Event Detection in Twitter Stream using Weighted Dynamic Heartbeat Graph Approach

Zafar Saeed
Department of Computer Science, Quaid-i-Azam University, Islamabad, Pakistan

Rabeeh Ayaz Abbasi
Department of Computer Science, Quaid-i-Azam University, Islamabad, Pakistan

Muhammad Imran Razzak
Advanced Analytics Institute, University of Technology, Sydney, Australia

Guandong Xu
Advanced Analytics Institute, University of Technology, Sydney, Australia

Abstract

Tweets about everyday events are published on Twitter. Detecting such events is a challenging task due to the diverse and noisy contents of Twitter. In this paper, we propose a novel approach named Weighted Dynamic Heartbeat Graph (WDHG) to detect events from the Twitter stream. Once an event is detected in a Twitter stream, WDHG suppresses it in later stages, in order to detect new emerging events. This unique characteristic makes the proposed approach sensitive to capture emerging events efficiently. Experiments are performed on three real-life benchmark datasets: FA Cup Final 2012, Super Tuesday 2012, and the US Elections 2012. Results show considerable improvement over existing event detection methods in most cases.

1 Introduction

The proliferation of social media and blogging networks has resulted in the unprecedented growth of their users across the world. The social media stream such as Twitter provides real-time information daily about worldwide events and generates a huge amount of data, in terms of content and diversity. In recent years, there has been growing interest in detecting, analyzing and exploiting data generated by Twitter; however the analysis of the large datasets generated by Twitter is challenging due to its diverse and noisy content, and challenges associated analysis such as scalability, accuracy, and efficiency [1], [2], [3].

Searching for event information in Twitter using keywords is a naive way, where the keyword-based search returns relevant documents matched to queried keywords. However, this approach does not guarantee that retrieved documents will represent the event accurately. Moreover, keywords selected in searches may vary from user to user, and keywords that appear in a data stream can change as the data stream evolves e.g., “earthquake” to “aftershock” or “casualties/death”. Currently, when an event occurs, there are a few keywords that show bursty behavior (i.e., a sudden increase in usage) and the frequency of publishing micro-documents increases rapidly (Figure 1). One simple way to find event-related information is to use trending keywords suggested by Twitter. However, trending keywords may not necessarily identify all the topics required to describe an event. Thus, sophisticated techniques are required to detect events on Twitter.

Recent work on event detection heavily relies on bursty features [4], [5]. However, bursty features may not be able to detect events that are less popular. In this study, we develop a robust and sensitive graph-based approach. The proposed approach identifies significant changes in a text stream to detect events. This approach uses displaced temporal frequency, which is a change in frequency with respect to time, to characterize the event-related keywords and their dynamic relations.

1.1 Problem

The analysis of Twitter stream, which contains diverse and noisy content, requires that a number of challenges are addressed such as accuracy, efficiency, and scalability. Micro-documents that are published to report an event have a set of related keywords that can be used to detect an event and to identify a relevant description. To identify significant information patterns, many graph-based methods have been proposed to deal with real-life data [6], [7], [8], [9]; however many are inadequate for the analysis of complex, dynamic and non-stationary data. They focus on bursty features that are frequent, co-occurring, and biased toward highly weighted patterns. These methods are well-placed to detect event-related topics with high frequency but ignore the dominant nature of burstiness due to which smaller events in the data are overlooked.

Corresponding Author: Rabeeh Ayaz Abbasi (Email: rabbasi@qau.edu.pk)

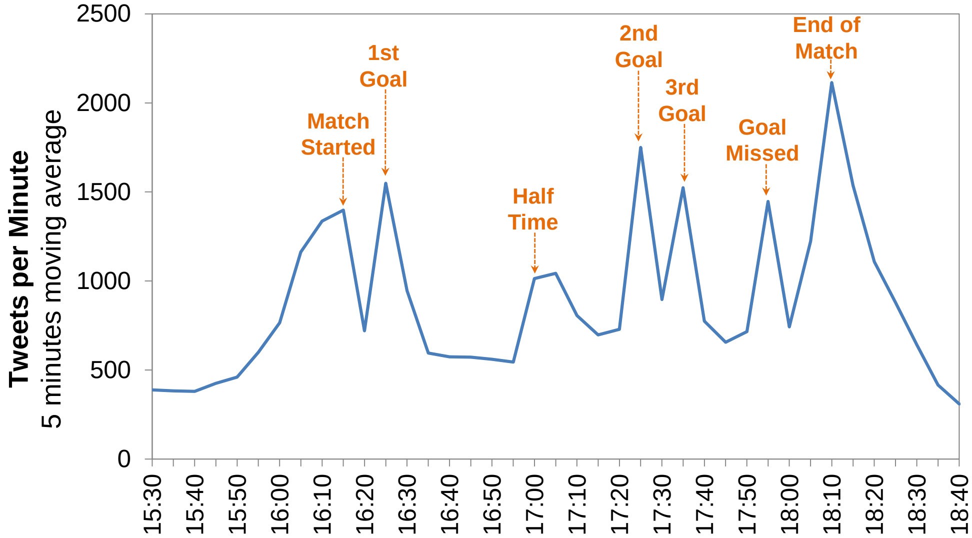


Fig. 1: Tweets traffic at the time of Final Match of Football Association Challenge Cup, 2012 (FA Cup) between 15:30 and 18:40.

1.2 Contributions

In this study, we propose an approach Weighted Dynamic Heartbeat Graph (WDHG) to detect events. Once an event is detected, WDHG approach suppresses the bursty keywords at subsequent time intervals. This enables other related information to be more visible and helps in capturing new and emerging events. The proposed approach is change-sensitive and detects event-related topics efficiently. It repeatedly captures the change-patterns in the time series data of Twitter streams and highlights key occurrences.

The key contributions of our study are, therefore:

- The proposed approach (WDHG) is change-sensitive, as it subsequently suppresses high-frequency events after their first detection.
- WDHG is computationally efficient. It detects events in polynomial time.
- Experiments on three real-life benchmark datasets: FA Cup, Super Tuesday and the US Election demonstrate that WDHG outperforms state-of-the-art methods.

