NRC·CNRC

NRC Publications Archive
Archives des publications du CNRC

Bursty event detection on social media

Cai, Yuanjing; Wang, Yunli; Larkin, Samuel; Goutte, Cyril

This publication could be one of several versions: author's original, accepted manuscript or the publisher's version. / La version de cette publication peut être l'une des suivantes : la version prépublication de l'auteur, la version acceptée du manuscrit ou la version de l'éditeur.

Publisher's version / Version de l'éditeur:

The 7th International Workshop on Natural Language Processing for Social Media, 2019-08-12

NRC Publications Archive Record / Notice des Archives des publications du CNRC :
https://nrc-publications.canada.ca/eng/view/object/?id=76d675b4-d734-4dfa-bd6a-8c05519e0c93
https://publications-cnrc.canada.ca/fra/voir/objet/?id=76d675b4-d734-4dfa-bd6a-8c05519e0c93

Access and use of this website and the material on it are subject to the Terms and Conditions set forth at
https://nrc-publications.canada.ca/eng/copyright
READ THESE TERMS AND CONDITIONS CAREFULLY BEFORE USING THIS WEBSITE.

L'accès à ce site Web et l'utilisation de son contenu sont assujettis aux conditions présentées dans le site
https://publications-cnrc.canada.ca/fra/droits
LISEZ CES CONDITIONS ATTENTIVEMENT AVANT D'UTILISER CE SITE WEB.

Questions? Contact the NRC Publications Archive team at PublicationsArchive-ArchivesPublications@nrc-cnrc.gc.ca. If you wish to email the authors directly, please see the first page of the publication for their contact information.

Vous avez des questions? Nous pouvons vous aider. Pour communiquer directement avec un auteur, consultez la première page de la revue dans laquelle son article a été publié afin de trouver ses coordonnées. Si vous n'arrivez pas à les repérer, communiquez avec nous à PublicationsArchive-ArchivesPublications@nrc-cnrc.gc.ca.

National Research Council Canada | Conseil national de recherches Canada

Canada

Bursty Event Detection on Social Media

Yuanjing Cai [1], **Yunli Wang** [2], **Samuel Larkin**[2] and **Cyril Goutte** [2]
[1] University of Waterloo
[2] National Research Council Canada
y48cai@edu.uwaterloo.ca, {Yunli.Wang, Samuel.Larkin, Cyril.Goutte}@nrc-cnrc.gc.ca

Abstract

Messages posted on social media such as Twitter and Instagram are a rich and promising source of information on real-life events. However, due to the high volume and the noisy nature of posts on social media, the messages reporting events are usually overwhelmed by unrelated daily chatter. To detect unspecified events, many topic modeling and wavelet signal processing methods have been proposed. In this paper, we propose an improved method, BCCED, using a *burstiness* index and *co-occurrence clustering* for event detection, that builds on the Event Detection with Clustering of Wavelet-based Signals (EDCoW) method of Weng et al. [2011]. We compare their performance with two topic modeling methods on two social media datasets. Experiments show that BCCED outperforms these alternatives for unspecified event detection from social media.

1 Introduction

User-generated messages on social media such as Twitter and Instagram are a good source of information on real-life events. Compared with traditional news articles, they contain more abundant information, ranging from global news to local activities, and are more timely. Twitter continues to be the preferred medium for breaking news, consistently leading Facebook or Google Plus [Osborne and Dredze, 2014]. Social media also provides different perspectives on news items than traditional media. Therefore, event detection on social media has received increased attention from researchers and organizations [Atefeh and Khreich, 2015; Hasan *et al.*, 2018].

We adopt the definition of event from Hasan et al. [2018]: an event, in the context of social media, is an occurrence of interest in the real world which instigates a discussion on the event-associated topic by various users of social media, either soon after the occurrence or, sometimes, in anticipation of it. Approaches to event detection can be classified according to event types: specified or unspecified events [Atefeh and Khreich, 2015]. For specified event detection, some information such as time, type or description of target events is known beforehands, while no prior information is available for unspecified event detection. For an example of specified events, Sakaki et al. [2010] detect earthquakes using a supervised method to collect messages related to the events and statistical models to identify the time and location. Identifying unspecified events from social media data requires more effort to filter noisy messages unrelated to actual events. We focus on unspecified event detection since it has broader applications in public safety, public health, etc. We also consider unspecified event detection as a first step to monitor social media data. Further techniques for processing specified events are applicable once they are detected.

For unspecified events, most research is based on monitoring for bursts or trends in social media streams. The challenge is how to extract words with bursty patterns from the vast amount of noise in the text stream, as the majority of messages on social media are unrelated daily chatter. Some studies used Latent Dirichlet Allocation (LDA) [Pozdnoukhov and Kaiser, 2011; Chae *et al.*, 2012] (LDA_MMPP and LDA_STL) or extension of LDA ([Zhou and Chen, 2014]) as an unsupervised methods for unspecified event detection. However, the performance of LDA on short and informal documents such as tweets is often problematic.

