Discovering Emerging Topics in Social Streams via Link-Anomaly Detection

Toshimitsu Takahashi, Ryota Tomioka, and Kenji Yamanishi, *Member*, *IEEE*

Abstract—Detection of emerging topics is now receiving renewed interest motivated by the rapid growth of social networks. Conventional-term-frequency-based approaches may not be appropriate in this context, because the information exchanged in social-network posts include not only text but also images, URLs, and videos. We focus on emergence of topics signaled by social aspects of theses networks. Specifically, we focus on mentions of users—links between users that are generated dynamically (intentionally or unintentionally) through replies, mentions, and retweets. We propose a probability model of the mentioning behavior of a social network user, and propose to detect the emergence of a new topic from the anomalies measured through the model. Aggregating anomaly scores from hundreds of users, we show that we can detect emerging topics only based on the reply/mention relationships in social-network posts. We demonstrate our technique in several real data sets we gathered from Twitter. The experiments show that the proposed mention-anomaly-based approaches can detect new topics at least as early as text-anomaly-based approaches, and in some cases much earlier when the topic is poorly identified by the textual contents in posts.

Index Terms—Topic detection, anomaly detection, social networks, sequentially discounted normalized maximum-likelihood coding, burst detection

1 Introduction

Communication over social networks, such as Facebook and Twitter, is gaining its importance in our daily life. Since the information exchanged over social networks are not only texts but also URLs, images, and videos, they are challenging testbeds for the study of data mining. In particular, we are interested in the problem of detecting emerging topics from social streams, which can be used to create automated "breaking news", or discover hidden market needs or underground political movements. Compared to conventional media, social media are able to capture the earliest, unedited voice of ordinary people. Therefore, the challenge is to detect the emergence of a topic as early as possible at a moderate number of false positives.

Another difference that makes social media social is the existence of *mentions*. Here, we mean by mentions *links* to other users of the same social network in the form of message-to, reply-to, retweet-of, or explicitly in the text. One post may contain a number of mentions. Some users may include mentions in their posts rarely; other users may be mentioning their friends all the time. Some users (like celebrities) may receive mentions every minute; for others, being mentioned might be a rare occasion. In this sense, *mention is like a language* with the number of words equal to the number of users in a social network.

We are interested in detecting emerging topics from social network streams based on monitoring the mentioning behavior of users. Our basic assumption is that a new (emerging) topic is something people feel like discussing, commenting, or forwarding the information further to their friends. Conventional approaches for topic detection have mainly been concerned with the frequencies of (textual) words [1], [2]. A term-frequency-based approach could suffer from the ambiguity caused by synonyms or homonyms. It may also require complicated preprocessing (e.g., segmentation) depending on the target language. Moreover, it cannot be applied when the contents of the messages are mostly nontextual information. On the other hand, the "words" formed by mentions are unique, require little preprocessing to obtain (the information is often separated from the contents), and are available regardless of the nature of the contents.

Fig. 1 shows an example of the emergence of a topic through posts on social networks. The first post by Bob contains mentions to Alice and John, which are both probably friends of Bob, so there is nothing unusual here. The second post by John is a reply to Bob but it is also visible to many friends of John that are not direct friends of Bob. Then in the third post, Dave, one of John's friends, forwards (called retweet in Twitter) the information further down to his own friends. It is worth mentioning that it is not clear what the topic of this conversation is about from the textual information, because they are talking about something (a new gadget, car, or jewelry) that is shown as a link in the text.

In this paper, we propose a probability model that can capture the normal mentioning behavior of a user, which consists of both the number of mentions per post and the frequency of users occurring in the mentions. Then this model is used to measure the *anomaly* of future user behavior. Using the proposed probability model, we can quantitatively measure the novelty or possible impact of a post reflected in the mentioning behavior of the user. We aggregate the anomaly scores obtained in this way over

- *T. Takahashi is with the Institute of Industrial Science, The University of Tokyo, Tokyo 153-8505, Japan. E-mail: takahashi@tauhat.com.*
- *R. Tomioka and K. Yamanishi are with the Department of Mathematical Informatics, The University of Tokyo, 7-3-1, Hongo, Bunkyo-ku, Tokyo 113-8656, Japan. E-mail: {tomioka, yamanishi}@mist.i.u-tokyo.ac.jp.*

Manuscript received Dec. 2011; revised 15 June 2012; accepted 21 Nov. 2012; published online 12 Dec. 2012.
Recommended for acceptance by B.C. Ooi.
For information on obtaining reprints of this article, please send e-mail to: tkde@computer.org, and reference IEEECS Log Number TKDE-2011-12-0803.
Digital Object Identifier no. 10.1109/TKDE.2012.239.

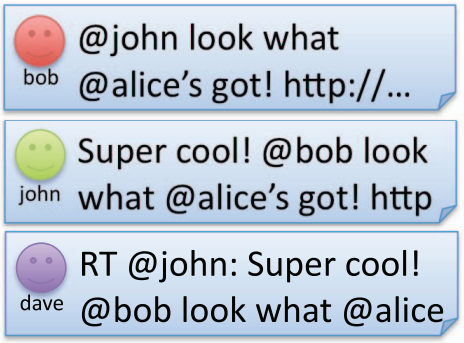


Fig. 1. Example of the emergence of a topic in social streams.

hundreds of users and apply a recently proposed change-point detection technique based on the sequentially discounting normalized maximum-likelihood (SDNML) coding [3]. This technique can detect a change in the statistical dependence structure in the time series of aggregated anomaly scores, and pinpoint where the topic emergence is; see Fig. 2. The effectiveness of the proposed approach is demonstrated on four data sets we have collected from Twitter. We show that our mention-anomaly-based approaches can detect the emergence of a new topic at least as fast as text-anomaly-based counterparts. Furthermore, we show that in three out of four data sets, the proposed mention-anomaly-based methods can detect the emergence of topics much earlier than the text-anomaly-based methods, which can be explained by the keyword ambiguity we mentioned above.

2 Related Work

Detection and tracking of topics have been studied extensively in the area of topic detection and tracking (TDT) [1]. In this context, the main task is to either classify a new document into one of the known topics (tracking) or to detect that it belongs to none of the known categories. Subsequently, temporal structure of topics has been modeled and analyzed through dynamic model selection [4], temporal text mining [5], and factorial hidden Markov models [6].

Another line of research is concerned with formalizing the notion of "bursts" in a stream of documents. In his seminal paper, Kleinberg modeled bursts using the time-varying Poisson process with a hidden discrete process that controls the firing rate [2]. Recently, He and Parker developed a physics-inspired model of bursts based on the change in the momentum of topics [7].

All the above-mentioned studies make use of textual content of the documents, but not the social content of the documents. The social content (links) has been utilized in the study of citation networks [8]. However, citation networks are often analyzed in a stationary setting.

The novelty of the current paper lies in focusing on the social content of the documents (posts) and in combining this with a change-point analysis.

3 Proposed Method

The overall flow of the proposed method is shown in Fig. 2. Each step in the flow is described in the corresponding subsection. We assume that the data arrive from a social network service in a sequential manner through some API. For each new post, we use samples within the past time interval of length T for the corresponding user for training the mention model we propose below (Step 1). We assign an anomaly score to each post based on the learned probability distribution (Step 2). The score is then aggregated over users (Step 3) and further fed into SDNML-based change-point analysis (Steps 4 and 5). We also describe Kleinberg's burst-detection method, which can be used instead of the SDNML-based change-point analysis in Section 3.6 (Step 6). We also discuss the scalability of the proposed approach in Section 3.8.

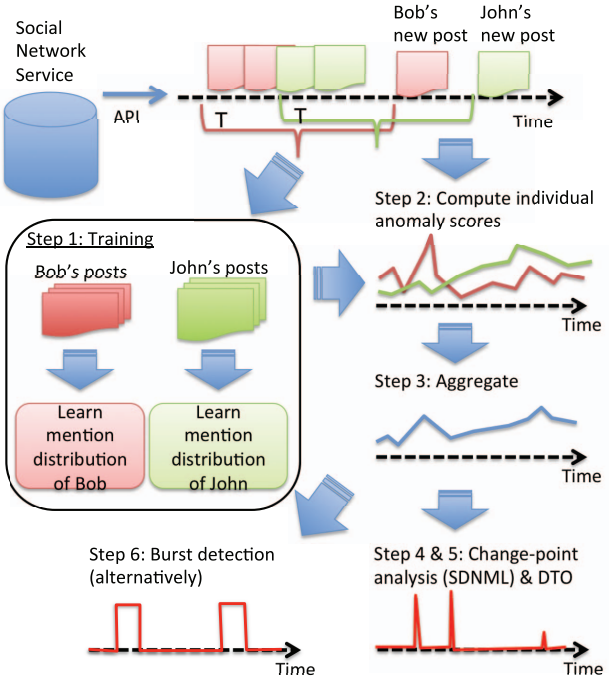


Fig. 2. Overall flow of the proposed method. Each step is described in the corresponding subsection of Section 3.