2 Weighted Dynamic Heartbeat Graph (WDHG) Approach

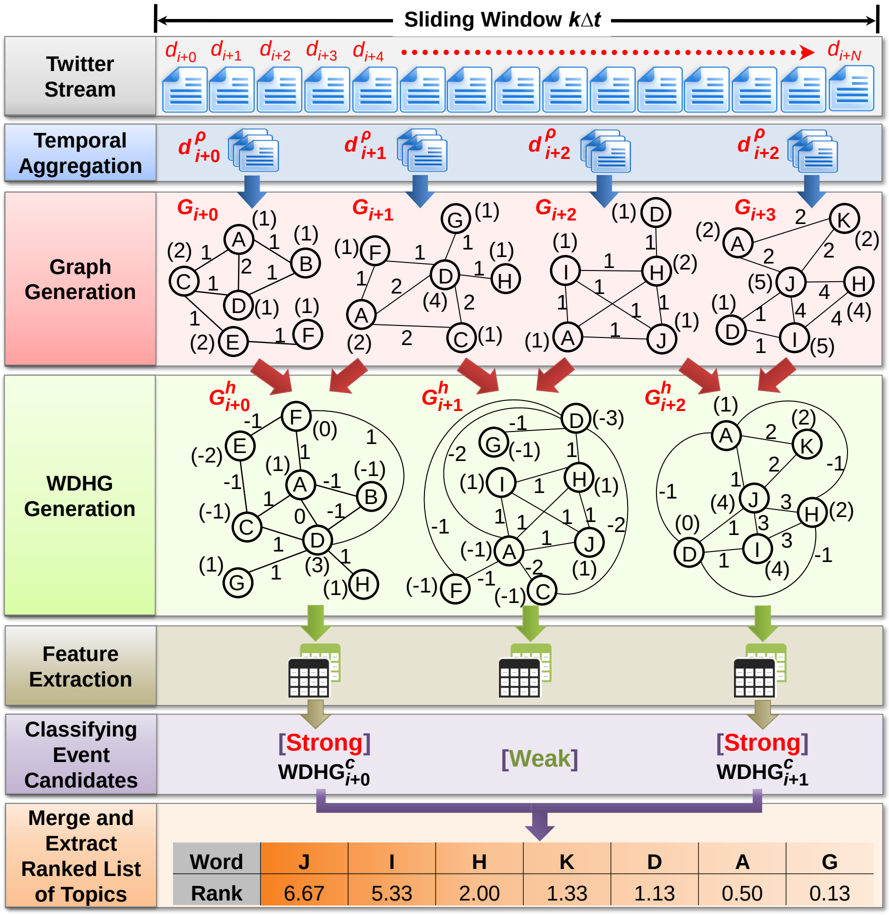
Figure 2 shows the workflow of the proposed approach and steps involved to transform the Twitter text stream into temporal graphs for extracting event-related topics. The data undergoes several transformations starting from the text stream as input. Preliminary concepts and definitions can be found here [10]. Micro-documents in the text stream are accumulated to generate super-documents. A series of graph inheriting word co-occurrence relationships from the micro-documents are generated using a set of super-documents. Furthermore, each adjacent pair of graphs is mapped onto a WDHG (see Subsection 2.2). Afterward, features are extracted (see Subsection 2.3). A rule-based classifier labels the candidate WDHGs for event representation. To classify WDHGs as event candidates, we use aggregated centrality as a key feature. Finally, all the candidate WDHGs are merged to extract event-related topics (see Subsection 2.4).

This study is an extension to our previous work [11]. Unlike our previous work, we construct a weighted graph structure and use three different approaches for the event detection model (see Subsection 2.3). We evaluate the performance of each detection model (see Subsection 3.4). We extend experiments with a larger benchmark dataset (i.e., the US Election) for the evaluation. We also perform a detailed execution time analysis to determine the efficiency of the proposed approach (see Subsection 3.5).

2.1 Assumptions

For detecting events, the proposed approach works with the following assumptions:

- There is a significant change in the temporal frequency of words, *or* new words appear between t_{i-1} and t_i time intervals.
- Significant words remain connected and result in a cohesive structure in the graph.


Sliding Window kΔt
Twitter Stream
d_{i+0} d_{i+1} d_{i+2} d_{i+3} d_{i+4} d_{i+N}
Temporal Aggregation
d^{ρ}_{i+0} d^{ρ}_{i+1} d^{ρ}_{i+2} d^{ρ}_{i+2}
Graph Generation
G_{i+0} G_{i+1} G_{i+2} G_{i+3}
WDHG Generation
G^{h}_{i+0} G^{h}_{i+1} G^{h}_{i+2}
Feature Extraction
Classifying Event Candidates
[Strong] WDHG^{c}_{i+0}
[Weak]
[Strong] WDHG^{c}_{i+1}
Merge and Extract Ranked List of Topics

Word	J	I	H	K	D	A	G
Rank	6.67	5.33	2.00	1.33	1.13	0.50	0.13

Fig. 2: Work-flow diagram illustrating data transformation and process starting from input as a text stream to output as a ranked list of event-related topics.

2.2 WDHG Series

Instead of trying to compute the features directly from the text stream, we developed a flexible approach by capturing the co-occurrence relationship among the words in the form of a graph series. We create a graph series G_i using the sliding window paradigm. Each node in G_i is a word. An edge represents the co-occurrence relationship between nodes. A graph series is a set of graphs $\mathcal{G} = \{G_1, G_2, G_3, \ldots, G_{|\mathcal{D}^{\rho}|}\}$, where each $G_i \in \mathcal{G}$ is generated for $d_i^{\rho} \in \mathcal{D}^{\rho}$ such that G_i is a labeled graph. Each graph G_i in the graph series is temporally aligned with the Twitter stream and possesses the coherence relationship among the words of each micro-document d_i. Clique among words of each micro-document is created to increase the central tendency of topics within the graph structure.

After transforming the text stream into a graph series, we create a weighted dynamic heartbeat graph series $\mathcal{G}^h$ by linearly combining and mapping every pair of adjacent graphs (G_{i-1} and G_i) from the graph series $\mathcal{G}$. A WDHG G_i^h is a difference graph formed by combining two subsequent graphs from $\mathcal{G}$. It discriminates new and existing topics.

Transformation to WDHG is not straightforward. Adjacent graphs differ in their structures. Canonical order of the nodes in the graphs is not identical. Graphs also differ in the number of nodes. To overcome this issue, we align the dimensions of adjacent graphs (G_{i-1} and G_i) by regenerating their adjacency matrices in equal dimensions and canonical order. The procedure of a WDHG generation is briefly described in Algorithm 1. Figure 3 illustrates an example to generate a WDHG from two adjacent graphs G_{i-1} and G_i. Node weights are shown in "()". Edge weights can be seen beside the edges.

Computational complexity improves significantly from $O(K|V|^4)$ to $O(K|V|^2)$ using above mentioned transformation method, where $K = |\mathcal{G}|$ is the number of graphs in the graph series $\mathcal{G}$.

