Sub-event Detection on Twitter Network

Chao Chen and Gabriel Terejanu(✉)

Department of Computer Science and Engineering, University of South Carolina, Columbia, SC 29208, USA
chen288@email.sc.edu, terejanu@cse.sc.edu

Abstract. This work addresses the online detection of sub-events using Twitter stream data. We formulate the process of sub-event identification as an outlier detection problem using three statistical methods: Kalman Filter, Gaussian Process, and Probabilistic Principal Component Analysis. These methods are used to construct the probability distribution of percentage change in the number of tweets. Outliers are identified as future observations that do not fit these predicted probability distributions. Five real-world case studies are investigated to test the effectiveness of the methods. Finally, we discuss the limitations of the proposed frame-work and provide future directions for improvement.

Keywords: Outlier detection · Time-series analysis · Social media mining

1 Introduction

Launched in 2006, Twitter serves as a microblogging platform in which people can publish at most 140 character-long tweets or 10,000 character-long direct messages [1]. Due to its popularity, portability, and ease of use, Twitter quickly has grown into a platform for people sharing daily life updates, chatting, and recording or spreading news. As of September 2015, Twitter announced that there were more than 320 million monthly active users worldwide1[1]. In comparison to conventional news sources, Twitter favors real-time content and breaking news, and it thus plays an important role as a dynamic information source for individuals, companies, and organizations [2].

Since its establishment, Twitter has generously opened a portion of its data to the public and has attracted extensive research in many areas [3–5]. In many studies, the primary task is to identify the event-related tweets and then exploit these tweets to build domain knowledge-related models for analysis. As defined by Atefeh [2], events are generally considered as "real-world occurrences that unfold over space and time". Compared to many data sources, tweets serve as a massive and timely collection of facts and controversial opinions related to specific events [2]. Furthermore, events discussed on Twitter vary in both scale and category, while some may reach to global audiences such as presidential election [6], and others, such as wildfire [7, 8], appeal to local users. In general, studies of Twitter events can be categorized into natural events [3], political events [9], social events [10], and others [11].

[1] https://about.twitter.com/company

L. Iliadis et al. (Eds.): AIAI 2018, IFIP AICT 519, pp. 50–60, 2018.
https://doi.org/10.1007/978-3-319-92007-8_5

Originated from the Topic Detection and Tracking (TDT) program, detection of retrospective or new events has been addressed over two decades from a collection of news stories [12]. Historically, there exist a number of systems developed to automatically detect events from online news [13–15].

An event usually consists of many sub-events, which can describe various facets of it [7]. Furthermore, users tend to post new statuses of an event to keep track of the dynamics of it. Within an event, some unexpected situations or results may occur and surprise users, such as the bombing during the Boston Marathon and the verdict moment of the Zimmerman trial. By building an intelligent system, we can identify these sub-events to quickly respond to them, thus avoiding crisis situations or maximizing marketing impact.

2 Background

Traditionally, unsupervised models and supervised models have been widely applied to detect events from news sources. Clustering methods have been a classic approach for both Retrospective Event Detection (RED) and New Event Detection (NED) since 1990s. According to Allan et al. [12], they designed a single pass clustering method with a threshold model to detect and track events from a collection of digitalized news sources. Chen and Roy [16] also applied clustering approaches such as DBSCAN to identify events for other user-generated contents such as photos.

Additionally, supervised algorithms such as naive Bayes, SVM, and gradient boosted decision trees, have been proposed for event detection. Becker et al. [17] employed the Naive Bayes classifier to label the clustered tweets into event-tweets or non-events tweets with derived temporal features, social features, topical features, and Twitter-centric features, while the tweets are grouped using an incremental clustering algorithm. Sakaki et al. [3] applied the Support Vector Machine to classify tweets into tweets related to target events or not with three key features. Subsequently, they designed a spatial-temporal model to estimate the center of an earthquake and forecast the trajectory of a hurricane using Kalman filtering and particle filtering. Popescu and Pennacchiotti [18] proposed a gradient boosted decision tree based model integrated with a number of custom features to detect controversial events from Twitter streams.

