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*

Abstract

A huge amount of data is generated every second on social media. Event and topic detection must address both scalability and accuracy challenges when using enormous and noisy data collections from social media. Documents describing the same event and story have a similar set of collocated keywords that can be used to identify the event time and its description. In this work, we propose a novel graph-based approach, called the Enhanced Heartbeat Graph (EHG), which does not only detect events at an early stage but also suppresses event-related topics in the upcoming text stream in order to highlight other micro details. We have compared the proposed approach with ten state-of-the-art approaches for event detection. Experiment results on real-world data (i.e., Football Association Challenge Cup Final, Super Tuesday, and the US Election 2012) show considerable improvement in most cases, while computational complexity remains very attractive.

Keywords: event detection, twitter, text stream, emerging trends, dynamic graph, time series network, big data

1. Introduction

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. In addition to its huge volume and diversity, much of the data is inter-dependent in nature. 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

[*] Rabeeh Abbasi is the corresponding author.

Email addresses: zsaeed@cs.qau.edu.pk (Zafar Saeed), rabbasi@qau.edu.pk (Rabeeh Ayaz Abbasi), mirpak@gmail.com (Imran Razzak), onaiza@qau.edu.pk (Onaiza Maqbool), abida.sadaf@qau.edu.pk (Abida Sadaf), guandong.xu@uts.edu.au (Guandong Xu)

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 detection of emerging events involves the identification of trending topics related to the event by monitoring and processing the text stream (Panagiotou et al., 2016). Event detection from social media has recently 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 and produce an abundance of meaningful information on social media text streams such as Twitter. For example, Figure 1 shows the tweet traffic behavior for the events occurred during the final of "The Football Association Challenge Cup" (FA Cup) 2012. Labeled spikes in the figure represent corresponding events occurred during the match. The instant public reaction to the match events can be observed through an increase in tweet frequency.

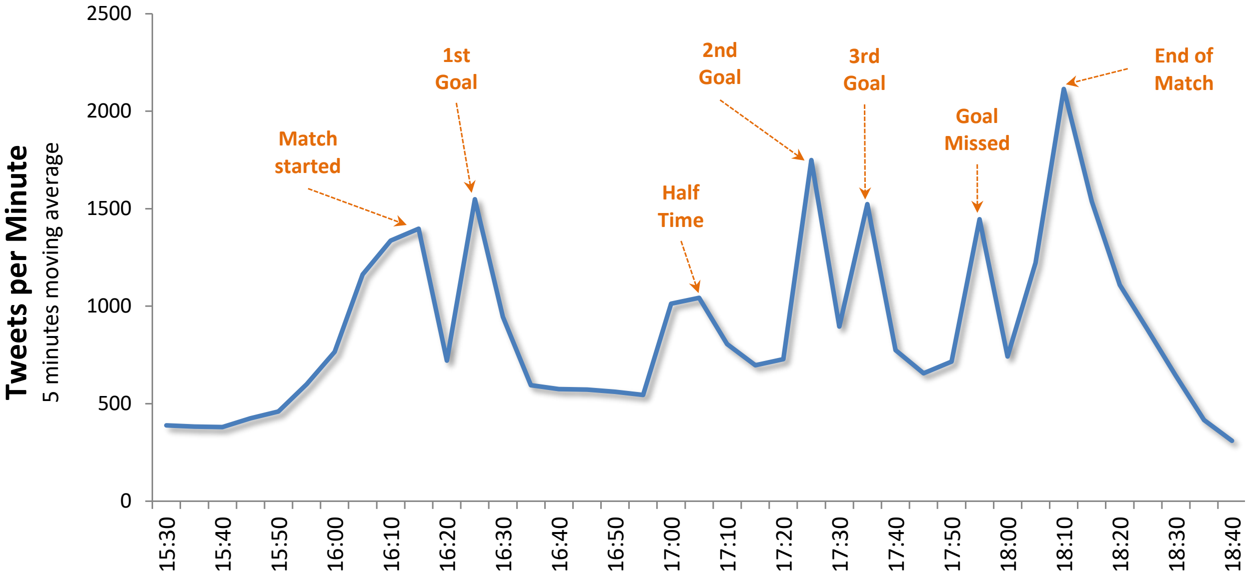


Figure 1: ~~Tweet frequency for the FA Cup Final, 2012 between time 15:30 and 18:40.~~

It is interesting to analyze such data to extract meaningful information which can be used for searching, discovering, and sensing events as well as identifying their nature. However, analyzing large, diverse and noisy data is difficult due to the challenges such as scalability, accuracy and efficiency (Earle et al., 2012; Jarwar et al., 2017). Generally, there are two approaches for detecting events: *document pivoted* and *feature pivoted* (discussed in detail in Section 2). 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, rapidly changing contents, etc. 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 and Jung, 2015; Shamma et al., 2011; Yang and Leskovec, 2011). Burstiness often dominates other details in the data which can be equally crucial for detecting a different event which is not so bursty.

We address the drawback of feature pivot approach by measuring the Kullback-Leibler (KL) divergence score (Kullback, 1997), also known as relative entropy. 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 characteristic mentioned above can be observed in Figure 2 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 Figure 2(A_0) and Figure 2(B_0) are based on KL-divergence score, whereas in Figure 2(A_1) and Figure 2(B_1) are based on term-frequency. In Figure 2(A_1) it can be observed that keywords "romney" and "win" are dominating, whereas in Figure 2(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 Figure 2(B_0), whereas they are dominated by "romney" in Figure 2(B_1) at time interval 10:15. Furthermore, Section 7.2 empirically discusses the effect of modified KL-divergence on data distribution in detail.

This paper extends our previous work (Saeed et al., 2018). Unlike our previous work, we develop a detection model incorporating a modified KL-divergence for generating graph structures (see Section 4.4). This results in a new feature i.e., *Divergence Factor* (see Section 4.3). The above model is further extended to weighted graph structure as well. Furthermore, a new model is also created to enhance our previous work. We create and compare four event detection model and the winning model is compared with ten different baseline methods (see Section 7.4). 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 7.2) and as well as a detailed time complexity analysis is conducted to evaluate the efficiency of the proposed approach (see Section 5).

The goal of this study is to find unusual data patterns in Twitter stream to detect events while overcoming some of the weaknesses of the existing methods. The core idea of proposed approach is to address the dominating nature of burstiness in the data by suppressing the bursty topics once captured. Therefore, topological and temporal relationship in the data are measured using a modified KL-divergence of words and their co-occurrences with respect to time. The aforementioned characteristics are inherent in the proposed feature design for the detection model.

We create temporal graphs based on feature pivot. Selecting useful edges in the graph