Another group of work exploits the temporal patterns or signal of Twitter streams [Weng and Lee, 2011; Li *et al.*, 2012; Schubert *et al.*, 2014]. Weng et al. [2011] proposed Event Detection with Clustering of Wavelet-based Signals (EDCoW): they build wavelet signals which capture the burst of words occurrence, and filter trivial words based on their auto-correlation. Li et al. [2012] used tweet segmentation, event segment detection and event segment clustering three steps to detect unspecified events (Twevent). Although Twevent performs better than EDCoW, Twevent used *newsworthiness* in candidate event filtering, which relied on prior knowledge on Wikipedia. Schubert et al. [Schubert *et al.*, 2014] used EWMA (exponentially weighted average) to evaluate the significance of emerging and trending topics by monitoring the word pair co-occurrences.

In this study, we capture the bursty temporal patterns of tweets in the signal processing step, and also the semantic similarity of word pairs in clustering step. We build our system based on EDCoW, and add two improvements (1) In addition to the burst similarity, we include the co-occurrence similarity to generate word clusters; (2) we rank words clusters using *burstiness* in order to separate events and non-events.

This new method is called *Burstiness and Co-occurrence Clustering for Event Detection*, or BCCED for short. We applied BCCED on two generic short-text streams crawled from Twitter and Instagram. Experimental results show that BCCED improves the precision of event detection on these two datasets, compared to EDCoW and the topic modeling approaches LDA_MMPP and LDA_STL.

2 Methods

In this section, we describe EDCoW and introduce the improvements used in BCCED. Event detection consists of four stages: signal construction using wavelet analysis (Sec. 2.1), signal filtering (Sec. 2.2), graph partitioning (Sec. 2.3), and event ranking (Sec. 2.4). Signal construction and graph partitioning are common to both methods.

2.1 Signal Construction with Wavelet Analysis

For each unique word w in the corpus, the signal is built from the stream of messages in two stages. Assuming the stream starts at time t_0 and ends at time t_{max}, the first stage splits it into U intervals of length ℓ: $t_0 < t_1 < \ldots < t_U = t_{\text{max}}$, with $(t_u - t_{u-1}) = \ell, \forall u$. The first-stage signal for word w is constructed as

$$S_w = (s_w(1), s_w(2), ..., s_w(U)) \tag{1}$$

where, $s_w(u), u = 1 \ldots U$ is the DF-IDF (Document Frequency-Inverse Document Frequency) score for word w in time interval $[t_{u-1}, t_u]$, defined as

$$s_w(u) = \frac{N_w(u)}{N(u)} \log \frac{\sum_{i=1}^{U} N(i)}{\sum_{i=1}^{U} N_w(i)} \tag{2}$$

where $N_w(u)$ is the number of messages between t_{u-1} and t_u that contain word w, and $N(u)$ is the total number of messages in the same time interval. Eq. 2 measures the importance of word w in $[t_{u-1}, t_u]$, i.e. $s_w(u)$ takes a high value if word w is used in more messages in this time interval than elsewhere and a lower value otherwise (Fig. 2).

The second-stage signal is built with a non-overlapping sliding window, which covers a number of 1st-stage sample points (Fig. 1). For a sliding window of size Δ, let D_v denote the signal fragment in the sliding window. Each value $s'_w(v)$ in the second-stage signal

$$S'_w = (s'_w(1), s'_w(2), ..., s'_w(V)) \tag{3}$$

is constructed as follows:

1. We compute the Discrete Wavelet Transformation (DWT) of the 1st-stage signal in time window D_v, which is a non-redundant representation of the signal fragment. Let $C_j(k)$ be the k-th wavelet coefficient at scale j obtained from the DWT. The wavelet energy at each scale j can be computed as

$$E_j = \sum_k |C_j(k)|^2 \tag{4}$$

and normalized as *relative wavelet energy*

$$\rho_j = \frac{E_j}{\sum_j E_j}. \tag{5}$$

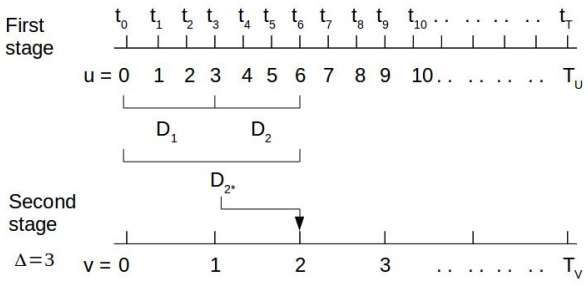

Figure 1: Two stages of signal construction

The distribution $\{\rho_j\}$ is the wavelet energy distribution across different scales j.