Furthermore, ensemble approaches are also employed to address the event detection problem. Sankaranarayanan et al. [19] first employed a classification scheme to classify tweets into different groups, and then applied a clustering algorithm to identify events.

As argued by Meladianos et al. [1], sub-event detection has been receiving more and more attention from the event research community. For the time being, there are a number of studies dealing with sub-event detection in an offline mode [20]. Zhao et al. [21] adopted a simple statistical approach to detect sub-events during NFL games when tweeting rate suddenly rose higher than a prior threshold. Chakrabarti and Punera [22] developed a two-phase model with a modified Hidden Markov Model to identify sub-events and then derived a summary of the tweets stream. However, their approach has a severe deficiency because it fails to work properly under situations when unseen event types are involved. Zubiaga et al. [20] compared two different approaches for sub-event detection. The first approach measured recent tweeting activities and

identified sub-events if there was a sudden increase of the tweeting rate by at least 1.7 compared to the previous period. The second approach relied on all previous tweeting activities and detected sub-events if the tweeting rate within 60 s exceeded 90% of all previously tweeting rates. As claimed by the authors, the latter outlier-based approach outperformed the first increase-based approach since it neglected situations when there existed low tweeting rates preceded by even lower rates [20].

Nichols et al. [23] provided both an online approach and an offline approach to detect sub-events as well as summarizing important events moments by comparing slopes of statuses updates with a specific slope threshold, which was defined as the sum of the median and three times the standard deviation (median + 3*standard deviation) in their experiment. Shen et al. [24] incorporated "burstiness" and "cohesiveness" properties of tweets into a participant-based sub-event detection framework, and developed a mixture model tuned by EM which yielded the identification of important moments of an event. Chierichetti et al. [25] proposed a logistic regression classifier, to capture the new sub-events with the exploration of the tweet and retweet rates as the features.

In this study, we formalize sub-event detection as an outlier detection problem, where a set of statistical models, Kalman filter (KF), Gaussian process (GP), and probabilistic principle component analysis (PPCA), are used to construct the probability distribution of future observables. Outliers are identified as observations that do not fit these predicted probability distributions. Three real-world case studies (2013 Boston marathon, 2013 NBA AllStar, Zimmerman trial) are investigated to test the effectiveness of the methods. Finally, we discuss the limitations of the proposed framework and provide future directions for improvement.

3 Methodology

Our goal is to model the evolution of the probability distribution of the tweeting change rate (increase/decrease) from period t – 1 to t as defined in the following equation. Each period t spans 30 min and #tweets represents the total number of tweets within that period and filtered for the particular event of interest.

$$v_t = \frac{\#tweets_t - \#tweets_{t-1}}{\#tweets_{t-1} + 1} \tag{1}$$

Three methods (KF, GP, PPCA) described in the following subsections, are evaluated in constructing the probability density function $p(v_{t+1}|v_{1:t} = \{v_t^* \ldots v_1^*\})$. All three approximate the target using a Gaussian density function. This probability distribution is then used to determine whether an observation v_{t+1}^* is an outlier (*denotes the actual observation of percentage change). An observation is labeled as unexpected sub-event when it is identified as an outlier at the 0.025 significance test for the one-tail test.

$$p(v_{t+1} \geq |v^*_{t+1}||v_{1:t}) < 0.025 \tag{2}$$

3.1 Kalman Filter (KF)

Kalman filter and its variants are widely applied in dynamic systems to estimate the state of a system [26, 27]. In this study, we assume that a latent variable h_t related to our quantity of interest - percentage change v_t, evolves with time using the following linear dynamical system.

$$h_t = Ah_{t-1} + \eta^h_t \tag{3}$$

$$v_t = Bh_t + \eta^v_t \tag{4}$$

$$h_1 \sim N(\mu_0, \sigma^2_0) \tag{5}$$

Here, η^h_t is the process noise and η^v_t is the measurement noise. They are as assumed to be independent of one another, temporally independent, and normally distributed according to $N(0, \Sigma_H)$ and $N(0, \Sigma_V)$ respectively. The model parameters $A, B, \Sigma_H, \Sigma_V, \mu_0, \sigma^2_0$ are learned from the data using the Expectation Maximization (EM) algorithm [28].

