Non-linear Time-series Analysis of Social Influence

Thinh Minh Do
Kumamoto University
do@dm.cs.kumamoto-u.ac.jp
Supervised by Yasushi Sakurai
Expected graduation date: 31st March, 2020

ABSTRACT

In this paper, we present Δ-SPOT, a non-linear model for analysing large scale web search data, and its fitting algorithm. Δ-SPOT can forecast long-range future dynamics of the keywords/queries. We use the Google Search, Twitter and MemeTracker data set for extensive experiments, which show that our method outperforms other non-linear mining methods. We also provide an online algorithm contributing to the need of monitoring multiple co-evolving data sequences.

Categories and Subject Descriptors: H.2.8 [**Social Networks and Graph Analysis**]: Database applications–*Data mining*

Keywords: Time-series Analysis; Social Influence Analysis; Parameter-free;

1. INTRODUCTION

Our goal is to detect patterns, rules and outliers in a huge set of web search data, consisting of tuples of the form: *(keyword, location, time)*. How can we find meaningful information, such as the external events? Do those events have any relationship between each others (cyclic events or not?) Also, can we detect global/local-level patterns? Are there countries that react differently from the global trend? Can we forecast the dynamics of future events? Thus, the most fundamental requirement is the efficient monitoring of the data sequences. In this paper, we propose Δ-SPOT, which is sense-making, automatic, scalable and parameter-free, and provides a good summary of large collections of local online activities to solve the following problem automatically and effectively:

Informal problem. Given a large collection of online activities, which consists of d keywords in l locations of duration n with missing values and external shocks, we want to detect external shocks (important events in reality), find global and local patterns, and forecast future activities.

Besides, we also introduce an incremental online algorithm, Δ-STREAM, which enhance the application of Δ-SPOT for online analysis.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

SIGMOD'16 PhD Symp, June 25-July 01 2016, San Francisco, CA, USA

© 2016 ACM. ISBN 978-1-4503-4192-9/16/06…$15.00

DOI: http://dx.doi.org/10.1145/2926693.2929902

Table 1: Capabilities of approaches. Only our approach meets all specifications.

	SI/ ++	AR/ ++	FUNNEL	Δ-SPOT/ Δ-STREAM
Non-linear	√		√	√
Outliers detection			√	√
Online activities				√
Cyclic events/shocks				√
Local analysis			√	√
Parameter-free			√	√
Forecasting		√	√	√
Online processing				√

2. RELATED WORK

Pattern discovery in time series. In recent years, there has been a huge interest in mining time-stamped data [9, 4]. Traditional approaches typically use linear methods, such as auto-regression (AR), linear dynamical systems (LDS), and their variants [11, 2, 3]. TriMine [6] is a scalable method for forecasting complex time-stamped events, while, [5] developed AutoPlait, which is a fully-automatic mining algorithm for co-evolving sequences.

Social activity analysis. The work described in [7] studied the rise and fall patterns in the information diffusion process through online social media. FUNNEL [8] is a non-linear model for spatially co-evolving epidemic tensors.

Contrast to the competitors. Table 1 illustrates the relative advantages of our method. Only our Δ-SPOT matches all requirements.

3. PROPOSED MODEL

3.1 Intuition behind our method

We have a collection of sequences with d unique keywords, l countries with duration n. We can treat this set of $d \times l$ sequences as a 3rd-order tensor, i.e., $\mathcal{X} \in \mathbb{N}^{d \times l \times n}$, where the element $x_{ij}(t)$ of $\mathcal{X}$ shows the total number of entries of the i-th keyword in the j-th country at time-tick t. For example, ('Harry Potter', 'US', 'July 15-21, 2007', 100), means that the search volume for 'Harry Potter' in 'US' on 'July 15-21 in 2007' is '100'. We refer to each sequence of the i-th keyword in the j-th location: $\boldsymbol{x}_{ij} = \{x_{ij}(t)\}_{t=1}^{n}$, as a "local/country"-level web search sequence. Similarly, we can turn these local sequences into "global/world"-level web search sequences: $\bar{\boldsymbol{x}}_i = \{\bar{x}_i(t)\}_{t=1}^{n}$, where $\bar{x}_i(t)$ shows the total count of the i-th keyword at time-tick t, i.e., $\bar{x}_i(t) = \sum_{j=1}^{l} x_{ij}(t)$. (A count is defined as the activity of searching for the keyword via Google Search in a specific location and time period.)