One of the inherent characteristics of WDHG is to suppress and handle bias in the data caused by the burstiness of dominating topics. Each WDHG is formulated by estimating the displaced temporal frequency of nodes and edges between each pair of graphs G_{i-1} and G_i to highlight other event-related information, which might also be important but less frequent at the same time. Algorithm 1 calculates the change in the corresponding pair of graphs G_{i-1} and G_i. It generates a WDHG G_i^h by assigning new weights to all the nodes and edges. The graph series $\mathcal{G}$ is transformed into WDHG series $\mathcal{G}^h$ in a streaming fashion without affecting the temporal alignment of the data. Furthermore, these WDHGs are used

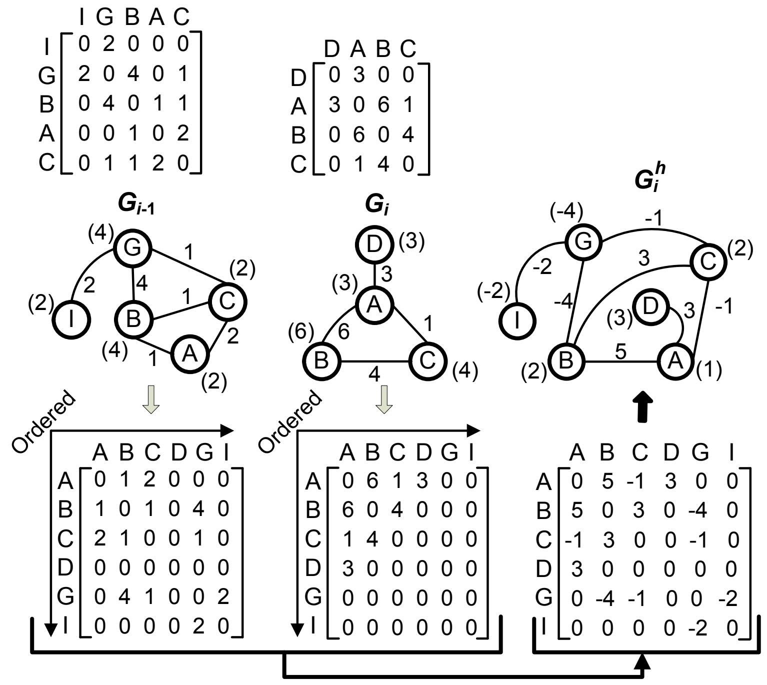


Fig. 3: The example illustrates the process to generate a WDGH.

Algorithm 1: Generate Set of WDHG

```
input  : $\mathcal{G} = \{G_1, G_2, G_3, \ldots, G_{|P|}\}$ a set of temporal graphs generated against a set of super-documents $D^\rho$
output: $\mathcal{G}^h = \{G_1^h, G_2^h, \ldots, G_{|\mathcal{G}|-1}^h\}$

foreach $G_i \in \mathcal{G}$ do
    merge set of nodes $V^{G_i}$ and $V^{G_{i+1}}$ by taking union $U$
    regenerate adjacency matrix $A$ for $G_i$ using $U$
    regenerate adjacency matrix $B$ for $G_{i+1}$ using $U$
    calculate difference graph $G_i^h$ using $A, B, V^{G_i}$, and $V^{G_{i+1}}$
end
```

to detect emerging events at precise time descriptions. The methodology of the event detection process is described in the next section.

2.3 Event Detection

The text stream has diverse contents and the heartbeat graph inherits the structural and co-occurrence relationship of a text stream. To understand our event detection model, let $\psi = G_i^h$ where G_i^h is *i-th* heartbeat graph for simplification. To detect the strong event candidates, we extract *growth factor* and *aggregated centrality* as key features from each WDHG and perform three experiments to compute corresponding heartbeat scores.

2.3.1 Growth Factor Based:

In the first experiment, we use growth factor $GF(\psi)$ to compute the heartbeat. $GF(\psi)$ is an accumulated score of node weights in a WDHG ψ which shows the intensity of drift in topics and their popularity in the text stream. It also shows how previously observed topics are trending, in terms of popularity and if new topics are emerging at the time interval t_i compared to t_{i-1} (1).

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

where $\vartheta(v_k^\psi)$ is the k^{th} node weight which represents the displaced temporal frequency of a word between G_i and G_{i-1}. Heartbeat score based on growth factor is calculated as shown in (2).

$$\mathscr{H}(\psi) = GF(\psi). \tag{2}$$

2.3.2 Aggregated Centrality Based:

Aggregated centrality represents the central tendency of different topics and their coherence in the WDHG ψ. We use aggregated topic centrality to compute the heartbeat score. Topic centrality $TC(v_k^{\psi})$ expresses the central tendency of words in each WDHG ψ which signifies the theme of discussion in the text stream at a certain time interval t_i. It is calculated using (3).

$$TC(v_k^{\psi}) = \frac{\sum_{i=1}^{|\varepsilon^{\psi}|} \pi_3(\varepsilon_i^{\psi})[\pi_1(\varepsilon_i^{\psi}) = k \vee \pi_2(\varepsilon_i^{\psi}) = k)]}{|V^{\psi}|}, \tag{3}$$

where v_k^{ψ}, ε^{ψ}, and $|V^{\psi}|$ represent a node, indexed edge vector, and the total number of nodes in the WDHG 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}, and $\pi_3(\varepsilon_i^{\psi})$ is the weight of the edge. The centrality scores of all nodes that contain at least one positive edge in the WDHG ψ are accumulated to calculate the aggregated centrality score $AC(T^{\psi})$ using (4) and (5), where T^{ψ} is a set of indexes of those nodes that are connected to at least one positive edge.

$$T^{\psi} = \bigcup_{i=1}^{|\varepsilon^{\psi}|} \left(\pi_1(\varepsilon_i^{\psi}) \cup \pi_2(\varepsilon_i^{\psi})\right), \tag{4}$$

$$AC(T^{\psi}) = \sum_{k=1}^{|T^{\psi}|} TC(v_{T_k^{\psi}}^{\psi}). \tag{5}$$

The indexed edge vector ε^{ψ} is used to calculate aggregated centrality. It contains only those edges that have positive weights. Due to the initial assumption (see Section 2) in the proposed approach, all the negative edges are dropped. It improves the centrality of newly emerging topics in the graph structure with respect to existing ones. It also reduces the number of passes significantly, thereby improving execution time. A high aggregated centrality score of a WDHG ψ shows that the keywords are coherent and emerging topics are concurrently reported in the Twitter stream at a certain time interval t_i. Heartbeat score based on aggregated centrality is calculated using (6).

$$\mathscr{H}(\psi) = AC(T^{\psi}). \tag{6}$$

2.3.3 Aggregated Centrality and Growth Factor Based:

We also combine aggregated centrality and growth factor by multiplying both features to calculate heartbeat score (7).

$$\mathscr{H}(\psi) = GF(\psi) \times AC(T^{\psi}). \tag{7}$$

2.3.4 Emerging Event Identification:

To detect event candidates, two labels *Strong* or *Weak* are assigned to each WDHG ψ. *"Strong"* means WDHG ψ contains emerging event descriptions and *"Weak"* means WDHG ψ is insignificant. A rule-based classification function $Est(\psi)$ (8) estimates and assigns class labels to each WDHG $\psi \in \mathcal{G}^h$.