2. We evaluate the Shannon Wavelet Entropy (SWE) of the distribution $\{\rho_j\}$,

$$\text{SWE}(D_v) = -\sum_j \rho_j \cdot \log \rho_j \tag{6}$$

which measures the degree of order/disorder of the signal [Rosso *et al.*, 2001], e.g. the wavelet energy of a disordered signal will be evenly distributed across all scales, resulting in a large value of the Shannon Entropy. The H-Measure of the signal is a normalized value of SWE, defined as

$$H_v = \text{SWE}(D_v) / \text{SWE}_{\max} \tag{7}$$

where $\text{SWE}_{\max}$ is the maximum possible Shannon Entropy, $\log N_J$, with N_J the number of scales in the DWT [Rosso *et al.*, 2002].

3. Segments D_{v-1} and D_v are concatenated into a larger segment D_{v*}, whose H-Measure H_{v*} is also obtained. The value of $s'_w(v)$ is calculated as:

$$s'_w(v) = \begin{cases} \frac{H_{v*} - H_{v-1}}{H_{v-1}}, & \text{if } H_{v*} > H_{v-1} \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

$s'_w(v)$ measures how much of the change in wavelet entropy in D_{v*} results from the part of the signal in D_v. If there is no significant difference in 1st-stage signal between D_{v-1} and D_v, $s'_w(v)$ will be low (Fig. 3).

2.2 Signal Filtering and Similarity

Similar burst patterns are detected using the cross-correlation between the 2nd-stage signals for different words. The cross-correlation at time lag 0 between two signals $f(t)$ and $g(t)$ is defined as:

$$(f * g) = \sum_t f(t)g(t) \tag{9}$$

and $(f * f)$ is the auto-correlation of signal f. Calculating cross-correlation between all pairs of words would be prohibitively expensive. Non relevant words are first filtered out by thresholding the auto-correlation using a threshold θ_1.

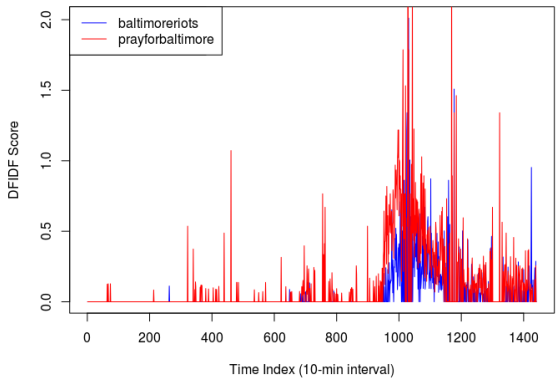

Figure 2: The first stage signals of two hashtags

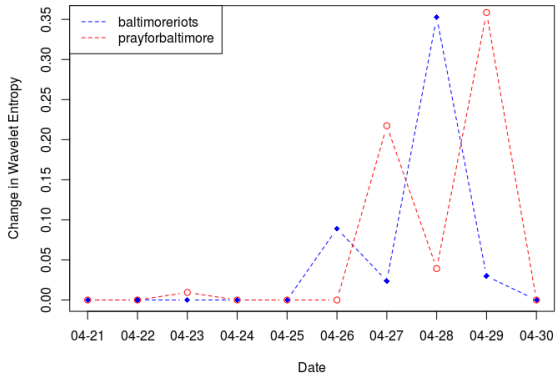

Figure 3: The second stage signals of two hashtags

This exploits the fact that the distribution of auto-correlations is highly skewed, i.e. the majority of auto-correlations are close to zero. Let $A_w = (s'_w * s'_w)$ denote the auto-correlation for word w, and $\mathcal{A}$ the set of all auto-correlations. We pick:

$$\theta_1 = \text{median}(\mathcal{A}) + \gamma_1 \cdot \text{MAD}(\mathcal{A}) \quad (10)$$

where the median absolute difference $\text{MAD}(\mathcal{A}) = \text{median}(|A_w - \text{median}(\mathcal{A})|)$ is a robust measure of the variability in $\mathcal{A}$. We thus filter out the majority of non-relevant signals, leaving only those with extremely high auto-correlations, which makes it possible to perform all pairwise operations. The cross-correlations between all signals $s'_w(v)$ for remaining words w are computed and stored in matrix $\mathbf{X} = [X_{wm}]$, with $X_{wm} = (s'_w * s'_m)$ for all pairs of words (w, m).

EDCoW

The similarity matrix $\mathbf{M}_E$ is simply defined as

$$\mathbf{M}_E = \mathbf{X} \quad (11)$$

Since $\mathbf{M}_E$ is a symmetric matrix, we only need to examine the values in the upper triangular part of it, $\mathbf{M}^{\text{Up}}$. The distribution of elements in $\mathbf{M}^{\text{Up}}$ is also highly skewed. Since we only want to keep the words which are highly associated with other words, and "sparsify" the corresponding word graph, another threshold θ_2 is applied to the elements of $\mathbf{M}^{\text{Up}}$:

$$\theta_2 = \text{median}(\mathbf{M}^{\text{Up}}) + \gamma_2 \cdot MAD(\mathbf{M}^{\text{Up}}) \quad (12)$$

Each cell of $\mathbf{M}_E$ is set to 0 if its value is lower than θ_2, resulting in a sparse symmetric matrix representing the pairwise similarity between words.