3.1 Probability Model

In this subsection, we describe the probability model that we used to capture the normal mentioning behavior of a user and how to train the model; see Step 1 in Fig. 2.

We characterize a post in a social network stream by the number of mentions k it contains, and the set V of names (IDs) of the mentionees (users who are mentioned in the post).

There are two types of infinity we have to take into account here. The first is the number k of users mentioned in a post. Although, in practice a user cannot mention hundreds of other users in a post, we would like to avoid putting an artificial limit on the number of users mentioned in a post. Instead, we will assume a geometric distribution and integrate out the parameter to avoid even an implicit limitation through the parameter. The second type of infinity is the number of users one can possibly mention. To avoid limiting the number of possible mentionees, we use a Chinese Restaurant Process (CRP; see [9]) based estimation; see also Teh et al. [10] who use CRP for infinite vocabulary.

Formally, we consider the following joint probability distribution:

$$P(k, V \mid \theta, \{\pi_v\})) = P(k \mid \theta) \prod_{v' \in V} \pi_{v'}. \quad (1)$$

Here, the joint distribution consists of two parts: the probability of the number of mentions $k = |V|$ and the probability of each mention given the number of mentions. The probability of the number of mentions $P(k \mid \theta)$ is defined as a geometric distribution with parameter θ as follows:

$$P(k \mid \theta) = (1 - \theta)^k \theta. \quad (2)$$

The probability of mentioning user $v \in V$ is denoted by π_v (where the sum of π_v over all users must be 1, $\sum_v \pi_v = 1$) and we assume that the k users in V are independently and identically mentioned. In other words, we ignore the dependences between mentionees and model them as bag of words [11].

Suppose that we are given n past posts $\mathcal{T} = \{(k_1, V_1), \ldots, (k_n, V_n)\}$ from a user, and we would like to learn the predictive distribution

$$P(k, V \mid \mathcal{T}) = P(k \mid \mathcal{T}) \prod_{v \in V} P(v \mid \mathcal{T}) \quad (3)$$

for that user; see Step 1 in Fig. 2.

First, we compute the predictive distribution with respect to the number of mentions $P(k \mid \mathcal{T})$. This can be obtained by assuming a beta distribution as a prior and integrating out the parameter θ. The density function of the beta prior distribution is written as follows:

$$p(\theta \mid \alpha, \beta) = \frac{(1 - \theta)^{\beta - 1} \theta^{\alpha - 1}}{B(\alpha, \beta)},$$

where α and β are parameters of the beta distribution and $B(\alpha, \beta)$ is the beta function. By the Bayes rule, the predictive distribution can be obtained as follows:

$$\begin{aligned} P(k \mid \mathcal{T}, \alpha, \beta) &= P(k \mid k_1, \ldots, k_n, \alpha, \beta) \\ &= \frac{P(k, k_1, \ldots, k_n \mid \alpha, \beta)}{P(k_1, \ldots, k_n \mid \alpha, \beta)} \\ &= \frac{\int_0^1 (1 - \theta)^{\sum_{i=1}^n k_i + k + \beta - 1} \theta^{n+1+\alpha-1} \mathrm{d}\theta}{\int_0^1 (1 - \theta)^{\sum_{i=1}^n k_i + \beta - 1} \theta^{n+\alpha-1} \mathrm{d}\theta}. \end{aligned}$$

Both the integrals on the numerator and denominator can be obtained in closed forms as beta functions and the predictive distribution can be rewritten as follows:

$$P(k \mid \mathcal{T}, \alpha, \beta) = \frac{B(n + 1 + \alpha, \sum_{i=1}^n k_i + k + \beta)}{B(n + \alpha, \sum_{i=1}^n k_i + \beta)}.$$

Using the relation between the beta function and gamma function, we can further simplify the expression as follows:

$$P(k \mid \mathcal{T}, \alpha, \beta) = \frac{n + \alpha}{m + k + \beta} \prod_{j=0}^{k} \frac{m + \beta + j}{n + m + \alpha + \beta + j}, \quad (4)$$

where $m = \sum_{i=1}^n k_i$ is the total number of mentions in the training set $\mathcal{T}$. We use parameters $\alpha = \beta = 0.5$, which correspond to the Jeffreys prior for the Bernoulli likelihood, in the experiments. Note that Jeffreys prior for the geometric likelihood is not integrable.

Next, we derive the predictive distribution $P(v \mid \mathcal{T})$ of mentioning user v. The maximum-likelihood (ML) estimator is given as $P(v \mid \mathcal{T}) = m_v / m$, where m is the number of total mentions and m_v is the number of mentions to user v in the data set $\mathcal{T}$. The ML estimator, however, cannot handle users that did not appear in the training set $\mathcal{T}$; it would assign probability zero to all these users, which would appear infinitely anomalous in our framework. Instead, we use a CRP-based estimation; see [9]. The CRP-based estimator assigns probability to each user v that is proportional to the number of mentions m_v in the training set $\mathcal{T}$; in addition, it keeps probability proportional to γ for mentioning someone who was not mentioned in the training set $\mathcal{T}$. Accordingly, the probability of known users is given as follows:

$$P(v \mid \mathcal{T}) = \frac{m_v}{m + \gamma} \quad (\text{for } v\text{: } m_v \geq 1). \quad (5)$$

On the other hand, the probability of mentioning a new user is given as follows:

$$P(\{v : m_v = 0\} \mid \mathcal{T}) = \frac{\gamma}{m + \gamma}. \quad (6)$$

We use $\gamma = 0.5$ in the experiments.

3.2 Computing the Link-Anomaly Score

In this subsection, we describe how to compute the deviation of a user's behavior from the normal mentioning behavior modeled in the previous subsection; see Step 2 in Fig. 2.

To compute the anomaly score of a new post $\boldsymbol{x} = (t, u, k, V)$ by user u at time t containing k mentions to users V, we compute the probability (3) with the training set $\mathcal{T}_u^{(t)}$, which is the collection of posts by user u in the time period $[t - T, t]$ (we use $T = 30$ days in this paper). Accordingly, the link-anomaly score is defined as follows:

$$\begin{aligned} s(\boldsymbol{x}) &= -\log\left(P\big(k \mid \mathcal{T}_u^{(t)}\big) \prod_{v \in V} P\big(v \mid \mathcal{T}_u^{(t)}\big) \right) \\ &= -\log P\big(k \mid \mathcal{T}_u^{(t)}\big) - \sum_{v \in V} \log P\big(v \mid \mathcal{T}_u^{(t)}\big). \end{aligned} \quad (7)$$

The two terms in the above equation can be computed via the predictive distribution of the number of mentions (4), and the predictive distribution of the mentionee (5)-(6), respectively.

3.3 Combining Anomaly Scores from Different Users

In this subsection, we describe how to combine the anomaly scores from different users; see Step 3 in Fig. 2.

The anomaly score in (7) is computed for each user depending on the current post of user u and his/her past behavior $\mathcal{T}_u^{(t)}$. To measure the general trend of user behavior, we propose to aggregate the anomaly scores obtained for posts $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ using a discretization of window size $\tau > 0$ as follows:

$$s'_j = \frac{1}{\tau} \sum_{t_i \in [\tau(j-1), \tau j]} s(\boldsymbol{x}_i), \quad (8)$$

where $\boldsymbol{x}_i = (t_i, u_i, k_i, V_i)$ is the post at time t_i by user u_i including k_i mentions to users V_i.

3.4 Change-Point Detection via SDNML Coding

In this subsection, we describe how to detect change points from the sequence of aggregated anomaly scores; see Step 4 in Fig. 2.