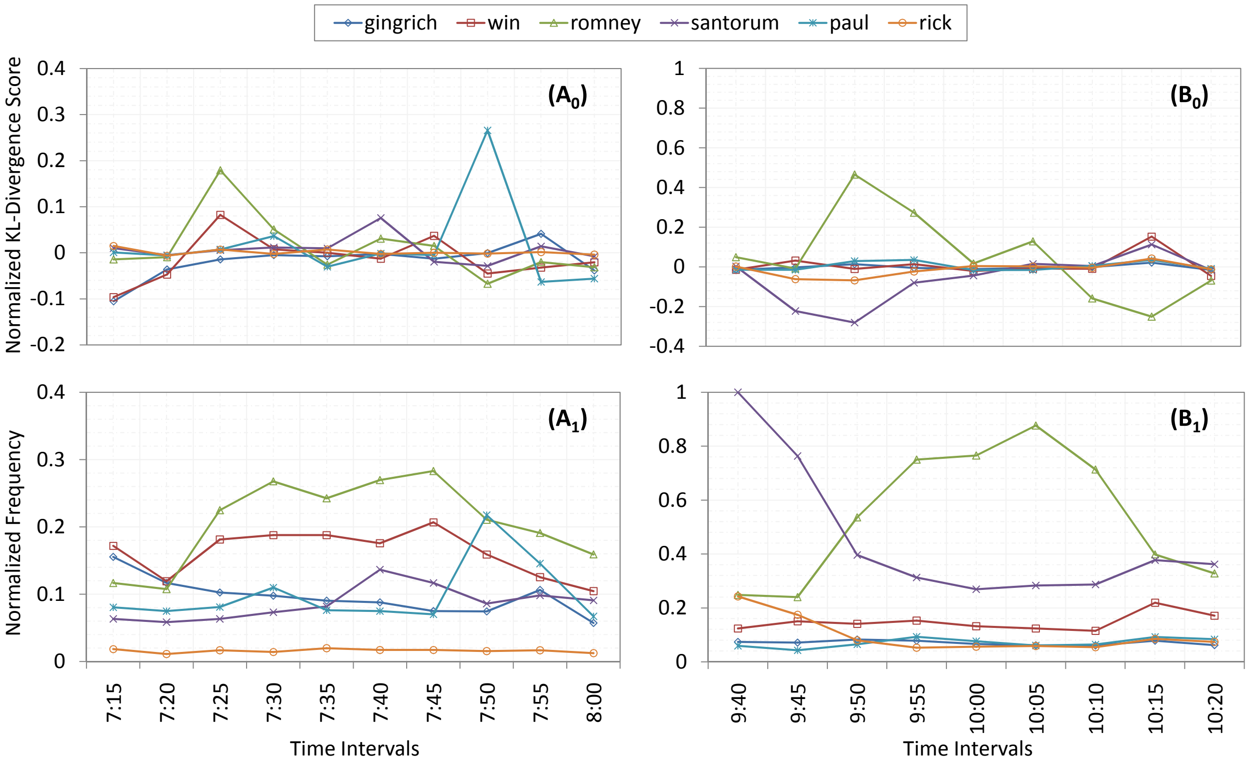


Figure 2: Motivation: Some of the event related topics are dominating the text stream

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 et al., 2017; Katragadda et al., 2016, 2017), we select unique words having the highest ranking scores from candidate graphs to extract the event related topics without compromising the performance.

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

- A novel graph-based approach named the Enhanced Heartbeat Graph (EHG) which is efficient in the detection of events from Twitter stream at early stages.
- EHG-based feature design for event detection and 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.

The rest of the paper is organized as follows: Section 2 describes background. Section 3 describes the preliminaries that formulate the concepts essentially required to understand our approach. Section 4 describes our proposed approach that includes transformation of Twitter stream into series of EHGs, feature design and detection method. Section 5 describes the time and space complexity analysis of the proposed approach. Section 6 describes the benchmark datasets. We discuss the observations, parameter selection, results and evaluation of the proposed approach in Section 7. Finally, Section 8 concludes the paper and highlight directions for the expected future work.

2. Background

In this section, we review the recent developments in the areas of event detection on Twitter. Generally, there are two approaches (Weng and 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.

2.1. Document Pivot Methods

Pivoted document is a classic approach that groups documents into clusters based on their similarity. In this regard, Petrović et al. (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 and 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 et al. (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 et al. (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 et al. (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.

2.2. 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 and Leskovec, 2011). Generally, a text stream is segmented into single words (or n-grams) often represented by bag-of-words (BoW) model. In order to identify significant keywords representing events, several research studies (Li et al., 2012c; Nguyen and Jung, 2015; Shamma et al., 2011; Yang and Leskovec, 2011) have used frequency signals while processing the text stream. The keywords that have high frequency/burst are retained and further processed to segregate the information which is later used to identify the occurrence of events.

To find abnormal spikes in the keyword-based frequency signals, He et al. (2007) uses Discrete Fourier Transformation (DFT) method for grouping keywords based on features extracted from periodicity and strength of the power spectrum (He et al., 2007). The method is extended by Weng and Lee (2011) using wavelet analysis on the word frequencies to obtain new features for every word. Based on low signal auto-correlations, trivial words are filtered out (Weng and Lee, 2011). Events are identified by clustering the remaining words using graph partitioning. However, these methods (He et al., 2007; Weng and Lee, 2011) are unable to keep track of the temporal information which is a significant aspect for detecting events.

Li et al. (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, users 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 et al., 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 et al. (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 similar patterns are grouped using k-nearest-neighbors to represent emerging events. Similarly, Adedoyin-Olowe et al. (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.

Generally, 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 widely used (Sethi and Kantardzic, 2017; Velampalli and Eberle, 2017).

Long et al. (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

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

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 and 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.

Considering the limitations in the existing methods, we present a novel graph-based approach named "Enhanced Heartbeat Graph" (EHG) for detecting events. EHG is suitable for microblogging text streams like Twitter and Weibo. We formulate the text stream as a series of temporal graphs and extract three features: divergence factor, trend probability, and aggregated centrality from each temporal graph. Extracted features are fused to generate a heartbeat signal. Furthermore, an adaptive measure is proposed for assigning binary class labels for detecting emerging events in the text stream. After detecting the event at an early stage, subsequently EHG suppresses the burstiness of detected event-related topics in order to identify further emerging events. This unique characteristic makes EHG approach robust in detecting events and related topics.

3. 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 build our method over BoW model to 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 and Kalita, 2013; Buntain, 2015; Nguyen and Jung, 2017; Zhou and Chen, 2014; Zhou et al., 2015). 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.

3.1. Micro-Document

In this study, a micro-document and micro-blog refer to a tweet and Twitter respectively. A micro-document d_i is a short textual content consisting of words that are published online through a micro-blog. 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.

3.2. 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 $(i-1)^{th}$ micro-documents published at time $\pi_1(d_i)$ and $\pi_1(d_{i-1})$ respectively, such that $\pi_1(d_i) \geq \pi_1(d_{i-1})$.

3.3. Super-Document

The length of micro-documents is limited hence, statistical inference do 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 create k partitions of micro-documents in the text stream $\mathcal{D}^\rho = \{\{d_1, d_2, ..., d_p\}, \{d_{p+1}, ..., d_{p+q}\}, ..., \{..., d_n\}\}$. Each partition 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 could be used later to generate graph series (mentioned in Section 4.1) which increases the cohesiveness among the topics that co-occur. Thus, a set of super-documents consists of k discrete partitions, where each partition $d_i^\rho \in \mathcal{D}^\rho$ such that $d_i^\rho \subset \mathcal{D}$ and $\bigcup\limits_i d_i^\rho = \mathcal{D}$ and $\bigcap\limits_i d_i^\rho = \varnothing$. 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.

3.4. 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.

4. 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 framework of the proposed EHG approach is illustrated in Figure 3 which shows data status and processing at each step. The data undergoes five transformations from the data source to detected 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 3.3). A series of graphs $\mathcal{G}$ is then created (as described in Section 4.1) using the set of super-documents. As a result, the graph represents all the unique words, their frequencies and co-occurrence relations between them. 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 4.2). The EHG inherits 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 4.3). 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 4.4).

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

4.1. 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 implicitly leads to co-occurrence relationship between the words of all the micro-documents in a super-document d_i^ρ because each word is unique in the graph G_i as shown in Figure 4. 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 $v_m \neq v_n$
- $\mathscr{W} : V \rightarrow \mathbb{R}$ and $\mathscr{S} : E \rightarrow \mathbb{R}$ are the functions that assign weights to each node and edge in the graph G_i using Equations 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 $v_m \neq v_n$. The cliques between the nodes of each micro-document also increase the centrality of words in the graph structure and express the importance due to their co-occurrence with diverse set of words in a text stream.