3.2 Δ-SPOT - with a single sequence

The model we propose has nodes (=users) of three classes. Those are **S**usceptible: nodes in this class can get influ-

Table 2: Symbols and definitions

Symbol	Definition
d	Number of keywords/queries
l	Number of locations/countries
n	Duration of sequences
$\mathcal{X}$	3rd-order tensor ($\mathcal{X} \in \mathbb{N}^{d \times l \times n}$)
$\boldsymbol{x}_{ij}$	Local-level sequence of keyword i in location j i.e., $\boldsymbol{x}_{ij} = \{x_{ij}(t)\}_{t=1}^{n}$
$\bar{\boldsymbol{x}}_i$	Global-level sequence of keyword i i.e., $\bar{\boldsymbol{x}}_i = \sum_{j=1}^{l} \boldsymbol{x}_{ij}$
$S_{ij}(t)$	Count of (S)usceptibles i in location j at time t
$I_{ij}(t)$	Count of (I)nfectives i in location j at time t
$V_{ij}(t)$	Count of (V)igilants i in location j at time t
$\mathbf{B_G}$	Base global matrix ($d \times 4$)
$\mathbf{B_L}$	Base local matrix ($d \times l$)
$\mathbf{R_G}$	Growth effect global matrix ($d \times 2$)
$\mathbf{R_L}$	Growth effect local matrix ($d \times l$)
$\mathcal{S}$	External shock tensor i.e., $\mathcal{S} = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_k\}$
$\mathcal{F}$	Complete set of Δ-SPOT i.e., $\mathcal{F} = \{\mathbf{B_G}, \mathbf{B_L}, \mathbf{R_G}, \mathbf{R_L}, \mathcal{S}\}$

enced by their neighboring nodes who have searched for it ; **I**nfective: nodes who already searched for the keywords, also capable of influencing other available nodes; and **V**igilant (i.e., busy/unavailable): nodes in this class are immune to the influence of the trend.

Figure 1 shows a diagram of our model. Here, β represents the rate of effective contacts between citizens in infective and susceptible classes; δ is the rate at which infected citizens lost interest in the topic and stop searching for it; and γ is the immunization loss probability for a change in status: being ready to search for the topic. $\epsilon(t)$ and $\eta(t)$ represent the external shock effect and growth effect. The number of the susceptible class $S(t)$ is the count of users available for infection, and if there is an external shock event , the infection becomes stronger than usual. Each infective pair would lead to a new infective citizen, and will eventually cause a major spike. With respect to the growth effect, it starts at time t_η and make the number of infectives rise quickly to a new base.

MODEL 1 (Δ-SPOT-SINGLE). *Our model is described by the following equations:*

$$\begin{aligned} S(t+1) = \;& S(t) - \beta S(t)\epsilon(t)I(t)(1+\eta(t)) + \gamma V(t) \\ I(t+1) = \;& I(t) + \beta S(t)\epsilon(t)I(t)(1+\eta(t)) - \delta I(t) \\ V(t+1) = \;& V(t) + \delta I(t) - \gamma V(t) \end{aligned} \quad (1)$$

The growth effect $\eta(t) = \begin{cases} 0 & (t < t_\eta) \\ \eta_0 & (t \geq t_\eta) \end{cases}$

In addition, we introduce the temporal susceptible rate $\epsilon(t) = 1 + \sum_{i=1}^{k} f(t; \boldsymbol{s}_i)$, *and*

$$f(t; \boldsymbol{s}) = \begin{cases} \epsilon_0 & (t_s + t_p\lceil t/t_p \rceil < t < t_s + t_p\lceil t/t_p \rceil + t_w) \\ 0 & (else) \end{cases}$$

where, k is the number of shocks. If k = 0, then $\epsilon(t) = 1$.

Each external shock consists of $\boldsymbol{s} = \{t_p, t_s, t_w, \epsilon_0\}$, i.e.,

- t_p: Periodicity of the event (if $t_p = \infty$, the event is non-cyclic).
- t_s: Starting point of the event.
- t_w: Duration of the event.
- ϵ_0: Strength of the external shock.

3.3 Δ-SPOT - with multi-evolving sequences

We extract important patterns with respect to the following separated aspects: (**P1**), (**P2**): base properties of global and local dynamics $\mathbf{B_G}$, $\mathbf{B_L}$; (**P3**): the sudden change of popularity (if any) $\mathbf{R_G}$, $\mathbf{R_L}$; and (**P4**): external shock events (cyclic and non-cyclic) $\mathcal{S}$ (see Figure 2).