Given an aggregated measure of anomaly (8), we apply a change-point detection technique proposed in [3]. This technique is an extension of ChangeFinder proposed in [12], [13] that detects a change in the statistical dependence structure of a time series by monitoring the compressibility of a new piece of data. Urabe et al. [3] proposed to use a sequential version of normalized maximum-likelihood (NML) coding called SDNML coding [3] as a coding criterion instead of the plug-in predictive distribution used in [12], [13]. Specifically, a change point is detected through two layers of scoring processes. The first layer detects outliers and the second layer detects change-points (see [12], [13]). In each layer, predictive loss based on the SDNML coding distribution for an autoregressive (AR) model is used as a criterion for scoring. Although the NML code length is known to be optimal [14], it is often hard to compute. The SNML proposed in [15], [16] is an approximation to the NML code length that can be computed in a sequential manner. The SDNML proposed in [3] further employs discounting in the learning of the AR models. The notion of SDNML has also been proposed by Giurcăneanu et al. [17], [18], independent of [3].

Algorithmically, the change-point detection procedure can be outlined as follows: For convenience, we denote the aggregated anomaly score as x_j instead of s'_j.

1. *First-layer learning.* Let $x^{j-1} := \{x_1, \ldots, x_{j-1}\}$ be the collection of aggregated anomaly scores from discrete time 1 to $j-1$. Sequentially learn the SDNML density function $p_{\mathrm{SDNML}}(x_j \mid x^{j-1})(j = 1, 2, \ldots)$; see Section 3.7 for details.
2. *First-layer scoring.* Compute the intermediate change-point score by smoothing the log loss of the SDNML density function with window size κ as follows:

$$y_j = \frac{1}{\kappa} \sum_{j'=j-\kappa+1}^{j} \left(-\log p_{\mathrm{SDNML}}\left(x_j \mid x^{j-1}\right)\right).$$

3. *Second-layer learning.* Let $y^{j-1} := \{y_1, \ldots, y_{j-1}\}$ be the collection of smoothed change-point score obtained as above. Sequentially learn the second layer SDNML density function $p_{\mathrm{SDNML}}(y_j \mid y^{j-1})(j = 1, 2, \ldots)$; see Section 3.7 for details.
4. *Second-layer scoring.* Compute the final change-point score by smoothing the log loss of the SDNML density function as follows:

$$\mathrm{Score}(y_j) = \frac{1}{\kappa} \sum_{j'=j-\kappa+1}^{j} \left(-\log p_{\mathrm{SDNML}}(y_j \mid y^{j-1})\right). \quad (9)$$

Bayesian theory is employed in the calculation of the predictive distribution described in Section 3.1. Meanwhile, the MDL principle is employed in the calculation of the predictive distribution for the change-point score (9). These theories are consistently combined in our framework, because we measure both the anomaly score (7) and the change-point score (9) in terms of the logarithmic loss based on the predictive distributions.

3.5 Dynamic Threshold Optimization (DTO)

As a final step in our method, we need to convert the change-point scores into binary alarms by thresholding. Since the distribution of change-point scores may change over time, we need to dynamically adjust the threshold to analyze a sequence over a long period of time. In this subsection, we describe how to dynamically optimize the threshold using the method of dynamic threshold optimization proposed in [19]; see Step 5 in Fig. 2.

In DTO, we use a one-dimensional histogram for the representation of the score distribution. We learn it in a sequential and discounting way. Then the dynamically optimized threshold $\eta(j)$ at time step j is defined as the least score value such that the tail probability above $\eta(j)$ is no greater than ρ. We call ρ a *significance-level parameter*.

The details of DTO are summarized as follows: Let N_H be a given positive integer. Let $\{q(h)(h = 1, \ldots, N_H) : \sum_{h=1}^{N_H} q(h) = 1\}$ be a one-dimensional histogram with N_H bins, where h is an index of bins, with a smaller index indicating a bin having a smaller score. For given a, b such that $a < b$, N_H bins in the histogram are set as $\{(-\infty, a); [a + \{(b-a)/(N_H - 2)\}\ell, [a + \{(b-a)/(N_H - 2)\}(\ell + 1)(\ell = 0, 1, \ldots, N_H - 3)$ and $[b, \infty)$. Let $\{q^{(j)}(h)\}$ be a histogram updated after seeing the jth score. The procedures of updating the histogram and DTO are given in Algorithm 1.

Algorithm 1. Dynamic Threshold Optimization (DTO) [19].
Given: $\{Score_j \mid j = 1, 2, \ldots\}$: scores, N_H: total number of cells, ρ: parameter for threshold, λ_H: estimation parameter, r_H: discounting parameter, M: data size
Initialization: Let $q_1^{(1)}(h)$ (a weighted sufficient statistics) be a uniform distribution.
for $j = 1, \ldots, M-1$ **do**
Threshold optimization: Let l be the least index such that $\sum_{h=1}^{l} q^{(j)}(h) \geq 1 - \rho$. The threshold at time j is given as

$$\eta(j) = a + \frac{b-a}{N_H - 2}(l + 1).$$

Alarm output: Raise an alarm if $Score_j \geq \eta(j)$.
Histogram update:

$$q_1^{(j+1)}(h) = \begin{cases} (1 - r_H) q_1^{(j)}(h) + r_H & \text{if } Score_j \text{ falls into the } h\text{th cell,} \\ (1 - r_H) q_1^{(j)}(h) & \text{otherwise.} \end{cases}$$

$q^{(j+1)}(h) = (q_1^{(j+1)}(h) + \lambda_H)/(\sum_h q_1^{(j+1)}(h) + N_H \lambda_H)$.
end for

3.6 Kleinberg's Burst-Detection Method

In addition to the change-point detection based on SDNML followed by DTO described in previous sections, we also test the combination of our method with Kleinberg's burst-detection method [2]. More specifically, we implemented a two-state version of Kleinberg's burst-detection model. The reason we chose the two-state version was because in this experiment we expect no

hierarchical structure. The burst-detection method is based on a probabilistic automaton model with two states, burst state and nonburst state. Some events (e.g., arrival of posts) are assumed to happen according to a time-varying Poisson processes whose rate parameter depends on the current state. The burst-detection method estimates the state transition sequence $i_t \in \{\text{nonburst}, \text{burst}\}$ $(t = 1, \ldots, n)$ that maximizes the likelihood

$$p_{\text{sw}}^{b}(1 - p_{\text{sw}})^{n-b} \prod_{t=1}^{n} f_{\exp}(x_t; \alpha_{i_t}),$$

where p_{sw} is a given state transition probability, b is the number of state transitions in the sequence i_t $(t = 1, \ldots, n)$, $f_{\exp}(x; \alpha)$ is the probability density function of the exponential distribution with rate parameter α, and x_t is the tth interevent interval. The optimal sequence can be efficiently obtained by dynamic programming [2].

To obtain the event times and their intervals, we define an event as a point in time when the aggregated link-anomaly score (8) exceeds a threshold θ_{burst}.

3.7 Computation of the SDNML Code Length

The SDNML density p_{SDNML} in the change-point detection algorithm described in Section 3.4 is obtained by applying the SNML proposed by Roos et al. [15], [16] to the class of AR model with a *discounted* ML estimation, which makes the SDNML-based change-point detection algorithm more flexible than an SNML-based one. Let $x_t \in \mathbf{R}$ for each t. We define the pth-order AR model as follows:

$$p\big(x_t \mid x_{t-k}^{t-1} : \boldsymbol{\theta}\big) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2} \left(x_t - \sum_{i=1}^{p} a^{(i)} x_{t-i} \right)^2 \right),$$

where $\boldsymbol{\theta}^\top = (\boldsymbol{a}^\top, \sigma^2) = ((a^{(1)}, \ldots, a^{(p)}), \sigma^2)$ is the parameter vector.