For instance, consider a super-document is to be processed to create a graph. The super-document d_i^ρ contains three micro-documents and each micro-document consists of some

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

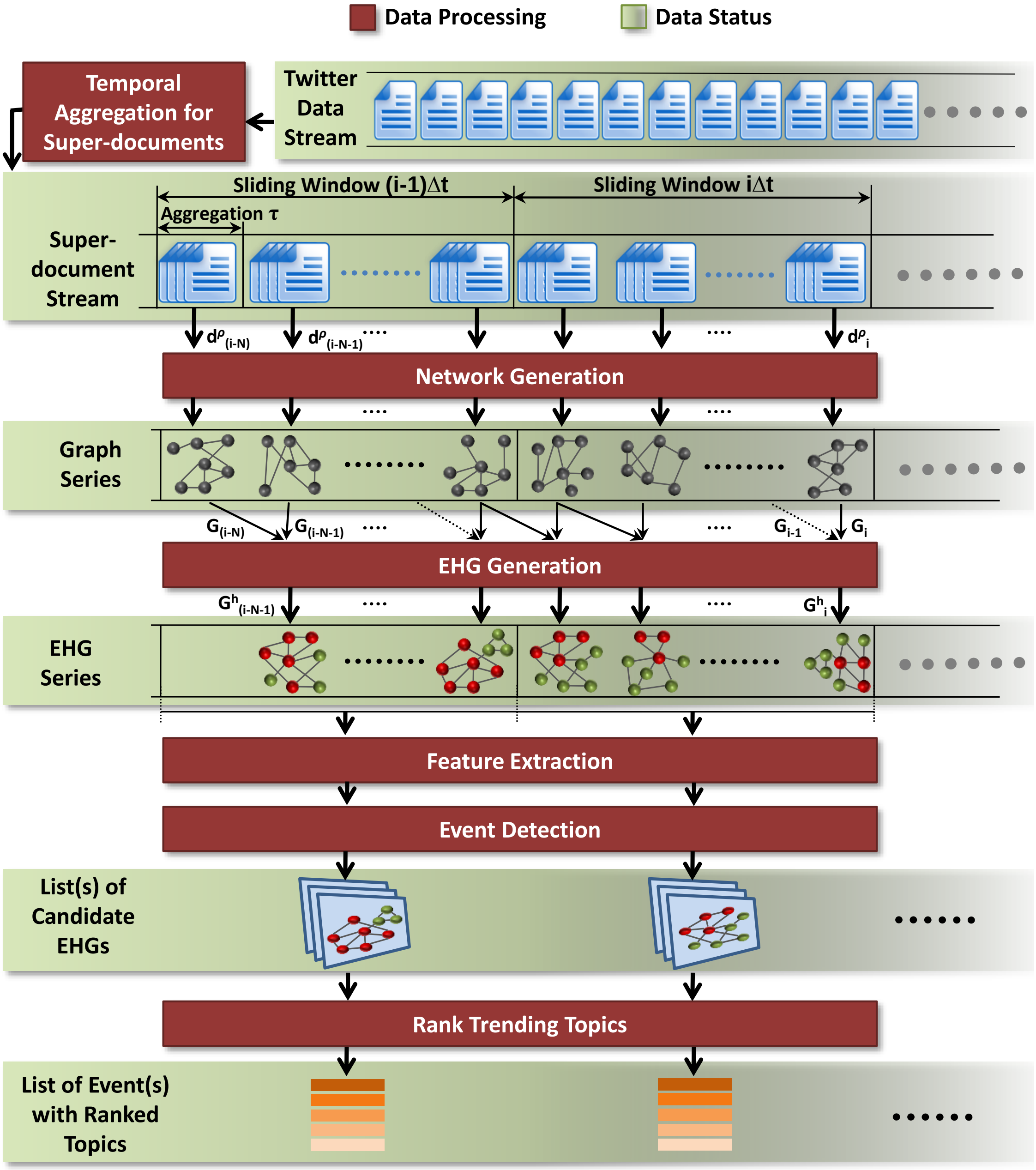


Figure 3: Work-flow of Twitter stream processing for event detection. The figure shows the step-wise transformation of text stream into corresponding graph representation and data processing modules.

words, as shown in Table 1. Each micro-document d_i, where $d_i \in d_i^\rho$, is embedded sequentially into graph G_i. Figure 4 shows the creation of graph for an example super-document d_i^ρ and elaborates the structural updates for the embedding of 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.

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

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

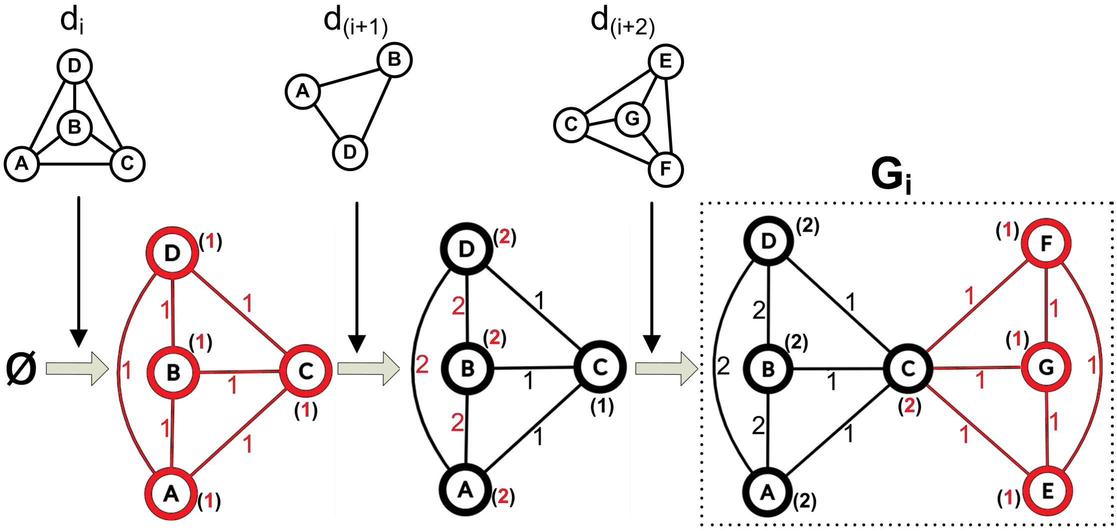


Figure 4: Graph creation against an example super-document $d_i^\rho = \{d_i, d_{(i+1)}, d_{(i+2)}\}$ shown in Table 1. The red nodes and edges are newly embedded nodes and edges. The red labels are new or updated weights. The figure shows the status of graph G_i with the embedding of each micro-document d_i.

From the given example, it can be observed that nodes A, B, and C have the same frequency in the super-document, but C has greater number of unique edges as compared to A and B, which makes C more important in the graph. Therefore, a clique among the words of each micro-document increases the significance 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.

4.2. 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 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, 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 resultant matrices. This might result in isolated nodes in both G_{i-1} and G_i as shown in Figure 5, 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 Equation 3. It is commonly called a distance measure, but unlike distance measures, it is asymmetric. It is very 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 Figure 5), 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 could 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 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 Equations 4 and 5 respectively.

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

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

Finally, the divergence between both distributions is calculated as shown in Equation 8 and used as a key feature in our model. The extracted features (described in Section 4.3) 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.

Figure 5 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 5 for details).