(P1) Base trends and global influence. Basically, we assume that the following parameters are the same for all

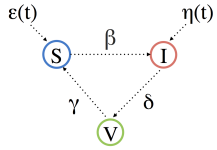


Figure 1: Δ-SPOT diagrams: classes of population: susceptibles, infectives, and vigilants.

l countries. Here, $\mathbf{B_G}$ is the set of global parameters of d keywords/queries, where $\{N_i, \beta_i, \delta_i, \gamma_i\}$ is the parameter set of the i-th keyword.

(P2) Area specificity. We share the parameters of the global-level matrices for all l countries. with one exception, N_{ij}, which describes the total population of users i in the j-th country. Specifically, we set the invariant, $N_{ij} = S_{ij}(t) + I_{ij}(t) + V_{ij}(t)$. Here, $\mathbf{B_L}$ is a parameter set of the potential population of d keywords and l countries, i.e., $\mathbf{B_L} = \{b^{(L)}{}_{ij}\}_{i,j=1}^{d,l}$, where $b^{(L)}{}_{ij}$ is the potential population of susceptibles of the i-th keyword in the j-th country.

(P3) Population growth effect. The growth effect appears due to the launch of new products and services that raise the interest of users, which should have the same starting time all over the world. Here, $\mathbf{R_G}$ is the set of global growth effect parameters of d keywords, where $\{\eta_{0_i}, t_{\eta_i}\}$ is the parameter set of the i-th keyword. $\mathbf{R_L}$ is a parameter set of the potential population of d keywords and l countries, i.e., $\mathbf{R_L} = \{r^{(L)}{}_{ij}\}_{i,j=1}^{d,l}$, where $r^{(L)}{}_{ij}$ is the population growth rate of the i-th keyword in the j-th country.

(P4) External shock events. To describe each external shock event, we create a new parameter set, namely external shock tensor $\mathcal{S}$, which consists of a set of k external shock events, as described in Figure 2 (b). A single external shock event $\mathbf{s}$ can be described as three components: $\mathbf{s} = \{\mathbf{s^{(D)}}, \mathbf{s^{(N)}}, \mathbf{s^{(L)}}\}$: $\mathbf{s^{(D)}}$ represents the external view for d keywords; $\mathbf{s^{(N)}}$ describes the periodicity (t_p), the starting time (t_s), and the duration (t_w) of the external event; and $\mathbf{s^{(L)}}$ expresses the strength of the external shocks of one event in l countries, where $\lceil n/t_p \rceil$ is the number of shocks belonging to that event.

Figure 3 compares the global fitting results of the keyword "Amazon", in four different cases to demonstrate the influence of the growth effect (**P3**) and external shocks (**P4**). The result shows the benefit of treating the growth effect differently from external shock effect, as well as combining these two effects to achieve good fitting results.

4. ALGORITHM

4.1 Model quality and data compression

We apply the minimum description length (MDL) principle to find an optimal representation $\mathcal{F}$.

Model description cost. The total code length for $\mathcal{X}$ with respect to a given parameter set $\mathcal{F}$ can be described in the following equation, which we want to minimize:

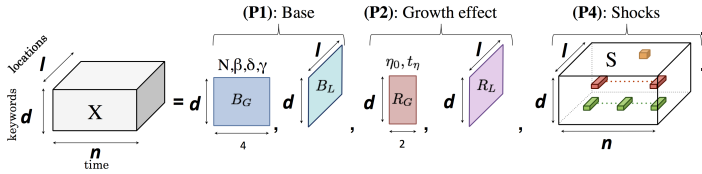
$$\begin{aligned} & Cost_T(\mathcal{X}; \mathcal{F}) = \log^*(d) + \log^*(l) + \log^*(n) \\ & + Cost_M(\mathbf{B_G}) + Cost_M(\mathbf{B_L}) + Cost_M(\mathbf{R_G}) \\ & + Cost_M(\mathbf{R_L}) + Cost_M(\mathcal{S}) + Cost_C(\mathcal{X}|\mathcal{F}) \end{aligned} \quad (2)$$

4.2 Multi-layer optimization

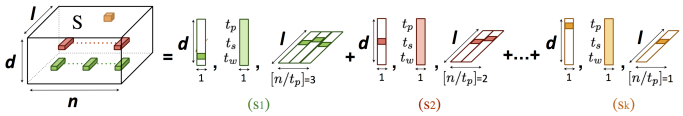
Algorithm 1 shows an overview of Δ-SPOT to find the full set of Δ-SPOT parameters given a tensor $\mathcal{X}$.