BCCED

In addition to the cross-correlation between burst patterns, co-occurrence also plays an important role in measuring word similarity. Words associated with the same event are more likely to be used together. Therefore, BCCED also accounts for word co-occurrence when constructing the similarity matrix.

We measure the co-occurrence similarity C_{wm} between words w and m, by the cosine similarity of two sparse vectors $\boldsymbol{\delta}_w$ and $\boldsymbol{\delta}_m$, where the $k-th$ element of $\boldsymbol{\delta}_w$ is 1 if w appears in message k, and 0 otherwise:

$$C_{wm} = \frac{\sum_k \delta_{wk}\delta_{mk}}{\|\boldsymbol{\delta}_w\|\|\boldsymbol{\delta}_m\|}, \quad \|\boldsymbol{\delta}_\cdot\| = \sqrt{\sum_k \delta_{\cdot k}^2} \quad (13)$$

The co-occurrence similarities are also arranged in a square matrix $\mathbf{C}$, and the overall similarity for BCCED is defined as:

$$\mathbf{M}_B = \mathbf{X} + \alpha \cdot \mathbf{C} \quad (14)$$

Here α controls the trade-off between the burst similarity and the co-occurrence similarity. We finally apply the threshold θ_2 as previously (Eq. 12) in order to sparsify the similarity matrix $\mathbf{M}_B$.

2.3 Graph Partitioning

From a graph theoretical point of view, the similarity matrix $\mathbf{M}$[1] can be viewed as the adjacency matrix of a sparse undirected weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{M})$. The vertex set $\mathcal{V}$ contains the words after filtering. The edge set is $\mathcal{E} = \mathcal{V} \times \mathcal{V}$. The edge between two vertices v_w and v_m has weight M_{wm}.

Word clustering can then be formulated as a graph partitioning problem. Each subgraph corresponds to a potential event, which contains a subset of words that are highly similar, while the similarity between words in different subgraphs are expected to be low. We used the *walktrap* algorithm [Pons and Latapy, 2005] to find communities in this undirected graph $\mathcal{G}$, where *modularity* [Newman, 2004] is used to measure the quality of the partitioning.

2.4 Event Ranking

Graph partitioning results in a number of word clusters $\mathcal{C}$ (see Fig. 4) that each may relate to an event. The last stage of the methods is to decide which clusters actually do relate to events.

[1]This may be $\mathbf{M}_E$ or $\mathbf{M}_B$ depending on the method.

EDCoW

The EDCoW method uses the *significance* ϵ of a cluster $\mathcal{C}$ to select it as an event:

$$\epsilon = \left(\sum_{w,m \in \mathcal{C}} M_{wm} \right) \times \frac{e^{1.5|\mathcal{C}|}}{(2|\mathcal{C}|)!}, \tag{15}$$

where $|\mathcal{C}|$ is the number of words in the cluster. This sums all cross-correlations between each pair of words in the cluster, from similarity matrix $\mathbf{M}_E$, with a penalty against large clusters. Clusters with ϵ above a threshold are output as events.

BCCED

We observe that many highly correlated words (yielding high ϵ) are not associated with an event. For example, "*friday*" and "*tgif*" have similar burst patterns, but do not indicate a significant, atypical event. They just relate to a regular Twitter topic. In addition, the penalty in Eq. 15 assumes that events are not likely to be associated with many words. However, thresholds θ_1 and θ_2 in section 2.1 have a large impact on the size of word clusters. If the thresholds are lower and many words remain after filtering the word clusters will be larger, and it seems unreasonable that they become less significant due to the size penalty.

In BCCED, we use the *burstiness* of word clusters to rank them and separate events from non-events. Our working assumption for events of interest is that the DF-IDF score of event keywords is significantly higher during a time interval that covers the event, than outside, cf. Fig. 2. For event word w, we expect S_w to peak during the event and be low and stable the rest of the time. We define the *burstiness index* to reflect this intuition. The DF-IDF score for cluster $\mathcal{C}$ at interval u is

$$s_{\mathcal{C}}(u) = \frac{N_{\mathcal{C}}(u)}{N(u)} \log \frac{\sum_{i=1}^{U} N(i)}{\sum_{i=1}^{U} N_{\mathcal{C}}(i)} \tag{16}$$

This is analogous to Eq. 16, where $N_{\mathcal{C}}(u)$ is the number of words in cluster $\mathcal{C}$ used in messages from time interval u, summed over all messages and divided by the number of words in $\mathcal{C}$. The *burstiness* of cluster $\mathcal{C}$ is given by

$$B(\mathcal{C}) = \frac{\sigma_s(\mathcal{C}) - \mu_s(\mathcal{C})}{\sigma_s(\mathcal{C}) + \mu_s(\mathcal{C})}, \tag{17}$$

where μ_s (resp. σ_s) is the avergae (resp. standard deviation) of $s_{\mathcal{C}}(u)$, over u. The *burstiness* index is bounded between -1 and +1, and its magnitude correlates with the signal's burstiness, as bursty signals have a large standard deviation w.r.t. their average [Goh and Barabási, 2008]. We are only interested in clusters that exhibit at least some burstiness in the time period of interest, i.e. the word clusters with $B(\mathcal{C}) > 0$.

