1054897

27. Almeida Aitor, and Gorka Azkune. "Predicting human behaviour with recurrent neural networks." Applied Sciences 8.2 (2018): 305. https://doi.org/10.3390/app8020305

28. Liu Yuxin, et al. "A statistical approach to participant selection in location-based social networks for offline event marketing." Information Sciences 480 (2019): 90–108. https://doi.org/10.1016/j.ins.2018.12.028

29. Chapelle, Olivier. "Modeling delayed feedback in display advertising." Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2014.

30. Lorenzen, F. Analysis of order clustering using high frequency data: A point process approach. Working Paper, 2012.

31. Rizoiu, Marian-Andrei, and Lexing Xie Xie. "Online popularity under promotion: Viral potential, forecasting, and the economics of time." Eleventh International AAAI Conference on Web and Social Media. 2017.

32. Saxena, Abhinav, et al. "Damage propagation modeling for aircraft engine run-to-failure simulation." 2008 international conference on prognostics and health management. IEEE, 2008.

33. McKinnon Ken IM. "Convergence of the Nelder–Mead Simplex Method to a Nonstationary Point." SIAM Journal on Optimization 9.1 (1998): 148–158. https://doi.org/10.1137/S1052623496303482

34. Berahas, Albert S., Jorge Nocedal, and Martin Takác. "A multi-batch L-BFGS method for machine learning." Advances in Neural Information Processing Systems. 2016.

35. Hestenes Magnus Rudolph, and Eduard Stiefel. Methods of conjugate gradients for solving linear systems. Vol. 49. No. 1. Washington, DC: NBS, 1952.

36. Granville Vincent, Mirko Krivánek, and Rasson J-P. "Simulated annealing: A proof of convergence." IEEE transactions on pattern analysis and machine intelligence 16.6 (1994): 652–656. https://doi.org/10.1109/34.295910

37. Jiang Zilong, Shu Gao, and Mingjiang Li. "An improved advertising CTR prediction approach based on the fuzzy deep neural network." PloS one 13.5 (2018). https://doi.org/10.1371/journal.pone.0190831