To compute the SDNML density function, we need the discounted ML estimators of the parameters in $\boldsymbol{\theta}$. We define the discounted ML estimator of the regression coefficient $\hat{\boldsymbol{a}}_t$ as follows:

$$\hat{\boldsymbol{a}}_t = \underset{\boldsymbol{a} \in \mathbf{R}^p}{\operatorname{argmin}} \sum_{j=t_0+1}^{t} w_{t-j} \big(x_j - \boldsymbol{a}^\top \bar{\boldsymbol{x}}_j\big)^2, \tag{10}$$

where $w_{j'} = r(1-r)^{j'}$ is a sequence of sample weights with the discounting coefficient r $(0 < r < 1)$; t_0 is the smallest number of samples such that the minimizer (10) is unique, $\bar{\boldsymbol{x}}_j := (x_{j-1}, x_{j-2}, \ldots, x_{j-k})^\top$. Note that the error terms from older samples receive geometrically decreasing weights in (10). The larger the discounting coefficient r, the smaller the weights of the older samples become; thus, we have stronger discounting effect. Moreover, we obtain the discounted ML estimator of the variance $\hat{\tau}_t$ as follows:

$$\hat{\tau}_t := \sum_{j=t_0+1}^{t} w_{t-j} \hat{e}_j^2,$$

where we define $\hat{e}_j^2 = (x_j - \hat{\boldsymbol{a}}_j^\top \bar{\boldsymbol{x}}_j)^2$. Clearly, when the discounted estimator of the AR coefficient $\hat{\boldsymbol{a}}_j$ is available, $\hat{\tau}_t$ can be computed in a sequential manner. Note that we introduce discounting not only in the estimation of the AR coefficients (10), but also in the estimation of the variance $\hat{\tau}_t$, while Urabe et al. [3] employed no discounting for the variance.

In the sequel, we first describe how to efficiently compute the AR estimator $\hat{\boldsymbol{a}}_j$. Finally, we derive the SDNML density function using the discounted ML estimators $(\hat{\boldsymbol{a}}_t, \hat{\tau}_t)$.

The AR coefficient $\hat{\boldsymbol{a}}_j$ can simply be computed by solving the least-squares problem (10). It can, however, be obtained more efficiently using the iterative formula described in [15], [16]. Here, we repeat the formula for the discounted version presented in [3]. First, define the sufficient statistics $V_t \in \mathbf{R}^{p \times p}$ and $\boldsymbol{\chi}_t \in \mathbf{R}^p$ as follows:

$$V_t := \sum_{j=t_0+1}^{t} w_j \bar{\boldsymbol{x}}_j \bar{\boldsymbol{x}}_j^\top, \quad \boldsymbol{\chi}_t := \sum_{j=t_0+1}^{t} w_j \bar{\boldsymbol{x}}_j x_j.$$

Using the sufficient statistics, the discounted AR coefficient $\hat{\boldsymbol{a}}_j$ from (10) can be written as follows:

$$\hat{\boldsymbol{a}}_t = V_t^{-1} \boldsymbol{\chi}_t.$$

Note that $\boldsymbol{\chi}_t$ can be computed in a sequential manner. The inverse matrix V_t^{-1} can also be computed sequentially using the Sherman-Morrison-Woodbury formula as follows:

$$V_t^{-1} = \frac{1}{1-r} V_t^{-1} - \frac{r}{1-r} \frac{V_t^{-1} \bar{\boldsymbol{x}}_t \bar{\boldsymbol{x}}_t^\top V_t^{-1}}{1 - r + c_t},$$

where $c_t = r \bar{\boldsymbol{x}}_t^\top V_t^{-1} \bar{\boldsymbol{x}}_t$.

Finally, the SDNML density function is written as follows:

$$p_{\text{SDNML}}\big(x_t \mid x^{t-1}\big) = \frac{1}{K_t(x^{t-1})} \frac{\hat{\tau}_t^{-(t-t_0)/2}}{\hat{\tau}_{t-1}^{-(t-t_0-1)/2}},$$

where the normalization factor $K_t(x^{t-1})$ is calculated as follows:

$$K_t(x^{t-1}) = \frac{\sqrt{\pi}}{1 - d_t} \sqrt{\frac{1-r}{r}} (1-r)^{-\frac{t-m}{2}} \frac{\Gamma((t - t_0 - 1)/2)}{\Gamma((t - t_0)/2)},$$

with $d_t = c_t/(1 - r + c_t)$.

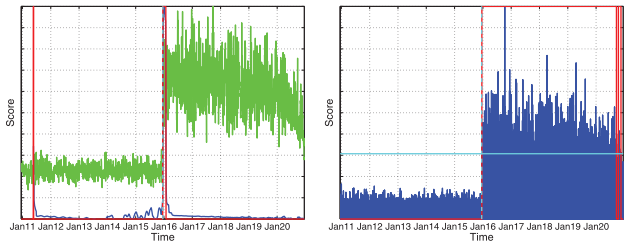
3.8 Scalability of the Proposed Algorithm

The proposed link-anomaly-based change-point detection is highly scalable. Every step described in the previous subsections (Step 1-Step 6) requires only linear time against the length of the analyzed time period. Computation of the predictive distribution for the number of mentions (4) can be computed in linear time against the number of mentions. Computation of the predictive distribution for the mention probability in (5) and (6) can be efficiently performed using a hash table. Aggregation of the anomaly scores from different users takes linear time against the number of users, which could be a computational bottle neck but can be easily parallelized. SDNML-based change-point detection requires two swipes over the analyzed time period. Kleinberg's burst-detection method can be efficiently implemented with dynamic programming.

4 Synthetic Experiments

4.1 Experimental Setup

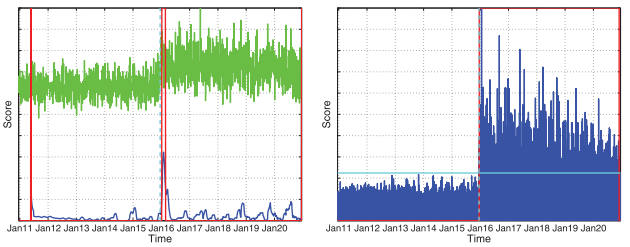
We generated synthetic data sets over 20 days from 100 users as we describe below. For each user, we assume



(a) Link-anomaly-based change-point analysis. Green: Aggregated anomaly score (8) at $\tau = 10$ minutes. Blue: Change-point score (9). Red: Alarm time.

(b) Link-anomaly-based burst detection. Blue: Aggregated anomaly score (8) at $\tau = 1$ second. Cyan: threshold for the filtering step in Kleinberg's burst model. Red: Burst state.

Fig. 3. Results for the synthetic data set "Synthetic100". The vertical dashed cyan line shows the true change-point at 9:00, Jan 16.



(a) Link-anomaly-based change-point analysis. Green: Aggregated anomaly score (8) at $\tau = 10$ minutes. Blue: Change-point score (9). Red: Alarm time.

(b) Link-anomaly-based burst detection. Blue: Aggregated anomaly score (8) at $\tau = 1$ second. Cyan: threshold for the filtering step in Kleinberg's burst model. Red: Burst state.

Fig. 4. Results for the synthetic data set "Synthetic20". The vertical dashed cyan line shows the true change-point at 9:00, Jan 16.

that posts arrive from a Poisson process and draw the interpost intervals from an exponential distribution with a fixed rate. The average interpost interval (=1/rate parameter of the Poisson process) is drawn from a Gamma distribution with shape parameter 1 and scale parameter 1 hour for each user. The number of mentions in each post is drawn from a geometric distribution with parameter 0.5, which corresponds to one mention per post in average. We identify the users by their IDs ranging from 0 to 99, and we measure the distance between them by the absolute difference in their IDs modulo 100, i.e., the users are organized on a circle. The ID j of each mentionee of a post at time t by the ith user is drawn independently as $j = \mathrm{round}(i + \xi) \bmod 100$, where ξ is drawn from the Gaussian distribution $\mathcal{N}(0, \sigma_t^2)$. Note that the parameter σ_t that controls how far users communicate with each other depends on time.

We generated two data sets. In the first data set, which we call "Synthetic100", we set $\sigma_t = 1$ from the first day to the 15th day and $\sigma_t = 10$ from the 16th day to the 20th day for all 100 users; therefore, we create an artificial change-point in the communication pattern of the users at the 16th day. In the second data set, which we call "Synthetic20", the parameter σ_t was varied for the first 20 users as in "Synthetic100", whereas for the rest of the users it was set $\sigma_t = 1$ for all t. Thus, it simulates the more realistic setting in practice where only some of the users react to the emerging topic.

We report the result of combining our proposed approach with SDNML-based change-point analysis and DTO, and with Kleinberg's burst-detection model. The proposed link-based anomaly detection method was implemented with parameters $\alpha = \beta = \gamma = 0.5$ and $T = 10$ days. SDNML-based change-point analysis was implemented with smoothing parameter $\kappa = 15$ and AR model order 30. The parameters in DTO were set as $\rho = 0.05, N_H = 20, \lambda_H = 0.01, r = 0.005$.