4.2.1 Global-level parameter fitting

Given a tensor $\mathcal{X}$, our sub-goal is to find the optimal global-level parameter set: $\mathcal{F}_\mathcal{G}$, to minimize the cost function (i.e., Equation 2). As shown in Algorithm 2, we pro-

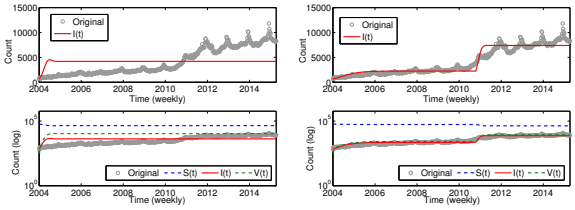


(a) Δ-SPOT structure (i.e., $\mathcal{F} = \{\mathbf{B_G}, \mathbf{B_L}, \mathbf{R_G}, \mathbf{R_L}, \mathcal{S}\}$)



(b) External shock tensor (i.e., $\mathcal{S} = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_k\}$)

Figure 2: Δ-SPOT structure: (a) important properties extracted from tensor $\mathcal{X}$. Also, (b) external shock tensor $\mathcal{S}$ consists of a set of k components.



(a) None (b) (**P3**)

(c) (**P4**) (d) (**P3**) + (**P4**)

Figure 3: Influence of combining growth effect and external shock effect: compared with the case of using (a) none of above, (b) only growth effect, (c) only external shock effect, and (d) the combination of both effects. Clearly, (d) fits the data very well.

vide a detailed algorithm of the global-level fitting. Given a tensor $\mathcal{X}$, it creates a set of d global sequences: $\{\bar{\boldsymbol{x}}_i\}_{i=1}^{d}$. The goal is to fit the global-level parameter set, as well as find the appropriate number of external-shocks. We apply the *Levenberg-Marquardt (LM)* [1] algorithm to minimize the cost function. Note that the extra tensor $\mathcal{S}$ consists of k entries $\{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_k\}$, Algorithm 2 can find only the global-level entry, which consists of $(keyword, time)$. We will introduce the local-level parameter fitting algorithm in Algorithm 3, to describe how the local-level entries can be computed. Also, the cost function (2) includes the cost of local-level parameters such as $\mathbf{B_L}$, $\mathbf{R_L}$ but these terms are independent of the global model fitting. Hence, we can simply consider them to be constant.

4.2.2 *Local-level parameter fitting*

Given a set of $d \times l$ local-level sequences, $\{\boldsymbol{x}_{ij}\}_{i,j=1}^{d,l} \in \mathcal{X}$, and a set of global-level parameters, $\mathcal{F}_\mathcal{G}$, our next goal is to fit the individual parameters of each disease in each state, that is, $\mathcal{F}_\mathcal{L} = \{\mathbf{B_L}, \mathcal{S}\}$. We propose an iterative optimization algorithm (see Algorithm 3). Our algorithm searches for the optimal solution with respect to (a) the base local matrix $\mathbf{B_L}$ and (b) the local-level external shocks $\mathcal{S}$, so that the total coding cost is minimized.

Algorithm 1 Δ-SPOT($\mathcal{X}$)

Input: Tensor $\mathcal{X}$ ($d \times l \times n$)
Output: Full parameters, i.e., $\mathcal{F} = \{\mathbf{B_G}, \mathbf{B_L}, \mathbf{R_G}, \mathbf{R_L}, \mathcal{S}\}$
$\{\mathcal{F}_\mathcal{G}\}$ =GlobalFit ($\mathcal{X}$); /* Global-level parameter fitting */
$\{\mathcal{F}_\mathcal{L}\}$ =LocalFit ($\mathcal{X}, \mathcal{F}_\mathcal{G}$); /* Local-level parameter fitting */
return $\mathcal{F} = \{\mathcal{F}_\mathcal{G}, \mathcal{F}_\mathcal{L}\}$;

Algorithm 2 GlobalFit($\mathcal{X}$)