Algorithm 1: Generate set of Enhanced Heartbeat Graphs

```
input : $\mathcal{G} = \{G_1, G_2, G_3, ..., G_{|\mathcal{D}^\rho|}\}$ set of a graph series where $G_i \in \mathcal{G}$ is created for
        corresponding super-document $d_i^\rho \in \mathcal{D}^\rho$
output: $\mathcal{G}^h = \{G_1^h, G_2^h, ..., G_{|\mathcal{G}|-1}^h\}$

for $i \leftarrow 2$ to $|\mathcal{G}|$ do
    $V^\psi \leftarrow V^{G_i} \cup V^{G_{i-1}}$
    $V^{G_i} \leftarrow$ recreate vertices using $V^\psi$
    $V^{G_{i-1}} \leftarrow$ recreate vertices using $V^\psi$
    $A \leftarrow$ create adjacency matrix for $G_{(i-1)}$ using $V^\psi$
    $B \leftarrow$ 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
end
```

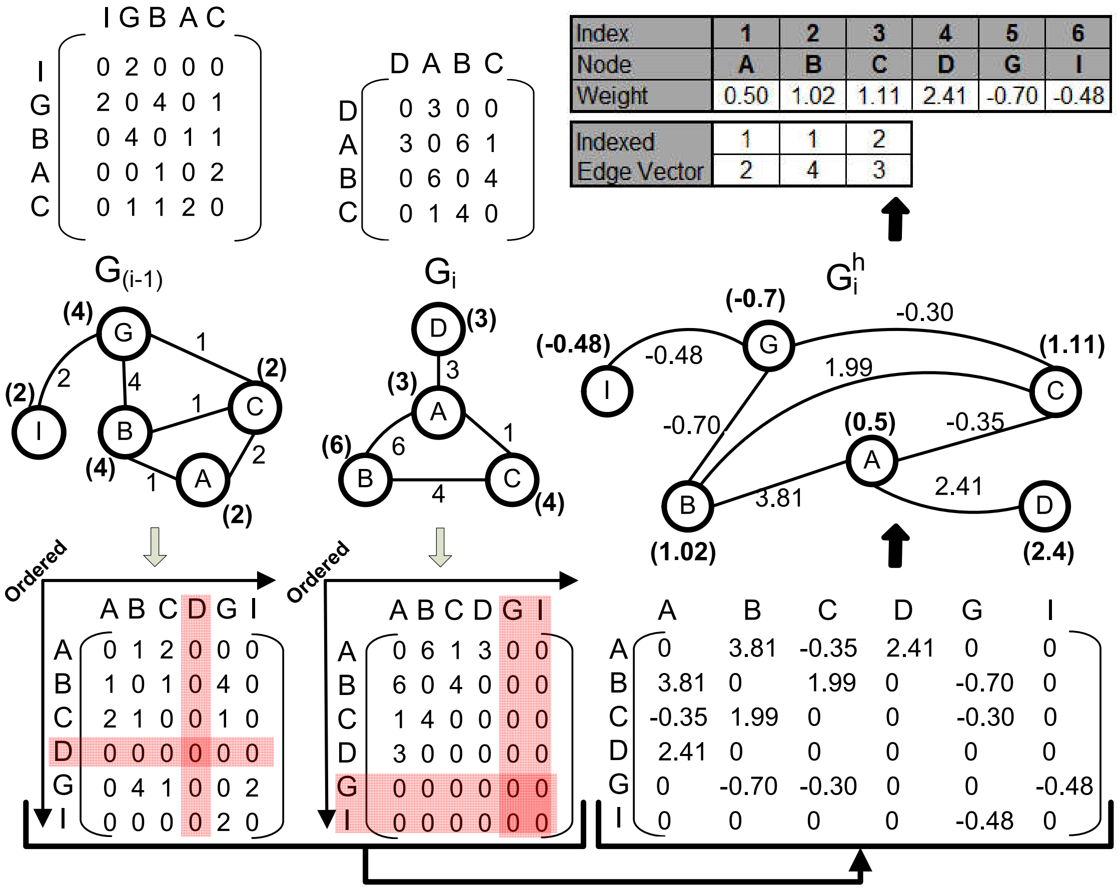
The EHG series is generated in a streaming fashion; therefore, it is temporally well aligned with the text stream. The EHG approach implicitly suppresses and handles the dominance of bursty topics by calculating modified KL-divergence of nodes and edges between each pair of adjacent graphs G_{i-1} and G_i. The weights of the nodes and the edges in an EHG G_i^h have the following possibilities 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

Algorithm 2: Generate Enhanced Heartbeat Graphs

```
input  : $A, B$ are adjacency matrices that represent $G_{(i-1)}$ and $G_{(i)}$ respectively.
         $V^A$ and $V^B$ are sets of vertices in $G_{(i-1)}, G_{(i)}$ respectively
output: $EHG$ against $G_{(i-1)}$ and $G_{(i)}$ containing an index edge vector $\varepsilon$ and list of
         weighted vertices $V^H$

$V^H \leftarrow List()$
for $k \leftarrow 1$ to $|V^B|$ do
    $V^H_{(k)} \leftarrow \left(V^B_{(k)} + 1\right) \times \log \frac{V^B_{(k)}+1}{V^A_{(k)}+1}$        ▶Using Equation 4
end
$\varepsilon \leftarrow List()$
for $r \leftarrow 2$ to $|V^H|$ do
    for $c \leftarrow 1$ to $r - 1$ do
        edgeWeight $\leftarrow \left(B_{(r,c)} + 1\right) \times \log \frac{B_{(r,c)}+1}{A_{(r,c)}+1}$        ▶Using Equation 5
        if $edgeWeight > 0$ AND $edge(r,c) \notin \varepsilon$ then
            $\varepsilon \leftarrow \varepsilon \cup edge(r, c)$
        end
    end
end
```



$$\begin{array}{c|ccccc} & I & G & B & A & C \\ \hline I & 0 & 2 & 0 & 0 & 0 \\ G & 2 & 0 & 4 & 0 & 1 \\ B & 0 & 4 & 0 & 1 & 1 \\ A & 0 & 0 & 1 & 0 & 2 \\ C & 0 & 1 & 1 & 2 & 0 \end{array}$$

$$\begin{array}{c|cccc} & D & A & B & C \\ \hline D & 0 & 3 & 0 & 0 \\ A & 3 & 0 & 6 & 1 \\ B & 0 & 6 & 0 & 4 \\ C & 0 & 1 & 4 & 0 \end{array}$$

Index	1	2	3	4	5	6
Node	A	B	C	D	G	I
Weight	0.50	1.02	1.11	2.41	-0.70	-0.48

Indexed Edge Vector	1	1	2
	2	4	3

$$\begin{array}{c|cccccc} & A & B & C & D & G & I \\ \hline A & 0 & 1 & 2 & 0 & 0 & 0 \\ B & 1 & 0 & 1 & 0 & 4 & 0 \\ C & 2 & 1 & 0 & 0 & 1 & 0 \\ D & 0 & 0 & 0 & 0 & 0 & 0 \\ G & 0 & 4 & 1 & 0 & 0 & 2 \\ I & 0 & 0 & 0 & 0 & 2 & 0 \end{array}$$

$$\begin{array}{c|cccccc} & A & B & C & D & G & I \\ \hline A & 0 & 6 & 1 & 3 & 0 & 0 \\ B & 6 & 0 & 4 & 0 & 0 & 0 \\ C & 1 & 4 & 0 & 0 & 0 & 0 \\ D & 3 & 0 & 0 & 0 & 0 & 0 \\ G & 0 & 0 & 0 & 0 & 0 & 0 \\ I & 0 & 0 & 0 & 0 & 0 & 0 \end{array}$$

$$\begin{array}{c|cccccc} & A & B & C & D & G & I \\ \hline A & 0 & 3.81 & -0.35 & 2.41 & 0 & 0 \\ B & 3.81 & 0 & 1.99 & 0 & -0.70 & 0 \\ C & -0.35 & 1.99 & 0 & 0 & -0.30 & 0 \\ D & 2.41 & 0 & 0 & 0 & 0 & 0 \\ G & 0 & -0.70 & -0.30 & 0 & 0 & -0.48 \\ I & 0 & 0 & 0 & 0 & -0.48 & 0 \end{array}$$

Figure 5: Shows an example of how an Enhanced Heartbeat Graph (EHG) is generated from two subsequent graphs which are adjacent.