For the burst-detection approach, we used the firing rate parameter of the Poisson point process 0.0001 (1/s) for the nonburst state and 0.001 (1/s) for the burst state, and the transition probability $p_{\mathrm{sw}} = 0.3$. The threshold θ_{burst} was set to 0.999-quantile point of the link-anomaly score (8) with $\tau = 1$ second; $\tau = 1$ second means aggregation over population without temporal averaging. We consider the transition from the nonburst state to the burst state as an "alarm".

4.2 Results

Fig. 3a shows that the proposed approach combined with SDNML-based change-point analysis, and DTO correctly identifies the change point at 9:00, January 16, for "Synthetic100" data set. We can clearly see that the proposed link-based anomaly score (green curve in Fig. 3a) is low in the period Jan 11-Jan 15 and high in the period Jan 16-Jan 20. The SDNML-based change-point analysis (the blue curve in Fig. 3a) sharply rises at the change-point and goes down to zero quickly. DTO converts the rise in change-point score into a binary sequence of alarms (the red curve in Fig. 3a). The first detection time of SDNML+DTO was 9:00, Jan 16, ignoring the initial instability around Jan 11. Fig. 3b further demonstrates that the proposed link-based anomaly score can be combined with burst analysis. The two-state burst model correctly identifies the low state of Jan 11-Jan 15 and the high state of Jan 16-Jan 20. The first detection time of the burst approach was 9:01, Jan 16.

Figs. 4a and 4b show the same plots for "Synthetic20" data set. Although the change in the link-based anomaly score at Jan 16 was smaller because of the reduced number of users who reacted to the topic, the proposed SDNML+DTO successfully raised an alarm at 10:30, January 16, ignoring the initial instability around January 11. The burst-detection approach raised an alarm at 9:13, January 16, which was earlier than the SDNML-based approach.

The above results show that the proposed approach can detect changes in the communication patterns of users even in a realistic setting when only some part of the users react to the emerging topic.

5 Real Data Experiments

5.1 Experimental Setup

We collected four data sets from Twitter. Each data set is associated with a list of posts in a service called Togetter[1]; Togetter is a collaborative service where people can tag Twitter posts that are related to each other and organize a list of posts that belong to a certain topic. Our goal is to evaluate whether the proposed approach can detect the emergence of the topics recognized and collected by people.

1. http://togetter.com/.

TABLE 1
Parameter Values We Used in the Real Data Experiments

Model	Parameter name	value
Mention model	Beta distribution	$\alpha = \beta = 0.5$
	CRP parameter	$\gamma = 0.5$
	Training period	$T = 30$ days
SDNML-based change-point detection	AR model order	$p = 30$
	Smoothing parameter	$\kappa = 15$
Dynamic threshold optimization (DTO)	Number of bins	$N_H = 20$
	Smoothing parameter	$\lambda_H = 0.01$
	Discount rate	$r = 0.005$
	Significance level parameter	$\rho = 0.05$
	Normal data upper limit	a =average+3σ of the input data
	Normal data lower limit	b =minimum of the input data
Kleinberg's burst detection model	Rate parameter (nonburst state)	$\alpha_{\mathrm{nonburst}} = 0.001$ (1/s)
	Rate parameter (burst state)	$\alpha_{\mathrm{burst}} = 0.01$ (1/s)
	State transition probability	$p_{\mathrm{sw}} = 0.3$
	Threshold parameter	θ_{burst} =0.9995-quantile point of the aggregated anomaly scores

TABLE 2
Data Sets

Data set	Number of participants	Number of posts	Keyword (in English)	Keyword (unicode)
"Job hunting"	200	415,814	Job hunting	u'\u5c31\u8077'
"Youtube"	160	547,287	Senkaku	u'\u5c16\u95a3'
"NASA"	90	278,156	Arsenic	u'\u30d2\u7d20'
"BBC"	47	89,887	British	u'\u30a4\u30ae\u30ea\u30b9'

The four data sets we collected are called "Job hunting", "Youtube", "NASA", "BBC" and each of them corresponds to a user organized list in Togetter. For each list, we extracted a list of Twitter users that appeared in the list, and collected Twitter posts from those users. See Table 2 for the number of participants and the number of posts we collected for each data set. Note that we collected Twitter posts up to 30 days before the time period of interest for each user; thus, the number of posts we analyzed was much larger than the number of posts listed in Togetter.

For the proposed mention-anomaly model, we used $\alpha = \beta = \gamma = 0.5$ and $T = 30$ days.

For the SDNML-based change-point detection, we use the smoothing parameter $\kappa = 15$, and the order of the AR model 30 in the experiments; the parameters in DTO were set as $\rho = 0.05$, $N_H = 20$, $\lambda_H = 0.01$, $r_H = 0.005$.

For Kleinberg's burst model, the firing rate parameters of the Poisson point process we used were 0.001 (1/s) for the nonburst state and 0.01 (1/s) for the burst state. The transition probability was $p_{\mathrm{sw}} = 0.3$. The threshold parameter θ_{burst} was set to 0.9995-quantile point of the link-anomaly score (8) with $\tau = 1$ second. We consider the transition from the nonburst state to the burst state as an "alarm".

All the parameters we used in our experiments are listed in Table 1.

We compared our proposed link-based anomaly detection approach against a text-based anomaly detection approach. More specifically, we removed mentions and retweets, extracted words (including URLs) from each post, and considered k as the number of words and V as the bag-of-words representation of the post. We used the same SDNML-based change detection approach and Kleinberg's burst-detection approach in combination with the text-based anomaly score. The threshold parameter θ_{burst} was set in the same way as above. We used MeCab[2] to extract words from twitter posts.

Furthermore, we include a keyword-based change-point detection method in the comparison. In the keyword-based method, we looked at a sequence of frequency (observed within 1 minute) of a keyword related to the topic; the keyword was manually selected to best capture the topic. Then we applied DTO described in Section 3.5 to the sequence of keyword frequencies. In our experience, the sparsity of the keyword frequency seems to be a bad combination with the SDNML method; therefore, we did not use SDNML in the keyword-based method. We also applied Kleinberg's burst-detection method to the arrival times of the keyword. We set $\theta_{\mathrm{burst}} = 0$, and used all posts that include the keyword for the burst analysis.

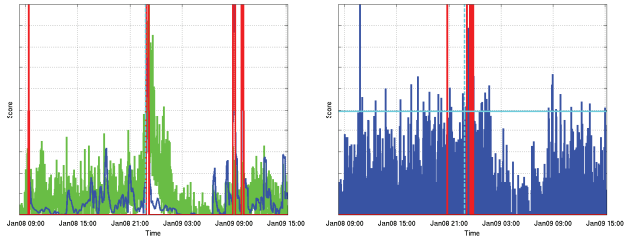
The keyword-based approach can only be used when we are expecting a burst of tweets mentioning the prespecified keyword, which could happen if we were making an advertisement campaign or any other kind of manipulation. However, here it should be regarded as a sanity check, since we are interested in automatically detecting the emergence of a topic without any intervention. Therefore, our goal is to detect emerging topics as early as the keyword-based methods. The keywords we used for the four data sets are summarized in Table 2.

5.2 "Job Hunting" Data Set

This data set is related to a controversial post by a famous person in Japan that "the reason students having difficulty finding jobs is, because they are stupid" and various replies to that post.

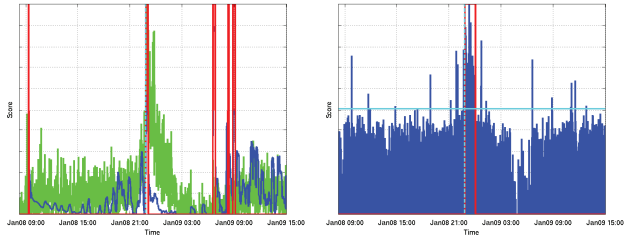
The keyword used in the keyword-based methods was "Job hunting." Figs. 5a and 5b show the results of the

2. http://mecab.googlecode.com/svn/trunk/mecab/doc/index.html.



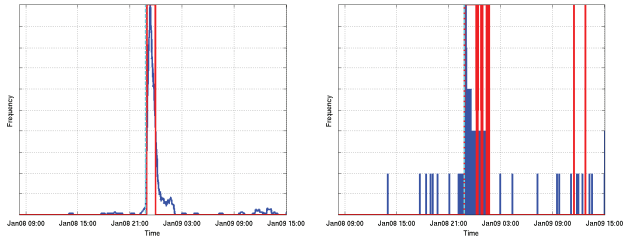
(a) Link-anomaly-based change-point analysis. Green: Aggregated anomaly score (8) at $\tau = 1$ minute. Blue: Change-point score (9). Red: Alarm time.