3 Experiments and Results

We compare the performance of BCCED, EDCoW and two topic modeling methods on two datasets collected in house: a Baltimore dataset from Instagram and a Toronto dataset from Twitter. Since we focus on unspecified event detection, we collected all messages within the geographic boundary of these two cities, and aim to detect any significant events during the time spanned of the messages.

3.1 Data Sets

The Baltimore dataset contains 385,595 Instagram messages collected in Baltimore, MD, USA from April 1 to May 31, 2015. After removing all non-ASCII characters, URLs, mentions of Instagram users (@username), stop words, and words with certain patterns repeated more than twice (e.g. "booo", "hahahaaa"), there are 358,458 messages and 218,281 unique words left. Since rare words are not likely to be associated with any event, we remove words that appear less than 5 times overall, and 9,033 unique words are left. The Toronto dataset contains 312,836 Twitter messages collected in Toronto from May 17 to May 31, 2018. After a similar preprocessing, there are 231,773 messages and 81,351 unique words left.

3.2 BCCED on Baltimore Data Set

In Baltimore dataset, we set $\ell = 10$ minutes as interval length for the 1st-stage signal, and $\Delta = 144$ so that each 2nd-stage time point captures the daily burst patterns of individual words. Figure 3 shows the second stage signals for "*baltimoreriots*" and "*prayforbaltimore*" from April 21–30, 2015. During that time, a protest broke out in Baltimore following the arrest for of Freddie Gray and his subsequent death.

After filtering signals with auto correlation less than θ_1 ($\gamma_1 = 15$, Eq. 10), there are 204 words left on which the similarity matrix is constructed ($\alpha = 5$). We filter the similarity matrix by setting all values below θ_2 to zero ($\gamma_2 = 15$, Eq. 12).

We construct a weighted graph with $\mathbf{M}_B$ as adjacency matrix and partition it into subgraphs (Fig. 4). Each subgraph of size ≥ 2 corresponds to a potential event. Finally, we classify word clusters as event or non-event based on their burstiness index. Table 1 displays all the detected events in decreasing order of burstiness. They include major political issues and local entertainment events. The major event, 2015 Baltimore riots, is split across several events that cover sub-events with different timelines (e.g. news reporting). Bursty events related to local music (2015 Deathfest) and sport (baseball) events also gathered attentions from social media users at that time.

3.3 BCCED on Toronto Dataset

In Toronto dataset, we also used a 10-minute interval in the first stage of signal construction. In the second stage, we used $\Delta = 72$, i.e. 12 hour distance between 2 consecutive time points, as the time span for this dataset is relatively short (only 13 days). Using a smaller Δ provides more data points for the 2nd stage signal. After filtering the signals based on auto correlations ($\gamma_1 = 12$), there are 389 remaining non-trivial words. Setting $\alpha = 7$ and filtering the similarity matrix with $\gamma_2 = 15$, only 81 words with high similarity to others are left.

Results for the Toronto dataset are shown in Table 2. Similarly, the detected events for the Toronto dataset include political campaign, sport events or accidents.

3.4 Topic Modeling Methods

LDA_STL and LDA_MMPP both use LDA to detect topics first, and then use statistical methods to filter noise. Their comparison highlights the effectiveness of the underlying statistical methods STD and MMPP in filtering non-event topics.