4.3. Feature Design

Existing techniques such as (Becker et al., 2011; Chen et al., 2013; Chierichetti et al., 2014; Gao et al., 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 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 7.2. 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 Equation 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 $\overline{\mathscr{S}}(e_k^\psi)$ and $\delta(e_k^\psi)$ are the weight and normalized weight of k^{th} edge in the EHG ψ respectively.

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

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

4.3.1. Divergence Factor

Divergence factor $DF(\psi)$ is an accumulated score of the nodes weight in the EHG ψ which shows the intensity of drift in the topics and their popularity in the text stream. Drift appears when new topics are observed in the data and the popularity indicates an increase in the divergence score of previously observed topics. A divergence factor score shows how previously observed topics are trending, in terms of popularity, and if new topics are emerging at time interval $i\tau$ compared to $(i-1)\tau$.

The divergence factor is calculated by accumulating the weights of all the nodes in the EHG ψ using Equation 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).

4.3.2. Trend Probability

A node in the EHG ψ can have a negative or positive weight. To calculate trend probability $TP(\psi)$, 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 Equation 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 that has positive and negative weights respectively. The probability distribution over the positive and negative weights of the nodes are then linearly combined using Equation 11, which shows the convergence of EHG ψ towards 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 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 Figure 6.

4.3.3. Topic Centrality

A node/word v_k^{ψ} connected with many positive edges in the EHG ψ shows that the word v_k^{ψ} is highly co-occurent with diverse set of words and making it important, hence topic centrality $TC(v_k^{\psi})$ expresses the central tendency of words in the EHG ψ. It signifies the theme of discussion in the Twitter stream at a certain time interval $i\tau$. It is calculated using Equation 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 v_k^ψ, ε^ψ, and $|V^\psi|$ represent a node, index edge vector, and the total number of nodes in the EHG ψ respectively. $\pi_1(\varepsilon_i^\psi)$ and $\pi_2(\varepsilon_i^\psi)$ are the indexes of the nodes connected to the edge ε_i^ψ. The centrality scores of all nodes, connected to at least one positive edge in the EHG ψ, are accumulated to calculate the aggregated centrality score $AC(T^\psi)$ using Equation 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:

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

To calculate aggregated centrality, the index edge vector is used. The index edge vector ε^ψ contains only those edges which have positive weights as shown in Figure 5. The edges with negative weights are dropped because of the initial assumption in the event detection method (see Section 4.4), as a result, it positively influences the centrality of newly emerging topics with respect to existing ones and as well as reducing the number of passes significantly that improves execution time. A higher aggregated centrality score depicts that the emerging topics are coherent and concurrently appear in text stream at a certain time interval $i\tau$.

4.4. Event Detection Method

In the following section, we present event detection method built on a feature set extracted from EHGs. The event detection method works on 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 Equations 8, 11, and

14 respectively and calculates the heartbeat score $HB(\psi)$ using Equation 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 among different words in the EHG ψ.

Heartbeat $HB(\psi)$ of an EHG ψ is the product of divergence factor $DF(\psi)$, trend probability $TP(\psi)$, and aggregated centrality $AC(T^{\psi})$.

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

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

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

- $TP(\psi) \leq 0$ shows that the words are losing their importance due to the decline in their popularity compared to earlier adjacent time interval, thus the EHG ψ is labeled as *Weak*. For example, if the weights of certain topics are decreasing at time $i\tau$ compared to $(i-1)\tau$ and there is no significant increase in the weights of other words, as a result, trend probability score would be negative which indicates the EHG ψ is not significant
- Otherwise, if the heartbeat score (as shown in Equation 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.

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

Here, θ is an adaptive measure that finds the threshold value in each sliding window $k\Delta t$ locally using Equation 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 Δt and τ, are the temporal coverage of each sliding window and super-document d_i^ρ respectively. $\mathcal{N}$ is 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 Equation 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 Equation 20.

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

Figure 6 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.

5. 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 Equation 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 Equation 22.

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

Algorithm 3: Event Detection Algorithm

```
Assign Binary-Class Labels
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 in the sliding window $k\Delta t$.

$AC \leftarrow List()$                          ►set of aggregated centrality
$weight \leftarrow 0$                           ►absolute weight of all nodes in a $G^h_m$
$DF \leftarrow List()$                          ►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 $G^h_m \in \mathcal{G}^{h(k\Delta t}$ do
    $i \leftarrow i + 1$
    $DF_{(i)} \leftarrow 0$
    $TP_{(i)} \leftarrow 0$
    $AC_{(i)} \leftarrow 0$
    $HB_{(i)} \leftarrow 0$
    for each vertex $v_n \in G^h_m$ do
        $DF_{(i)} \leftarrow DF_{(i)} + \vartheta(v_n)$                  ►Using Equation 8
        $weight \leftarrow weight + |\vartheta(v_n)|$
    end
    for each vertex $v_n \in G^h_m$ do
        $TP_{(i)} \leftarrow TP_{(i)} + \frac{\vartheta(v_n)}{weight}$   ►Using Equation 11
    end
    for each index $l \in T^\psi$ do
        $AC_{(i)} \leftarrow AC_{(i)} + TC(v^\psi_{T^\psi_l})$          ►Using Equation 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
```

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 $\mathcal{C}$
```

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 $\mathcal{L}$
```

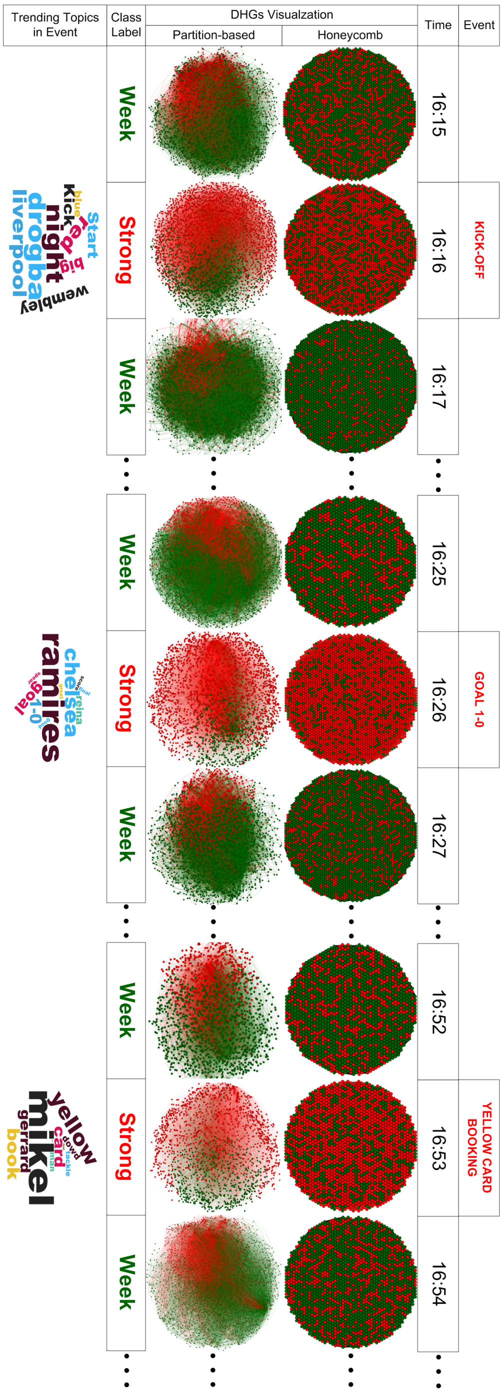


Figure 6: The graph visualization (Beehive and Partition-based) of three events (i.e., starting with the match, first goal, and a yellow card booking) at different time intervals of the FA Cup dataset. Red nodes are the words either new or appeared previously but gaining popularity at current time interval. 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.