The initial mean μ_0 and variance σ^2_0 are obtained using data from a 12 h window, and the EM is run on a 12 h moving window to determine the rest of the parameters. After the parameters are obtained, we make a prediction for the next 30 min to compute the probability $p(v_{t+1}|v_{1:t} = \{v^*_t \ldots v^*_1\})$ and test whether the next incoming observation is a sub-event.

3.2 Gaussian Process (GP)

To better capture the non-linearity in the data, we have also tested Gaussian processes. GP is a generalization of a multivariate Gaussian distribution to infinitely many variables [29]. Specifically, a GP defines a distribution over functions, $p(f)$, and f is a mapping function. In this study we use GP to capture the nonlinear relation between several past observations of percentage change and future ones. Namely, we consider the following model.

$$v_t = f(v_{t-1}, v_{t-2}, v_{t-3}) + \epsilon_t \tag{6}$$

Here $f(\cdot) \sim GP(\cdot|0, k)$ and $\varepsilon \sim N(\cdot|0, \sigma^2)$, where $k(\cdot, \cdot)$ is the kernel function. Common choices for kernel function include the squared exponential kernel function, polynomial kernel functions, and sigmoid kernel functions. In this work we have used cubic covariance function, parameters of which are determined using maximum likelihood estimation for each 24 h moving window, where the training data consists of inputs $\{X = \left(v^*_t, v^*_{t-1}, v^*_{t-2}\right)_{t=3\ldots47}, Y = \left(v^*_t\right)_{t=4\ldots48}\}$.

Once the training is completed, the probability density function corresponding to a new input $x_* = (v_t^*, v_{t-1}^*, v_{t-2}^*)$ is obtained via conditioning of the joint as follows.

$$p(v_{t+1}|x_*, X, y) = N(\mu_*, \sigma_*^2)$$
$$\mu_* = K_{*N}(K_N + \sigma^2 I)^{-1} y$$
$$\sigma_*^2 = K_{**} - K_{*N}(K_N + \sigma^2 I)^{-1} K_{N*} + \sigma^2$$

Here, K_N represents the Gram matrix whose entries are given by the kernel function evaluated at the corresponding pairs of inputs in the training data. K_{*N} is a row vector corresponding with kernel function evaluated between the new input x_* are all the training data points, and K_{**} is kernel function evaluated at the new input point.

3.3 Probabilistic Principle Component Analysis (PPCA)

A third model is tested by simply approximating the joint distribution $p(v_t, v_{t-1}, v_{t-2}, v_{t-3})$ using a Gaussian distribution based on 48 samples corresponding to each 24-h moving window. The prediction for the quantity of interest is obtained via conditioning the joint using the past three observations. Since we need to approximate the covariance in 4 dimensions using only 46 samples, we propose to use a more robust estimator such as PPCA than simply computing the sample covariance matrix.