$$Est(\psi) = \begin{cases} \text{“Strong”}, & \text{if } \mathscr{H}(\psi) \geq \theta_{(k\Delta t)}. \\ \text{“Weak”}, & \text{otherwise.} \end{cases} \tag{8}$$

Here, θ is a dynamic threshold computed based on heartbeat scores of WDHGs. It sets an optimum value for classification function $Est(\psi)$ (11) in the sliding window $k\Delta t$.

$$\mathcal{N} = \frac{\Delta t}{t_i}, \tag{9}$$

$$\varpi = \frac{\sum_{i=k}^{\mathcal{N}+k} (HB(\psi))}{\mathcal{N}}, \tag{10}$$

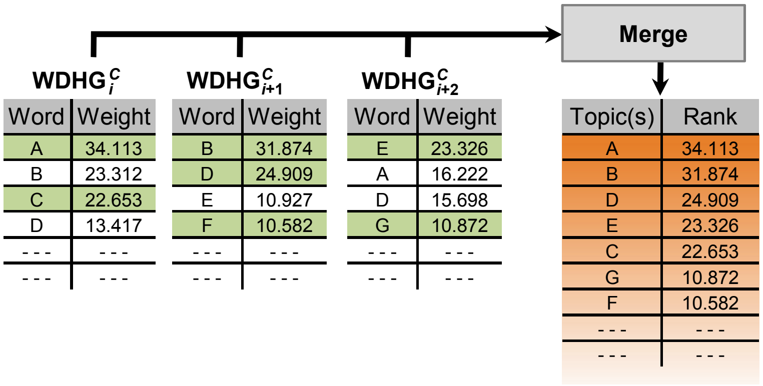
$$\theta_{(k\Delta t)} = \varpi + \omega \sqrt{\frac{\sum_{i=k}^{\mathcal{N}+k} (HB(\psi) - \varpi)^2}{\mathcal{N}}}, \tag{11}$$

where Δt and t_i are the temporal coverages of the sliding window and super-document d_i^{ρ}, respectively. $\mathcal{N}$ is the number of WDHGs in the sliding window (9). ϖ is the average heartbeat score within a sliding window (10). ω is the adjustment parameter that deals with the data dispersion. k is the index of the first WDHG in the sliding window under consideration, and $HB(\psi)$ is the heartbeat score of WDHG ψ.

2.4 Trending Topics

Multiple candidate WDHGs within a sliding window can have duplicate words. We select only unique words that have the highest weights among candidate graphs to extract the event-related topics without compromising performance. The weight for each word is calculated by fusing degree centrality and displaced temporal frequency scores. The example shown in Figure 4 further elaborates the process of merging multiple candidate WDHGs, where $WDHG_i^c$ is the *i-th* candidate graph in the sliding window under process. In the sliding window $k\Delta t$, a ranked list of topics is extracted from all candidate WDHGs and classified as *"Strong"*. Topic ranks are calculated using (12).

$$Rank(v_k^{\psi}) = \mathscr{C}(v_k^{\psi}) \times \mathscr{W}(v_k^{\psi}). \tag{12}$$



WDHG$_i^c$

Word	Weight
A	34.113
B	23.312
C	22.653
D	13.417
- - -	- - -
- - -	- - -

WDHG$_{i+1}^c$

Word	Weight
B	31.874
D	24.909
E	10.927
F	10.582
- - -	- - -
- - -	- - -

WDHG$_{i+2}^c$

Word	Weight
E	23.326
A	16.222
D	15.698
G	10.872
- - -	- - -
- - -	- - -

Topic(s)	Rank
A	34.113
B	31.874
D	24.909
E	23.326
C	22.653
G	10.872
F	10.582
- - -	- - -
- - -	- - -

Fig. 4: The example illustrates the unification of multiple candidate WDHGs in a sliding window to produce a final list of topics. Each copy that has the highest weight among its duplicates is selected for the final ranked list (as highlighted in the figure).

Figure 5 shows the visualization of three consecutive WDHGs with their class labels and the top ten keywords from an event (i.e., Goal) in the FA Cup dataset. The event detection model is described in Algorithm 2. It consists of feature extraction, event candidate detection, and finally extraction of an event-related ranked list of topics.

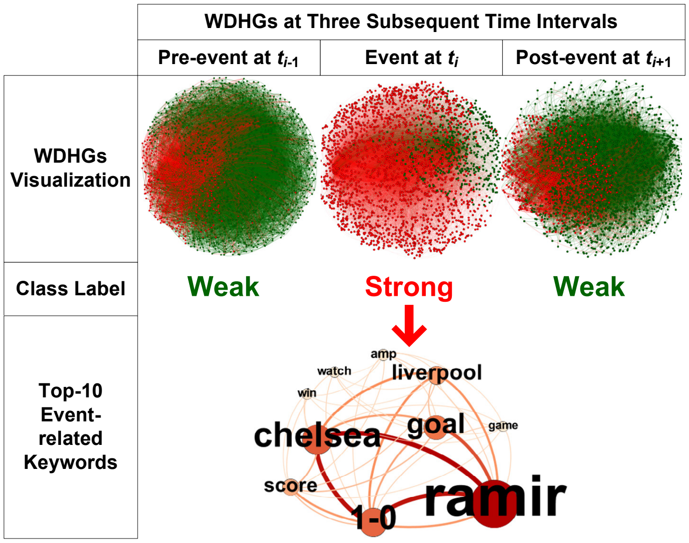


Fig. 5: Visualization of three WDHGs G_{i-1}^h, G_i^h, and G_{i+1}^h at time t_{i-1}, t_i, and t_{i+1}, respectively using the FA Cup dataset. A significant event "Goal" occurred at interval t_i. The nodes and edges with positive and negative weights are shown in red and green, respectively. A large portion of the graph is affected when an event occurs. It shows that WDHG is hyper-sensitive to detect newly emerging topics.

Algorithm 2: Event Detection Algorithm

```
Input  : $\mathcal{G}^{h(k\Delta t)}$– Set of WDHGs within the sliding window $k\Delta t$
Output: $\mathcal{L}$– List of ranked topics
foreach $G_i^h \in \mathcal{G}^{h(k\Delta t)}$ do
    calculate growth factor $GF_i$ using (1)
    calculate aggregated centrality $AC_i$ using (5)
    calculate heartbeat score $HB_i$ using (7)
end
calculate $\theta_{(k\Delta t)}$ using (11) for $\mathcal{G}^{h(k\Delta t)}$
foreach WDHG $G_i^h \in \mathcal{G}^{h(k\Delta t)}$ do
    assign binary class label $C$ to $G_i^h$ for the corresponding heartbeat score $HB_i$ using (8)
    if $C =$"Strong" then
        assign weight to each keyword in $G_i^h$ using (12)
        merge keywords from $G_i^h$ into the topic list $\mathcal{L}$
        remove duplicates from $\mathcal{L}$ while keeping keywords that have maximum weight
    end
end
sort $\mathcal{L}$ based on topic ranking
```

3 Experiments and Evaluation

We propose three event detection approaches. These approaches are based on the measures growth factor and topic centrality and their combination (see Subsection 2.3). To test and evaluate these three approaches, we first compare their performances with each other; then the best approach is selected and compared with nine state-of-the-art methods (see Subsection 3.4). We perform experiments on three benchmarks: FA Cup, Super Tuesday and the US Election datasets. An existing evaluation framework is used for comparative analysis [12].

