Expert Systems With Applications 136 (2019) 115–132

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Enhanced Heartbeat Graph for emerging event detection on Twitter using time series networks

Zafar Saeed[a,b], Rabeeh Ayaz Abbasi[a,*], Imran Razzak[b], Onaiza Maqbool[a], Abida Sadaf[c], Guandong Xu[b]

[a] Department of Computer Science, Quaid-i-Azam University, Islamabad, Pakistan
[b] Advanced Analytics Institute, University of Technology Sydney, Australia
[c] Institute of Information Technology, Quaid-i-Azam University, Islamabad, Pakistan

ARTICLE INFO

Article history:
Received 8 October 2018
Revised 9 April 2019
Accepted 3 June 2019
Available online 11 June 2019

Keywords:
Event detection
Twitter
Text stream
Emerging trends
Dynamic graph
Time series network
Big data

ABSTRACT

With increasing popularity of social media, Twitter has become one of the leading platforms to report events in real-time. Detecting events from Twitter stream requires complex techniques. Event-related trending topics consist of a group of words which successfully detect and identify events. Event detection techniques must be scalable and robust, so that they can deal with the huge volume and noise associated with social media. Existing event detection methods mostly rely on burstiness, mainly the frequency of words and their co-occurrences. However, burstiness sometimes dominates other relevant details in the data which could be equally significant. Besides, the topological and temporal relationships in the data are often ignored. In this work, we propose a novel graph-based approach, called the Enhanced Heartbeat Graph (EHG), which detects events efficiently. EHG suppresses dominating topics in the subsequent data stream, after their first detection. Experimental results on three real-world datasets (i.e., Football Association Challenge Cup Final, Super Tuesday, and the US Election 2012) show superior performance of the proposed approach in comparison to the state-of-the-art techniques.

1. Introduction and related work

Unprecedented growth of social media and microblogging services in recent years has resulted in mounds of diverse types of data being generated everyday. The value of information generated by people on such platforms is increasing enormously. People use online services to share content about various events they experience in their daily lives.

An *event* is a way of referring to an observable activity at a certain time and place which involves or affects a group of people. Online communication services, such as Twitter and Facebook, hold abundant and diverse contents shared by different people across the world regarding events, hence becoming a new source of information. It is interesting if a large number of users are experiencing and sharing similar content at a specific time interval. Such data contains real-life information with temporal characteristics which evolve over time. Therefore, a temporal text stream is useful in detecting events as it contains information which is shared and propagated by users on social media platforms. Micro-documents related to the same event have a similar set of collocated keywords that can be used to identify the time of the occurrence and its description.

The process of event detection involves processing and monitoring the text stream to identify event-related trending topics (Panagiotou, Katakis, & Gunopulos, 2016). Event detection from social media has become a focus of interest because people share their opinions, experiences, and news on such media. The active users instantly publish and report their experiences when participating in various real-life events. An abundance of meaningful information is produced on social media text streams such as Twitter.

It is interesting to analyze such data in order to extract useful information that can be used to discover events and to identify their characteristics. However, it is difficult to analyze large, noisy, and diverse data due to the challenges of efficiency, accuracy, and scalability (Earle, Bowden, & Guy, 2012; Jarwar et al., 2017).

Generally, there are two approaches (Weng & Lee, 2011): (1) document pivoted and (2) feature pivoted. Methodologically, document pivoted techniques work by grouping documents and feature pivoted techniques cluster important keywords representing event-related information.

* Corresponding author.
E-mail addresses: zsaeed@cs.qau.edu.pk (Z. Saeed), rabbasi@qau.edu.pk (R.A. Abbasi), mirpak@gmail.com (I. Razzak), onaiza@qau.edu.pk (O. Maqbool), abida.sadaf@qau.edu.pk (A. Sadaf), guandong.xu@uts.edu.au (G. Xu).

https://doi.org/10.1016/j.eswa.2019.06.005
0957-4174/

1.1. Document pivot methods

Pivoted document is a classic approach that groups documents into clusters based on their similarity. In this regard, Petrović, Osborne, and Lavrenko (2010) proposed a technique to detect events at early stages (Petrović et al., 2010). Locality Sensitive Hashing (LSH) is used to find the nearest neighbours for clustering. The emerging event is detected if incoming documents in the Twitter stream have low similarity with all the clusters previously detected. Kaleel and Abhari (2015) proposed a technique based on classic IR features (term vector) with a novel indexing mechanism (Kaleel & Abhari, 2015). Due to huge data size, updating the vector when a new word arrives is challenging. A combined approach is used for updating the term vector with incremental tf-idf (term frequency-inverse document frequency). A high dimensional vector is converted into a k-bit signature while preserving the cosine similarity between term vectors which is further used for the clustering. Most frequent terms within a cluster are used for defining the centroid and labeling the cluster.

Similarly, Ozdikis, Senkul, and Oguztuzun (2012) proposed a technique that expands the tf-idf based vector and assigns weights. The weights are not only assigned to existing terms, but also to semantically related terms. To expand the term vectors, two different expansion methods are presented. The first one calculates the co-occurrence of words from the corpus and then term vector of a document is expanded with the words that are co-occurring. The second method creates co-occurrence vector for each word in a document. The cosine similarity between vectors are used to cluster tweets to detect events. The method is further improved by utilizing only tweets containing hashtags. The term vector for each document is expanded by exploiting the co-occurrence of words with hashtags and found improvement in the results especially for targeted events (Ozdikis et al., 2012).

Kumar, Liu, Mehta, and Subramaniam (2015) proposed a method that uses term vector and a social feature *user diversity* which is defined as the entropy of users (Kumar et al., 2015). Tweets are clustered using an online one pass clustering algorithm. The cluster is identified as an event if user diversity is more than a certain threshold. A similar technique based on textual similarity is discussed by Becker, Naaman, and Gravano (2011) that additionally classifies tweets with a binary label referring to event or not. The classifier is trained on Twitter-specific and social features. A drawback of the proposed method is manual annotation of data.

Zhang et al. (2017) proposed a two-step event detection method called TrioVecEvent to detect local events from geo-tagged tweets. First, the geo-topic clusters are obtained using time, location, and text message by employing Bayesian mixture model. Second, clusters are considered as event candidates. The method extracts semantic and spatio-temporal features to characterize local events. Finally, a logistic regression based binary classifier decides a list of local events. Similar spatio-temporal features have been used by Guo and Gong (2017) to propose a parameter-free event discovery model called DP-density. The proposed model uses a Dirichlet process to determine the event diversity and density estimation method learns the influence of temporal stream.

Yin and Wang (2016) proposed a document clustering algorithm called FGSDMM+. The proposed algorithm processes the documents one by one with the underlying assumption of creating K_{max} clusters. A Dirichlet multinomial mixture model is used to calculate the probability of adding a document to a cluster. Each time the algorithm finds a cluster for a document, a new cluster is created to store that document. This process decreases the probability of remaining documents for the potential cluster. Later, the algorithm uses a collapsed Gibbs sampling algorithm iteratively to finalize the clusters.

1.2. Feature pivot methods

Event detection methods based on feature pivot approach focus on statistical modeling of bursty features to extract set of keywords for detecting event-related topics. The main idea behind such techniques is to capture the emerging topics that are previously unseen or rapidly gain attention in the social stream (Yang & Leskovec, 2011). Generally, a text stream is segmented into single words (or n-grams) often represented by bag-of-words (BoW) model. Many research studies (Li, Lei, Khadiwala, & Chang, 2012c; Nguyen & Jung, 2015; Shamma, Kennedy, & Churchill, 2011; Yang & Leskovec, 2011) have used word frequency signals to identify event-related keywords.

To find unusual spikes in the word frequency signals, He, Chang, and Lim (2007) uses Discrete Fourier Transformation (DFT) method to extract strength and periodicity of the power spectrum. The detected anomalous words are then grouped. Weng and Lee (2011) further extend the method to obtain new features using wavelet analysis. Insignificant words are dropped using low signal auto-correlation. A graph-partitioning algorithm is used to identify the event clusters (Weng & Lee, 2011).

Li, Sun, and Datta (2012a) targeted an event detection method based on tweet segments (or phrases) (Li et al., 2012a). The tweet content is split into segments using algorithm proposed in Li et al. (2012b). Tweet segments are grouped into clusters using a content-temporal similarity measure. In addition to the bursty segments, usersâ participation within a time window hints towards a possible event. A threshold (*newsworthiness*) eliminates clusters not related to events, and remaining clusters are identified as events. A similar work is proposed by Aiello et al. (2013) while comparing six different state-of-the-art approaches on three benchmarked datasets and concluded that n-grams produce better results than uni-grams.

Latent Dirichlet Allocation (LDA) (Blei, Ng, & Jordan, 2003) is a probabilistic model that builds over BoW and is widely used for topic modeling. Word frequencies are extracted from documents to create probability distribution of words that are likely to be found in the given topics. Cordeiro (2012) combined LDA with wavelet transform of term frequency signals (Cordeiro, 2012). The BoW only contains hashtags retrieved from the tweets and then grouped over a five minute time interval to generate temporal frequency signals using wavelet transformation. Spikes in the temporal frequency signals are detected using wavelet peak and local maxima detection. Finally, topics are extracted for event description by applying LDA to all the tweets containing hashtags responsible for the peaks in the corresponding time interval. Similarly, Cheng and Wicks (2014) combined LDA and Space-Time Scan Statistics (STSS)[1] (Kulldorff, 2010) to determine a method for detecting events that are spatio-temporal in nature. Initially, Space-Time Permutation Model (STPM) finds the geo-associations to create clusters with respect to the space and time regardless of the contents of the tweets. LDA model is further used on each cluster to classify key terms into groups of topics. The topics discovered by LDA are then mapped on to spatial-temporal clusters to describe spatial-temporal events.

Aforementioned approaches group similar words based on their individual frequency burst. However, it is also important to find a set of terms that are co-occurring frequently. Frequent pattern mining is of particular significance in this regard. Huang, Peng, and Wang (2015) propose a framework called High Utility Pattern Clustering (HUPC) based on association rule mining (Huang et al., 2015). After sorting the frequent patterns in decreasing order with respect to their *support*, the top-k highly

[1] https://www.satscan.org/ (accessed on Septermber 5, 2018).

similar patterns are grouped using k-nearest-neighbors to represent emerging events. Similarly, Adedoyin-Olowe, Gaber, Dancausa, Stahl, and Gomes (2016) consider hashtags that evolve over time as primary feature to define rules for detecting events (Adedoyin-Olowe et al., 2016). The corpus is divided into multiple time-frames of equal temporal coverage. The support and confidence are set to 0.001, which despite being low, allow abundant item-sets of hashtags related to the event to be extracted. The hashtags returned by the association rules are sorted and matched with the ground truth given by Aiello et al. (2013). Event detection occurs when top-k item-sets have at least one keyword similar to the ground truth in the same time frame. However, it is a fact that subsets of a pattern will always have equal or greater *support*. Therefore, subsets of any pattern remain viable to qualify for being topics and it becomes difficult to prune redundant patterns. In the case of Twitter stream, such cases are more likely to appear due to retweeting the popular tweets. Moreover, these techniques are biased toward the highly frequent patterns and often capture misleading associations between keywords.

The methods that detect events using anomalous or similarity patterns are often influenced by the bursty features and ignore the topological and temporal relationship between the keywords in the data. To capture such relationships graph-based methods have been used (Sethi & Kantardzic, 2017; Velampalli & Eberle, 2017).

Long, Wang, Chen, Jin, and Yu (2011) extracted topical words, using bursty features that include word frequency, hashtag frequency, and word entropy. To create co-occurrence graph in which nodes represent micro-documents (tweets) and an edge is created between two micro-documents, if topical words co-occur in both of them (Long et al., 2011). A top-down hierarchical clustering is employed to create event clusters. To observe the change among event clusters at different time windows, a bipartite graph matching algorithm is employed to link clusters across different time windows. Finally, micro-documents are grouped together using cosine similarity from the interlinked clusters to find relevant posts for the event description.

Similarly, Zhang et al. (2015) extracted BoW from micro-documents and weights are assigned to the words using tf-idf and *user authority score* based on follower count (Zhang et al., 2015). To find bursty words, a Hidden Markov Model (HMM) is employed on BoW and binary labels (i.e., *high* and *low*). Words that are labeled as *high* are taken to generate word relation graph. The nodes and the edges represent bursty words and their co-occurrence within each micro-document respectively. Each strongly connected component is considered as an event.

For extracting event-information from live data stream, Nguyen and Jung (2017) extracted meta-information, that includes *posting time, diffusion information, diffusion sensitivity*, and *diffusion degree* from micro-documents in the text stream (Nguyen & Jung, 2017). A directed graph between micro-documents is created where the nodes are tweets and the edges are measured by the similarity between nodes using normalized cross-correlation based on the tweet's meta-features. Density-based spatial clustering is employed to determine event clusters.

1.3. Research gaps and motivation

Traditional event detection methods based on document pivot approach are however less applicable to micro-blogs such as Twitter due to the several reasons such as short document size, abundance of noise, and rapidly changing contents. Such techniques require processing complete data to cluster documents based on their similarities, hence are not scalable. Similarly such methods often depend on an arbitrary threshold for including a new document into an existing event cluster. On the other hand, most of the existing event detection methods which are based on feature pivot focus on frequent patterns, referred to as burstiness (Li et al., 2012c; Nguyen & Jung, 2015; Shamma et al., 2011; Yang & Leskovec, 2011). However, burstiness dominates smaller but relevant details in the data which can be equally significant.

We address the drawback of feature pivot approach by systematically embedding micro-documents into a graph structure. Node and edge weights are measured using a modified Kullback–Leibler (KL) divergence score, also known as relative entropy (Kullback, 1997). The modified KL-divergence makes bursty words and their co-occurrence relationships less dominant in subsequent time intervals. This makes the proposed graph-based features efficient and more sensitive to capture the change in the Twitter stream.

The core idea of proposed approach is to address the dominating nature of burstiness in the data by suppressing the bursty topics once captured. Temporal relationships in the data are measured using a modified KL-divergence of words and their co-occurrences with respect to time. Therefore, the feature design implicitly inherits the characteristics of relative entropy.