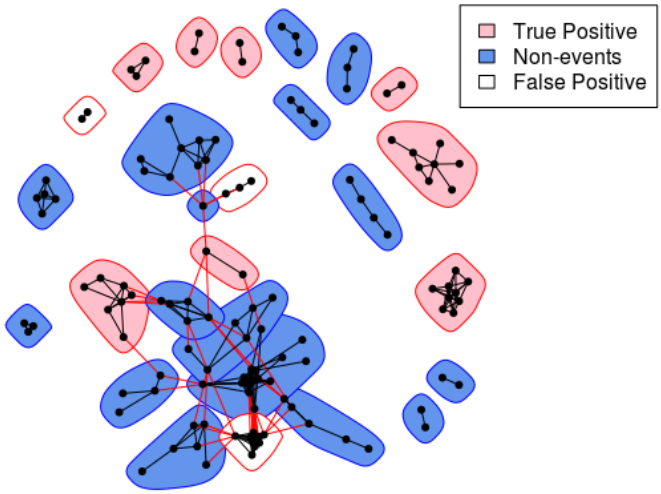

Figure 4: The results of graph partitioning

Words	Event	Date	Burstiness
payback, wwebaltimore	"WWE Payback" wrestling match	17-may	0.684
nosebleed, anb, agoraphobic	2015 Maryland Deathfest	21-may 24-may	0.676
deathmetal, electricwizard, doom, blackmetal, portal, marylanddeathfest, suffocation,demilich	2015 Maryland Deathfest	21-may 24-may	0.534
mobilephotography signofzodiac	- -	- -	0.510
paparazzi, cartel, chroma	-	-	0.261
city, police, justice, prayforbaltimore, freddegray, baltimoreriots, baltimoreuprising	2015 Baltimore Protest	18-apr 03-may	0.256
onedirection, imagine, harry,zayn, louis, sauce, fans, london, follow, sos	2015 Baltimore Protest	-	0.183
news, protests	2015 Baltimore Protest	18-apr 03-may	0.156
mondawmin, exclusive	Violence students vs. police Mondawmin Mall	27-apr	0.080
tatoo, eastcoast	2015 Baltimore Tattoo Arts Convention	10-apr 12-apr	0.054
game, baseball, orioles, baltimoreorioles, camdenyards, o's, birdland, os, letsgoos	Baseball games	misc.	0.013

Table 1: Detected events in Baltimore dataset using BCCED

LDA_STL is based on locally-weighted regression (LOESS). The time series of each topic can be modeled as the sum of three components: a trend component, a seasonal component, and a remainder [Chae *et al.*, 2012]. The remainder is used to detect anomalous outliers within each topic time series. A z-score is calculated as a multiple-day moving average of the remainder values. A value of $|z| > 1.96$ at a specific date indicates an event. The results of LDA_STL on the Baltimore dataset are shown in Table 3. Out of these 10 topics, LDA_STL identifies 23 different events ("Date" column). Only about half of these correspond to true identified unspecified events. In Toronto dataset, the number of topics is again set to 10. LDA_STL detected a total of 14 events with $|z| > 1.96$, out of which only 3 are recognized as actual events.

Words	Event	Date	Burstiness
orton, ladysbridge	Fire near Orton Park road & Ladysbridge drive in Scarborough	17-may	0.510
savelucifer, lucifer	Campaign against Fox's cancellation of "Lucifer" show	misc.	0.385
nba, warriors, finals, cavs, rockets	2018 NBA finals	31-may	0.172
vegas, cup, golden, knights, stanley	2018 Stanley Cup finals game 1	28-may	0.030
uber, app, checked	-	-	0.020
onpoli, onelxn	2018 Ontario General Election	26-may 27-may	0.004

Table 2: Detected events in Toronto dataset using BCCED

In LDA_MMPP, the time series of each topic is modeled as a Markov Modulated non-homogeneous Poisson process. Let $N(t)$ be the observed volume of messages at time t for a topic. $N(t)$ is decomposed into a periodic routine $N_0(t)$ and an additive novel process $N_E(t)$ [Pozdnoukhov and Kaiser, 2011]. For event-related topics, $N_E(t)$ is very localized in time, we improve LDA_MMPP by using K-shape time series clustering to separate event from non-event topics. The results of LDA_MMPP on the Baltimore dataset are shown in Figure 5. Seven topics are clustered in the event group, out of which five topics are identified as true positives. In Toronto dataset, using LDA_MMPP, all ten topics have burstiness above 0, but only 3 of them are true positives.

Topic	z-score	Date	Event
orioles,birdland, game,o's,baseball	3.84	10-apr	Baltimore Orioles vs. Toronto Blue Jays
dinner,chicken,food, lunch,breakfast	2.00 2.78 -2.33	18-apr 5-may 10-may	- - -
mdf, fashion, marylanddeathfest, style, louis	2.51 2.15 3.14 -2.50	4-may 23-may 24-may 26-may	2015 Maryland Deathfest
shit,good, nigga,fuck,wit	2.54 2.34	27-apr 28-apr	- -
tickets,party, size,event,shop	2.02 3.10	10-apr 18-apr	- -
life,god,people, things,love	2.75	10-may	-
freddiegray,city, blacklivesmatter, baltimoreuprising, prayforbaltimore	-2.11 2.41 4.49 -1.98	26-apr 27-apr 28-apr 4-may	2015 Baltimore protest
happy,birthday, love,day,mother's	4.38 2.40	10-may 16-may	Mother's Day -
hair,free, tattoo, ladies,hookah	2.94 -2.23 2.33	11-apr 13-apr 4-may	2015 Baltimore Tattoo Arts Convention
fitness,health,herbalife, workout,nutrition	2.29 2.29	11-apr	- -