3.1 Parameters

Three parameters (i.e., Δt, t_i, and ω) are used for optimization. Δt is the temporal coverage of the sliding window that contains a batch of data to be processed independently. As per ground-truth, the temporal coverage Δt of sliding window is set to one minute, one hour, and ten minutes for FA Cup, Super Tuesday, and the US Election datasets, respectively. The small size of the micro-document in the Twitter stream did not yield useful information; therefore, micro-documents are accumulated together that appear within a specific time span to form a super-document. The parameter t_i, a temporal coverage of super-documents, is set to one minute, ten minutes, and one minute for the FA Cup, Super Tuesday, and the US Election datasets, respectively. ω is the adjustment parameter that deals with the data dispersion. Its values of 1, 0.6, and 0.6 are used for FA Cup, Super Tuesday and the US election datasets, respectively.

3.2 Evaluation Measures

To evaluate the performance, we compare the results with the ground truth using two metrics:

- *Topic-Recall@K* (T-Recall): Fraction of topics detected successfully from the ground truth
- *Keyword-Precision@K* (K-Precision): Fraction of keywords detected successfully from the top-K retrieved keywords

As there are multiple topics in the benchmark datasets (see Table 1), the final evaluation metrics are calculated by micro-averaging the *T-Recall* and *K-Precision* of individual topics. The ground truth is created based on the events reported in the mainstream media. Topic precision cannot be used for the evaluation as the text stream contains several newsworthy events which are not included in the ground truth [12].

3.3 Dataset

The experiments are conducted on three well-known benchmarks: FA Cup, Super Tuesday, and the US Election datasets [12]. Many recent studies [13], [14], [15], [2], [16], [17], [18] use these benchmarks to evaluate the performance of their approaches. The details of the three datasets are given in Table 1.

TABLE 1: Datasets detail and temporal coverage

	Temporal Coverage	No. of Tweets	Total Topics
FA Cup	6 hours	124,524	13
Super Tuesday	24 hours	540,241	22
US Election	36 hours	2,335,105	64

The "FA Cup" dataset contains tweets posted during the final match of the Football Association Challenge Cup held on May 5, 2012. FA Cup is one of the oldest football competitions with a huge fan base. The match was played between the Chelsea and Liverpool teams. Chelsea won the Cup 2-1, Ramirez and Drogba scored each of the two goals from the winning team. The only goal from Liverpool was scored by Carrol. The ground truth for the FA Cup dataset comprises 13 topics, including kick-off, goals, half-time, fouls, bookings, and the end of the match.

The "Super Tuesday" dataset consists of tweets posted during the US primary elections, which were held on the first Tuesday of March 2012 in ten US states. The ground truth comprises 22 topics, which represents the key moments of the elections and projections of the voting results in different states.

The US Election dataset contains tweets posted during the United States presidential election of 2012 which was held on November 6, 2012. The ground truth consists of 64 topics. The topics were related to the outcomes of the presidential election, derived from mainstream media.

The ground-truth comprised several event-related topics as shown in Table 1. Each topic in the ground-truth is expressed by a set of keywords which are further divided into mandatory and optional categories. To detect a ground-truth topic, it is essential to detect all mandatory keywords; however optional words are more expressive in terms of event description and are used collectively in evaluating keyword precision along with mandatory keywords.

The datasets used in the study undergo number of pre-processing steps to improve their content quality. Duplicate tweets are removed. Tweet characters are converted to lower case and all the special characters are removed. Tokenization is performed to separate all words using white-spaces. Then, stop words and common words are removed. Words that consist of less than three letters are removed.

3.4 Results

A taxonomy of event detection techniques is proposed in a recent study [2]. Ibrahim et al. explored and classified event detection techniques into five major categories. 1) Probabilistic Models, 2) Clustering, 3) Frequent Pattern Mining, 4) Matrix Factorization, and 5) Exemplar-based. We consider some of the recent and state-of-the-art approaches from each of the categories to compare and evaluate the performance of the proposed approach. Since the proposed approach is graph-based, a Graph-feature pivot method is also included in the baselines.

We perform a two-step evaluation. First, we compare the performance of our three event detection methods (growth factor, aggregated centrality and a combination of growth factor with aggregated centrality) as shown in Figure 6. Results show that event detection based on aggregated centrality is better than growth factor or a combination of both.

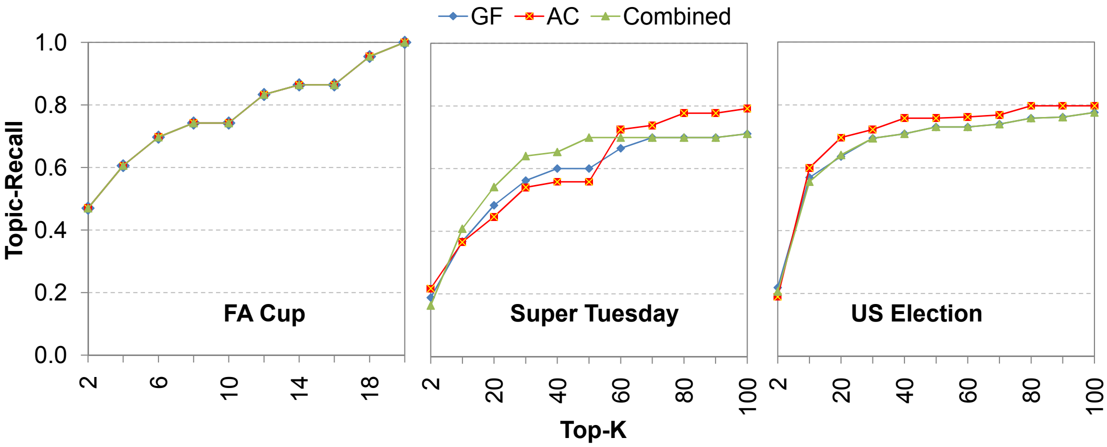


Fig. 6: Comparison of event detection methods Growth Factor (GF), Aggregated Centrality (AC) and Combined for FA Cup, Super Tuesday and the US Election datasets.