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 Equation 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, as aforementioned. 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 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. In the worst case scenario $O(|V^{\psi}| + |E^{\psi}|) = O(|V^{\psi}| + N)$ if and only if there is a continuous increase in temporal frequency of all the words in the text stream. However, the occurrence of such scenarios is not possible due to the evolutionary pattern of real-world events (Iyengar et al., 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 Equation 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 4.1 and EHG in Section 4.2, $O(K(|V^{G_i}| + |E^{G_i}|))$ 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 $k\Delta t$.

6. Dataset Collection

We conducted experiments on three well-known benchmark datasets[3] (FA Cup, Super Tuesday, and US Election) that are crawled for the targeted data streams of events. The

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

three datasets used for evaluation were first collected and made available by Aiello et al. (2013). Many recent studies (Adedoyin-Olowe et al., 2016; Elbagoury et al., 2015; Ibrahim et al., 2017; Nguyen and Jung, 2017; Nur'aini et al., 2015; Papadopoulos et al., 2014; Prabandari and 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 in total. The majority of these topics were added to the ground truth by anticipating the announcements of US television regarding the outcomes of the presidential election.

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	124524
Super Tuesday	06 March 2012 17:00	07 March 2012 17:00	22	540241
US Election	06 November 2012 17:00	08 November 2012 05:00	64	2335105

7. Result and Discussion

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

7.1. 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:

- 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 also was 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 3 in detail. Individual documents lose their identity when using the existing methods for aggregation, as tweets are merged in to a single super-document such as in existing studies (Aiello et al., 2013; Adedoyin-Olowe et al., 2016). 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 example Figure 4.

7.2. 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 could 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; 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 4.3) 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. Figure 7 shows the comparison of top six keywords associated with different event-related topics using the Super Tuesday dataset from 06:50AM to 11:30AM against time interval of five minutes. It can be observed that the signals generated using EHG, shown in Figure 7(top), are sharper when compared to signals in Figure 7(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 then suppressed. As a result, at time 07:50, the keyword "paul" could 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 Figure 7(top) but are suppressed by the EHG after their early burst, as shown in Figure 7(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 user participation, graph size, and the heartbeat score. User participation is the total number of unique users who published at least one micro-document, graph size is the total number of unique words in the EHG ψ, and heartbeat score represents the intensity of the occurrence of an event at time $i\tau$ as shown in Figure 8. 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 Figure 8 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.

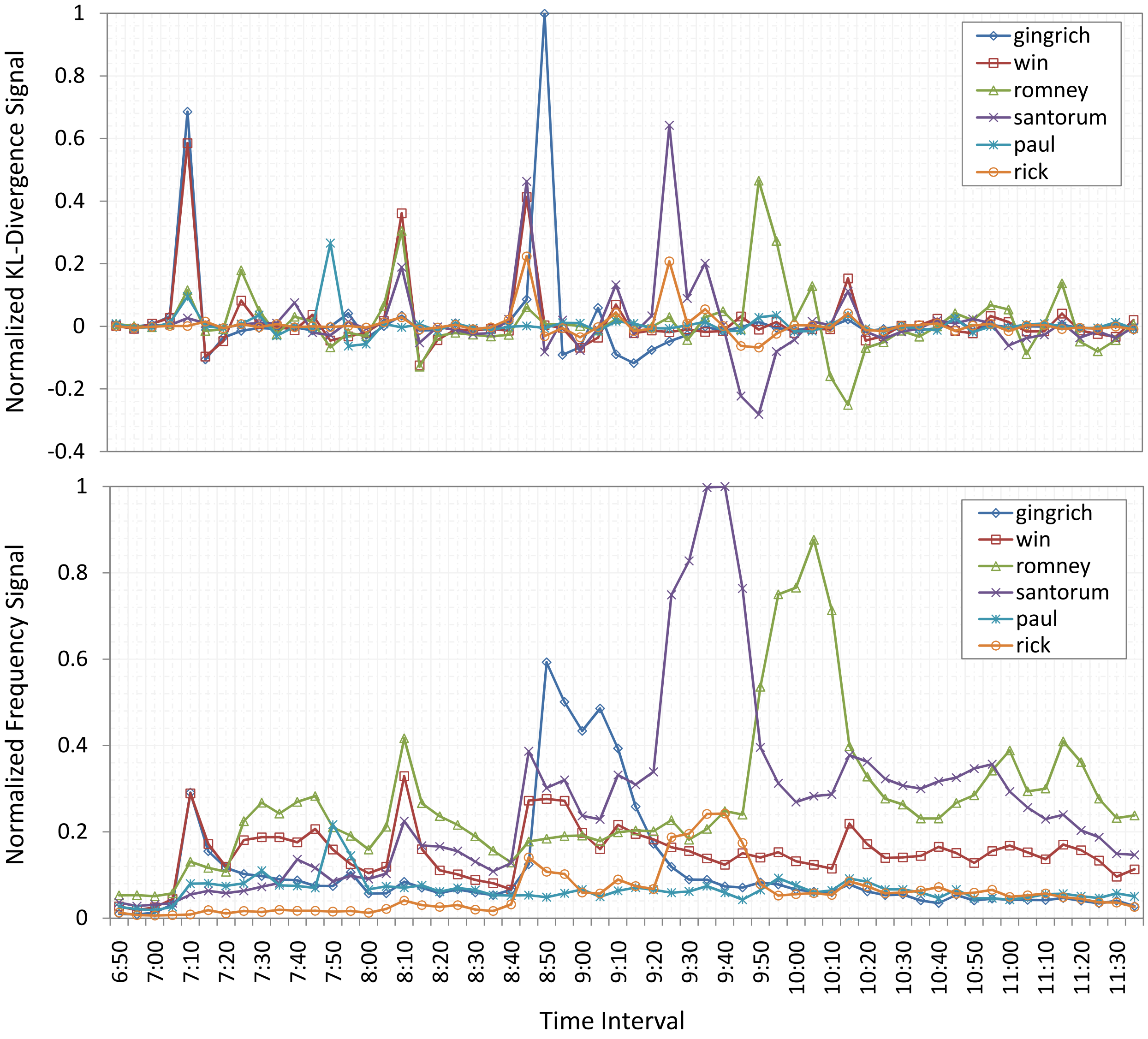


Figure 7: Comparison between term frequency and EHG-based signals. The signals are generated over a five minutes time interval against top six event-related keywords reported in the ground truth for the Super Tuesday dataset. **Figure (top)** represents the signals based on modified KL-divergence and **Figure (bottom)** represents the signals based on term frequency.