Input: Tensor $\mathcal{X}$
Output: Set of global-level parameters $\mathcal{F}_\mathcal{G}$
i.e., $\mathcal{F}_\mathcal{G} = \{\mathbf{B_G}, \mathbf{R_G}, \mathcal{S}\}$
for $i = 1 : d$ **do**
  Create $\bar{\boldsymbol{x}}_i$ from $\mathcal{X}$; /* Global sequence $\bar{\boldsymbol{x}}_i$ of i-th keyword */
  /* Initialize external shocks for keyword i */
  $\mathbf{s}_i = \emptyset$;
  while improving the cost **do**
    $\mathbf{b^{(G)}}_\mathbf{i} = \arg\min_{\mathbf{b^{(G)}}'_\mathbf{i}} \mathbf{Cost_C}(\bar{\mathbf{x}}_\mathbf{i} | \mathbf{b^{(G)}}'_\mathbf{i}, \mathbf{r^{(G)}}_\mathbf{i}, \mathbf{s_i})$; /* Base */
    $\mathbf{r^{(G)}}_\mathbf{i} = \arg\min_{\mathbf{r^{(G)}}'_\mathbf{i}} \mathbf{Cost_C}(\bar{\mathbf{x}}_\mathbf{i} | \mathbf{b^{(G)}}_\mathbf{i}, \mathbf{r^{(G)}}'_\mathbf{i}, \mathbf{s_i})$; /* Growth */
    $\mathbf{s}_i = \emptyset$; /* Initialize values */
    /* Find external shocks for keyword i */
    while improving the cost **do**
      $\mathbf{s} = \arg\min_{\mathbf{s}'} Cost_C(\bar{\boldsymbol{x}}_i | \mathbf{b^{(G)}}_\mathbf{i}, \mathbf{r^{(G)}}_\mathbf{i}, \{\mathbf{s_i} \cup \mathbf{s}'\})$;
      $\mathbf{s}_i = \mathbf{s}_i \cup \mathbf{s}$;
    end while
  end while
  /* Update parameter set of i-th keyword */
  $\mathbf{B_G} = \mathbf{B_G} \cup \mathbf{b^{(G)}}_\mathbf{i}$; $\mathbf{R_G} = \mathbf{R_G} \cup \mathbf{r^{(G)}}_\mathbf{i}$; $\mathcal{S} = \mathcal{S} \cup \mathbf{s}_i$;
end for
return $\mathcal{F}_\mathcal{G} = \{\mathbf{B_G}, \mathbf{R_G}, \mathcal{S}\}$;

5. ONLINE PROCESSING

Algorithm 4 shows an overview of Δ-STREAM. Given a new tensor $\mathcal{X}'$, our first task is to find the appropriate parameter set in both global level ($\mathcal{F}_\mathcal{G}'$), and local level ($\mathcal{F}_\mathcal{L}'$), thus we use them to update the original sequence's global parameter set $\mathcal{F}_\mathcal{G}$ and local parameter set $\mathcal{F}_\mathcal{L}$. As we described in Section 4, the newly captured external shock tensor $\mathcal{S}'$ includes both global and local shocks. The challenge here is to synchronize them to the external shock events of the old sequence including the cyclic events. We create a new parameter, the cyclic external shock candidate set $\mathcal{C}$ to further reduce the processing time. The candidate set includes multiple optimal cyclic shocks with different period, time-shift and duration. If a new captured spike forms with the old ones a potential cyclic shock (with specified period and duration), a new candidate is added to $\mathcal{C}$. When dealing with a new sequence, the next spike of the cyclic shock is automatically generated, and fit with its strength (height).

6. EXPERIMENTS

6.1 Sense-making

We demonstrate the global fitting results of three datasets: Figure 4 shows the results of model fitting on 2 trending keywords in various categories; Figure 5 (a) shows the results the popular hashtags "#apple"; and Figure 5 (b) shows the results of Meme#3: "yes we can yes we can" from the *MemeTracker* dataset. We show the original sequences (i.e., black dots) and estimated sequences: $I(t)$ (i.e., the Infectives, in red line) in linear-linear scales. Also, we made several important observations, which correspond to the properties mentioned above.

(P1) Base trends and global influence. As shown in Figure 4, Δ-SPOT successfully captures long-range nonlinear dynamics of user activities, as well as fit the data

Algorithm 3 LocalFit($\mathcal{X}, \mathbf{B_G}, \mathbf{R_G}, \mathcal{S}$)

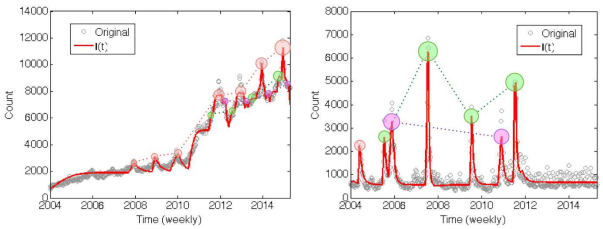
Input: (a) Tensor $\mathcal{X}$, (b) global-level parameter set $\mathcal{F_G}$
Output: Set of local-level parameters, i.e., $\mathcal{F_L}$
while improving the cost **do**
  /* For each local sequence $\boldsymbol{x}_{ij}$ (i-th keyword, j-th country) */
  for $i = 1 : n$ **do**
    for $j = 1 : l$ **do**
      $b^{(L)}{}_{ij} = \arg\min_{b^{(L)}{}'_{ij}} Cost_C(\boldsymbol{x}_{ij} | \mathbf{B_G}, \mathbf{R_G}, b^{(L)}{}'_{ij}, \mathcal{S})$;
      $r^{(L)}{}_{ij} = \arg\min_{r^{(L)}{}'_{ij}} Cost_C(\boldsymbol{x}_{ij} | \mathbf{B_G}, \mathbf{R_G}, r^{(L)}{}'_{ij}, \mathcal{S})$;
    end for
  end for
  for each external shock **s** in $\mathcal{S}$ **do**
    Update **s** to minimize the cost
  end for
end while
return $\mathcal{F_L} = \{\mathbf{B_L}, \mathbf{B_G}, \mathcal{S}\}$;