The characteristic mentioned above can be observed in Fig. 1 showing the signals of six keywords associated with top events. The data across time slices (i.e. 7:15-8:00 & 9:40-10:20) from the Super Tuesday dataset is visualized against five minutes time interval. Signals in Figs. 1(A_0) and (B_0) are based on a modified KL-divergence score, whereas in Figs. 1(A_1) and (B_1) are based on term-frequency. In Fig. 1(A_1) it can be observed that keywords "romney" and "win" are dominating, whereas in Fig. 1(A_0) both keywords are suppressed once they gain peak. Hence, the other keywords "santorum" and "paul" become visible at 7:45 and 7:50 respectively. Similarly "win" and "santorum" become visible in Fig. 1(B_0), whereas they are dominated by "romney" in Fig. 1(B_1) at time interval 10:15. Furthermore, Section 6.2 empirically discusses the effect of modified KL-divergence on data distribution in detail.

1.4. Contributions

We create temporal graphs based on feature pivot. Selecting useful edges in the graph reduces the graph density, hence improves time complexity of the algorithm as compared to existing co-occurrence graph based approaches (Long et al., 2011; Zhang et al., 2015). Moreover, instead of following the computationally expensive process of merging the event candidate graphs based on a similarity measures (Edouard, Cabrio, Tonelli, & Le Thanh, 2017; Katragadda, Benton, & Raghavan, 2017; Katragadda, Virani, Benton, & Raghavan, 2016), we select unique keywords across the candidate graphs having the highest ranking scores among duplicates to extract the event-related topics.

We develop a detection model incorporating a modified KL-divergence for generating graph structures (see Section 3.4). This results in a new feature i.e., *Divergence Factor* (see Section 3.3). The above model is further extended to weighted graph structure as well. Furthermore, a weighted graph model is also derived from our previous work (Saeed, Abbasi, Sadaf, Razzak, & Xu, 2018). We create and compare four event detection model and the winning model is compared with nine different baseline methods (see Section 6.4). The experiments are extended with a bigger benchmark dataset (US Elections, 2012). We have also performed a detailed analysis to observe the effect of the proposed approach on the data distribution of the Twitter stream (see Section 6.2) and as well as a detailed time complexity analysis is conducted to evaluate the efficiency of the proposed approach (see Section 4).

We describe the theoretical and empirical **key contributions** of this work as follow:

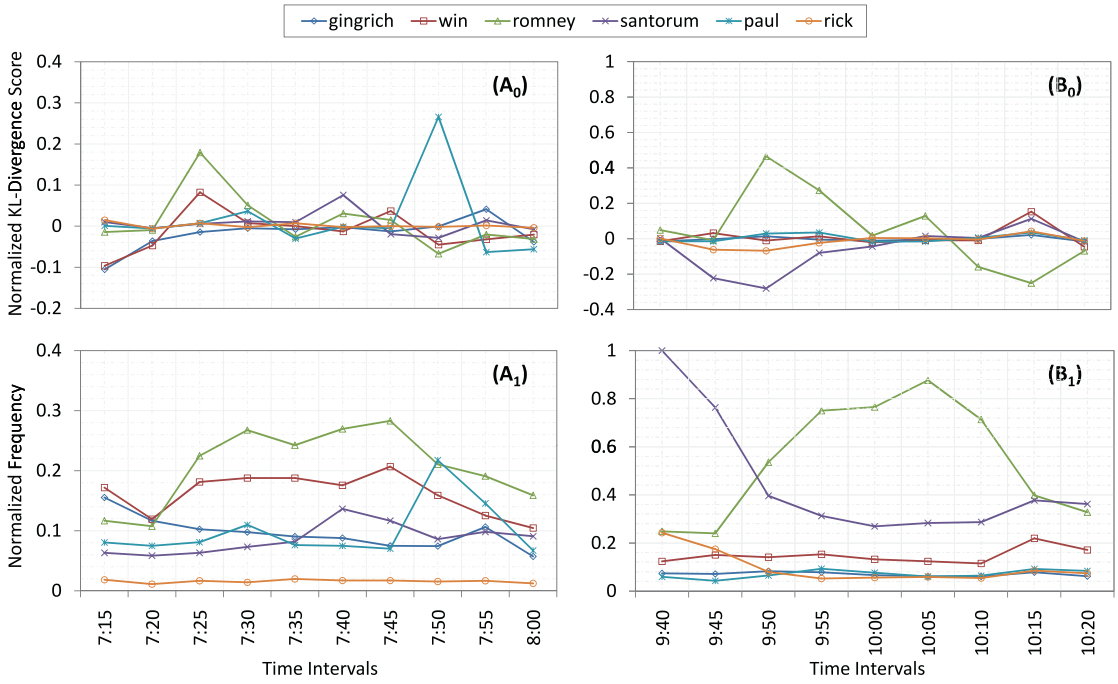


Fig. 1. Motivation: some of the event-related topics are dominating the text stream (in Figs. A_1 and B_1).

- A novel graph-based approach named the Enhanced Heartbeat Graph (EHG) which is efficient in the detection of events from Twitter stream.
- EHG-based feature design for event detection and topic extraction.
- Low computational complexity of the proposed approach which initially transforms the data into a series of EHGs in polynomial time. Later, it detects events in linear time, thus it could also be applied efficiently on live text stream.
- Empirical evaluation of the proposed method on the benchmark event detection datasets: FA Cup, Super Tuesday, and the US Election datasets.
- Comparison of the proposed approach with state-of-the-art event detection approaches.

2. Preliminaries

This section defines all the preliminaries used in the formation of the proposed approach. Our approach specifically focuses on micro-sized documents, such as those published on micro-blogging services, e.g., Twitter and Facebook. We create a series of temporal graphs to detect emerging events in the data stream. Due to the representation of textual data in a series of graphs, where each node in a graph is a unique word, we use the term "word(s)" and "node(s)" interchangeably (Benhardus & Kalita, 2013; Buntain, 2015; Nguyen & Jung, 2017; Zhou, Chen, & He, 2015; Zhou & Chen, 2014). Let: $U = \{u_1, u_2, u_3, ..., u_k\}$ be the set of all users who have published at least one micro-document, $T = \{t_1, t_2, t_3, ..., t_l\}$ be the set of all time instances where at least one micro-document has been published, and $W = \{w_1, w_2, w_3, ..., w_m\}$ be the set of all unique words appeared in the entire text stream.

2.1. Micro-document

In this study, a micro-document refers to a tweet. A micro-document d_i is a short textual content consisting of words that are published online on a micro-blogging service. Given the sets U, T, and W, micro-document d_i is defined as 3-tuple, $d_i = (t, u, \mathcal{W}) \in \mathcal{D}^{T\times U\times\mathcal{W}:\mathcal{W}\subset W}$, where u is a user who publishes a micro-document d_i with a set of words $\mathcal{W}$ at a specific time instance t_i.

2.2. Text stream

A text stream is a set of micro-documents $\mathcal{D} = \{d_1, d_2, d_3, ..., d_n\}$, where d_i and $d_{(i-1)}$ are the i^{th} and $(i-1)^{\text{th}}$ micro-documents published at time $\pi_1(d_i)$ and $\pi_1(d_{i-1})$ respectively, such that $\pi_1(d_i) \geq \pi_1(d_{i-1})$.

2.3. Super-document

The document size is very small in micro-blogs and it is difficult to analyze and detect meaningful patterns among small-sized document (Aiello et al., 2013). The length of a micro-document is limited, hence, statistical inference over the feature set does not yield good results. This limitation is resolved by temporal aggregation of micro-documents and creating a super-document. Let $\mathcal{D} = \{d_1, d_2, d_3, ..., d_n\}$ be the set of all micro-documents available in a text stream and given a temporal coverage τ, a super-document d_i^ρ is a continuous temporal aggregation of micro-documents collected in a certain time interval of length τ.

Instead of merging the micro-documents into one core document, we partition the set of micro-documents into k subsets $\mathcal{D}^\rho = \{\{d_1, d_2, ..., d_p\}, \{d_{p+1}, ..., d_{p+q}\}, ..., \{..., d_n\}\}$. Each subset d_i^ρ is considered as a temporal aggregation of micro-documents in sequence collected at time t_i until $t_i + \tau$. The micro-documents are able to retain their identity in a super-document that is used later to generate graph series (see Section 3.1) which increases the cohesiveness among the topics that co-occur. Thus, a set of super-documents consists of k mutually exclusive subsets, where each subset $d_i^\rho \in \mathcal{D}^\rho$ such that $d_i^\rho \subset \mathcal{D}$ and $\bigcup_i d_i^\rho = \mathcal{D}$ and $\bigcap_i d_i^\rho = \varnothing$. We refer temporal aggregation of the micro-documents from time t_i to $t_i + \tau$ as time interval $i\tau$ later in the paper.

Table 1
A super-document consists of three micro-documents.

Super Document d_i^ρ	
Micro-Document	Words
d_i	F C D G
$d_{(i+1)}$	D C
$d_{(i+2)}$	A D
$d_{(i+3)}$	C E A

2.4. Sliding window

A sliding window is a specific time interval within which data is processed and analyzed independently. Given a temporal coverage Δt a sliding window is a time interval from t_i to $t_i + \Delta t$ when data is collected from the text stream and monitored for possible events, where t_i and $t_i + \Delta t$ (we refer as $k\Delta t$ later in this paper) represent the starting and ending time of a sliding window. A sliding window temporally covers all the super-documents acquired during the given time interval $k\Delta t$ in temporal order. The set of super-documents covered in each sliding window is in temporal order, therefore, each super-document has a temporal characteristic and contributes individually in the feature design of EHG approach.

3. Enhanced Heartbeat Graph (EHG) approach

The proposed EHG approach creates a series of graphs. Each graph produces a heartbeat which signals the possibility for the occurrence of an event at a certain time interval $i\tau$. The work-flow of the proposed EHG approach is illustrated in Fig. 2 which shows data status and processing at each step. The data undergoes five transformations from the data source to the event-related topics: (1) Twitter data stream, (2) super-document stream, (3) graph series, (4) EHG series, and (5) events with ranked topics.

In the first step, we create a set of super-documents $\mathcal{D}^\rho$ by aggregating micro-documents from the text stream $\mathcal{D}$ (as described in Section 2.3). A series of graphs $\mathcal{G}$ is then created (as described in Section 3.1) using the set of super-documents. The graphs in $\mathcal{G}$ represent all the unique words, their frequencies and co-occurrence relations in the time series data. To identify the change in the text stream and the topological relations of words with respect to time, we calculate the KL-divergence score of the words and the relationships between a pair of adjacent graphs, and derive a new graph called *EHG* (as described in Section 3.2). The EHG inherits its temporal as well as structural characteristics of parent graphs. Afterwards, we extract three novel features to compute heartbeat score for each EHG (as described in Section 3.3). Finally, EHGs with a significant heartbeat score are labeled as candidates for events and then used to extract event-related topics (as described in Section 3.4).

In the following sections, we briefly explain transformation of text stream into series of temporal graphs and event detection.[2]

3.1. Graph series

For each super-document d_i^ρ (where $d_i^\rho \in \mathcal{D}^\rho$), a graph G_i is created in such a way that nodes are words and an edge between two nodes represents co-occurrence relationship within a micro-document which leads to co-occurrence relationship between the words of all the micro-documents in a super-document d_i^ρ as shown in Fig. 3. A graph series is a set of temporal graphs $\mathcal{G} = \{G_1, G_2, G_3, ..., G_{|\mathcal{D}^\rho|}\}$, where each graph G_i is created against d_i^ρ such that G_i is a labeled graph, i.e., $G_i = (V, E, \mathscr{W}, \mathscr{S})$, where:

- $V = \{v_1, v_2, v_3, ..., v_n\}$ such that v_i is a unique word that appears in d_i^ρ
- $E \subseteq V \times V$ is a set of edges such that $e_k = (v_m, v_n)$ and $v_m \neq v_n$
- $\mathscr{W} : V \to \mathbb{R}$ and $\mathscr{S} : E \to \mathbb{R}$ are the functions that assign weights to each node and edge in the graph G_i using Eqs. (1) and (2) respectively.

$$\mathscr{W}(v_k) = |d_i^\rho(v_k)| \tag{1}$$

$$\mathscr{S}(e_k) = |d_i^\rho(v_m, v_n)| \tag{2}$$

$|d_i^\rho(v_k)|$ is the term-frequency of v_k and $|d_i^\rho(v_m, v_n)|$ is the number of the co-occurrences of nodes v_m and v_n in the super-document d_i^ρ. The co-occurrences between nodes are enforced by creating a clique among the words of micro-documents. In a clique, each node v_m is connected to every other node v_n only if the words v_m and v_n appear in a micro-document $d_i \in d_i^\rho$ and $v_m \neq v_n$.

For instance, consider a super-document (as shown in Table 1) is to be processed to create a graph. The super-document d_i^ρ contains four micro-documents and each micro-document consists of some words. Each micro-document d_i, where $d_i \in d_i^\rho$, is embedded sequentially into graph G_i. Fig. 3 shows the creation of graph for d_i^ρ and elaborates the structural updates while embedding micro-documents. The nodes and edges are labeled with the weights (the node weights are given in parentheses "(n)"). Once a micro-document d_i is embedded, the graph structure and its weights are updated. Newly embedded nodes, edges, and updated weights are represented in red. The graph creation is completed when all the micro-documents in super-document d_i^ρ are embedded.

From the given example, it can be observed that C has more unique edges than the other nodes due to its occurrence in multiple documents with diverse set of words. It makes C a suitable candidate for representing a theme of users' discussions. A clique among the words of each micro-document captures the co-occurrence relationship. Embedding each micro-document sequentially into graph structure increases the centrality of those words which not only appear frequently but also with a larger set of diverse words in the text stream at a certain time interval. The resultant graph series is further used to generate EHG series by combining each pair of adjacent graph.