PPCA model is defined as a linear relationship between the 4-dimensional observable $[v_t, v_{t-1}, v_{t-2}, v_{t-3}]^T$ and the M-dimensional latent variable z_n which follows a zero-mean normal distribution with unit covariance matrix [30]. In this study we have set $p = 2$.

$$[v_t, v_{t-1}, v_{t-2}, v_{t-3}]^T = Wz_n + \mu + \epsilon_n \tag{7}$$

Here, W is a 4×2 matrix, μ is the data offset, and ϵ is the projection error, which assumed to follow isotropic Gaussian distribution $\varepsilon \sim N(0, \sigma^2 I)$. We can then obtain the joint distribution of the features by integrating out the latent variables:

$$p[v_t, v_{t-1}, v_{t-2}, v_{t-3}]^T \sim N(\mu, C) \tag{8}$$

Here, the covariance matrix $\mathrm{C} = WW^T + \sigma^2 I$. The parameters W, μ, and σ^2 can be either estimated using the EM approach or by maximizing the following likelihood function [30].

$$\mathrm{L} = -\frac{N}{2} d\ln(2\pi) + \ln|\mathrm{C}| + \mathrm{tr}(\mathrm{C}^{-1}\mathrm{S}) \tag{9}$$

$$\mathrm{S} = \frac{1}{N} \sum\nolimits_{n=1}^{N} (t_n - \mu)(t_n - \mu)^T \tag{10}$$

4 Experiments

Twitter data were collected from Jan. 2, 2013 to Oct. 7, 2014 using Twitter streaming APIs. Then we handpicked three national events during this period, including the 2013 Boston marathon event, the 2013 NBA AllStar event, and the Zimmerman trial event. For these events, we filtered out relevant tweets with pre-specified keywords and hashtags, and provided basic summary of the events shown in Table 1.

For two of the three events, we detected sub-events using data retrieved in one week. However, for the Zimmerman trial event, we missed partial data and thus used data collected in three days that were relevant to the event. Based upon the data, we developed an online detection system that could capture outliers. Figure 1 shows a daily pattern of the number of users and number of tweets for the collected tweets. As the figure indicates, there exist periodic patterns for both the number of tweets and the number of users.

5 Results

As shown in Figs. 2, 3 and 4, the upper sub-plot to the lower sub-plot are outliers identified by the KF, GP, and PPCA algorithms, respectively. Red color indicates actual percentage change of tweets, green color indicates the confidence interval, and cyan color indicates the identified outliers by each algorithm. For the Boston marathon event, as shown in Fig. 2, there were 90, 4, and 7 sub-events detected by the KF, GP, and PPCA algorithms, respectively. 4 of the 90 sub-events identified by KF were labelled as real sub-events, 2 sub-events identified by GP were labelled as real sub-events, and 3 sub-events identified by PPCA were labelled as real sub-events. For this particular event, GP yielded the best precision and 2 of the 4 identified sub-event were real sub-events. Meanwhile, KF achieved the best recall but with many false positives (Table 3).

Outliers of the Zimmerman trial event are visualized in Fig. 3. In terms of the recall value, both KF and PPCA captured 3 of the 10 sub-events, but PPCA achieved a slightly better precision value. In contrast, GP achieved the best precision value.

Figure 4 indicated the identified outliers of the NBA AllStar event. For this event, KF outperformed the other two methods and yielded slightly better recall and precision value. It captured 3 of the 12 sub-events and 3 of the 5 predicted sub-events were real.

A summary of the three picked events are provided in Table 3. Overall, GP and PPCA yield similar F1 score, while GP achieves better recall value and PPCA achieves better precision. KF, in compared to the other two methods, yields the best recall value. This performance is most affected by the Boston event, in which many false positives are identified. More interestingly, we notice that GP provides robust estimates of the uncertainty while the other two methods yield higher uncertainty estimates for time windows after outliers. This observation can be illustrated by the large green confidence bounds after the two spikes in Figs. 2 and 3.