(b) Link-anomaly-based burst detection. Blue: Aggregated anomaly score (8) at $\tau = 1$ second. Horizontal cyan: threshold for the filtering step in Kleinberg's burst model. Red: Burst state.



(c) Text-anomaly-based change-point analysis. Green: Aggregated anomaly score (8) at $\tau = 1$ minute. Blue: Change-point score (9). Red: Alarm time.

(d) Text-anomaly-based burst detection. Blue: Aggregated anomaly score (8) at $\tau = 1$ second. Horizontal cyan: threshold for the filtering step in Kleinberg's burst model. Red: Burst state.



(e) Keyword-frequency-based change-point analysis. Blue: Frequency of keyword "Job hunting" per one minute. Red: Alarm time.

(f) Keyword-frequency-based burst detection. Blue: Frequency of keyword "Job hunting" per one second. Red: Burst state (burst or not).

Fig. 5. Result of "Job hunting" data set. The initial controversial post was posted at 22:50, Jan 08 (indicated by vertical cyan dashed lines).

proposed link-anomaly-based change detection and burst detection, respectively. Figs. 5c and 5d show the results of the text-anomaly-based change detection and burst detection, respectively. Figs. 5e and 5f show the results of the keyword-frequency-based change detection and burst detection, respectively.

The first alarm time of the proposed link-anomaly-based change-point analysis was 22:55, whereas that of the text-anomaly-based counterpart was 22:51; see also Table 3. The earliest detection was achieved by the keyword-frequency-based burst-detection method. Nevertheless, from Fig. 5, we can observe that the proposed link-anomaly-based methods were able to detect the emerging topic almost as early as the text-anomaly-based methods and keyword-frequency-based methods with only 5 minutes of delay from the initial controversial post at 22:50.

5.3 "Youtube" Data Set

This data set is related to the recent leakage of some confidential video by the Japan Coastal Guard officer.

The keyword used in the keyword-based methods was "Senkaku." Figs. 6a and 6b show the results of link-anomaly-based change detection and burst detection, respectively. Figs. 6c and 6d show the results of text-anomaly-based change detection and burst detection, respectively. Figs. 6e and 6f show the results of keyword-frequency-based change detection and burst detection, respectively.

The first alarm times of the proposed link-anomaly-based change-point analysis and the text-anomaly-based change-point analysis were both 08:44, Nov. 05, which were almost 9 hours after the first post about the video leakage. Although there is an elevation in the aggregated anomaly score (8) in Fig. 6a around midnight, Nov 05, it seems that SDNML fails to detect this elevation as a change-point. In fact, the link-anomaly-based burst detection (Fig. 6b) raised an alarm at 00:07, which is earlier than the keyword-frequency-based dynamic thresholding and closer to the keyword-frequency-based burst detection at 23:59, Nov 04. The alarm time of the text-anomaly-based burst detection was 01:24, Nov 05.

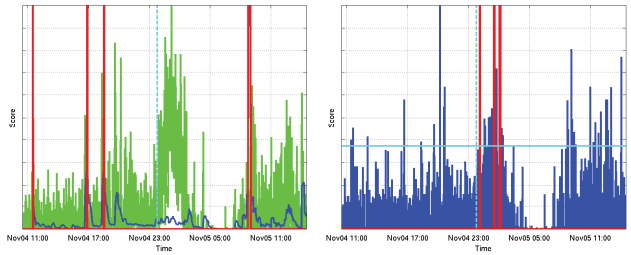
5.4 "NASA" Data Set

This data set is related to the discussion among Twitter users interested in astronomy that preceded NASA's press conference about discovery of an arsenic-eating organism.

TABLE 3
Detection Time and the Number of False Detections

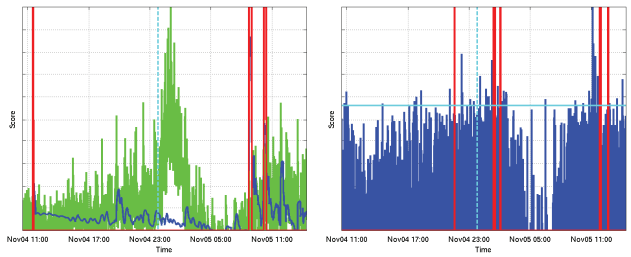
Method		"Job hunting"	"Youtube"	"NASA"	"BBC"
	Event time	22:50, Jan 08	23:48, Nov 04	21:39, Nov 30	17:08, Jan 21
Link-anomaly-based change-point detection	♯ of false alarms	3	3	13	2
	1st detection time	22:55, Jan 08	08:44, Nov 05	**22:20, Nov 30**	**19:52, Jan 21**
Link-anomaly-based burst detection	♯ of false alarms	2	2	26	1
	1st detection time	23:04, Jan 08	*00:07, Nov 05*	*22:44, Nov 30*	*20:51, Jan 21*
Text-anomaly-based change-point detection	♯ of false alarms	4	2	20	2
	1st detection time	*22:51, Jan 08*	08:44, Nov 05	08:17, Dec 01	22:37, Jan 21
Text-anomaly-based burst detection	♯ of false alarms	0	4	18	1
	1st detection time	*22:51, Jan 08*	01:24, Nov 05	00:54, Dec 01	23:14, Jan 21
Keyword-frequency-based dynamic thresholding	♯ of false alarms	0	0	5	0
	1st detection time	22:57, Jan 08	00:30, Nov 05	04:10, Dec 03	22:41, Jan 21
Keyword-frequency-based burst detection	♯ of false alarms	5	14	10	0
	1st detection time	**22:50, Jan 08**	**23:59, Nov 04**	23:59, Dec 02	22:32, Jan 21

The first detection time is defined as the time of the first alert after the event/post that initiated each topic; see also Figs. 5, 6, 7, 8.



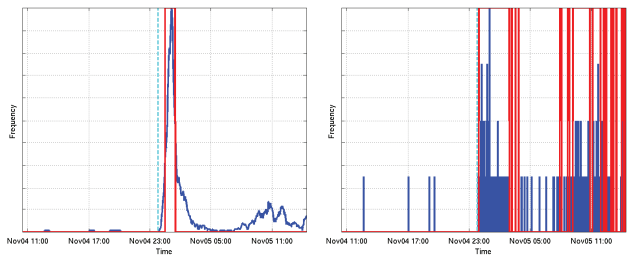
(a) Link-anomaly-based change-point analysis. Green: Aggregated anomaly score (8) at $\tau = 1$ minute. Blue: Change-point score (9). Red: Alarm time.

(b) Link-anomaly-based burst detection. Blue: Aggregated anomaly score (8) at $\tau = 1$ second. Horizontal cyan: threshold for the filtering step in Kleinberg's burst model. Red: Burst state.



(c) Text-anomaly-based change-point analysis. Green: Aggregated anomaly score (8) at $\tau = 1$ minute. Blue: Change-point score (9). Red: Alarm time.

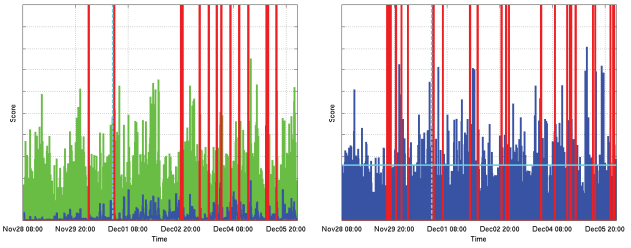
(d) Text-anomaly-based burst detection. Blue: Aggregated anomaly score (8) at $\tau = 1$ second. Horizontal cyan: threshold for the filtering step in Kleinberg's burst model. Red: Burst state.



(e) Keyword-frequency-based change-point analysis. Blue: Frequency of keyword "Senkaku" per one minute. Red: Alarm time.

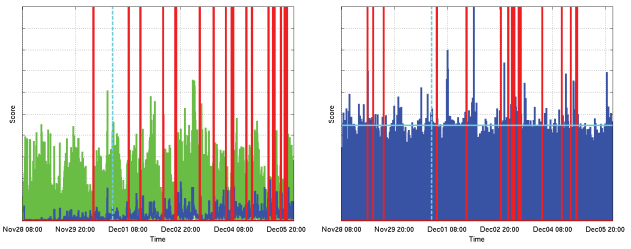
(f) Keyword-frequency-based burst detection. Blue: Frequency of keyword "Senkaku" per one second. Red: Burst state (burst or not).