3.2. Enhanced Heartbeat Graph (EHG) series

The EHG series is a set of graphs $\mathcal{G}^h$ where each EHG $G_i^h \in \mathcal{G}^h$ corresponds to a pair of adjacent graphs G_i and G_{i-1}. An EHG G_i^h expresses time-based relative entropy of words and their co-occurrence relations. This characteristic suppresses the topic(s) which have been identified in the previous time interval $(i-1)\tau$. Thus, makes it sensitive to detect emerging topics at time interval $i\tau$. To create the EHG series, Algorithm 1 linearly combines and maps each pair of adjacent graphs G_i and G_{i-1} onto a new EHG G_i^h. Finally, a subset $\mathcal{G}^{h(k\Delta t)}$, which is temporally covered by a sliding window $k\Delta t$, is used independently to detect emerging events with respect to the temporal characteristics of the text stream. The step-by-step implementation to generate an EHG is given in Algorithm 1.

The Algorithm 1 takes a graph series $\mathcal{G}$ as input and generates EHG series. Since the graph series is created using the set of super-documents $\mathcal{D}^\rho$ which are mutually exclusive, thus, the set of nodes in the pair of adjacent graphs G_{i-1} and G_i could be different due to the dynamic nature of Twitter data stream. Furthermore,

[2] For ease of understanding, mathematical notations and their descriptions are provided in Table A1.

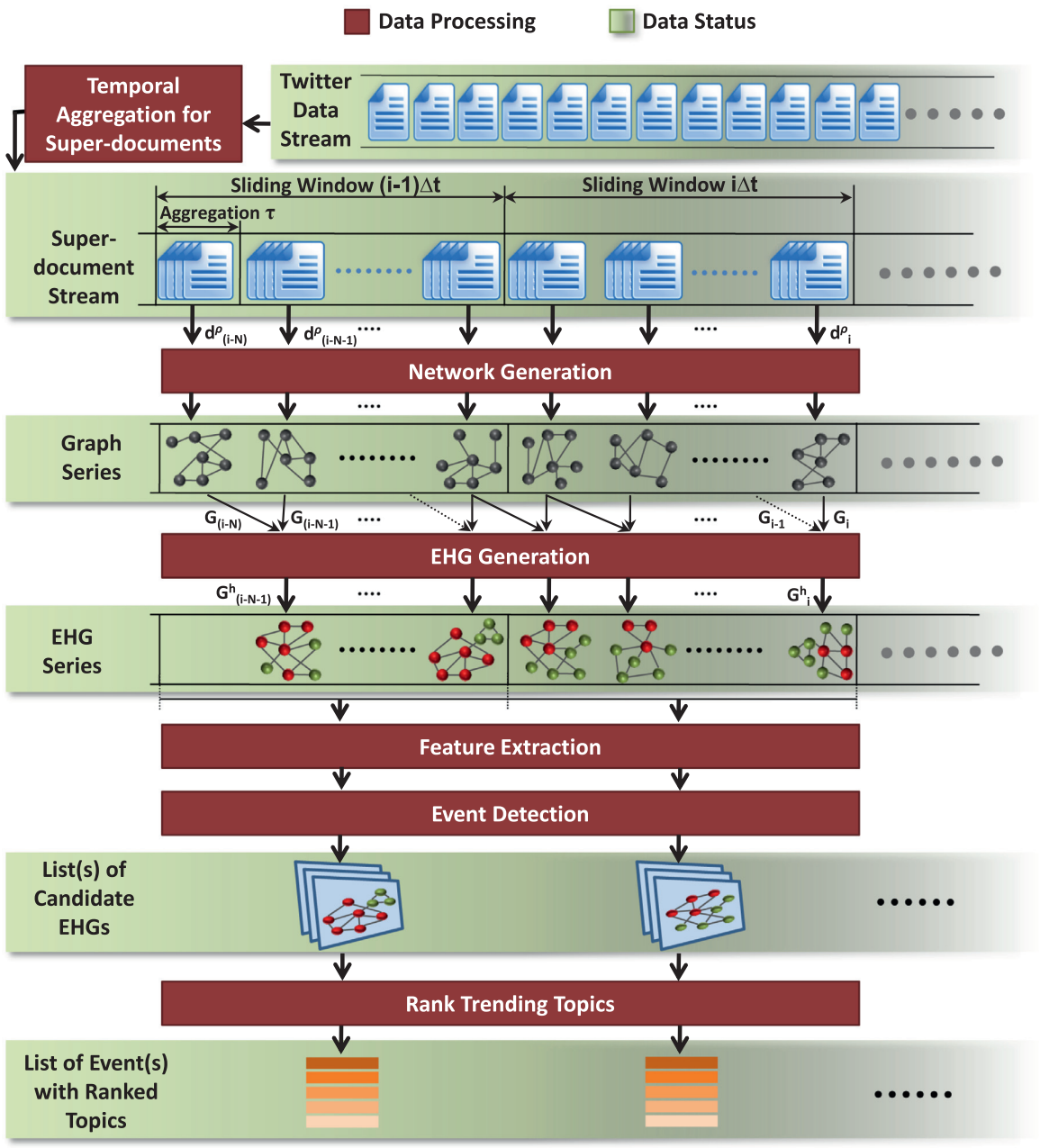


Fig. 2. Work-flow of Twitter stream processing for event detection. The figure shows the step-wise transformation of text stream into corresponding graph representation and data processing modules.

there is no canonical order between the nodes, hence, computing KL-divergence for words and their co-occurrences is not possible and remains unpredictable. To address this computational challenge, we aligned the dimensions of the adjacency matrices of G_{i-1} and G_i by taking a union of the sets of nodes in both graphs and then reordering them canonically. The edges of both graphs (G_{i-1} and G_i) are then mapped onto new matrices. This might result in isolated nodes in both G_{i-1} and G_i as shown in Fig. 4, but it doesn't affect the structure of the resultant EHG.

KL-divergence is a well-known measure in the field of information theory. It is used to find the difference between two data distributions (Kullback, 1997) as shown in Eq. (3). It is commonly called a distance measure, but unlike distance measures, it is asymmetric. It is suitable for measuring the change in a data distribution over time.

$$D_{KL}(Q||P) = \sum_i Q(i) \times \log\left(\frac{Q(i)}{P(i)}\right) \tag{3}$$

For generating an EHG (see Algorithm 1 and Fig. 4), our model uses frequency distributions of words appearing within fixed-sized time intervals, where P and Q are the frequency distributions of words at time interval $(i-1)\tau$ and $i\tau$ respectively. Some words may have zero value in the distribution P, thus KL-divergence would result in an undefined value. Similarly, distribution Q can also have zero values. A zero-value in the time series data indicate that the popularity of the word in terms of frequency $Q(i)$ has reduced as compared to $P(i)$, KL-divergence would be undefined in such a scenario. Undefined values would result in loss of information, therefore, the mathematical expression of KL-divergence for

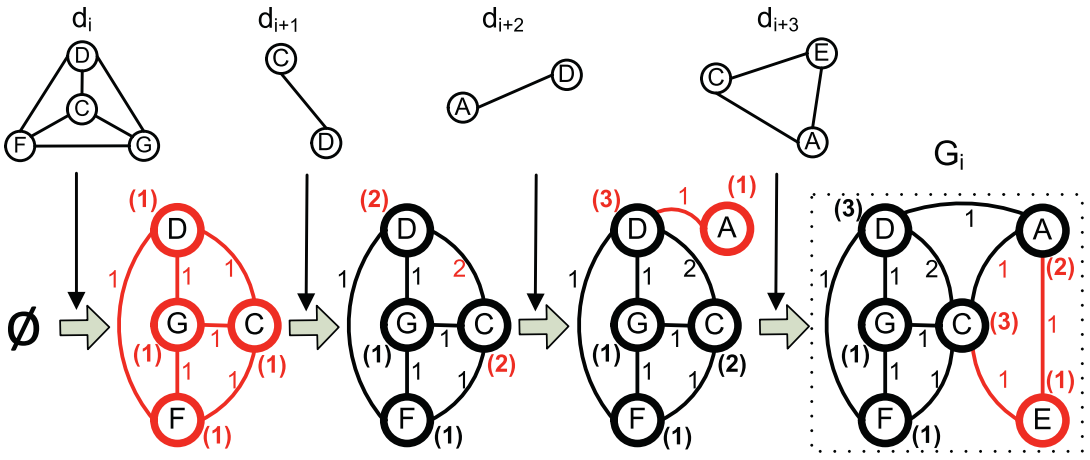
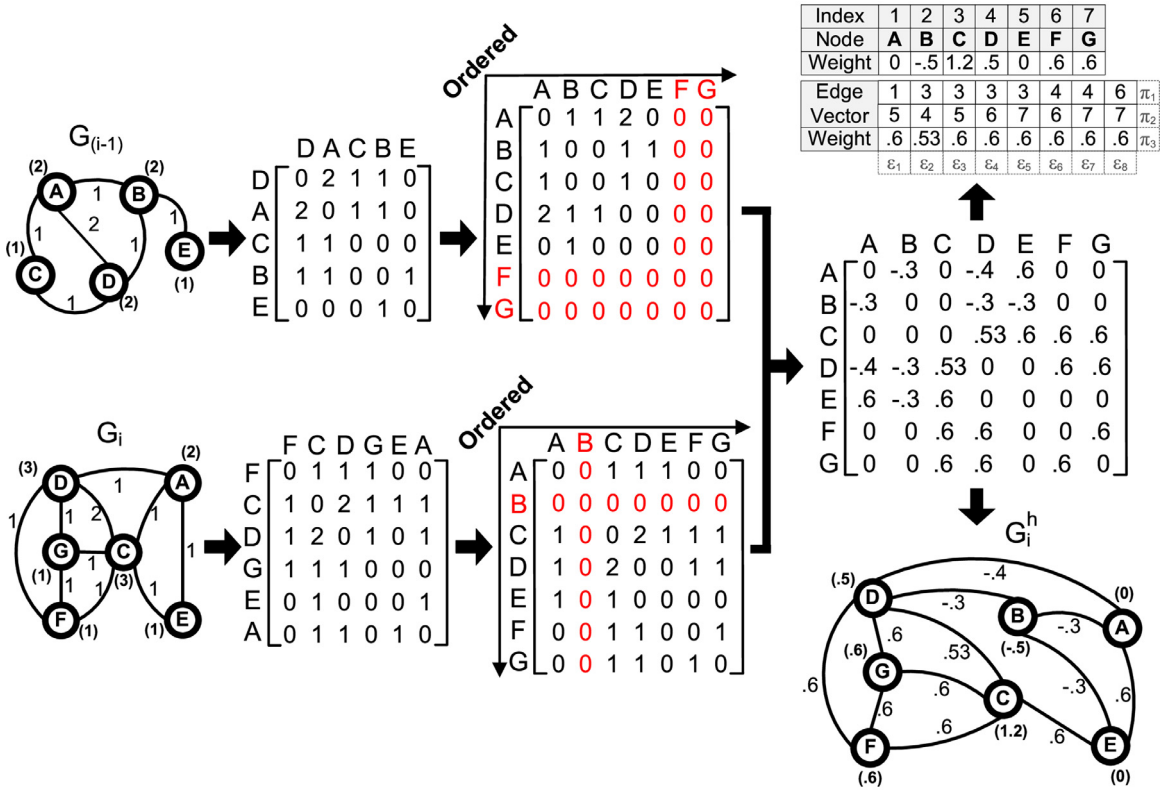


Fig. 3. Graph creation for an example super-document $d_i^{\rho} = \{d_i, d_{(i+1)}, d_{(i+2)}\}$ shown in Table 1. The red nodes and edges are newly embedded nodes and edges. The red labels are new or updated weights. The figure shows the status of graph G_i with the embedding of each micro-document d_i.



Index	1	2	3	4	5	6	7	
Node	A	B	C	D	E	F	G	
Weight	0	-.5	1.2	.5	0	.6	.6	
Edge	1	3	3	3	3	4	4	6	π_1
Vector	5	4	5	6	7	6	7	7	π_2
Weight	.6	.53	.6	.6	.6	.6	.6	.6	π_3
	ε_1	ε_2	ε_3	ε_4	ε_5	ε_6	ε_7	ε_8	

Fig. 4. Shows an example of how an Enhanced Heartbeat Graph (EHG) is generated from two subsequent graphs which are adjacent.

the data points $Q(i)$ and $P(i)$ is modified for the nodes and edges in the graph. Each node and edge in G_i^h is assigned new weights based on modified KL-divergence as shown in Eqs. (4) and (5) respectively.

$$\overline{\mathscr{W}}(v_k) = \left(\mathscr{W}(v_k^{G_i}) + 1\right) \times \log \frac{\left(\mathscr{W}(v_k^{G_i}) + 1\right)}{\left(\mathscr{W}(v_k^{G_{i-1}}) + 1\right)} \tag{4}$$

$$\overline{\mathscr{S}}(e_k) = \left(\mathscr{S}(e_k^{G_i}) + 1\right) \times \log \frac{\left(\mathscr{S}(e_k^{G_i}) + 1\right)}{\left(\mathscr{S}(e_k^{G_{i-1}}) + 1\right)} \tag{5}$$

Finally, the divergence between both distributions is calculated as shown in Eq. (8) and used as a key feature in our model. The extracted features (described in Section 3.3) from an EHG express the significance of the change in the text stream at a certain time interval $i\tau$ in terms of a heartbeat score which is further used to detect event candidate graphs.

Fig. 4 demonstrates the formation of EHG between two graphs. Node weights (given in parentheses "(n)"), and edge weights can be seen in the corresponding graphs as well as in their adjacency matrices. The canonical arrangement of both graphs G_{i-1} and G_i significantly improves the computational complexity (see Section 4 for details).

Each graph G_i^h in EHG series $\mathcal{G}^h$ inherits the divergence of text stream by calculating a modified KL-divergence of nodes and edges between each pair of adjacent graphs G_{i-1} and G_i. Therefore, it implicitly handles the dominance of bursty topics. The weights of the nodes and the edges in an EHG G_i^h have the following possibili-

Algorithm 1: Generate a set of EHGs.