Table 1. Basic information for the picked events

Event	Collection starting time	Event time	Collection ending time	Key words/ hashtags
2013 Boston marathon	04/12/2013 00:00:00	04/15/2013 14:49:00	04/18/2013 23:59:59	marathon, #marathon
2013 NBA AllStar	02/14/2013 00:00:00	02/17/2013 20:30:00	02/20/2013 23:59:59	allstar, all-star
Zimmerman trial	07/12/2013 11:30:00	07/13/2013 22:00:00	07/15/2013 11:30:00	trayvon, zimmerman

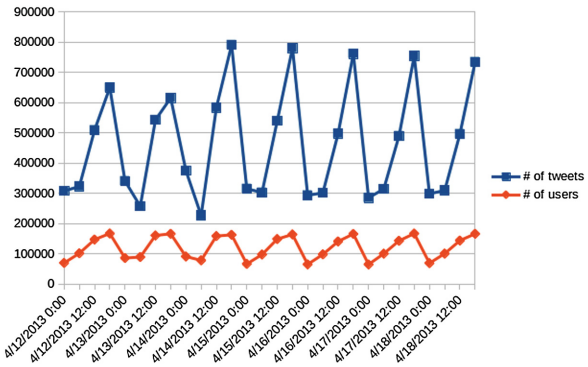

Fig. 1. Daily patterns of the collected tweets during 04/12/2013 and 04/18/2013.

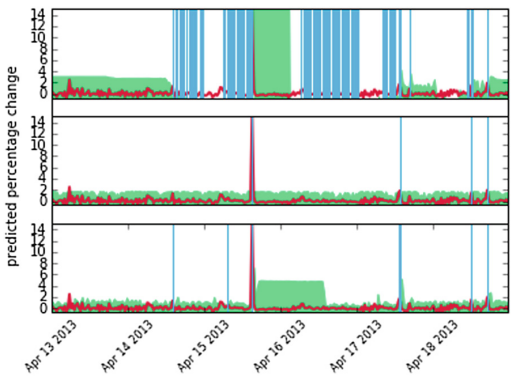

Fig. 2. Predicted sub-events with KF, GP, and PPCA, for the 2013 Boston marathon event. The cyan color indicates the sub-events identified by each algorithm. (Color figure online)

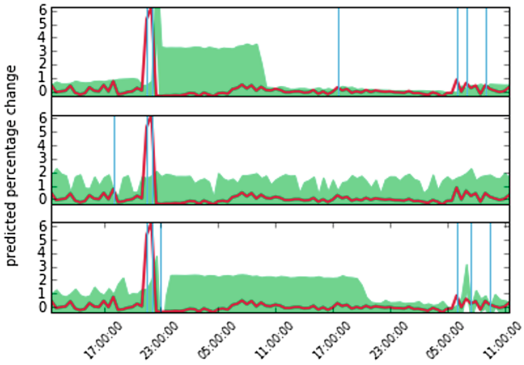

Fig. 3. Predicted sub-events with KF, GP, and PPCA, for the Zimmerman trial event. The cyan color indicates the sub-events identified by each algorithm. (Color figure online)

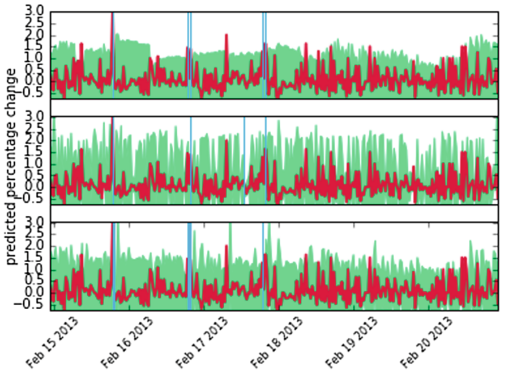

Fig. 4. Predicted sub-events with KF, GP, and PPCA, for the 2013 NBA AllStar event. The cyan color indicates the sub-events identified by each algorithm. (Color figure online)

Table 2. Evaluation metrics of the picked events

	Recall	Precision	F1
KF	0.34	0.10	0.15
GP	0.21	0.55	0.30
PPCA	0.28	0.42	0.33

Table 3. Evaluation metrics of the picked events

			Predicted	
			Sub-event	Non sub-event
KF	Real	Sub-event	10	19
		Non sub-event	92	552
GP	Real	Sub-event	6	23
		Non sub-event	5	639
PPCA	Real	Sub-event	8	21
		Non sub-event	11	633