Table 3: Detected events in Baltimore dataset using LDA_STL

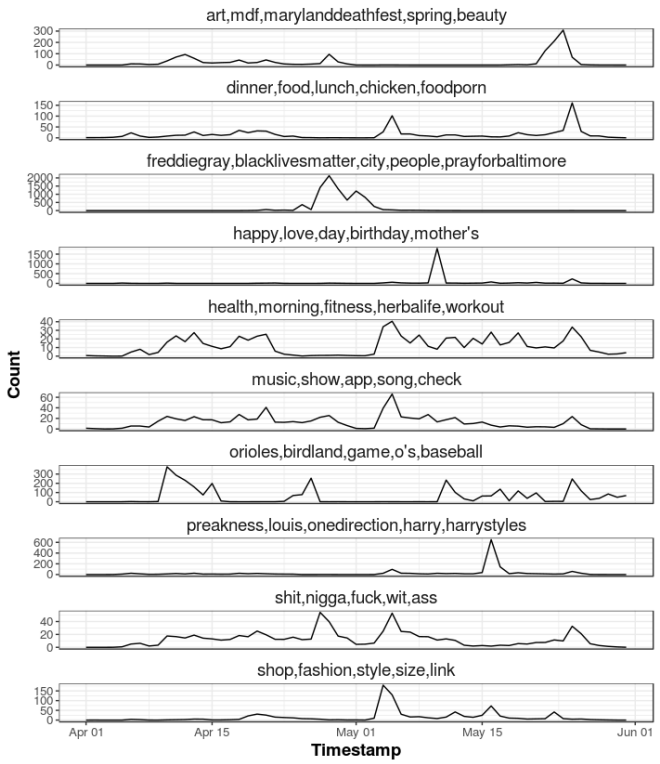

Figure 5: Detected topics in Baltimore dataset using LDA_MMPP

3.5 Evaluation

Reference events in Baltimore (Apr-May, 2015) and Toronto (May, 2018) are not available. For unspecified events, collecting a complete list of local events is always a challenge. Therefore, we only use precision to evaluate and compare the performance of these event detection methods. The precision is defined as the ratio between detected true events and detected events. The detected events were manually examined to check whether they were true events by searching from Wikipedia and Google.

The overall performance of LDA_STL, LDA_MMPP, EDCoW, BCCED is summarized in Table 4. BCCED reaches the best performance on both datasets, often by a large margin. On the Baltimore dataset, the performance of LDA_MMPP is quite close to BCCED. On the Toronto dataset, all alternatives perform clearly below BCCED.

Dataset	LDA_STL	LDA_MMPP	EDCoW	BCCED
Baltimore	0.435	0.714	0.50	0.730
Toronto	0.214	0.300	0.143	0.830

Table 4: Performance (precision) of event detection methods

4 Discussion

In this section, we discuss some observations from our experiments, our contributions, and limitations of BCCED.

4.1 BCCED vs. EDCoW

BCCED brings two key improvements over EDCoW. First it adds word co-occurrence to the cross-correlation between burst patterns in the computation of the word similarity matrix. Second, it uses a burstiness index to rank potential events and separate them from non-events.

Word co-occurrence similarity represents semantic similarity between words. We tested to generate the co-occurrence similarity matrix using pre-trained GloVe word embedding on Twitter [Pennington *et al.*, 2014]. The precision of using the co-occurrence similarity from pre-trained word embedding dropped to 33%. Also, we trained two hundred dimensional fastText word embedding [Mikolov *et al.*, 2017] on Baltimore dataset and generate co-occurrence similarity matrix using fastText word embedding, the performance did not improve. Pre-trained word embeddings do not seem to work for our task, maybe due to two reasons: many words such as "baltimoreriots" cannot be found in pre-trained word embedding; pre-trained word embedding only reflects the global semantic similarity between words. Hashtags and compound words are generated on social media with emerging of new events, so word embedding does not capture the semantic similarity between words at dynamic social media environment. On the other hand, word embedding trained on our datasets might represent the local semantic similarity between words on the entire dataset, but does not represent the co-occurrence similarity between words in a focused time window. Using co-occurrence similarity in BCCED is a straightforward approach representing semantic similarity between words with similar temporal patterns.

In our datasets, we observe many words have relatively large auto-correlation but do not relate to events. They typically represent regular social media content such as hashtags that are more active on specific days of the week (e.g. *#tgif*, *#wcw*, *#mcm*). They exhibit weekly bursts, which results in large auto-correlations. Thus, among word clusters from graph partitioning, many clusters actually correspond to regular topics on social media. We use *burstiness* to filter non-event word clusters and further improve the precision of BCCED.

4.2 Limitations

EDCoW and BCCED are both sensitive to parameters. Among the five parameters (ℓ, Δ, γ_1, α and γ_2), the length ℓ of the time window in the 1st stage of signal construction has the largest impact. If we vary the time window ℓ, and adjust Δ so that two consecutive points in the 2nd stage signal stay 1 day apart, ℓ has a large impact on the auto correlation of word signals: as it increases, the distribution of the auto-correlations becomes less heavy-tailed, so auto-correlations increase. Since the filtering stage using auto-correlation directly impacts which words will remain for clustering, it is essential that event-related words have a relatively high auto-correlation, and thus the choice of ℓ will be critical.

4.3 Bursty patterns vs. LDA methods

BCCED detects events in short-text data such as tweets more effectively compared to the two topic modeling methods. One key advantage of methods such as BCCED is that they capture

the bursty patterns of words and filter most irrelevant words before clustering semantic similar words. By contrast, topic modelling approaches group co-occurrence words first and then filter noise using statistical methods.