Input : $\mathcal{G} = \{G_1, G_2, G_3, ..., G_{|\mathcal{D}^\rho|}\}$– A set of a graphs, where $G_i \in \mathcal{G}$ corresponds to super-document $d_i^\rho \in \mathcal{D}^\rho$

Output: $\mathcal{G}^h = \{G_1^h, G_2^h, \ldots, G_{|\mathcal{G}|-1}^h\}$

1 **for** $i \leftarrow 2$ **to** $|\mathcal{G}|$ **do**
2 $V^\psi \leftarrow V^{G_i} \cup V^{G_{i-1}}$
3 $V^{G_i} \leftarrow$ recreate vertices using V^ψ
4 $V^{G_{i-1}} \leftarrow$ recreate vertices using V^ψ
5 $A \leftarrow$ create adjacency matrix for $G_{(i-1)}$ using V^ψ
6 $B \leftarrow$ create adjacency matrix for $G_{(i)}$ using $V^\psi G_{(i)}^h \leftarrow$ `GenerateEHG` $(A, B, V^{G_i}, V^{G_{i-1}})$; ►`Using Algorithm 2`
7 **end**

ties to signify the event-related topics with respect to the temporal characteristics:

- $\overline{\mathscr{W}}(v_k) > 0$ means the word is gaining in popularity
- The existence of an edge e_k in an EHG shows that the connected nodes v_m and v_n are not only gaining in co-existential popularity but also expresses that the micro-documents in the text stream is themed around v_m and v_n. Therefore, the edge e_k makes v_m and v_n significant to be detected as event-related trending topic

3.3. Feature design

Existing techniques such as Becker et al. (2011), Chen, Amiri, Li, and Chua (2013), Chierichetti, Kleinberg, Kumar, Mahdian, and Pandey (2014) and Gao, Cao, He, and Li (2013) use bursty features such as term frequency (tf) and inverse document frequency (idf) to detect events in Twitter data stream. A reason behind the use of bursty features is that when a popular event emerges in the real world, people report and publish event-related information. A large number of people reporting the same event produces burstiness in a data stream. However, such bursty features often dominate other less-frequent but relevant information and induce bias in event-related information extraction; therefore, bursty features are not always effective, especially when the data is extremely diverse such as in Twitter stream. Moreover, bursty features do not provide the topological relationship between the different words. Similarly, approaches (Adedoyin-Olowe et al., 2016; Aiello et al., 2013) that use frequent pattern mining, focus on the co-occurrence frequency of a set of words but do not consider the words that are recurring with diverse set of words. Consider the example given in Table 1 where the term *C* is recurring with diverse set of words hence, would be important to identify the topic in the text stream.

Instead of bursty features based on tf and idf, we focus on change in temporal burstiness of the words and their topological relations in the temporal graphs. The node and edge weights in the EHG G_i^h are based on a modified KL-divergence score between graphs G_{i-1} and G_i which are created at time interval $(i-1)\tau$ and $i\tau$ respectively. Due to this unique characteristic, the proposed approach detects events at an early stage and once detected it implicitly suppresses the event-related bursty topics in subsequent time intervals. The limitations and characteristics of burstiness, compared to our approach, are empirically discussed in Section 6.2. Fusion of temporal and topological features (discussed later in this section) improves the performance of our approach.

Based on the aforementioned characteristics of the EHG, we extract three features *divergence factor, trend probability*, and *aggregated centrality*. The EHG is generated using a pair of graphs that are adjacent at time interval $i\tau$ and $(i-1)\tau$, thus, these features identify the change in the burstiness of topics, possibility of occurrence of an event, and the theme in the text stream respectively.

For the simplification of notations, let $\psi = G_i^h$ where G_i^h is i^{th} heartbeat graph in $\mathcal{G}^h$. A node in the EHG ψ can have negative or positive weights. We normalized the node and edge weights in the EHG ψ between [-1,1] using Eqs. (6) and (7) respectively where $\overline{\mathscr{W}}(v_k^\psi)$ and $\vartheta(v_k^\psi)$ are the weight and normalized weight of k^{th} node, similarly $\overline{\mathscr{F}}(e_k^\psi)$ and $\delta(e_k^\psi)$ are the weight and normalized weight of k^{th} edge in the EHG ψ respectively.

$$\vartheta(v_k^\psi) = \frac{\overline{\mathscr{W}}(v_k^\psi)}{\max\limits_{1 \le j \le |V^\psi|} \overline{\mathscr{W}}(v_j^\psi)} \tag{6}$$

$$\delta(e_k^\psi) = \frac{\overline{\mathscr{F}}(e_k^\psi)}{\max\limits_{1 \le j \le |E^\psi|} \overline{\mathscr{F}}(e_j^\psi)} \tag{7}$$

3.3.1. Divergence factor

Divergence factor $DF(\psi)$ measures the intensity of temporal change and the popularity in the trending topics. A change appears when new topics are observed in the data. The popularity indicates an increase in the divergence score of previously observed topics. $DF(\psi)$ expresses the emergence of new topics as well as the gain in popularity of previously observed topics. The divergence factor is calculated by accumulating the weights of all the nodes in the EHG ψ using Eq. (8).

$$DF(\psi) = \sum_{k=1}^{|V^\psi|} \vartheta(v_k^\psi) \tag{8}$$

Where $\vartheta(v_k^\psi)$ is the k^{th} node weight that represents the divergence score of a word between G_i and G_{i-1} (see Algorithm 2, Step 3).

Algorithm 2: Generate Enhanced Heartbeat Graph.

Input : A, B– Adjacency matrices correspond to $G_{(i-1)}$ and $G_{(i)}$ respectively
V^A, V^B– Sets of vertices correspond to $G_{(i-1)}, G_{(i)}$ respectively

Output: *EHG* containing an index edge vector ε and a list of weighted vertices V^H

1 $V^H \leftarrow List()$
2 **for** $k \leftarrow 1$ **to** $|V^B|$ **do**
3 $V_{(k)}^H \leftarrow \left(V_{(k)}^B + 1\right) \times \log \frac{V_{(k)}^B + 1}{V_{(k)}^A + 1}$; ►`Using Equation 4`
4 **end**
5 $\varepsilon \leftarrow List()$ **for** $r \leftarrow 2$ **to** $|V^H|$ **do**
6 **for** $c \leftarrow 1$ **to** $r - 1$ **do**
7 edgeWeight $\leftarrow \left(B_{(r,c)} + 1\right) \times \log \frac{B_{(r,c)}+1}{A_{(r,c)}+1}$; ►`Using Equation 5`
8 **if** *edgeWeight* > 0 ***AND*** $edge(r, c) \notin \varepsilon$ **then**
9 $\varepsilon \leftarrow \varepsilon \cup edge(r, c)$
10 **end**
11 **end**
12 **end**

3.3.2. Trend probability

Trend probability $TP(\psi)$ expresses the chances of an EHG ψ as an event candidate at time interval $i\tau$. A node weight in the EHG ψ can be a positive or a negative value. The probability distribution against the positive $\vartheta(v_k^{\psi+})$ and negative $\vartheta(v_k^{\psi-})$ weights of the nodes are calculated within the EHG ψ using Eqs. (9) and (10).

$$P(\vartheta(v_k^{\psi+})) = \frac{\vartheta(v_k^{\psi+})}{\sum_{l=1}^{|V^{\psi}|} |\vartheta(v_l^{\psi})|} \tag{9}$$

$$P(\vartheta(v_k^{\psi-})) = \frac{|\vartheta(v_k^{\psi-})|}{\sum_{l=1}^{|V^{\psi}|} |\vartheta(v_l^{\psi})|} \tag{10}$$

Where $v_k^{\psi+}$ and $v_k^{\psi-}$ are k^{th} nodes with positive and negative weights respectively. The probability distribution over the positive and negative weights of the nodes are then linearly combined using Eq. (11), which characterize the convergence of EHG ψ as an emerging event.

$$TP(\psi) = \beta_1 \sum_{k=1}^{|V^{\psi+}|} P(\vartheta(v_k^{\psi+})) + \beta_2 \sum_{l=1}^{|V^{\psi-}|} P(\vartheta(v_l^{\psi-})) \tag{11}$$

Where β_1 and β_2 are set to 1 and -1 respectively. $TP(\psi) > 0$ indicates the possibility of an emerging event because the probability distribution over the positive words is greater, hence showing that a major sub-graph in the EHG ψ has changed. This can also be observed in Fig. 5.

3.3.3. Topic centrality

Topic centrality $TC(v_k^{\psi})$ expresses the central tendency of words in the EHG ψ. It characterize the theme of discussion in the Twitter stream at a certain time interval $i\tau$. A node v_k^{ψ} connected with many positive edges in the EHG ψ shows that the word v_k^{ψ} is highly co-occurrent with diverse set of words and making it a topical keyword. $TC(v_k^{\psi})$ is calculated using Eq. (12).

$$TC(v_k^{\psi}) = \frac{\sum_{i=1}^{|\varepsilon^{\psi}|} [\pi_1(\varepsilon_i^{\psi}) = k \vee \pi_2(\varepsilon_i^{\psi}) = k)]}{|V^{\psi}|} \tag{12}$$

Where ε^{ψ}, v_k^{ψ}, and $|V^{\psi}|$ represent an index edge vector, a node, and the total number of nodes in the EHG ψ respectively. $\pi_1(\varepsilon_i^{\psi})$ and $\pi_2(\varepsilon_i^{\psi})$ represent the indexes of the nodes connected to the edge ε_i^{ψ}. The aggregated centrality score $AC(T^{\psi})$ is calculated by accumulating the topic centrality scores of all the nodes in the EHG ψ using Eqs. (13) and (14). Where T^{ψ} is a set of indexes of those nodes that are connected to at least one positive edge.

$$T^{\psi} = \bigcup_{i=1}^{|\varepsilon^{\psi}|} \left(\pi_1(\varepsilon_i^{\psi}) \cup \pi_2(\varepsilon_i^{\psi}) \right) \tag{13}$$

Then the aggregated centrality of EHG ψ is calculated as:

$$AC(T^{\psi}) = \sum_{k=1}^{|T^{\psi}|} TC(v_{T_k^{\psi}}^{\psi}) \tag{14}$$

An index edge vector ε^{ψ} is used to calculate aggregated centrality. The index edge vector contains only those edges which have positive weights as shown in Fig. 4. Removal of negative edges results in improving the quality of graph structure for newly emerging topics. It also reduces the number of passes for calculating aggregated centrality. A higher aggregated centrality score depicts that the emerging topics are cohesive and concurrently appear in the text stream.

3.4. Event detection method

The event detection method uses the feature set extracted from EHGs and works with the following assumptions:

- A text stream has diverse content that changes dynamically, however an event can only occur when there is a significant increase in the popularity of existing topics or new topic(s) appear in the text stream at time interval $i\tau$ compared to $(i-1)\tau$
- The occurrence of an event is not only dependant on the relative entropy of the topics, it also relies on the change in the probability distribution of words as well as the cohesion in the topological structure of graph.

The detection method (as given in Algorithm 3) fuses the three key features, these being divergence factor, trend probability, and aggregated centrality as shown in Eqs. (8), (11), and (14) respectively and calculates the heartbeat score $HB(\psi)$ using Eq. (15). Divergence factor $DF(\psi)$ shows how significant change occurred, trend probability $TP(\psi) > 0$ shows that the words are either gaining in popularity or are newly emerging, whereas aggregated centrality $AC(T^{\psi})$ represents the coherence and central tendency

Algorithm 3: Event Detection Algorithm.

Input : $\mathcal{G}^{h(k\Delta t)}$ – Set of EHGs temporally covered by the sliding window $k\Delta t$.

Output: $\mathcal{L}$ – List of ranked topics.

$AC \leftarrow List()$; ▶set of aggregated centrality
$weight \leftarrow 0$; ▶absolute weight of all nodes in a G_m^h
$DF \leftarrow List()$; ▶set of divergence factors
$TP \leftarrow List()$; ▶set of trend probabilities
$HB \leftarrow List()$; ▶set of heartbeat scores of all the EHGs in $\mathcal{G}^{h(k\Delta t)}$
$\mathcal{C} \leftarrow List()$; ▶set of class-labels
$\mathcal{L} \leftarrow List()$; ▶set of keywords as ranked topic(s)
$i \leftarrow 0$
for *each* $G_m^h \in \mathcal{G}^{h(k\Delta t)}$ **do**
  $i \leftarrow i + 1$
  $DF_{(i)} \leftarrow 0$
  $TP_{(i)} \leftarrow 0$
  $AC_{(i)} \leftarrow 0$
  $HB_{(i)} \leftarrow 0$
  for *each vertex* $v_n \in G_m^h$ **do**
    $DF_{(i)} \leftarrow DF_{(i)} + \vartheta(v_n)$; ▶Using Equation 8
    $weight \leftarrow weight + |\vartheta(v_n)|$
  end
  for *each vertex* $v_n \in G_m^h$ **do**
    $TP_{(i)} \leftarrow TP_{(i)} + \frac{\vartheta(v_n)}{weight}$; ▶Using Equation 11
  end
  for *each index* $l \in T^{\psi}$ **do**
    $AC_{(i)} \leftarrow AC_{(i)} + TC(v^{\psi}_{T_l^{\psi}})$; ▶Using Equations 12 and 14
  end
  $HB_{(i)} \leftarrow DF_{(i)} \times TP_{(i)} \times AC_{(i)}$; ▶Using Equation 15
end
$\mathcal{C} \leftarrow$ `AssignClassLabels`(TP, HB) ; ▶Algorithm 4
$\mathcal{L} \leftarrow$ `TopicRanking`$(\mathcal{G}^{h(k\Delta t)}, \mathcal{C})$; ▶Algorithm 5