6 Conclusion

In this study, we explore of building an intelligent system for sub-event detection with three probabilistic models. The sub-events of a point-of-interest event is captured by the system if a new observation is out of the confidence bound of the predictive distribution. We demonstrate the proposed system could capture sub-events with varying performance. The KF model is able to produce slightly better recall, while the GP model is most robust to outliers and yields the best precision. Compared to these two models, PPCA achieves a balanced performance on recall and precision, yielding the best overall F1 score. Nevertheless, we need to interpret the aggregated evaluation with caution because the performance is affected by individualized events, outliers, and the proper choice of parameters. In the future study, we will further tune the parameters, incorporate robust distributions (e.g. t distribution), and take content features into considerations.

References

1. Meladianos, P., Nikolentzos, G., Rousseau, F., Stavrakas, Y., Vazirgiannis, M.: Degeneracy based real-time sub-event detection in Twitter stream. In: Cha, M., Mascolo, C., Sandvig, C. (eds.) ICWSM, pp. 248–257. AAAI Press (2015)
2. Atefeh, F., Khreich, W.: A survey of techniques for event detection in Twitter. Comput. Intell. **31**(1), 132–164 (2015)
3. Sakaki, T., Okazaki, M., Matsuo, Y.: Earthquake shakes Twitter users: real-time event detection by social sensors. In: Proceedings of the 19th International Conference on World Wide Web, WWW 2010, pp. 851–860. ACM, New York (2010)
4. Guo, D., Chen, C.: Detecting non-personal and spam users on geo-tagged Twitter network. Trans. GIS **18**(3), 370–384 (2014)
5. Huang, Y., Guo, D., Kasakoff, A., Grieve, J.: Understanding us regional linguistic variation with Twitter data analysis. Comput. Environ. Urban Syst. **59**, 244–255 (2015)
6. Wang, H., Can, D., Kazemzadeh, A., Bar, F., Narayanan, S.: A system for real-time Twitter sentiment analysis of 2012 U.S. presidential election cycle. In: Proceedings of the ACL 2012 System Demonstrations, ACL 2012, pp. 115–120. Association for Computational Linguistics, Stroudsburg (2012)
7. Pohl, D., Bouchachia, A., Hellwagner, H.: Automatic sub-event detection in emergency management using social media. In: Proceedings of the 21st International Conference on World Wide Web, WWW 2012 Companion, pp. 683–686. ACM, New York (2012)
8. Palen, L., Starbird, K., Vieweg, S., Hughes, A.: Twitter-based information distribution during the 2009 red river valley flood threat. Bull. Am. Soc. Inf. Sci. Technol. **36**(5), 13–17 (2010)
9. Tumasjan, A., Sprenger, T.O., Sandner, P.G., Welpe, I.M.: Election forecasts with Twitter. Soc. Sci. Comput. Rev. **29**(4), 402–418 (2011)
10. Lee, R., Wakamiya, S., Sumiya, K.: Discovery of unusual regional social activities using geo-tagged microblogs. World Wide Web **14**(4), 321–349 (2011)
11. Mathioudakis, M., Koudas, N.: TwitterMonitor: trend detection over the Twitter stream. In: Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data, SIGMOD 2010, pp. 1155–1158. ACM, New York (2010)