Second, we compare the performance of the winner approach (i.e., aggregated centrality) as the event detection method, and perform a comparative analysis with the following state-of-the-art methods:

- *Probabilistic Model* - Latent Dirichlet Allocation (LDA) [19]
- *Clustering* - Document-pivot (Doc-p) [20], BN-gram [12]
- *Frequent Pattern Mining* - Soft Frequent Pattern Mining (SFPM) [12]

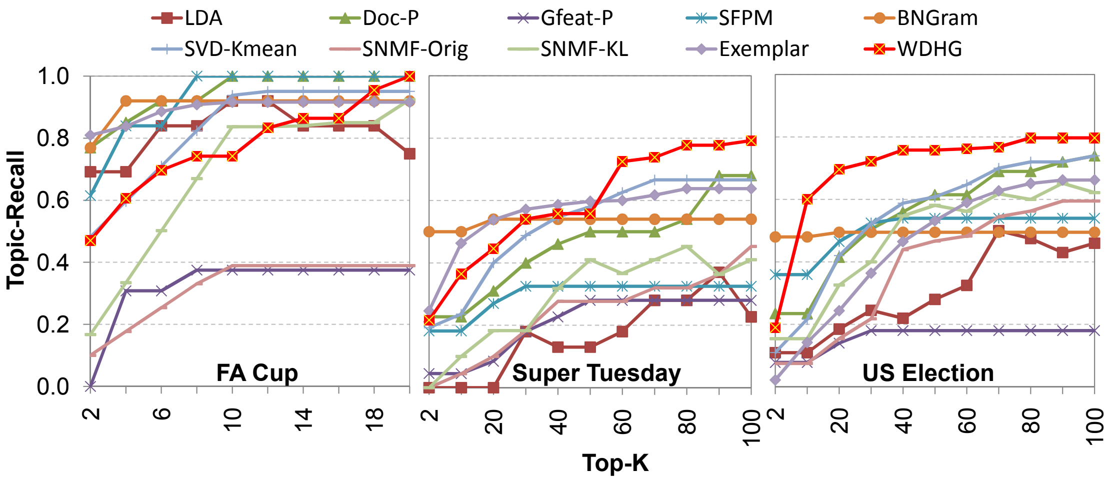


Fig. 7: Comparison with baseline methods for FA Cup, Super Tuesday and the US Election datasets

- *Matrix Factorization* - SVD-KMean [21], SNMF-Orig, SNMF-KL [18]
- *Exemplar-Based* - Exemplar [15]
- *Graph-based* - Graph-based Feature-pivot (GFeat-p) [22]

For the FA Cup dataset, Figure 7(left) shows the results for *T-Recall* at $K = \{2, 4, 6, \ldots, 20\}$. For all the approaches, the results for the FA Cup are the best among all the three datasets due to short duration and high popularity of the football match. Users posting tweets for such events are focused and consistent. Therefore, topics that appeared in the data are less diverse and easier to detect than in Super Tuesday and the US Election datasets. For smaller values of K, WDHG does not perform well; however, it quickly gains the maximum *T-Recall* and detects all the ground truth topics at $K = 20$ as shown in Figure 7(left).

Conversely, the Super Tuesday and the US Election datasets have more diversity in their text streams due to greater temporal coverage compared to the FA Cup. The fundamental characteristic of the proposed method is to sense change in the text stream and to extract coherent topics from the word-graph (see Section 2). Therefore, the proposed method outperforms all other baselines at $K > 50$ for the Super Tuesday dataset as shown in Figure 7(middle).

The US Election is one of the largest datasets in terms of temporal coverage which spans over 36 hours including sixty 64 topics. The proposed WDHG method significantly outperformed all the baseline methods at $K > 2$ as shown in Figure 7(right).

Table 2 shows the results of *K-Precision* at $K = 2$. The proposed WDHG method combines the scores of the displaced temporal frequency and the topic centrality in the word-graph; therefore it detects even-related keywords with high precision for all three datasets. The proposed WDHG method is an effective approach and performed consistently better compared to existing methods in terms of performance and accuracy.

TABLE 2: Comparison with baseline methods for Keyword-Precision@2 for FA Cup, Super Tuesday, and the US Election datasets. Best results for each dataset are in bold face.

Method	FA Cup	Super Tuesday	US Election
LDA	0.164	0.000	0.165
Doc-P	0.337	0.511	0.401
Gfeat-P	0.000	0.375	0.375
SFPM	0.233	0.471	0.241
BNGram	0.299	0.628	0.405
SVD-Kmean	0.242	0.367	0.300
SNMF-Orig	0.330	0.241	0.241
SNMF-KL	0.242	0.164	0.164
Exemplar	0.300	0.485	0.391
WDHG	**0.545**	**0.750**	**0.423**

We observe that whenever the text stream starts to change and deviate from its current trending topics, the proposed WDHG method quickly detects the emerging event due to its sensitivity towards the dynamic nature of the text stream. It also appears that user participation increases when an event occurs. The user participation feature is not used in the proposed WDHG method; however, it can be useful to improve the detection model further. We are planning to use it in future studies.

3.5 Execution Time Analysis

Twitter generates a great number of micro-documents within a short interval, and frequency of publishing micro-documents increases rapidly when a significant event occurs as shown in Figure 1. Therefore, we evaluate the execution time of event detection methods to analyze how quickly these methods detect event-related topics with top-k keywords. We conduct experiments on a machine having a CPU Intel Core i5-3210M (2.5 GHz) processor and 16GB-DDR3 memory. Usually, the primary objective of event detection methods is to produce results with high accuracy and then to achieve better computational time. Therefore, the methods SNMF-Orig, SNMF-KL, LDA and Gfeat-P with inferior performances (see Figure 7) are excluded from the execution time analysis.

Figure 8 shows the time required to process each event candidate sliding window by WDHG method. Figure 9 shows the average processing time required to produce the top 20, 100, and 100 keywords in the FA Cup, Super Tuesday, and the US Election datasets, respectively.

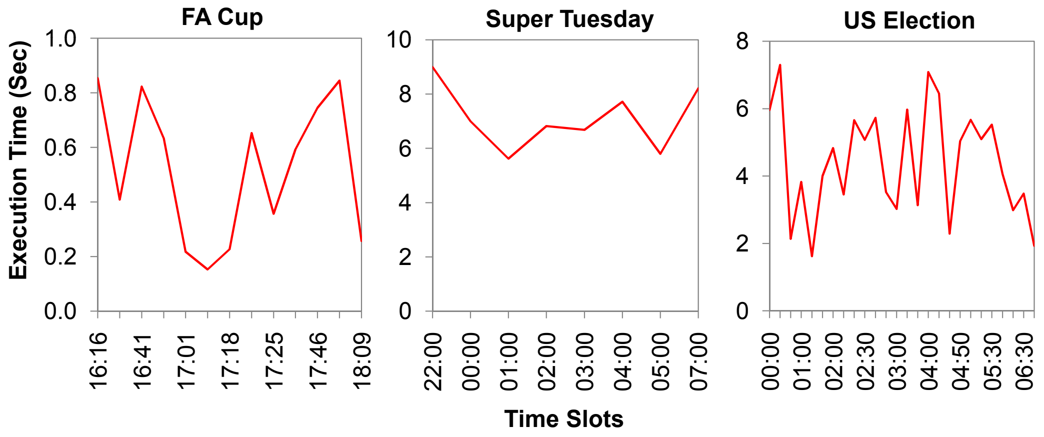


Fig. 8: Execution time of the proposed approach for event candidate sliding windows of FA Cup, Super Tuesday, and the US Election datasets.