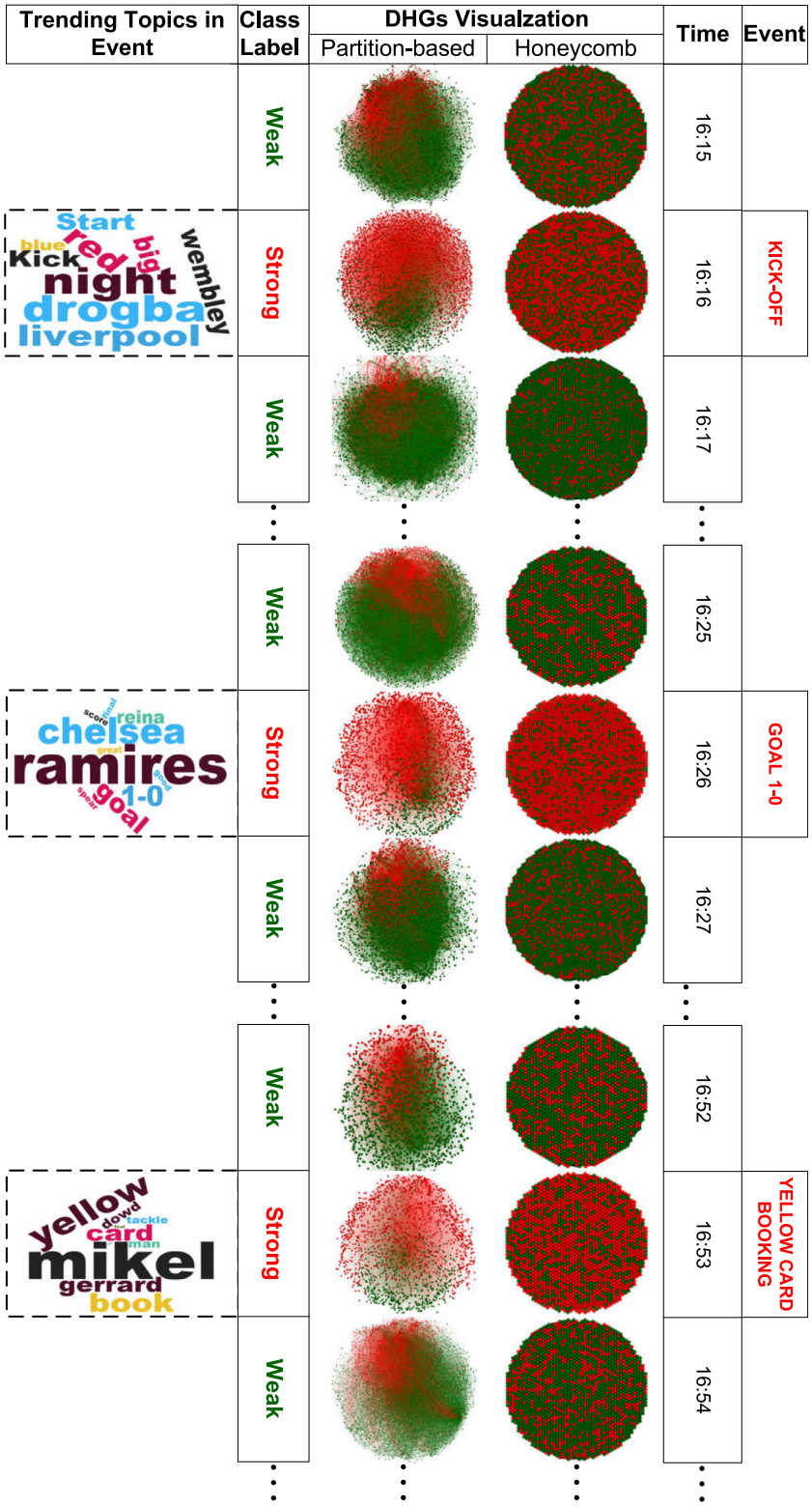


Fig. 5. The graph visualization (Beehive and Partition-based) of three events (i.e., starting with the match, first goal, and a yellow card booking) at different time intervals of the FA Cup dataset. Red nodes are the words either new or appeared previously but gaining popularity at current time interval. The EHG shows hyper sensitivity to event-related topics when an event emerges. The ranked lists of top-10 keywords against detected events are also shown here.

among different words in the EHG ψ.

$$\equiv \sum_{k=1}^{|V^{\psi}|} \left(\frac{(\vartheta(v_k^{\psi}))^2 \sum_{i=1}^{|\varepsilon^{\psi}|} [(\pi_1(\varepsilon_i^{\psi})=k) \vee (\pi_2(\varepsilon_i^{\psi})=k)]}{|V^{\psi}| \sum_{l=1}^{|V^{\psi}|} |\vartheta(v_l^{\psi})|} \right)$$

$$HB(\psi) = DF(\psi) \times TP(\psi) \times AC(T^{\psi}) \quad (15)$$

To find event candidates, a rule-based classification function (as shown in Eq. Eq. (16)) is used to label EHGs. The classification function (as given in Algorithm 4) works in a two-steps rule:

Algorithm 4: Assign Class Labels.

Input : $TP-$ set of trend probabilities.
$HB-$ set of heartbeats Correspond to each $G_i^h \in \mathcal{G}^{h(k\Delta t)}$.

Output: $\mathcal{C}-$ List of class labels.

$\theta_{(k\Delta t)} \leftarrow$ compute using Equation 19 over $\mathcal{G}^{h(k\Delta t)}$ **for** $i \leftarrow 1$ **to** $|HB|$ **do**
  if $TP_{(i)} \leq 0$ **then**
    $\mathcal{C}_{(i)} \leftarrow$ "Weak"
  else if $HB_{(i)} \geq \theta_{(k\Delta t)}$ **then**
    $\mathcal{C}_{(i)} \leftarrow$ "Strong"
  else
    $\mathcal{C}_{(i)} \leftarrow$ "Weak"
  end
end
Return $\mathcal{C}$

- $TP(\psi) \geq 0$ shows that the topics are gaining importance due to the increase in their popularity compared to the earlier adjacent time interval. As well as, if the heartbeat score (as shown in Eq. (15)) of an EHG ψ is greater than $\theta_{(k\Delta t)}$, assign *Strong* label which represents the existence of an event.
- Otherwise, assign *Weak* which indicates that the EHG ψ is insignificant.

$$Est(\psi) = \begin{cases} Strong, & \textbf{if} (TP(\psi) > 0) \wedge (HB(\psi) \geq \theta_{(k\Delta t)}) \\ Weak, & \textbf{otherwise} \end{cases} \tag{16}$$

Here, θ is an adaptive measure that finds the threshold value in each sliding window $k\Delta t$ using Eq. (19).

$$\mathcal{N} = \frac{\Delta t}{\tau} \tag{17}$$

$$\varpi = \frac{\sum_{i=k}^{\mathcal{N}+k} (HB(\psi))}{\mathcal{N}} \tag{18}$$

$$\theta_{(k\Delta t)} = \varpi + \omega \sqrt{\frac{\sum_{i=k}^{\mathcal{N}+k} (HB(\psi) - \varpi)^2}{\mathcal{N}}} \tag{19}$$

Where τ and Δt are the temporal coverage of each super-document d_i^ρ and sliding window respectively. $\mathcal{N}$ represents the number of super-documents in each sliding window such that $\Delta t (\text{mod } \tau) = 0$. ϖ is the average heartbeat score calculated using Eq. (18), ω is the adjustment parameter, k is the index of the first EHG in the sliding window under consideration, and $HB(\psi)$ is the heartbeat score of the EHG ψ.

Afterward, in each sliding window $k\Delta t$, Algorithm 5 generate a ranked list of topics from the candidate EHGs (i.e., labeled as *Strong*) by calculating the ranking score using Eq. (20).

$$Rank(v_k^\psi) = TC(v_k^\psi) \times \vartheta(v_k^\psi) \tag{20}$$

Fig. 5 shows the visualization of EHGs and their class labels with top ten trending topics. The visualization is created for three events (Kick-off, Goal, and Card-booking) from the FA Cup dataset. We have selected three consecutive time intervals (pre-event, event, and post-event) to understand the behavior of EHGs when an event emerges. For example, when the football match begins at time 16:16 (GMT), a large number of red nodes in the EHG shows that the event-related topics are gaining popularity compared to the previous time interval at 16:15. Once detected, EHG suppresses those topics in post-event time interval at 16:17. Similar behavior can be observed for *Goal* and *Card-booking* events.

Algorithm 5: Topic Ranking.

Input : $\mathcal{G}^{h(k\Delta t)}-$ set of EHGs in a sliding window.
$\mathcal{C}-$ set of class labels.

Output: $\mathcal{L}-$ List of ranked topics.

for $i \leftarrow 1$ **to** $|C|$ **do**
  if $\mathcal{C}_{(i)} =$"*Strong*" **then**
    $\mathcal{L} \leftarrow \mathcal{L} \cup$ Sort all keywords in G_i^h using Equation 20
  end
end
Keep top-ranked keywords from duplicates and remove all other in $\mathcal{L}$
Return $\mathcal{L}$

4. Complexity analysis

In order to generate an EHG ψ, we linearly combined two subsequent graphs G_i and G_{i-1} using their adjacency matrices $A^{G_i}_{[n_0 \times n_0]}$ and $A^{G_{i-1}}_{[n_1 \times n_1]}$, where A^{G_i}, $A^{G_{i-1}}$ represent matrices, and $[n_0 \times n_0]$, $[n_1 \times n_1]$ represent their dimensions respectively. Naturally, due to the diversity and dynamic nature of the text stream, the canonical order in the graph nodes does not exist. Therefore, a bijective function for A^{G_i} and $A^{G_{i-1}}$ to generate an EHG ψ does not exist and $(n \times n)$-dimensions for A^ψ remains unpredictable.

To avoid the computational challenge involved in above mentioned problem, we align both matrices canonically in equal dimensions without affecting the edges. We achieve this by taking union of the sets of nodes as shown in Eq. (21) with $O(Max(|V^{G_i}|, |V^{G_{i-1}}|))$.

$$V^\psi = V^{G_i} \cup V^{G_{i-1}} \tag{21}$$

where V^{G_i} and $V^{G_{i-1}}$ are the set of nodes in graphs G_i and G_{i-1} respectively. The adjacency matrices are then regenerated canonically with an extended set of nodes with $O((|V^\psi|)^2)$. The transformation function $\mathcal{T}$ maps G_i and G_{i-1} onto EHG ψ as shown in Eq. (22).

$$\mathcal{T} : G_{i-1}, G_i \rightarrow G_i^h \tag{22}$$

The computational complexity of generating an EHG is $O(Max(|V^{G_i}|, |V^{G_{i-1}}|) + 2(|V^\psi|)^2) \equiv O((|V^\psi|)^2)$ as $Max(|V^{G_i}|, |V^{G_{i-1}}|) \leq |V^\psi|$. Generating a series of EHGs in a sliding window $k\Delta t$, asymptotically we get $O(K(|V^\psi|)^2)$ where $K = \mathcal{N}$ which depends upon Δt and τ as shown in Eq. (17) and is a considerably small value.

In addition to the adjacency matrix structure, the sparseness increases even further due to the alignment of the matrices. A^ψ represents an undirected graph that is symmetric with all zeros in the diagonal therefore, only $\frac{|V|(|V|-1)}{2}$ possible edges are considered in edge distribution. Furthermore, adjacency matrix A^ψ is transformed into an index vector $\varepsilon^\psi = \{e_1, e_2, e_3, ..., e_n\}$ with respect to the canonical order of EHG ψ having n-dimensions. Each dimension e_k represents a positive edge containing a pair of indexes (i,j) that can be mapped back to adjacency matrix $A^\psi_{[n \times n]}$. The transformation into index edge vector does not take overhead into account

Table 2
Dataset statistics and temporal coverage.

	From (GMT)	To (GMT)	Total Topics	Tweet Count
FA Cup	05 May 2012 14:00	05 May 2012 20:00	13	124,524
Super Tuesday	06 March 2012 17:00	07 March 2012 17:00	22	540,241
US Election	06 November 2012 17:00	08 November 2012 05:00	64	2,335,105

Table 3
A sample of automatically detected event-related topics from FA Cup and Super Tuesday are compared with the ground truth keywords. Table also shows the relevant tweets against detected events.

Case Study	Time Inter-val	Extracted keywords	Ground Truth keywords	Relevant Tweets from the data corpus
FA Cup	17:25	drogba, chelsea, 2-0, goal, score, wembley, didier	goal, 2-0, didier, drogba, chelsea, score	2-0!!! Great goal from Drogba #FACupFinal. Didier Drogba has now scored 8 goals in 8 games at Wembley.
	18:09	chelsea, liverpool, win, congratulation, cup, champions, deserve, full, time, 2-1	full, time, final, whistle, gone, chelsea, champions, congratulations, 2-1, win	@chelseafc champions of #Facup 2012 congratulations it was a great game
Super Tuesday	01:00 - 01:59	#supertuesday, win, romney, project, state, cnn, call, news, primary, nbc, mitt, victory, @mittromney	mitt, romney, @mittromney, massachusetts, win, project, nbc, cnn, primary, home	Looking like #mitt #romney overwhelming victory in #massachusetts for obvious reasons. People there know and trust him
	05:00 - 05:59	#supertuesday, romney, ohio, win, mitt, primary, news, @mittromney, cnn	mitt, romney, @mittromney, ohio, ap, declare, primary	BREAKING: Romney wins primary in Ohio, a crucial Super Tuesday state

in the computation since ε^{ψ} was created during the EHG algorithm (see Algorithm 2).

The transformation of the EHGs into the index vector space results in reducing the computational overhead of calculating aggregated centrality from $O(|V^{\psi}|+|E^{\psi}|)$ to $O(|V^{\psi}|+N)$, where $N=|\varepsilon^{\psi}|$. Here, the value of $N \ll |E^{\psi}|$ because ε^{ψ} contains only those edges that have positive weights. $O(|V^{\psi}|+|E^{\psi}|)=O(|V^{\psi}|+N)$ if and only temporal frequency of all words continuously increases. However, such a case is less likely to occur due to the evolutionary pattern of real-world events (Iyengar, Finin, & Joshi, 2011).

Against each EHG, our detection method calculates the divergence factor by accumulating the weights of each node with $O(|V^{\psi}|)$, the aggregation of the probability distribution of words in each EHG ψ with $O(|V^{\psi}|)$, and aggregated centrality with $O(|V^{\psi}|+N)$. Thus, the total time complexity to calculate the feature set is $O(3|V^{\psi}|+N) \equiv O(|V^{\psi}|+N)$. In each sliding window $k\Delta t$, the threshold value $\theta_{k\Delta t}$ is calculated with $O(K)$ where $K=\mathcal{N}$ which depends upon Δt and τ as shown in Eq. (17). Conclusively, the detection method overall takes $O(K(|V^{\psi}|+N)+K) \equiv O(K(|V^{\psi}|+N))$.