12. Allan, J. (ed.): Introduction to Topic Detection and Tracking, pp. 1–16. Kluwer Academic Publishers (2002)
13. Zhang, C., Zhou, G., Yuan, Q., Zhuang, H., Zheng, Y., Kaplan, L.M., Wang, S., Han, J.: GeoBurst: real-time local event detection in geo-tagged tweet streams. In: Perego, R., Sebastiani, F., Aslam, J.A., Ruthven, I., Zobel, J. (eds.) SIGIR, pp. 513–522. ACM (2016)
14. Li, R., Lei, K.H., Khadiwala, R., Chang, K.C.C.: TEDAS: a Twitter-based event detection and analysis system. In: Proceedings of the 2012 IEEE 28th International Conference on Data Engineering, ICDE 2012, pp. 1273–1276. IEEE Computer Society, Washington, DC (2012)
15. Ifrim, G., Shi, B., Brigadir, I.: Event detection in Twitter using aggressive filtering and hierarchical tweet clustering. In: Papadopoulos, S., Corney, D., Aiello, L.M. (eds.) SNOW-DC@WWW, CEUR Workshop Proceedings, vol. 1150, 33–40. CEUR-WS.org (2014)
16. Chen, L., Roy, A.: Event detection from Flickr data through wavelet-based spatial analysis. In: Proceedings of the 18th ACM Conference on Information and Knowledge Management, CIKM 2009, pp. 523–532. ACM, New York (2009)
17. Becker, H., Naaman, M., Gravano, L.: Beyond trending topics: real-world event identification on Twitter. In: Fifth International AAAI Conference on Weblogs and Social Media (2011)
18. Popescu, A.M., Pennacchiotti, M.: Detecting controversial events from Twitter. In: Proceedings of the 19th ACM International Conference on Information and Knowledge Management, CIKM 2010, pp. 1873–1876. ACM, New York (2010)
19. Sankaranarayanan, J., Samet, H., Teitler, B.E., Lieberman, M.D., Sperling, J.: TwitterStand: news in tweets. In: Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, GIS 2009, pp. 42–51. ACM, New York (2009)
20. Zubiaga, A., Spina, D., Amigó, E., Gonzalo, J.: Towards real-time summarization of scheduled events from Twitter streams. In: Proceedings of the 23rd ACM Conference on Hypertext and Social Media, HT 2012, pp. 319–320. ACM, New York (2012)
21. Zhao, S., Zhong, L., Wickramasuriya, J., Vasudevan, V.: Human as real-time sensors of social and physical events: a case study of Twitter and sports games. CoRR abs/1106.4300 (2011)
22. Chakrabarti, D., Punera, K.: Event summarization using tweets. In: ICWSM (2011)
23. Nichols, J., Mahmud, J., Drews, C.: Summarizing sporting events using Twitter. In: Proceedings of the 2012 ACM International Conference on Intelligent User Interfaces, IUI 2012, pp. 189–198. ACM, New York (2012)
24. Shen, C., Liu, F., Weng, F., Li, T.: A participant-based approach for event summarization using Twitter streams. In: Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, 9–14 June 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA, pp. 1152–1162 (2013)
25. Chierichetti, F., Kleinberg, J.M., Kumar, R., Mahdian, M., Pandey, S.: Event detection via communication pattern analysis. In: Proceedings of the Eighth International Conference on Weblogs and Social Media, ICWSM 2014, Ann Arbor, Michigan, USA, 1–4 June 2014 (2014)
26. Terejanu, G., Singh, T., Scott, P.D.: Unscented Kalman filter/smoother for a CBRN puff - based dispersion model. In: 11th International Conference on Information Fusion, Quebec City, Canada, July 2007
27. Ozbek, L., Ozlale, U.: Employing the extended Kalman filter in measuring the output gap. J. Econ. Dyn. Control **29**(9), 1611–1622 (2005)
28. Fletcher, T.: The Kalman filter explained (2010). www.cs.ucl.ac.uk/sta/T.Fletcher/

29. Snelson, E., Ghahramani, Z.: Variable noise and dimensionality reduction for sparse Gaussian processes. In: UAI 2006, Proceedings of the 22nd Conference in Uncertainty in Artificial Intelligence, Cambridge, MA, USA, 13–16 July 2006 (2006)
30. Tipping, M.E., Bishop, C.M.: Probabilistic principal component analysis. J. Royal Stat. Soc. Ser. B **61**, 611–622 (1999)