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

Case Study	**Time Interval**	**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, cup, congratulation, 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	win, #supertuesday, 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	romney, ohio, win, mitt, #supertuesday, primary, news, @mittromney, cnn	mitt, romney, @mittromney, ohio, ap, declare, primary	BREAKING: Romney wins primary in Ohio, a crucial Super Tuesday state

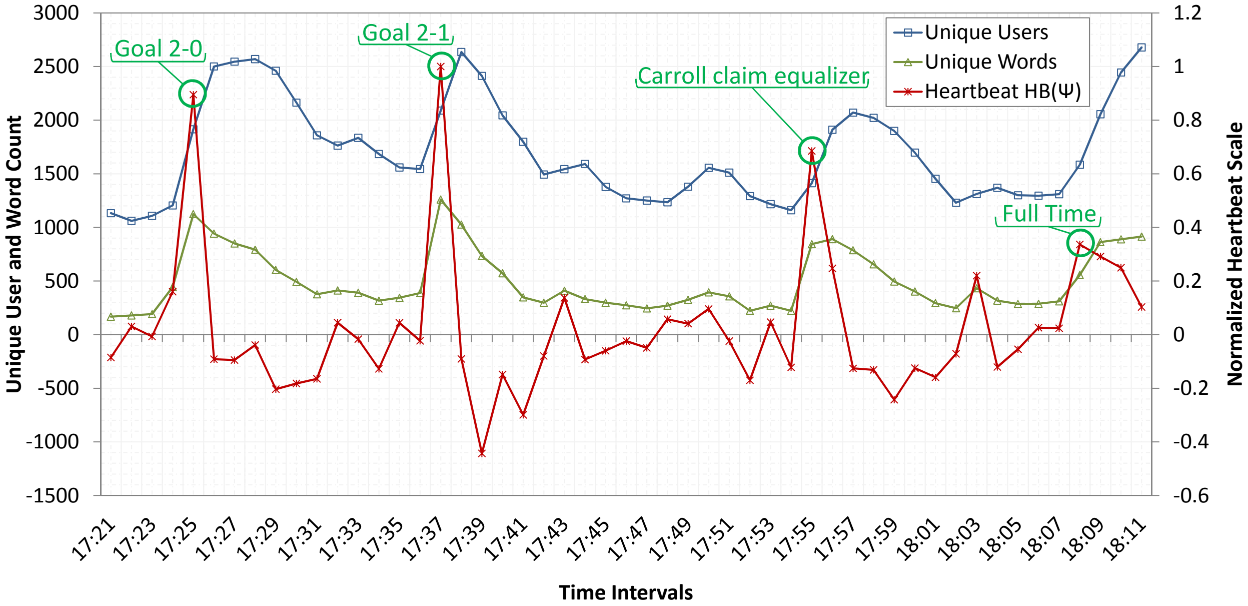


Figure 8: The heartbeat signal showing significant increase in heartbeat when an event occurs. The figure also shows the signals of unique words and users count across the text stream of the FA Cup 2012.

7.3. 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) that approximately generated 10 super-documents in each sliding window 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 would be labeled as *Strong* and Equation 19 works correctly in the detection model.

7.4. Results and Evaluation

We have extended the DHG approach (Saeed et al., 2018) by constructing a weighted DHG structure in which topic centrality feature is extracted using weighted edges. Another method namely 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 considered its weighted version namely 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 Figure 9.

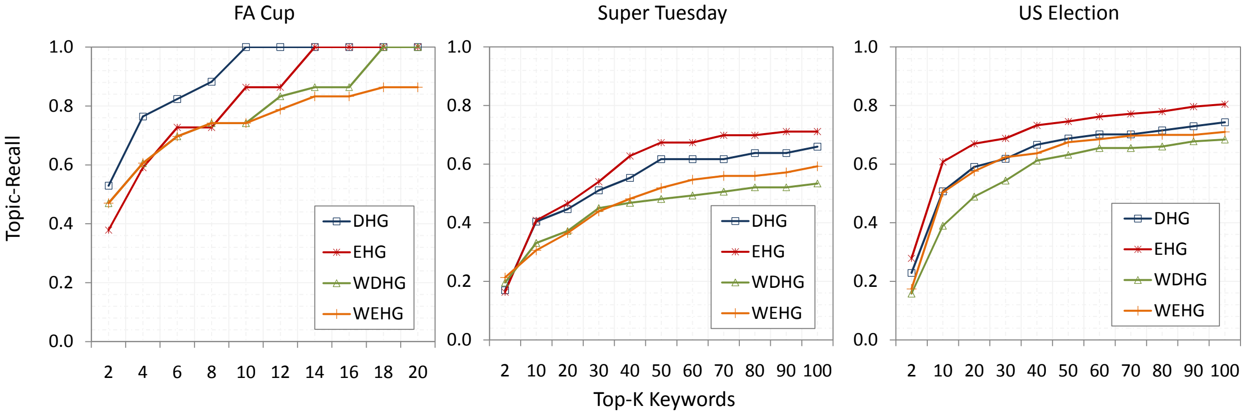


Figure 9: Performance comparison of four event detection methods for Topic-Recall against three benchmark datasets.

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) Probabilistic Models, 2) Clustering, 3) Frequent Pattern Mining, 4) Matrix Factorization, and 5) Exemplar-based (Ibrahim et al., 2017). We have considered the graph-based method as a separate category. Hence, the proposed approach is also compared with two existing graph-based methods 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 ten state-of-the-art approaches:

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

- Graph Feature-pivot (GFeat-p) (O'Connor et al., 2010), Dynamic Heartbeat Graph (DHG) (Saeed et al., 2018) - (*Graph-based*)

The ground truth is created based on the events reported in the mainstream media. We cannot use topic precision for the evaluation 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 micro-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 on FA Cup, because the users who published content on their micro-blogs are very focused, consistent, and to the point due to the popularity and limited time of this on-going event. Therefore, the topics appearing in the text stream are less diverse comparatively. 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 17:56 in the FA Cup dataset, the ground truth marks *andy, carroll* and *line* as mandatory keywords, whereas *header, cech, over, claim, equalize* are the optional keywords. Therefore, it is more likely that mandatory keywords are among the top trends but do not necessarily appear in the top-most position. Initially, the EHG approach has comparable T-Rec at $K = 2, 4, 6, ..., 12$ and achieves the maximum possible T-Rec at $K > 12$ for the FA Cup dataset.

Similarly, the EHG method outperforms the other detection methods after $K > 30$ for the Super Tuesday dataset. The results for T-Rec are 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 three datasets at $K = 2$, as shown in Table 7. Hence, the EHG is a robust detection approach in terms of performance and efficiency.

7.5. Limitations

The numerical 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 follows.

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.

Top-K Method	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 US Election dataset.

Top-K Method	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.

Top-K Method	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

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

Datasets Method	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

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 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 rather than one. Such cases are also less likely to occur, especially when the temporal coverage of sliding window is large.

8. Conclusion

In this paper, a novel Enhanced Heartbeat Graph (EHG)-based approach is developed that is efficient for text streams such as Twitter. We formulated the text stream as a series of temporal graphs that are further processed to generate heartbeats within each sliding window of fixed temporal coverage. Furthermore, we designed three unique features, divergence factor, trend probability, and topic centrality to identify emerging events using EHG. We evaluated the performance of EHG on three publicly available benchmark datasets (the FA Cup Final 2012, Super Tuesday 2012, and US Election 2012). The experimental results showed that the EHG approach is capable of dealing with dynamic nature of text streams and detected emerging events with improved precision and recall when compared to five state-of-the-art methods. The empirical evaluation showed that the EHG approach is robust in terms of computational complexity and scalability thus, it could be used for live streams as well. In future, we plan to automate parameter selection. We also plan to evaluate the proposed approach on a live stream.

Acknowledgment

The first author, Zafar Saeed is financially supported by Higher Education Commission (HEC) of Pakistan under International Research Support Initiative Program (IRSIP), grant PIN No. *IRSIP-35-Engg-06*. This research was partially conducted during Zafar Saeed's visit at the Advanced Analytics Institute, University of Technology, Sydney, Australia.

References

Adedoyin-Olowe, M., Gaber, M. M., Dancausa, C. M., Stahl, F., and 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., Goker, A., Kompatsiaris, I., and Jaimes, A. (2013). Sensing trending topics in twitter. *Multimedia, IEEE Transactions on*, 15(6):1268–1282.