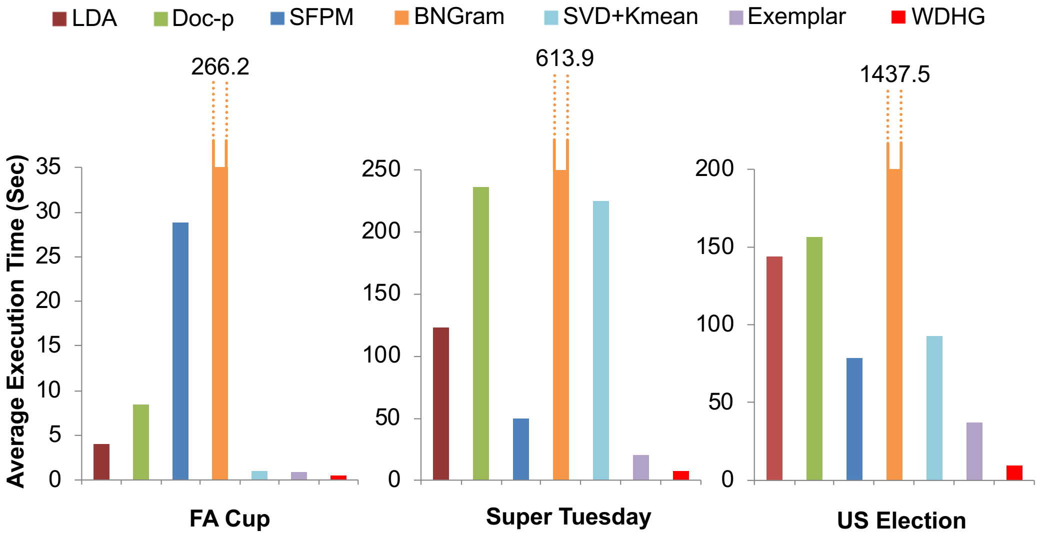


Fig. 9: Comparison of average execution time to produce top 20, 100, and 100 keywords for FA Cup, Super Tuesday, and the US Election datasets, respectively.

The average execution times (in seconds) for the proposed approach to detect event-related topics are 0.46, 7.24, and 9.52 for the FA Cup, Super Tuesday, and the US Election datasets, respectively, which is better than all of the baseline methods. BNGram has the worst execution times (266.25, 613.97, and 1437.50) for the FA Cup, Super Tuesday, and the US Election datasets, respectively. Exemplar has the second best execution times (0.88, 20.12, and 37.25) for the FA Cup, Super Tuesday, and the US Election datasets, respectively. Comparison of the execution times with Exemplar shows that the proposed approach is 47%, 64%, and 74% faster for the FA Cup, Super Tuesday, and the US Election datasets, respectively.

4 Related Work

Impressive efforts have been made to detect events in social streams. The detection of emerging events involves the identification of trending topics related to the event by monitoring and processing the text stream. Existing event detection techniques include bursty detection, topic model and clustering algorithms.

In order to identify significant keywords representing events, several research studies [23], [24], [25], [26], [27] have used frequency signals while processing the text stream. 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. Mathioudakis et al. identified the events based on a sharp increase in keyword frequency in specified time intervals [24]. However, detection based on the intensity of frequency could not distinguish different events that shared the same keywords in the bursty flow. To find abnormal spikes in keyword-based frequency signals, He et al. used the Discrete Fourier Transformation (DFT) method to group keywords based on features extracted from periodicity and strength of the power spectrum [28]. The method was extended by Weng and Lee who used wavelet analysis on word frequencies to obtain new features for every word. Based on low signal auto-correlations, trivial words were filtered out [29]. Events were identified by clustering the remaining words using graph partitioning. However, these methods [28], [29] are unable to keep track of the temporal information which is a significant aspect for detecting events.

Cheng et al. modeled the word co-occurrence patterns to learn topics that resulted in inference emerging topics [30]. Although the topic models worked well for short texts, they still required prior knowledge. Agarwal et al. discovered dense clusters in highly dynamic graphs by using a short-cycle property [31]. These dense clusters were considered social media events. In another work, a system named DYNDENS was developed which quantified the magnitude of change based on updates in edge weights. The system incrementally computed dense subgraphs to detect event stories [32]. DYNDENS is efficient and scalable to rapidly evolving datasets. Although the detection methods proposed by [31], [32] are efficient, despite rapid changes in microblog streams, they suffer from the loss of single-entity events.

Traditional event detection methods are not designed to process and detect events efficiently from such dynamic data, particularly when the data stream is noisy and consists of diverse events. In addition, most of the state-of-the-art approaches depend on highly weighted and frequent patterns to detect events [23], [25], [26], [27]. These approaches ignore the dominating nature of burstiness over small events in the data.

The proposed approach differs from existing approaches because it highlights dominating patterns at an early stage in the text stream and handles post-event effects by suppressing those patterns in the subsequent time interval, which provides an opportunity to discover new emerging events. Figure 5 visualizes the pre-event, event, post-event graphs to show the characteristics of the proposed approach. Instead of focusing on burstiness, we considered change in temporal frequency with respect to time which we named displaced temporal frequency. It captured the change in the frequencies of words appearing in text stream at an early stage and later suppressed their burstiness to highlight other topics. These characteristics are an inherent part of the proposed approach, which lead to a better performance in the event detection process.

5 Conclusion

In this paper, we presented a novel, sensitive and efficient Weighted Dynamic Heartbeat Graph (WDHG) method to detect events from a text stream. The text stream was systematically transformed into a series of temporal graphs. These graphs inherited temporal frequencies and co-occurrence relationships of the words appearing in the text stream. Each graph was further used to extract a heartbeat score using two features: growth factor and aggregated centrality. A rule-based classifier labeled the graphs as event candidates. Multiple event candidates were merged together to extract a list of ranked topics. For the performance evaluation of the proposed approach, three benchmarks: FA Cup, Super Tuesday, and the US Election were used. The quantitative evaluation showed that the proposed approach outperformed the state-of-the-art methods. The empirical evaluation showed that the proposed approach is computationally efficient and scalable. In the future, we plan to explore user participation and social network based features, as well as test the proposed approach on live text streams.

References

[1] P. S. Earle, D. C. Bowden, and M. Guy, "Twitter earthquake detection: earthquake monitoring in a social world," *Annals of Geophysics*, vol. 54, no. 6, pp. 708–715, 2012.