Algorithm 4 Δ-STREAM($\mathcal{X}', \mathcal{C}$)

Input: A subsequence $\mathcal{X}'$ ($d \times l \times n'$) of duration $n_{\mathcal{X}'}$, and a set of cyclic shock candidates $\mathcal{C}$
Output: An update of parameter set, i.e., $\mathcal{F} = \{\mathbf{B_G}, \mathbf{B_L}, \mathbf{R_G}, \mathbf{R_L}, \mathcal{S}\}$
/* Parameter fitting for global-level sequences */
$\{\mathcal{F_G}'\}$ =GlobalFit ($\mathcal{X}'$);
/* Parameter fitting for local-level sequences */
$\{\mathcal{F_L}'\}$ =LocalFit ($\mathcal{X}', \mathcal{F_G}'$);
for every candidate in $\mathcal{C}$ **do**
  Fit the shocks strength in $\mathcal{S}'$
end for
/* Update the external shocks tensor */
$\{\mathcal{S}\} = \{\mathcal{S}\} \cup \{\mathcal{S}'\}$;
/* Update new candidates */
for every shock in $\mathcal{S}$ **do**
  if there exists a new cyclic event $\mathbf{s}_0$ **then**
    Add a new candidate to $\mathcal{C}$
  end if
end for
/* Update the global parameter set */
$\{\mathcal{F_G}\} = \{\mathcal{F_G}\} \cup \{\mathcal{F_G}'\}$;
/* Update the global parameter set */
$\{\mathcal{F_L}\} = \{\mathcal{F_L}\} \cup \{\mathcal{F_L}'\}$;
return $\mathcal{F} = \{\mathcal{F_G}, \mathcal{F_L}\}$;

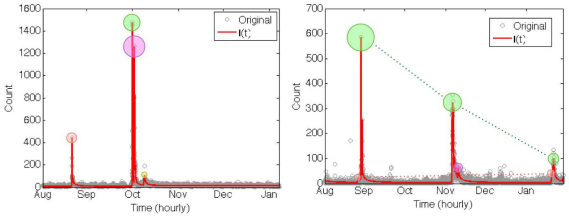
sequences in high accuracy. **(P2) Area specificity.** For example, Figure 6 (a) shows the local fitting results for keyword "Ebola". We detected some countries (GB,US,JP) that behave similarly to the global trend (i.e., the world reaction to the burst of Ebola Virus in 2014, shown in green circles). Besides, we also detected several outlier countries which have less capacities of network connection (LA,NP). **(P3) Population growth effect.** In Figure 4 (a), our model can detect the population growth effect, which is treated separatedly from the external shock effect. **(P4) External shock events.** Δ-SPOT can capture important external events relating to the keywords, including the cyclic events.

Moreover, we execute the online process experiments to evaluate the fitting capacity of Δ-STREAM. For each stage of the process, we input a subsequence, applying the fitting process in both linear and log scale. Then we synchronize the new parameter set with the last one. Similarly, we made some observations to confirm the quality of the data stream monitoring algorithm. Figure 7 (a) shows the monitoring result for keyword "Amazon". We set the window size of one-year-length (i.e, $wd = 52$ time-ticks). The monitoring algorithm, Δ-STREAM can capture the correct increasing pattern of the web search data stream, as well as detect the annual external events relating to the keyword. In Figure 7 (b)-(c), we set the window size of one-week-length (i.e, $wd =$



(a) "Amazon" (**P1**),(**P2**),(**P3**),(**P4**) (b) "Harry Potter" (**P1**),(**P2**),(**P4**)

Figure 4: Global fitting results for 2 keywords in *GoogleTrends* dataset



(a) "#apple" (**P1**),(**P2**),(**P4**) (b) Meme#3 (**P1**),(**P2**),(**P4**)

Figure 5: Global fitting results for (a) *Twitter* and (b) *MemeTracker* dataset.