Another noticeable difference between all methods is the last step of event ranking and/or detection: z-score for LDA_STL, k-shape clustering in LDA_MMPP, *significance* for EDCoW and *burstiness* in BCCED. The experiments show that *burstiness* separates events and non-events better than other methods.

5 Conclusion

In this study, we address the problem of detecting unspecified events from social media using unsupervised methods. Two classes of unsupervised methods were compared: singnal processing and topic modeling methods. Based on EDCoW, we proposed several improvements in the BCCED method. By combining the cross-correlation of word usage patterns and word co-occurrence similarity, we greatly improve word clustering results in two generic and noisy datasets extracted from social media. The use of the *burstiness* index helps rank and select relevant target events more accurately than EDCoW. In addition, we compared the signal processing methods with two alternatives relying on topic modeling, LDA_STL and LDA_MMPP. Results from empirical studies show that BCCED outperforms all alternatives, reaching 73% and 83% precision on our two social media datasets. For future work, we would like to explore the use of ensemble methods to further improve the precision of event detection, and seek better evaluation methods for unspecified event detection.

Acknowledgments

References

[Atefeh and Khreich, 2015] Farzindar Atefeh and Wael Khreich. A survey of techniques for event detection in twitter. *Computational Intelligence*, 31(1):132–164, 2015.

[Chae *et al.*, 2012] Junghoon Chae, Dennis Thom, Harald Bosch, Yun Jang, Ross Maciejewski, David S Ebert, and Thomas Ertl. Spatiotemporal social media analytics for abnormal event detection and examination using seasonal-trend decomposition. In *Visual Analytics Science and Technology (VAST), 2012 IEEE Conference on*, pages 143–152. IEEE, 2012.

[Goh and Barabási, 2008] K.-I. Goh and A.-L. Barabási. Burstiness and memory in complex systems. *EPL (Europhysics Letters)*, 81(4):48002, 2008.

[Hasan *et al.*, 2018] Mahmud Hasan, Mehmet A. Orgun, and Rolf Schwitter. A survey on real-time event detection from the twitter data stream. *Journal of Information Science*, 44(4):443–463, 2018.

[Li *et al.*, 2012] Chenliang Li, Aixin Sun, and Anwitaman Datta. Twevent: segment-based event detection from tweets. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 155–164. ACM, 2012.

[Mikolov *et al.*, 2017] Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. Advances in pre-training distributed word representations. *arXiv preprint arXiv:1712.09405*, 2017.

[Newman, 2004] Mark E.J. Newman. Fast algorithm for detecting community structure in networks. *Physical review E*, 69(6):066133, 2004.

[Osborne and Dredze, 2014] Miles Osborne and Mark Dredze. Facebook, twitter and google plus for breaking news: Is there a winner? In *Proc. Intl. Conf. on Web and Social Media*, 2014.

[Pennington *et al.*, 2014] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[Pons and Latapy, 2005] Pascal Pons and Matthieu Latapy. Computing communities in large networks using random walks. In *International symposium on computer and information sciences*, pages 284–293. Springer, 2005.

[Pozdnoukhov and Kaiser, 2011] Alexei Pozdnoukhov and Christian Kaiser. Space-time dynamics of topics in streaming text. In *Proceedings of the 3rd ACM SIGSPATIAL international workshop on location-based social networks*, pages 1–8. ACM, 2011.

[Rosso *et al.*, 2001] Osvaldo A Rosso, Susana Blanco, Juliana Yordanova, Vasil Kolev, Alejandra Figliola, Martin Schürmann, and Erol Başar. Wavelet entropy: a new tool for analysis of short duration brain electrical signals. *Journal of neuroscience methods*, 105(1):65–75, 2001.

[Rosso *et al.*, 2002] O. A. Rosso, M. T. Martin, and A. Plastino. Brain electrical activity analysis using wavelet-based informational tools. *Physica A: Statistical Mechanics and its Applications*, 313(3-4):587–608, 2002.

[Sakaki *et al.*, 2010] Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. Earthquake shakes twitter users: real-time event detection by social sensors. In *Proceedings of the 19th international conference on World wide web*, pages 851–860. ACM, 2010.

[Schubert *et al.*, 2014] Erich Schubert, Michael Weiler, and Hans-Peter Kriegel. Signitrend: scalable detection of emerging topics in textual streams by hashed significance thresholds. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 871–880. ACM, 2014.

[Weng and Lee, 2011] Jianshu Weng and Bu-Sung Lee. Event detection in twitter. In *Proc. Intl. Conf. on Web and Social Media*, volume 11, pages 401–408, 2011.

[Zhou and Chen, 2014] Xiangmin Zhou and Lei Chen. Event detection over twitter social media streams. *The VLDB Journal—The International Journal on Very Large Data Bases*, 23(3):381–400, 2014.