In contrast to the proposed approach, most of the existing approaches load the entire dataset into memory. This leads to scalability problems when the data size is huge. Due to the evolutionary pattern in temporal characteristics of the real-world events, our approach on the other hand processes the data in sliding windows, therefore producing results efficiently with respect to computational memory. Considering the given definition of graph in Section 3.1 and EHG in Section 3.2, $O(K(|V^{G_i}|+|E^{G_i}|))$ and $O(K(|V^{\psi}|+|\varepsilon^{\psi}|))$ are the space complexities of generating a graph series and EHG series respectively, where K represents the total number of graphs in a sliding window $k\Delta t$.

5. Dataset collection

We conducted experiments on three well-known benchmark datasets[3] (FA Cup, Super Tuesday, and the US Election) that are crawled for the targeted data streams of events. The three datasets were first collected and made available by Aiello et al. (2013). Many recent studies (Adedoyin-Olowe et al., 2016; Choi & Park, 2019; Elbagoury, Ibrahim, Farahat, Kamel, & Karray, 2015; Ibrahim, Elbagoury, Kamel, & Karray, 2018; Nguyen & Jung, 2017; Nur'aini, Najahaty, Hidayati, Murfi, & Nurrohmah, 2015; Papadopoulos, Corney, & Aiello, 2014; Prabandari & Murfi, 2017; Saeed et al., 2018) used these benchmark datasets for evaluating their event detection approaches, which makes the comparison with the proposed approach easy. The statistics of the datasets are given in Table 2 with the data coverage in GMT.

The FA Cup (Football Association Challenge Cup) is one of the oldest and famous knock-out competitions in English football. The FA Cup dataset contains data on the final match of 2012 between Chelsea and Liverpool. The ground truth consists of 13 topics, including match start, match half, match end, goals, and key bookings.

The Super Tuesday Primaries dataset contains data crawled on Tuesday 6 March 2012 including the key moment when it was likely that the party nominee is elected. The ground truth consists of 22 topics having events such as televised speeches and winning projections of candidates.

The US Election dataset was collected against United States presidential election of 2012 which was held on November 6. The ground truth consists of 64 topics. The majority of these topics were added to the ground truth by monitoring the announcements of US television regarding the outcomes of the presidential election.

6. Result and discussion

In this section, we discuss the data pre-processing, observations, results and evaluation of the DGH approach on the benchmark datasets.

6.1. Pre-processing

Micro-documents on social media often contain a large amount of noise, including a significant amount of misspelled words, emoticons, self-abbreviated words like "ty" and "OMG", and duplicate words. To reduce the noise, the data is pre-processed in two steps: (1) redundant and meaningless tweets are removed; (2) the classic IR approach is used to clean the data

Dropping Tweets

To improve data quality, certain tweets are removed from the data, based on the following criteria:

[3] http://www.socialsensor.eu/results/datasets/72-twitter-tdt-dataset (accessed on Septermber 5, 2018).

- retweets as they may add bias to the burstiness of topics
- tweets containing URLs
- tweets that do not contain any word other than hashtags and mentions
- tweets less than three words
- duplicate tweets
- tweets not written in English

Cleaning

In the cleaning process punctuation, special characters other than (#,@,-), stop words and common words are removed. Special character "#","@", and "-" are kept because "#" and "@" are meaningful prefixes, these being *hashtag* and *mention*, respectively. Though we treat both as part of the BoW, later we may use them as separate features in future work. Furthermore, "-" is also used to add a prefix to a word like "warm-up", "well-known", and "half-time". It was also beneficial to find keywords like "1-0" and "2-1" in FA Cup dataset. Words with less than three letters and duplicate words within a tweet were also removed. Furthermore, each word is reduced to its root form using stemming.

Aggregation

The clean tweets are aggregated into a set of super-documents, as described in Section 2 in detail. Individual documents lose their identity when aggregated, as tweets are merged into a single super-document such as in existing studies (Adedoyin-Olowe et al., 2016; Aiello et al., 2013). Rather than combining and merging all the micro-documents into one large document, we applied time-based aggregation by dividing a text stream into segments. Each segment is a super-document which contain a set of micro-documents, as a result each micro-document retains its identity. While creating a graph, each micro-document in a super-document is embedded in a way that each word is linked to all other words within a micro-document forming a clique as shown in Fig. 3.

6.2. Observations

Generally, real-world events progress in three phases (i.e., build-up, stable/peak, and decay) (Iyengar et al., 2011). The temporal coverage of each phase can be different depending on the nature and popularity of the event. The tf-idf which is widely used as a key feature in many studies (Becker et al., 2011; Chen et al., 2013; Chierichetti et al., 2014; Gao et al., 2013) does not capture the dynamics involved in the progress of real-world event, therefore, the key occurrences of the event often dominate other related information which may not have high frequency but can be important as well. Moreover, the approaches based on such bursty features, are biased towards the highly frequent patterns. For example, approaches like Li et al. (2012c), Nguyen and Jung (2015), Shamma et al. (2011) and Yang and Leskovec (2011) capture such highly frequent patterns to aggregate the data around the key occurrences and lose the small but relevant details due to the features which characterize the data based on burstiness. To address the limitations of the existing approaches, instead of tf-idf, the EHG approach relies on the KL-divergence score of words and their relationships with respect to time. We designed features (see Section 3.3) on top of this core characteristics of the EHG, which helps to detect the events at an early stage.

To further elaborate the above mentioned characteristic, we generate signals based on term frequencies and their corresponding KL-divergence score using EHG series. Fig. 6 shows the comparison of top six keywords associated with different event-related topics over a time interval of five minutes from 06:50AM to 11:30AM in the Super Tuesday dataset. It can be observed that the signals generated using EHG, shown in Fig. 6(top), are sharper when compared to signals in Fig. 6(bottom). At time interval 07:20-08:40 the keywords "win" and "romney" are trending which dominate the data stream during the mentioned time interval. On the other hand, using EHG, the keywords "win" and "romney" are detected early at time 07:20 and are suppressed in subsequent time intervals. As a result, at time 07:50, the keyword "paul" can easily be identified to detect the trending topic. A similar case can be seen at time 09:10 where "santorum" and "win" are visible when "gingrich" is suppressed after its early detection at time 08:50. Likewise, at time 10:15, 11:00, and 11:15, similar characteristics can be observed when the bursty keywords clearly dominate, as shown in Fig. 6(top) but are suppressed by the EHG after their early burst, as shown in Fig. 6(bottom), making room for other related but not so frequent topics to appear at the top. This behavior of the EHG is generic and observed on all the datasets we used in this study which makes proposed approach interesting and sensitive to continuously changing data streams, such as Twitter.

We observe an interesting correlation between heartbeat score, graph size, and user participation as shown in Fig. 7. When an event occurs, user participation is at its peak but the growth in graph size in terms of BoW is at an early stage. It is due to the fact that when an event occurs, a larger number of users publish contents with a focus on describing the ongoing event with related vocabulary. This results in event-related topics becoming prominent and it also increases cohesiveness in the topological structure of EHG. The heartbeat of EHG shows a significant score in such scenarios. Thus, the EHG approach is adept at detecting emerging events at an early stage and is able to detect relevant topics before the diversity in the text stream increases. The behavior of heartbeat, user participation, and graph size can be observed in Fig. 7 which is created using the FA Cup dataset on one minute time interval, and also marked with the detected events occurred at time 17:25, 17:37, 17:55, and 18:08.

Table 3 shows a sample events with extracted event-related and ground truth keywords with related tweets in the corpus at 17:25 and 18:09 for the FA Cup. Similary 01:00-01:59 and 05:00-05:59 for the Super Tuesday.

6.3. Parameter selection

There are three parameters Δt, ω, and τ in the proposed approach. To ensure our approach and results are comparable to the ground truth (Aiello et al., 2013), we set the temporal coverage Δt of the sliding windows to one minute, one hour, and ten minutes for the FA Cup, Super Tuesday, and US election datasets, respectively. For popular events that have a narrow scope and limited life span such as the FA Cup, users publish and report event-related information with consistent content. Inversely, events that are comparatively broader in scope and have a longer life span have a high entropy in the frequency distribution of words (Aiello et al., 2013). Therefore, to calculate the threshold value over a sliding window $\theta_{k\Delta t}$, adjustment parameter ω, which deals with the dispersion in data, is set to 1, 0.6, and 0.6 for the FA Cup, Super Tuesday, and US election datasets, respectively. The temporal coverage τ of super-document is set to one minute, five minutes and one minute for the FA Cup, Super Tuesday and US Election datasets, respectively. If temporal coverage τ of a super-document is less than one minute, it reduces the impact of feature set, therefore, we set $\tau \geq 1$ minute(s) in a way that each sliding window contains 10 super-documents on average with exception to the FA Cup dataset. In FA Cup dataset, each sliding window contains exactly one EHG. The only EHG in each sliding window of interest is labeled as *Strong* and Eq. (19) works correctly in the detection model.

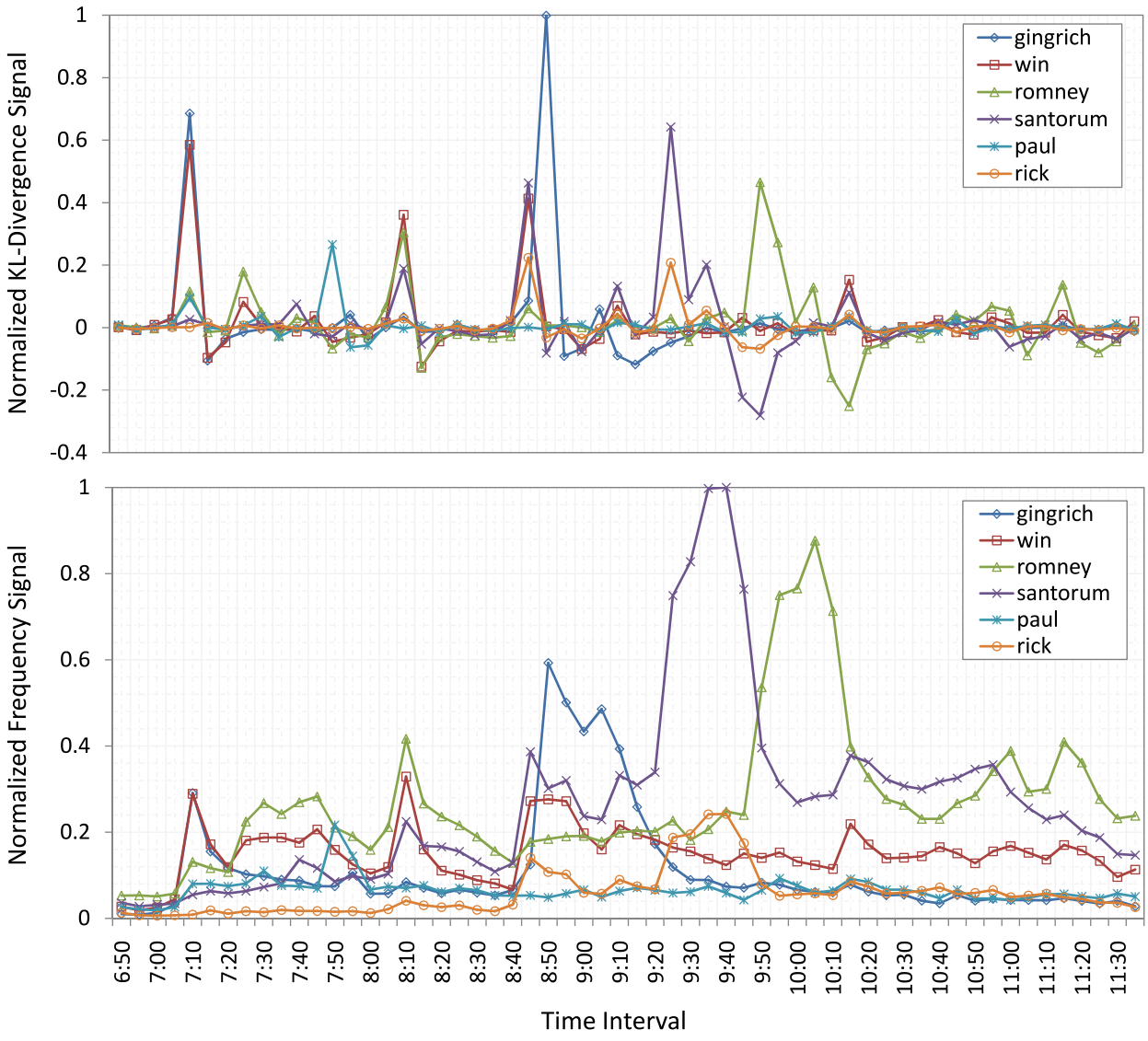


Fig. 6. Comparison between term frequency and EHG-based signals. The signals are generated over a five minutes time interval against top six event-related keywords reported in the ground truth for the Super Tuesday dataset. **Figure (top)** represents the signals based on modified KL-divergence and **Figure (bottom)** represents the signals based on term frequency.

6.4. Evaluation

We have extended the DHG approach (Saeed et al., 2018) by formulating a weighted DHG model in which topic centrality feature is extracted using weighted edges. The proposed approach *Enhanced Heartbeat Graph* (EHG) is constructed that generates a graph structure using modified KL-divergence and introduces a new feature *divergence factor*. Similarly, we have formulated its weighted model called WEHG that considers the edge weights based on modified KL-divergence. First, we compare four event detection methods that include DHG, WDHG, EHG, and WEHG for evaluating their performances based on topic-recall as shown in Fig. 8.

The results show that the EHG-based approach consistently performs better than the other three methods. Therefore, we take the best method for the next evaluation process with existing baseline methods.

Event detection techniques can be classified into five major categories. (1) Frequent Pattern Mining, (2) Probabilistic Models, (3) Clustering, (4) Exemplar-based, and (5) Matrix Factorization (Ibrahim et al., 2018).