Fig. 6. Result of "Youtube" data set. The first post about the video leakage was posted at 23:48, Nov 04 (indicated by vertical cyan dashed lines).



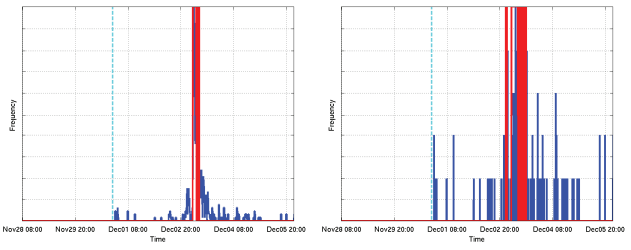
(a) Link-anomaly-based change-point analysis. Green: Aggregated anomaly score (8) at $\tau = 1$ minute. Blue: Change-point score (9). Red: Alarm time.

(b) Link-anomaly-based burst detection. Blue: Aggregated anomaly score (8) at $\tau = 1$ second. Horizontal cyan: threshold for the filtering step in Kleinberg's burst model. Red: Burst state.



(c) Text-anomaly-based change-point analysis. Green: Aggregated anomaly score (8) at $\tau = 1$ minute. Blue: Change-point score (9). Red: Alarm time.

(d) Text-anomaly-based burst detection. Blue: Aggregated anomaly score (8) at $\tau = 1$ second. Horizontal cyan: threshold for the filtering step in Kleinberg's burst model. Red: Burst state.



(e) Keyword-frequency-based change-point analysis. Blue: Frequency of keyword "arsenic" per one minute. Red: Alarm time.

(f) Keyword-frequency-based burst detection. Blue: Frequency of keyword "arsenic" per one second. Red: Burst state (burst or not).

Fig. 7. Result of "NASA" data set. The initial post predicting NASA's finding about arsenic-eating organism was posted at 21:39, Nov 30 much earlier than NASA's official press conference at 04:00, Dec 03.

The keyword used in the keyword-based models was "arsenic." Figs. 7a and 7b show the results of link-anomaly-based change detection and burst detection, respectively. Figs. 7c and 7d show the results of text-anomaly-based change detection and burst detection, respectively. Figs. 7e and 7f show the same results for the keyword-frequency-based methods.

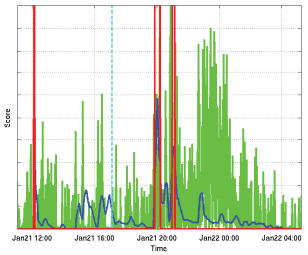
The first alarm times of the two link-anomaly-based methods were 22:20, Nov 30 (change-point detection) and 22:44, Nov 30 (burst detection), respectively. Both of these were earlier than NASA's official press conference (04:00, Dec 03) and were earlier than the text-anomaly-based methods (change-point detection at 08:17, Dec 01,** and burst detection at 00:54, Dec 01). The keyword-based methods are even slower. The keyword-frequency-based dynamic thresholding raised an alarm at 04:10, Dec 03,** after NASA's official press release; burst detection raised an alarm at 23:59, Dec 02; see Table 3.

5.5 "BBC" Data Set

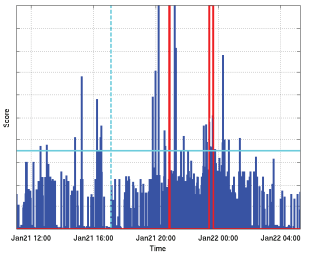
This data set is related to angry reactions among Japanese Twitter users against a BBC comedy show that asked "who is the unluckiest person in the world" (the answer is a Japanese man who got hit by nuclear bombs in both Hiroshima and Nagasaki but survived).

The keyword used in the keyword-based models was "British" (or "Britain"). Figs. 8a and 8b show the results of link-anomaly-based change detection and burst detection, respectively. Figs. 8c and 8d show the results of text-anomaly-based change detection and burst detection, respectively. Figs. 8e and 8f show the same results for the keyword-frequency-based methods.

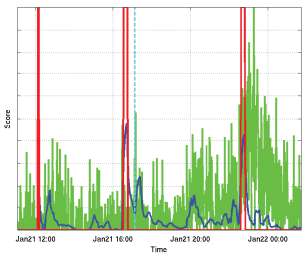
The first alarm times of the two link-anomaly-based methods were 19:52 (change-point detection) and 20:51 (burst detection), both of which were earlier than the text-anomaly-based counterparts at 22:37 (change-point detection) and 23:14 (burst detection). See Table 3.



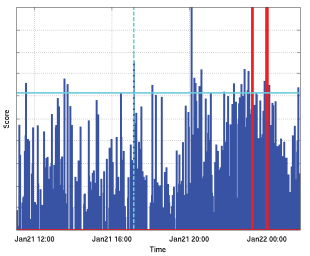
(a) Link-anomaly-based change-point analysis. Green: Aggregated anomaly score (8) at $\tau = 1$ minute. Blue: Change-point score (9). Red: Alarm time.



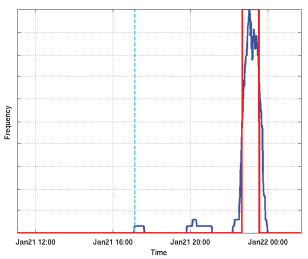
(b) Link-anomaly-based burst detection. Blue: Aggregated anomaly score (8) at $\tau = 1$ second. Horizontal cyan: threshold for the filtering step in Kleinberg's burst model. Red: Burst state.



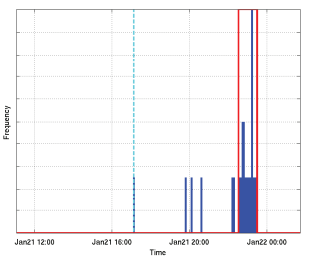
(c) Text-anomaly-based change-point analysis. Green: Aggregated anomaly score (8) at $\tau = 1$ minute. Blue: Change-point score (9). Red: Alarm time.



(d) Text-anomaly-based burst detection. Blue: Aggregated anomaly score (8) at $\tau = 1$ second. Horizontal cyan: threshold for the filtering step in Kleinberg's burst model. Red: Burst state.



(e) Keyword-frequency-based change-point analysis. Blue: Frequency of keyword "British" per one minute. Red: Alarm time.



(f) Keyword-frequency-based burst detection. Blue: Frequency of keyword "British" per one second. Red: Burst state (burst or not).

Fig. 8. Result of "BBC" data set. The first post about BBC's comedy show was posted at 17:08, Jan 21 (indicated by vertical cyan dashed lines).

5.6 Discussion

Within the four data sets we have analyzed above, the proposed link-anomaly-based methods compared favorably against the text-anomaly-based methods on "Youtube", "NASA", and "BBC" data sets. On the other hand, the text-anomaly-based methods were earlier to detect the topics on "Job hunting" data set. The proposed link-anomaly-based methods performed even better than the keyword-based methods on "NASA" and "BBC" data sets.

The above results support our hypothesis that the emergence of new topic is reflected in the anomaly of the way people communicate to each other, and also that such emergence shows up earlier and more reliably in the anomaly of the mentioning behavior than the anomaly of the textual contents. This is probably because the textual words suffer from variations caused by rephrasing, and also because the space of textual words is much larger than the space of twitter users.

TABLE 4
Number of False Alarms for the Proposed Change-Point Detection Method Based on the Link-Anomaly Score (8) for Various Significance Level Parameter Values ρ

ρ	"Job hunting"	"Youtube"	"NASA"	"BBC"
0.01	3	1	8	2
0.05	3	3	13	2
0.1	7	5	29	2

Compared to the keyword-based methods, the above observation is natural, because for "Job hunting" and "Youtube" data sets, the keywords seemed to have been unambiguously defined from the beginning of the emergence of the topics, whereas for "NASA" and "BBC" data sets, the keywords are more ambiguous. In particular, in the case of "NASA" data set, people had been mentioning "arsenic"-eating organism *earlier* than NASA's official release but only rarely (see Fig. 7f). Thus, the keyword-frequency-based methods could not detect the keyword as an emerging topic, although the keyword "arsenic" appeared earlier than the official release. For "BBC" data set, the proposed link-anomaly-based burst model detects two bursty areas (Fig. 8b). Interestingly, the link-anomaly-based change-point analysis only finds the first area (Fig. 8a), whereas the text-anomaly-based methods (Figs. 8c and 8d) and the keyword-frequency-based methods only find the second area (Figs. 8e and 8f). This is probably because there was an initial stage where people reacted individually using different words, and later there was another stage in which the keywords were more unified.