168 time-ticks). Our method can capture the basic trend of the data stream, and some external shocks during the scan.

6.2 Accuracy

We compared Δ-SPOT with the standard *SIRS* model, *SKIPS* [10], and *FUNNEL* [8]. Figure 8 (a) shows the root-mean-square error (RMSE) between the original and estimated counts of the global sequences $\{\bar{\boldsymbol{x}}_i(t)\}_{i,t}^{d,n}$. Similarly, Figure 8 (b) shows the results of the local counts $\{\boldsymbol{x}_{ij}(t)\}_{i,j,t}^{d,l,n}$, (i.e., each keyword in each country, at each time-tick). A lower value indicates a better fitting accuracy.

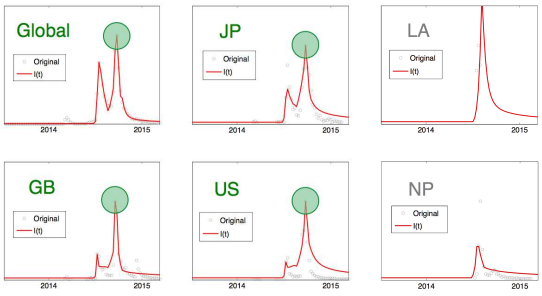
We also evaluated the accuracy of Δ-STREAM in terms of global/local fitting. Δ-STREAM still provides better fitting accuracy compared to the *SIRS* model, *SKIPS* and *FUNNEL*, while being close to the offline Δ-SPOT.

6.3 Scalability

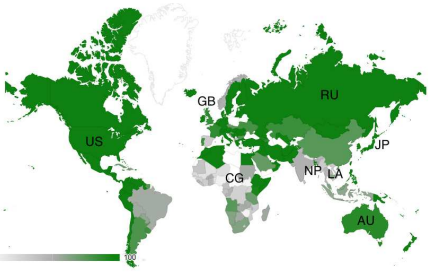
We varied the dataset size with respect to (a) keywords d, (b) countries l, and (c) duration n. Figure 9 shows the computational cost of Δ-SPOT in terms of the dataset size: Δ-SPOT is linear with respect to data size. More importantly, our proposed online streaming method, Δ-STREAM achieves a dramatic reduction in computational time: it requires constant; i.e., it does not depend on d, l or n.

7. Δ-SPOT AT WORK

As described in Section 4, Δ-SPOT can detect the exact periodicity of the cyclic events. Given the external shock tensor S, Δ-SPOT automatically generates the next spikes of the cyclic events in terms of the time and duration, respectively. Then we used the regression function to estimate the strengh of those spikes, given the strength of the previous spikes. As shown in Figure 10, we trained the model parameters by using the 400 time-ticks of the keyword "Grammy" (solid black lines in the figures), and then forecasted the following years (solid red lines). Δ-SPOT can predict the time-tick, the duration and the relative strength of incoming external events, which refer to the annual Grammy Awards, held every February. We also compared Δ-SPOT with the Auto Regressive (AR) model, and TBATS model. We ap-

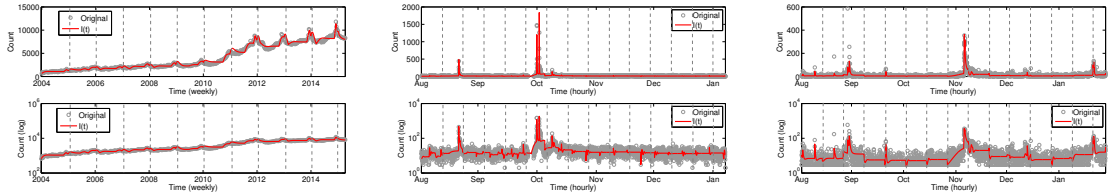


(a) Original/fitted sequences for "Ebola"



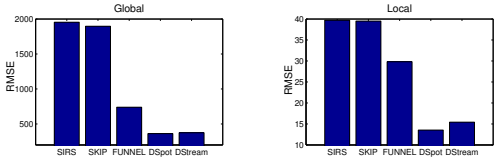
(b) World-wide reaction

Figure 6: Local fitting power of Δ-SPOT for the keyword "Ebola" which refers to the Ebola Virus bursting in 2014 (shown in green circles). (a) It can capture the local similar behaviors in the U.K. (GB), the U.S. (US) and Japan (JP). It also detects local outliers in Laos (LA) and Nepal (NP), in comparison to the global trend. We have a clearer observation in (b) the world map of user reaction to the disease burst in 2014.