We have considered the graph-based method as a separate category. Hence, the proposed approach is also compared with an existing graph-based method as well. For evaluation, we include at least one recent study from each of the categories mentioned above as baselines. To evaluate the performance of the proposed EHG approach, we compare the results on three benchmark datasets with the following nine state-of-the-art approaches:

- Soft Frequent Pattern Mining (SFPM) (Aiello et al., 2013) - (*Frequent Pattern Mining*)
- Latent Dirichlet Allocation (LDA) (Teh, Newman, & Welling, 2007) - (*Probabilistic Model*)
- Document-pivot (Doc-p) (Petrović et al., 2010), BN-gram (Aiello et al., 2013) - (*Clustering*)
- Exemplar (Elbagoury et al., 2015) - (*Exemplar-Based*)
- SVD-KMean (Nur'aini et al., 2015), SNMF-Orig, SNMF-KL (Prabandari & Murfi, 2017) - (*Matrix Factorization*)
- Graph Feature-pivot (GFeat-p) (O'Connor, Krieger, & Ahn, 2010) - (*Graph-based*)

The ground truth is created based on the events reported in the mainstream media. We cannot use topic precision for the evalua-

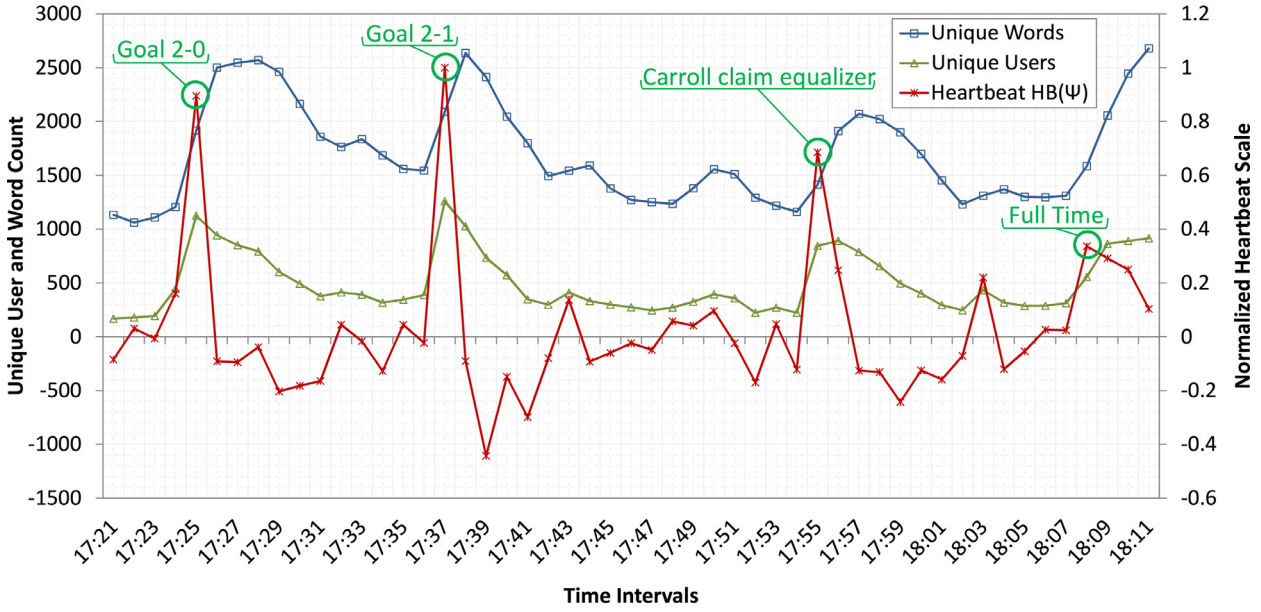


Fig. 7. The heartbeat signal showing significant increase in heartbeat score when an event occurs. The figure also shows the signals of unique words and users count across the text stream of the FA Cup 2012.

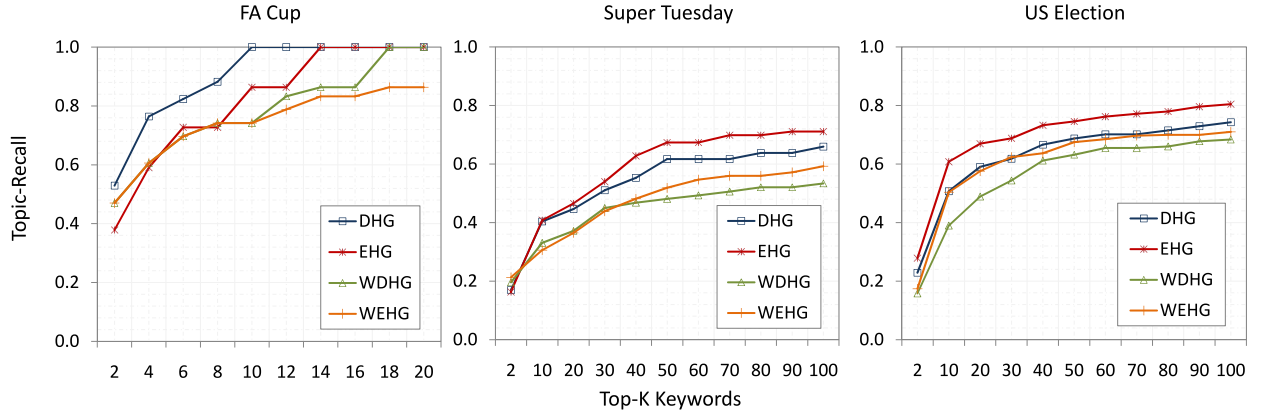


Fig. 8. Performance comparison of four event detection methods for Topic-Recall against three benchmark datasets.

tion as the text stream contains several newsworthy event-related topics which are not included in the ground truth (Aiello et al., 2013). The EHG approach also detected such topics e.g. "girl singing national anthem", "player injury", and "extra time added to the match" in the FA Cup dataset which are not present in the ground truth. Thus, topic precision cannot be truly measured.

Therefore, we have used two evaluation measures which are *Topic-Recall@K* (T-Rec) and *Keyword-Precision@K* (K-Pre). T-Rec is the percentage of ground truth topics detected correctly from top-K retrieved topics. In the ground truth, the topic-related keywords are divided into three groups *mandatory, optional* and *forbidden*. A topic is successfully detected if the detection method produces topic-related mandatory keywords but not forbidden as given in the ground truth, hence only mandatory keywords are used to calculate T-Rec. K-Pre is the percentage of keywords detected correctly out of the top-K number of words. For calculating K-Pre, all the keywords given in the set of mandatory and optional keywords are used. T-Rec and K-Pre are calculated by averaging the individual T-Rec and K-Pre scores from multiple event sliding windows. In comparison to the other two datasets, we obtained best results for the FA Cup dataset. Due to the limited and short-lived events in the FA Cup match, users publish micro-blogs with consistent vocabulary and focused intentions. Therefore, topics that appeared in the FA Cup are less diverse and easier to detect compared to Super Tuesday and the US Election datasets. We present the results for T-Rec at $K = [2, 4, 6, ..., 20]$ in the Table 4.

The mandatory keywords cover a broader semantic perspective and optional keywords provide descriptive information. For instance, at time 18:09 in the FA Cup dataset, the ground truth marks *full, time, final, whistle* and *gone* as mandatory keywords. Whereas *chelsea, champions, congratulations* and *win* are the optional keywords. Therefore, it is more likely that mandatory keywords are among the top trends but do not necessarily appear in the topmost position. Initially, the EHG approach has a comparable T-Rec at $2 \leq K \leq 12$ and gains the maximum possible T-Rec at $K > 12$.

For the Super Tuesday dataset, the EHG method shows similar superiority over all the other detection methods after $K > 30$ as shown in Table 5.

For the US Election, which is the largest dataset used in this experiment, the EHG approach produces better results and outperforms all other approaches after $K > 2$. The results for T-Rec are shown in Table 6.

Similarly, the proposed approach is able to detect relevant keywords with high precision compared to the other methods for all

Table 4
Performance comparison of ten event detection methods including proposed EHG approach. Table shows the topic-recall of each detection method at top-20 retrieved keywords for the FA Cup dataset.

Method	Top-K									
	2	4	6	8	10	12	14	16	18	20
LDA	.692	.692	.840	.840	.920	.920	.840	.840	.840	.750
Doc-P	**.769**	**.850**	**.920**	.920	**1**	**1**	**1**	**1**	**1**	**1**
Gfeat-P	.000	.308	.308	.375	.375	.375	.375	.375	.375	.375
SFPM	.615	.840	.840	**1**	**1**	**1**	**1**	**1**	**1**	**1**
BNGram	**.769**	**.920**	**.920**	.920	.920	.920	.920	.920	.920	.920
SVD+Kmean	.482	.596	.710	.824	.938	.951	.951	.951	.951	.951
SNMF-Orig	.100	.177	.254	.331	.389	.389	.389	.389	.389	.389
SNMF-KL	.167	.334	.502	.670	.837	.837	.840	.850	.850	.924
Exempler	.810	.838	.886	.908	.916	.916	.916	.916	.916	.916
EHG	.379	.591	.727	.727	.864	.864	**1**	**1**	**1**	**1**

Table 5
Performance comparison of ten event detection methods including proposed EHG approach. Table shows the topic-recall of each detection method at top-100 retrieved keywords for the Super Tuesday dataset.

Method	Top-K										
	2	10	20	30	40	50	60	70	80	90	100
LDA	.000	.000	.000	.180	.130	.130	.180	.280	.280	.370	.227
Doc-P	.227	.227	.310	.400	.460	.500	.500	.500	.540	.680	.680
Gfeat-P	.046	.045	.085	.180	.227	.280	.280	.280	.280	.280	.280
SFPM	.182	.182	.270	.325	.325	.325	.325	.325	.325	.325	.325
BNGram	**.500**	**.500**	**.540**	.540	.540	.540	.540	.540	.540	.540	.540
SVD+Kmean	.192	.236	.400	.488	.547	.580	.626	.666	.666	.666	.666
SNMF-Orig	.000	.045	.100	.183	.277	.277	.277	.320	.320	.363	.453
SNMF-KL	.000	.100	.183	.183	.318	.410	.366	.410	.453	.363	.410
Exempler	.246	.463	.538	**.572**	.586	.597	.600	.617	.638	.638	.638
EHG	.163	.408	.466	.540	**.628**	**.674**	**.674**	**.699**	**.699**	**.711**	**.711**

Table 6
Performance comparison of ten event detection methods including proposed EHG approach. Table shows the topic-recall of each detection method at top-100 retrieved keywords for the US Election dataset.

Method	Top-K										
	2	10	20	30	40	50	60	70	80	90	100
LDA	.109	.109	.185	.245	.220	.280	.325	.500	.475	.430	.460
Doc-P	.234	.234	.415	.505	.560	.615	.615	.690	.690	.720	.740
Gfeat-P	.078	.078	.140	.180	.180	.180	.180	.180	.180	.180	.180
SFPM	.359	.359	.465	.525	.540	.540	.540	.540	.540	.540	.540
BNGram	**.480**	.480	.495	.495	.495	.495	.495	.495	.495	.495	.495
SVD+Kmean	.110	.216	.420	.522	.588	.608	.647	.700	.720	.720	.740
SNMF-Orig	.075	.075	.154	.218	.439	.467	.483	.545	.563	.595	.595
SNMF-KL	.154	.154	.326	.400	.547	.581	.562	.618	.600	.652	.622
Exempler	.022	.142	.244	.364	.465	.532	.590	.628	.651	.662	.662
EHG	.279	**.608**	**.670**	**.688**	**.733**	**.746**	**.762**	**.772**	**.780**	**.796**	**.805**

three datasets at $K = 2$, as shown in Table 7. The results on all three benchmarks demonstrate that EHG is a superior approach in terms of performance and efficiency.

6.5. Limitations

The quantitative evaluation performed on the benchmark datasets shows that EHG is superior in comparison to the state-of-the-art approaches. However, there are a few limitations that need to be addressed as follow.

In the case of a sudden shift in the vocabulary appearing in text stream, the EHG is biased towards the negative (i.e., *Weak*) class at time $(i+1)\tau$ if and only if the heartbeat score of an EHG is greater at $i\tau$. For example, at time $i\tau$, the aggregation of positive and negative probability distribution in $P(\vartheta(v_l^{\psi^+})) = 0.89$, $P(\vartheta(v_l^{\psi^-})) = 0.11$, respectively. So, at time $(i+1)\tau$ if there is a major shift in the vocabulary of the text stream, then the probability distribution is affected negatively because our approach considers KL-divergence score, hence the weights of the bursty words at $i\tau$ will be negative at $(i+1)\tau$. There is a chance that at $(i+1)\tau$ a new event is emerging, but a greater heartbeat score at $i\tau$ might over-influence the probability distribution of the words in the EHG ψ at ,$(i+1)\tau$ therefore, labeling it *Weak*. However, the scenario

Table 7
Comparison of the EHG approach with nine state-of-the-art detection methods for K-Pre@2 for the FA Cup, Super Tuesday, and US Election datasets.

Method	Datasets		
	FA Cup	Super Tuesday	US Election
LDA	.164	.000	.165
Doc-P	.337	.511	.401
Gfeat-P	.000	.375	.375
SFPM	.233	.471	.241
BNGram	.299	.628	.405
SVD+Kmean	.242	.367	.300
SNMF-Orig	.330	.241	.241
SNMF-KL	.242	.164	.164
Exemplar	.300	.485	.391
EHG	**.442**	**.812**	**.591**

of sudden shift in the temporal frequency of the words is less likely to occur. The empirical evaluation shows that EHG works well on a targeted text stream where the data is crawled against seed words. Unlike targeted data, a live stream is different and consists of multiple events simultaneously. In such cases, it is challenging for the proposed approach to discriminate among multiple events. The EHG approach may not be able to associate and segregate different topics when multiple events appearing in the text stream concurrently.

The proposed approach detects events by processing and quantifying the data locally in each sliding window. However, there may be a case when an event occurs in sliding window $k\Delta t$ and keeps gaining in popularity in subsequent sliding window $(k+1)\Delta t$. In such a scenario, it considers both as two different emerging events. Such cases are also less likely to occur, especially when the temporal coverage of sliding window is large.