[2] R. Ibrahim, A. Elbagoury, M. S. Kamel, and F. Karray, "Tools and approaches for topic detection from Twitter streams: Survey," *Knowledge and Information Systems*, vol. 54, no. 3, pp. 511–539, 2018.

[3] M. A. Jarwar, R. A. Abbasi, M. Mushtaq, O. Maqbool, N. R. Aljohani, A. Daud, J. S. Alowibdi, J. R. Cano, S. García, and I. Chong, "Communiments: A framework for detecting community based sentiments for events," *International Journal on Semantic Web and Information Systems*, vol. 13, no. 2, pp. 87–108, 2017.

[4] F. Johansson, V. Jethava, D. Dubhashi, and C. Bhattacharyya, "Global graph kernels using geometric embeddings," in *Proceedings of the 31st International Conference on Machine Learning*, 2014, pp. 1–9.

[5] F. D. Johansson and D. Dubhashi, "Learning with similarity functions on graphs using matchings of geometric embeddings," in *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2015, pp. 467–476.

[6] E. Shabunina and G. Pasi, "A graph-based approach to ememes identification and tracking in social media streams," *Knowledge-Based Systems*, vol. 139, pp. 108–118, 2018.

[7] P. Yanardag and S. Vishwanathan, "Deep graph kernels," in *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2015, pp. 1365–1374.

[8] P. Yanardag and S. Vishwanathan, "A structural smoothing framework for robust graph comparison," in *Advances in Neural Information Processing Systems*, 2015, pp. 2134–2142.

[9] Y. Yao and L. B. Holder, "Detecting concept drift in classification over streaming graphs," in *KDD Workshop on Mining and Learning with Graphs (MLG)*, 2016, pp. 2134–2142.

[10] Z. Saeed, R. A. Abbasi, M. I. Razzak, and G. Xu, "Event detection in twitter stream using weighted dynamic heartbeat graph approach," *arXiv preprint arXiv:1902.08522*, 2019.

[11] Z. Saeed, R. A. Abbasi, A. Sadaf, M. I. Razzak, and G. Xu, "Text stream to temporal network - A dynamic Heartbeat graph to detect emerging events on Twitter," in *Proceedings of 22nd Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining*. Springer, 2018, pp. 534–545.

[12] L. M. Aiello, G. Petkos, C. Martin, D. Corney, S. Papadopoulos, R. Skraba, A. Göker, I. Kompatsiaris, and A. Jaimes, "Sensing trending topics in Twitter," *IEEE Transactions on Multimedia*, vol. 15, no. 6, pp. 1268–1282, 2013.

[13] M. Adedoyin-Olowe, M. M. Gaber, C. M. Dancausa, F. Stahl, and J. B. Gomes, "A rule dynamics approach to event detection in Twitter with its application to sports and politics," *Expert Systems with Applications*, vol. 55, pp. 351–360, 2016.

[14] H.-J. Choi and C. H. Park, "Emerging topic detection in Twitter stream based on high utility pattern mining," *Expert Systems with Applications*, vol. 115, pp. 27–36, 2019.

[15] A. Elbagoury, R. Ibrahim, A. K. Farahat, M. S. Kamel, and F. Karray, "Exemplar-based topic detection in Twitter streams," in *Proceedings of the Ninth International AAAI Conference on Web and Social Media*. AAAI Publications, 2015, pp. 610–613.

[16] D. T. Nguyen and J. E. Jung, "Real-time event detection for online behavioral analysis of big social data," *Future Generation Computer Systems*, vol. 66, pp. 137–145, 2017.

[17] S. Papadopoulos, D. Corney, and L. M. Aiello, "Snow 2014 data challenge: Assessing the performance of news topic detection methods in social media," in *SNOW-DC@ WWW*, 2014, pp. 1–8.

[18] R. Prabandari and H. Murfi, "Comparative study of original recover and recover KL in separable non-negative matrix factorization for topic detection in Twitter," in *AIP Conference Proceedings*, vol. 1862. AIP Publishing, 2017, p. 030144.

[19] Y. W. Teh, D. Newman, and M. Welling, "A collapsed variational bayesian inference algorithm for latent dirichlet allocation," in *Proceedings of the 19th International Conference on Neural Information Processing Systems*. MIT Press, 2007, pp. 1353–1360.

[20] S. Petrović, M. Osborne, and V. Lavrenko, "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*. Association for Computational Linguistics, 2010, pp. 181–189.

[21] K. Nur'aini, I. Najahaty, L. Hidayati, H. Murfi, and S. Nurrohmah, "Combination of singular value decomposition and k-means clustering methods for topic detection on Twitter," in *Proceedings of 7th International Conference on Advanced Computer Science and Information Systems*. IEEE, 2015, pp. 123–128.

[22] B. O'Connor, M. Krieger, and D. Ahn, "TweetMotif: Exploratory search and topic summarization for Twitter," in *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media*. AAAI Publications, 2010, pp. 384–385.

[23] R. Li, K. H. Lei, R. Khadiwala, and K. C.-C. Chang, "Tedas: A Twitter-based event detection and analysis system," in *Proceedings of 28th International Conference on Data Engineering*. IEEE, 2012, pp. 1273–1276.

[24] M. Mathioudakis and N. Koudas, "Twittermonitor: Trend detection over the Twitter stream," in *Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data*. ACM, 2010, pp. 1155–1158.

[25] D. T. Nguyen and J. J. Jung, "Real-time event detection on social data stream," *Mobile Networks and Applications*, vol. 20, no. 4, pp. 475–486, 2015.

[26] D. A. Shamma, L. Kennedy, and E. F. Churchill, "Peaks and persistence: Modeling the shape of microblog conversations," in *Proceedings of the 2011 ACM Conference on Computer Supported Cooperative Work*. ACM, 2011, pp. 355–358.

[27] J. Yang and J. Leskovec, "Patterns of temporal variation in online media," in *Proceedings of the fourth ACM International Conference on Web Search and Data Mining*. ACM, 2011, pp. 177–186.

[28] Q. He, K. Chang, and E.-P. Lim, "Analyzing feature trajectories for event detection," in *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2007, pp. 207–214.

[29] J. Weng and B.-S. Lee, "Event detection in Twitter," in *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*. AAAI Publications, 2011, pp. 401–408.

[30] X. Cheng, X. Yan, Y. Lan, and J. Guo, "BTM: Topic modeling over short texts," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 12, pp. 2928–2941, 2014.

[31] M. K. Agarwal, K. Ramamritham, and M. Bhide, "Real time discovery of dense clusters in highly dynamic graphs: Identifying real world events in highly dynamic environments," *in Proceedings of the VLDB Endowment*, vol. 5, no. 10, pp. 980–991, 2012.

[32] A. Angel, N. Sarkas, N. Koudas, and D. Srivastava, "Dense subgraph maintenance under streaming edge weight updates for real-time story identification," *in Proceedings of the VLDB Endowment*, vol. 5, no. 6, pp. 574–585, 2012.