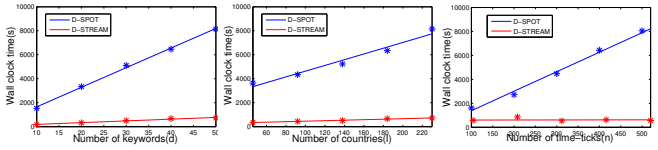
(a) "Amazon" (b) "#apple" (c) Meme#3

Figure 7: Online processing results for 3 queries: for each new coming subsequence, Δ-STREAM captures all important features, including the stream dynamics and patterns, as well as updates the external events.



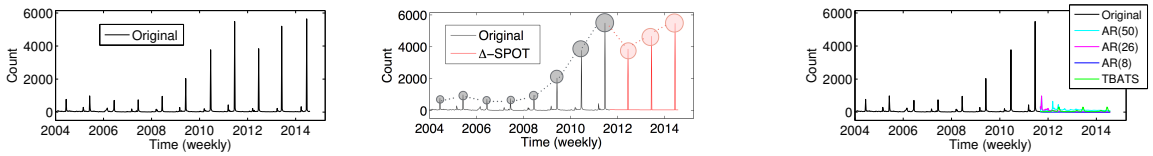
(a)Global fitting (b)Local fitting

Figure 8: Fitting accuracy (RMSE) for Δ-SPOT (lower is better).



(a) Keywords (d) (b) Countries (l) (c) Duration (n)

Figure 9: Δ-SPOT scales linearly: wall clock time vs. dataset size ($d \times l \times n$).



(a) Original sequence "Grammy" (b) Forecasted with Δ-SPOT (c) Forecasted with other methods (i.e., AR, TBATS)

Figure 10: Forecasting result: we train the model parameters using first 400 time-ticks of the sequences and do forecasting the remaining part.

plied several regression coefficients: $r = 8, 26, 50$ for AR. In Figure 10 (a,b,c), we show the original sequences, the forecast results of Δ-SPOT and AR with TBATS, respectively. Our method achieves high forecasting accuracy while AR and TBATS failed to forecast future patterns.

8. CONCLUSION

In this paper, we presented Δ-SPOT, which demonstrates the desirable properties: **Effective**: it can detect important hidden events that match the reality; **Automatic**: it requires no training set and no domain expertise, thanks to our coding scheme; **Scalable**: Δ-SPOT is linear to the data size (i.e., $O(dln)$); and **Practical**: Δ-SPOT can undertake long-range forecasting and outperforms existing methods.

9. REFERENCES

[1] K. Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly Journal of Applied Mathmatics*, II(2):164–168, 1944.

[2] L. Li, C.-J. M. Liang, J. Liu, S. Nath, A. Terzis, and C. Faloutsos. Thermocast: A cyber-physical forecasting model for data centers. In *KDD*, 2011.

[3] L. Li, J. McCann, N. Pollard, and C. Faloutsos. Dynammo: Mining and summarization of coevolving sequences with missing values. In *KDD*, 2009.

[4] Y. Matsubara, L. Li, E. E. Papalexakis, D. Lo, Y. Sakurai, and C. Faloutsos. F-trail: Finding patterns in taxi trajectories. In *PAKDD*, pages 86–98, 2013.

[5] Y. Matsubara, Y. Sakurai, and C. Faloutsos. Autoplait: automatic mining of co-evolving time sequences. In *SIGMOD*, pages 193–204, 2014.

[6] Y. Matsubara, Y. Sakurai, C. Faloutsos, T. Iwata, and M. Yoshikawa. Fast mining and forecasting of complex time-stamped events. In *KDD*, pages 271–279, 2012.

[7] Y. Matsubara, Y. Sakurai, B. A. Prakash, L. Li, and C. Faloutsos. Rise and fall patterns of information diffusion: model and implications. In *KDD*, pages 6–14, 2012.

[8] Y. Matsubara, Y. Sakurai, W. G. van Panhuis, and C. Faloutsos. FUNNEL: automatic mining of spatially coevolving epidemics. In *KDD*, pages 105–114, 2014.

[9] Y. Sakurai, Y. Matsubara, and C. Faloutsos. Mining and forecasting of big time-series data. In *SIGMOD*, pages 919–922, 2015.

[10] L. Stone, R. Olinky, and A. Huppert. Seasonal dynamics of recurrent epidemics. *Nature*, 446:533–536, March 2007.

[11] Y. Tao, C. Faloutsos, D. Papadias, and B. Liu. Prediction and indexing of moving objects with unknown motion patterns. In *SIGMOD*, pages 611–622, 2004.