7. Conclusion

Event detection from Twitter stream needs to address the interlinked dimensions of the data to characterize the emerging events. The contents of micro-documents, the temporal distribution of data, and the theme around which social media users post, are essential aspects to detect an event. To capture these attributes from data, we designed three unique features: divergence factor, aggregated centrality, and trend probability. We developed four novel graph-based event detection methods (DHG, WDHG, EHG, and WEHG). Each method expressed text stream into temporal graphs uniquely. The designed features were extracted from the temporal graphs to generate a heartbeat signal. A binary classifier found event candidates using heartbeat signal. Then, event-related topics were extracted and ranked. We evaluated the performance of our four methods and compared the best approach (EHG) with nine state-of-the-art methods. Three publicly available benchmarks (FA Cup Final 2012, Super Tuesday 2012, and the US Election 2012) were used for the experiments. The results showed that the proposed approach is capable of dealing with complex nature of text streams. It outperformed existing state-of-the-art methods with improved precision and recall. The empirical evaluation showed that EHG approach is robust in terms of computational complexity and scalability.

In future, we plan to explore several directions including optimal temporal coverage of sliding windows and super-documents to address existing limitations. We also plan to employ ranking on the detected events, enabling to identify most important events quickly. Finally, we intend to evaluate the proposed approach on a live stream.

Conflict of interest

None.

Credit authorship contribution statement

Zafar Saeed: Conceptualization, Formal analysis, Funding acquisition, Investigation, Methodology, Writing - original draft. **Rabeeh Ayaz Abbasi:** Conceptualization, Data curation, Investigation, Methodology, Supervision, Writing - review & editing. **Imran Razzak:** Visualization, Writing - review & editing. **Onaiza Maqbool:** Writing - review & editing. **Abida Sadaf:** Visualization, Writing - review & editing. **Guandong Xu:** Resources, Validation, Writing - review & editing.

Acknowlgedgments

The first author, Zafar Saeed is financially supported by Higher Education Commission (HEC) of Pakistan under International Research Support Initiative Program (IRSIP), grant no. *IRSIP-35-Engg-06* and Indigenous Ph.D Fellowship Program, grant no. *112-31367-2PS1-213*.

Appendix A. Mathematical Notations

Table A1
Mathematical notations and their descriptions.

Notation	Description
U	Set of all users who have published at least one micro-document
T	Set of all time instances where at least one micro-document has been published
W	Set of all unique words appeared in the data stream
$\mathcal{D}$	Set of micro-documents
$\mathcal{D}^{\rho}$	Set of super-documents in text stream
Δt	Temporal coverage of sliding window
$\mathcal{G}$	Set of graphs representing graph series
V	Set of vertices in a graph represent words
E	Set of edges in a graph represent co-occurent of words
$\mathcal{G}^h$	Set of heartbeat graphs representing EHG series
$\varepsilon^{G_i^h}$ / ε^{ψ}	index edge vector for the graph G_i^h
$\mathcal{G}^{h(k\Delta t)}$	Set of all the EHGs temporally covered in kth sliding window
$i\tau$	Time interval starting at time instance t_i until $t_i + \tau$
V^{ψ}	Set of vertices in the graph G_i^h
β_i	constants for linear combination for probability distribution of topics
T^{ψ}	Set of nodes that are connected to positive edges in G_i^h
$\theta_{(k\Delta t)}$	Threshold value for classification function $Est(\psi)$ for kth sliding window
$\mathcal{N}$	Total number of super-documents in a sliding window
τ	Temporal coverage of super-document
ϖ	Average heartbeat score in a certain sliding window
ω	Adjustment parameter for threshold θ
A^{G_i}	Adjacency matrix for the graph G_i
A^{ψ}	Adjacency matrix for the graph G_i^h
α_i	constants for the linear combination of pair of adjacent graph

References

Adedoyin-Olowe, M., Gaber, M. M., Dancausa, C. M., Stahl, F., & Gomes, J. B. (2016). A rule dynamics approach to event detection in twitter with its application to sports and politics. *Expert Systems with Applications, 55*, 351–360.

Aiello, L. M., Petkos, G., Martin, C., Corney, D., Papadopoulos, S., Skraba, R., & Jaimes, A. (2013). Sensing trending topics in twitter. *Multimedia, IEEE Transactions on, 15*(6), 1268–1282.

Becker, H., Naaman, M., & Gravano, L. (2011). Beyond trending topics: Real-world event identification on twitter. In *International AAAI conference on web and social media* (pp. 438–441). USA: AAAI.

Benhardus, J., & Kalita, J. (2013). Streaming trend detection in twitter. *International Journal of Web Based Communities, 9*(1), 122–139.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research, 3*(Jan), 993–1022.

Buntain, C. (2015). Discovering credible events in near real time from social media streams. In *Proceedings of the 24th international conference on world wide web* (pp. 481–485). New York, NY, USA. ACM: ACM.

Chen, Y., Amiri, H., Li, Z., & Chua, T.-S. (2013). Emerging topic detection for organizations from microblogs. In *Proceedings of the 36th international ACM SIGIR conference on research and development in information retrieval* (pp. 43–52). New York, NY, USA. ACM: ACM.

Cheng, T., & Wicks, T. (2014). Event detection using twitter: A spatio-temporal approach. *PloS One, 9*(6), e97807.

Chierichetti, F., Kleinberg, J., Kumar, R., Mahdian, M., & Pandey, S. (2014). Event detection via communication pattern analysis. In *Proceedings of 8th international AAAI conference on weblogs and social media* (pp. 51–60). USA: AAAI. Association for the Advancement of Artificial Intelligence.

Choi, H.-J., & Park, C. H. (2019). Emerging topic detection in twitter stream based on high utility pattern mining. *Expert Systems with Applications, 115*, 27–36.

Cordeiro, M. (2012). Twitter event detection: Combining wavelet analysis and topic inference summarization. In *Doctoral symposium on informatics engineering* (pp. 11–16).

Earle, P. S., Bowden, D. C., & Guy, M. (2012). Twitter earthquake detection: Earthquake monitoring in a social world. *Annals of Geophysics, 54*(6).

Edouard, A., Cabrio, E., Tonelli, S., & Le Thanh, N. (2017). Graph-based event extraction from twitter. In *RANLP17* (pp. 1–10).

Elbagoury, A., Ibrahim, R., Farahat, A. K., Kamel, M. S., & Karray, F. (2015). Exemplar-based topic detection in twitter streams. In *ICWSM* (pp. 610–613).

Gao, X., Cao, J., He, Q., & Li, J. (2013). A novel method for geographical social event detection in social media. In *Proceedings of the fifth international conference on internet multimedia computing and service* (pp. 305–308). New York, NY, USA. ACM: ACM.

Guo, J., & Gong, Z. (2017). A density-based nonparametric model for online event discovery from the social media data. In *Proceedings of the 26th international joint conference on artificial intelligence* (pp. 1732–1738). AAAI Press.

He, Q., Chang, K., & Lim, E.-P. (2007). Analyzing feature trajectories for event detection. In *Proceedings of the 30th annual international ACM SIGIR conference on research and development in information retrieval* (pp. 207–214). ACM.

Huang, J., Peng, M., & Wang, H. (2015). Topic detection from large scale of microblog stream with high utility pattern clustering. In *Proceedings of the 8th workshop on ph. d. workshop in information and knowledge management* (pp. 3–10). New York, NY, USA. ACM: ACM.

Ibrahim, R., Elbagoury, A., Kamel, M. S., & Karray, F. (2018). Tools and approaches for topic detection from twitter streams: Survey. *Knowledge and Information Systems, 54*(3), 511–539.

Iyengar, A., Finin, T., & Joshi, A. (2011). Content-based prediction of temporal boundaries for events in twitter. In *Privacy, security, risk and trust (PASSAT) and 2011 IEEE third inernational conference on social computing (SocialCom), 2011 IEEE third international conference on* (pp. 186–191). USA. IEEE: IEEE.

Jarwar, M. A., Abbasi, R. A., Mushtaq, M., Maqbool, O., Aljohani, N. R., Daud, A., & Chong, I. (2017). Communiments: A framework for detecting community based sentiments for events. *International Journal on Semantic Web and Information Systems (IJSWIS), 13*(2), 87–108.

Kaleel, S. B., & Abhari, A. (2015). Cluster-discovery of twitter messages for event detection and trending. *Journal of Computational Science, 6*, 47–57.

Katragadda, S., Benton, R. G., & Raghavan, V. V. (2017). Framework for real-time event detection using multiple social media sources. In *50th hawaii international conference on system sciences, HICSS 2017, Hilton Waikoloa village, Hawaii, USA, January 4-7, 2017* (pp. 1716–1725).

Katragadda, S., Virani, S., Benton, R., & Raghavan, V. (2016). Detection of event onset using twitter. In *2016 international joint conference on neural networks (IJCNN)* (pp. 1539–1546). doi:10.1109/IJCNN.2016.7727381.

Kullback, S. (1997). *Information theory and statistics*. Courier Corporation.

Kulldorff, M. (2010). Satscan user guide for version 9.0. https://www.satscan.org/. [Online; accessed September 5, 2018].

Kumar, S., Liu, H., Mehta, S., & Subramaniam, L. V. (2015). Exploring a scalable solution to identifying events in noisy twitter streams. In *Proceedings of the 2015 IEEE/ACM international conference on advances in social networks analysis and mining 2015* (pp. 496–499). ACM.

Li, C., Sun, A., & Datta, A. (2012a). Twevent: Segment-based event detection from tweets. In *Proceedings of the 21st ACM international conference on information and knowledge management* (pp. 155–164). ACM.

Li, C., Weng, J., He, Q., Yao, Y., Datta, A., Sun, A., & Lee, B.-S. (2012b). Twiner: Named entity recognition in targeted twitter stream. In *Proceedings of the 35th international ACM SIGIR conference on research and development in information retrieval* (pp. 721–730). ACM.

Li, R., Lei, K. H., Khadiwala, R., & Chang, K. C.-C. (2012c). Tedas: A twitter-based event detection and analysis system. In *Data engineering (ICDE), 2012 IEEE 28th international conference on* (pp. 1273–1276). IEEE.

Long, R., Wang, H., Chen, Y., Jin, O., & Yu, Y. (2011). Towards effective event detection, tracking and summarization on microblog data. In *International conference on web-age information management* (pp. 652–663). Springer.

Nguyen, D. T., & Jung, J. E. (2017). Real-time event detection for online behavioral analysis of big social data. *Future Generation Computer Systems, 66*, 137–145.

Nguyen, D. T., & Jung, J. J. (2015). Real-time event detection on social data stream. *Mobile Networks and Applications, 20*(4), 475–486.

Nur'aini, K., Najahaty, I., Hidayati, L., Murfi, H., & Nurrohmah, S. (2015). Combination of singular value decomposition and k-means clustering methods for topic detection on twitter. In *Advanced computer science and information systems (ICACSIS), 2015 international conference on* (pp. 123–128). IEEE.

O'Connor, B., Krieger, M., & Ahn, D. (2010). Tweetmotif: Exploratory search and topic summarization for twitter. In *ICWSM* (pp. 384–385).

Ozdikis, O., Senkul, P., & Oguztuzun, H. (2012). Semantic expansion of tweet contents for enhanced event detection in twitter. In *Proceedings of the 2012 international conference on advances in social networks analysis and mining (ASONAM 2012)* (pp. 20–24). IEEE Computer Society.

Panagiotou, N., Katakis, I., & Gunopulos, D. (2016). Detecting events in online social networks: Definitions, trends and challenges. In *Solving large scale learning tasks. challenges and algorithms* (pp. 42–84). Springer.

Papadopoulos, S., Corney, D., & Aiello, L. M. (2014). Snow 2014 data challenge: Assessing the performance of news topic detection methods in social media. In *SNOW-DC@WWW* (pp. 1–8).

Petrović, S., Osborne, M., & Lavrenko, V. (2010). Streaming first story detection with application to twitter. In *Human language technologies: The 2010 annual conference of the North American chapter of the association for computational linguistics* (pp. 181–189). Association for Computational Linguistics.

Prabandari, R., & Murfi, H. (2017). Comparative study of original recover and recover kl in separable non-negative matrix factorization for topic detection in twitter. In *AIP conference proceedings* (pp. 030144–030145). AIP Publishing.

Saeed, Z., Abbasi, R. A., Sadaf, A., Razzak, M. I., & Xu, G. (2018). Text stream to temporal network – A dynamic heartbeat graph to detect emerging events on twitter. In *Advances in knowledge discovery and data mining – 22nd pacific-asia conference, PAKDD 2018, Melbourne, Australia, June 3–6, 2018, proceedings* (pp. 534–545).

Sethi, T. S., & Kantardzic, M. (2017). On the reliable detection of concept drift from streaming unlabeled data. *Expert Systems with Applications, 82*, 77–99.

Shamma, D. A., Kennedy, L., & Churchill, E. F. (2011). Peaks and persistence: Modeling the shape of microblog conversations. In *Proceedings of the ACM 2011 conference on computer supported cooperative work* (pp. 355–358). ACM.

Teh, Y. W., Newman, D., & Welling, M. (2007). A collapsed variational bayesian inference algorithm for latent dirichlet allocation. In *Advances in neural information processing systems* (pp. 1353–1360).

Velampalli, S., & Eberle, W. (2017). Novel graph based anomaly detection using background knowledge. In *Proceedings of the thirtieth international florida artificial intelligence research society conference* (pp. 538–543).

Weng, J., & Lee, B.-S. (2011). Event detection in twitter. In *ICWSM: 11* (pp. 401–408).

Yang, J., & Leskovec, J. (2011). Patterns of temporal variation in online media. In *Proceedings of the fourth ACM international conference on web search and data mining* (pp. 177–186). ACM.

Yin, J., & Wang, J. (2016). A text clustering algorithm using an online clustering scheme for initialization. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1995–2004). ACM.

Zhang, C., Liu, L., Lei, D., Yuan, Q., Zhuang, H., Hanratty, T., & Han, J. (2017). Triovecevent: Embedding-based online local event detection in geo-tagged tweet streams. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 595–604). ACM.

Zhang, X., Chen, X., Chen, Y., Wang, S., Li, Z., & Xia, J. (2015). Event detection and popularity prediction in microblogging. *Neurocomputing, 149*, 1469–1480.

Zhou, D., Chen, L., & He, Y. (2015). An unsupervised framework of exploring events on twitter: Filtering, extraction and categorization. In *Proceedings of 29th AAAI conference on artificial intelligence* (pp. 2468–2475). USA: AAAI.

Zhou, X., & Chen, L. (2014). Event detection over twitter social media streams. *The VLDB Journal The International Journal on Very Large Data Bases, 23*(3), 381–400.