In our approach, the alarm was raised if the change-point score exceeded a dynamically optimized threshold based on the significance level parameter ρ. Table 4 shows the number of false alarms for different threshold parameter values. We see that as ρ increased, the number of false alarms also increased. Meanwhile, even when it was so small, our approach was still able to detect the emerging topics as early as the keyword-based methods. We set $\rho = 0.05$ as a default parameter value in our experiments. Although there are several alarms for "NASA" data set, most of them are more or less related to the emerging topic.

6 Conclusion

In this paper, we have proposed a new approach to detect the emergence of topics in a social network stream. The basic idea of our approach is to focus on the social aspect of the posts reflected in the mentioning behavior of users instead of the textual contents. We have proposed a probability model that captures both the number of mentions per post and the frequency of mentionee. We have combined the proposed mention model with the SDNML change-point detection algorithm [3] and Kleinberg's burst-detection model [2] to pinpoint the emergence of a topic. Since the proposed method does not rely on the textual contents of social network posts, it is robust to rephrasing and it can be applied to the case where topics are concerned with information other than texts, such as images, video, audio, and so on.

We have applied the proposed approach to four real data sets that we have collected from Twitter. The four data sets included a wide-spread discussion about a controversial topic ("Job hunting" data set), a quick propagation of news about a video leaked on Youtube ("Youtube" data set), a rumor about the upcoming press conference by NASA ("NASA" data set), and an angry response to a foreign TV show ("BBC" data set). In all the data sets, our proposed approach showed promising performance. In three out of four data sets, the detection by the proposed link-anomaly-based methods was earlier than the text-anomaly-based counterparts. Furthermore, for "NASA" and "BBC" data sets, in which the keyword that defines the topic is more ambiguous than the first two data sets, the proposed link-anomaly-based approaches have detected the emergence of the topics even earlier than the keyword-based approaches that use hand-chosen keywords.

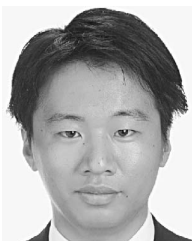
All the analysis presented in this paper was conducted offline, but the framework itself can be applied online. We are planning to scale up the proposed approach to handle social streams in real time. It would also be interesting to combine the proposed link-anomaly model with text-based approaches, because the proposed link-anomaly model does not immediately tell what the anomaly is. Combination of the word-based approach with the link-anomaly model would benefit both from the performance of the mention model and the intuitiveness of the word-based approach.

ACKNOWLEDGMENTS

This work was partially supported by MEXT KAKENHI 23240019, 22700138, Aihara Project, the FIRST program from JSPS, initiated by CSTP, Hakuhodo Corporation, NTT Corporation, and Microsoft Corporation (CORE6 Project).

REFERENCES

[1] J. Allan et al., "Topic Detection and Tracking Pilot Study: Final Report," *Proc. DARPA Broadcast News Transcription and Understanding Workshop,* 1998.
[2] J. Kleinberg, "Bursty and Hierarchical Structure in Streams," *Data Mining Knowledge Discovery,* vol. 7, no. 4, pp. 373-397, 2003.
[3] Y. Urabe, K. Yamanishi, R. Tomioka, and H. Iwai, "Real-Time Change-Point Detection Using Sequentially Discounting Normalized Maximum Likelihood Coding," *Proc. 15th Pacific-Asia Conf. Advances in Knowledge Discovery and Data Mining (PAKDD' 11),* 2011.
[4] S. Morinaga and K. Yamanishi, "Tracking Dynamics of Topic Trends Using a Finite Mixture Model," *Proc. 10th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining,* pp. 811-816, 2004.
[5] Q. Mei and C. Zhai, "Discovering Evolutionary Theme Patterns from Text: An Exploration of Temporal Text Mining," *Proc. 11th ACM SIGKDD Int'l Conf. Knowledge Discovery in Data Mining,* pp. 198-207, 2005.
[6] A. Krause, J. Leskovec, and C. Guestrin, "Data Association for Topic Intensity Tracking," *Proc. 23rd Int'l Conf. Machine Learning (ICML' 06),* pp. 497-504, 2006.
[7] D. He and D.S. Parker, "Topic Dynamics: An Alternative Model of Bursts in Streams of Topics," *Proc. 16th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining,* pp. 443-452, 2010.
[8] H. Small, "Visualizing Science by Citation Mapping," *J. Am. Soc. Information Science,* vol. 50, no. 9, pp. 799-813, 1999.
[9] D. Aldous, "Exchangeability and Related Topics," *École d'Été de Probabilités de Saint-Flour XIII—1983,* pp. 1-198, Springer, 1985.
[10] Y. Teh, M. Jordan, M. Beal, and D. Blei, "Hierarchical Dirichlet Processes," *J. Am. Statistical Assoc.,* vol. 101, no. 476, pp. 1566-1581, 2006.
[11] D. Lewis, "Naive (Bayes) at Forty: The Independence Assumption in Information Retrieval," *Proc. 10th European Conf. Machine Learning (ECML' 98),* pp. 4-15, 1998.
[12] K. Yamanishi and J. Takeuchi, "A Unifying Framework for Detecting Outliers and Change Points from Non-Stationary Time Series Data," *Proc. Eighth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining,* 2002.
[13] J. Takeuchi and K. Yamanishi, "A Unifying Framework for Detecting Outliers and Change Points from Time Series," *IEEE Trans. Knowledge Data Eng.,* vol. 18, no. 4, pp. 482-492, Apr. 2006.
[14] J. Rissanen, "Strong Optimality of the Normalized ML Models as Universal Codes and Information in Data," *IEEE Trans. Information Theory,* vol. 47, no. 5, pp. 1712-1717, July 2001.
[15] T. Roos and J. Rissanen, "On Sequentially Normalized Maximum Likelihood Models," *Proc. Workshop Information Theoretic Methods in Science and Eng.,* 2008.
[16] J. Rissanen, T. Roos, and P. Myllymäki, "Model Selection by Sequentially Normalized Least Squares," *J. Multivariate Analysis,* vol. 101, no. 4, pp. 839-849, 2010.
[17] C. Giurcăneanu, S. Razavi, and A. Liski, "Variable Selection in Linear Regression: Several Approaches Based on Normalized Maximum Likelihood," *Signal Processing,* vol. 91, pp. 1671-1692, 2011.
[18] C. Giurcăneanu and S. Razavi, "AR Order Selection in the Case When the Model Parameters Are Estimated by Forgetting Factor Least-Squares Algorithms," *Signal Processing,* vol. 90, no. 2, pp. 451-466, 2010.
[19] K. Yamanishi and Y. Maruyama, "Dynamic Syslog Mining for Network Failure Monitoring," *Proc. 11th ACM SIGKDD Int'l Conf. Knowledge Discovery in Data Mining,* pp. 499-508, 2005.



Toshimitsu Takahashi received the BEng degree from the University of Tokyo in 2011. He is working toward the master's degree in the Department of Information and Communication Engineering, the University of Tokyo, Japan. His interests include social media and data mining.

Ryota Tomioka received the PhD degree from the University of Tokyo, Japan, in 2008. During the PhD degree, he was a visiting research associate in the Intelligent Data Analysis Group at Fraunhofer FIRST and TU Berlin from 2005 to 2007. After the postdoctoral training in the Department of Computer Science at Tokyo Institute of Technology from 2008 to 2009, he joined the University of Tokyo in 2009. He is a research associate in the Department of Mathematical Informatics at the University of Tokyo, Japan. His research interests include machine learning and optimization.

Kenji Yamanishi received the ME degree from the University of Tokyo in 1987, the DrEng degree from the University of Tokyo in 1992. He is a professor in the Department of Mathematical Informatics at the University of Tokyo, Japan. He was with NEC Corporation from 1987 to 2008. He also worked for the NEC Research Institute in the USA as a visiting scientist from 1992 to 1995. Since 2009, he has been leading the Information-Theoretic Machine Learning and Data Mining Group at the University of Tokyo. He has been engaged in research and development of data mining technologies. He is a member of the IEEE, the ACM, the IEICE, the SITA, and the JSAI.