Becker, H., Naaman, M., and Gravano, L. (2011). Beyond trending topics: Real-world event identification on twitter. In *International AAAI Conference on Web and Social Media*, pages 438–441, USA. AAAI.

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

Blei, D. M., Ng, A. Y., and 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*, pages 481–485, New York, NY, USA. ACM, ACM.

Chen, Y., Amiri, H., Li, Z., and 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*, pages 43–52, New York, NY, USA. ACM, ACM.

Cheng, T. and Wicks, T. (2014). Event detection using twitter: a spatio-temporal approach. *PloS one*, 9(6):e97807.

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

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

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

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

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

Gao, X., Cao, J., He, Q., and 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*, pages 305–308, New York, NY, USA. ACM, ACM.

He, Q., Chang, K., and 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*, pages 207–214. ACM.

Huang, J., Peng, M., and 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*, pages 3–10, New York, NY, USA. ACM, ACM.

Ibrahim, R., Elbagoury, A., Kamel, M. S., and Karray, F. (2017). Tools and approaches for topic detection from twitter streams: survey. *Knowledge and Information Systems*, pages 1–29.

Iyengar, A., Finin, T., and 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*, pages 186–191, USA. IEEE, IEEE.

Jarwar, M. A., Abbasi, R. A., Mushtaq, M., Maqbool, O., Aljohani, N. R., Daud, A., Alowibdi, J. S., Cano, J. R., García, S., and 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. and 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., and 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*, pages 1716–1725.

Katragadda, S., Virani, S., Benton, R., and Raghavan, V. (2016). Detection of event onset using twitter. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 1539–1546.

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., and 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*, pages 496–499. ACM.

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

Li, C., Weng, J., He, Q., Yao, Y., Datta, A., Sun, A., and 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*, pages 721–730. ACM.

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

Long, R., Wang, H., Chen, Y., Jin, O., and Yu, Y. (2011). Towards effective event detection, tracking and summarization on microblog data. In *International Conference on Web-Age Information Management*, pages 652–663. Springer.

Nguyen, D. T. and 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. and 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., and 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*, pages 123–128. IEEE.

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

Ozdikis, O., Senkul, P., and 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)*, pages 20–24. IEEE Computer Society.

Panagiotou, N., Katakis, I., and Gunopulos, D. (2016). Detecting events in online social networks: Definitions, trends and challenges. In *Solving Large Scale Learning Tasks. Challenges and Algorithms*, pages 42–84. Springer.

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

Petrović, S., Osborne, M., and 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*, pages 181–189. Association for Computational Linguistics.

Prabandari, R. and 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*, pages 030144–5. AIP Publishing.

Saeed, Z., Abbasi, R. A., Sadaf, A., Razzak, M. I., and 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*, pages 534–545.

Sethi, T. S. and 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., and 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*, pages 355–358. ACM.

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

Velampalli, S. and Eberle, W. (2017). Novel graph based anomaly detection using background knowledge. In *Proceedings of the Thirtieth International Florida Artificial Intelligence Research Society Conference*, pages 538–543.

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

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

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

Zhou, D., Chen, L., and He, Y. (2015). An unsupervised framework of exploring events on twitter: Filtering, extraction and categorization. In *Proceedings of 29th AAAI Conference on Artificial Intelligence*, pages 2468–2475, USA. AAAI.

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

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
d_i	i^{th} micro-document in the set $\mathcal{D}$
$\mathcal{W}$	Set of words such that $\mathcal{W} \subset W$
$\mathcal{D}^\rho$	Set of super-documents in text stream
d_i^ρ	i^{th} super-document in the set $\mathcal{D}^\rho$
Δt	Temporal coverage of sliding window
$k\Delta t$	k^{th} sliding window in text stream
$\mathcal{G}$	Set of graphs representing graph series
G_i	i^{th} graph in the set $\mathcal{G}$
V	Set of vertices in a graph represent words
v_k	k^{th} vertex in a graph
E	Set of edges in a graph represent co-occurent of words
e_k	k^{th} edge in a graph
$\mathscr{W}(v_k)$	Weight associated to k^{th} vertex in a graph
$\mathscr{S}(e_k)$	Weight associated to k^{th} edge in a graph
$\overline{\mathscr{W}}(v_k)$	Weight associated to k^{th} vertex in the EHG G_i^h
$\overline{\mathscr{S}}(e_k)$	Weight associated to k^{th} edge in the EHG G_i^h
$\mathcal{G}^{\langle}$	Set of heartbeat graphs representing EHG series
G_i^h / ψ	i^{th} EHG in the set $\mathcal{G}^{\langle}$
$\varepsilon^{G_i^h}$ / ε^{ψ}	index edge vector for the graph G_i^h
$\lvert d_i^\rho(v_k) \rvert$	number of micro-documents containing word v_k which represents the temporal frequency of v_k
$\lvert d_i^\rho(v_m, v_n) \rvert$	number of micro-documents containing words v_m and v_n which represents the temporal co-occurrence frequency of an edge e_k
$\mathcal{G}^{h(k\Delta t)}$	Set of all the EHGs temporally covered in k^{th} sliding window
$\vartheta(v_k^\psi)$	Normalized weight associated to k^{th} vertex in G_i^h
$\delta(e_k^\psi)$	Normalized weight associated to k^{th} edge in G_i^h
$i\tau$	Time interval starting at time instance t_i until $t_i + \tau$
$DF(\psi)$	Divergence factor score of G_i^h represents the intensity of popularity of existing and emerging topics at a certain time interval

V^{ψ}	Set of vertices in the graph G_i^h
$\lvert V^{\psi}\rvert$	Total number of vertices in G_i^h represents unique word count at a certain time interval
$\lvert V^{\psi+}\rvert$	Total number of vertices which have positive weights in G_i^h
$\lvert V^{\psi-}\rvert$	Total number of vertices which have negative weights in G_i^h
$\vartheta(v_k^{\psi+})$	Normalized weight associated to k^{th} vertex in G_i^h which has positive value
$\vartheta(v_k^{\psi-})$	Normalized weight associated to k^{th} vertex in G_i^h which has negative value
$P(\vartheta(v_k^{\psi+}))$	Probability of k^{th} vertex in G_i^h which has positive weight
$P(\vartheta(v_k^{\psi-}))$	Probability of k^{th} vertex in G_i^h which has negative weight
β_i	constants for linear combination for probability distribution of topics
$TP(\psi)$	Trend probability score of G_i^h represents the probability of occurrence of an event at time interval
$TC(v_k^{\psi})$	Normalized degree centrality score of k^{th} vertex in G_i^h represents the occurrence of a topic with diverse set of words
$\lvert\varepsilon^{\psi}\rvert$	Total number of edges in G_i^h which have positive weights
ε_i^{ψ}	i^{th} positive edge in G_i^h
$\pi_1(\varepsilon_i^{\psi})$ and $\pi_2(\varepsilon_i^{\psi})$	indexes of nodes attached to the edge ε_i^{ψ}
T^{ψ}	Set of nodes that are connected to positive edges in G_i^h
$\lvert T^{\psi}\rvert$	Total number of unique nodes connected to positive edges in G_i^h
T_k^{ψ}	k^{th} node in G_i^h connected to at least one positive edge
$AC(T^{\psi})$	Aggregated centrality score of G_i^h represents the emerging topics are coherent and concurrently appearing in text stream at a certain time interval
$Est(\psi)$	Classification function that assigns class labels to an EHG
$\theta_{(k\Delta t)}$	Threshold value for classification function $Est(\psi)$ for k^{th} sliding window
$HB(\psi)$	Heartbeat score of G_i^h
$\mathcal{N}$	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
$\mathcal{T}(\psi)$	Transformation function that maps pair of adjacent graphs onto an EHG