Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 233 (2024) 481–491

Procedia Computer Science

www.elsevier.com/locate/procedia

5th International Conference on Innovative Data Communication Technologies and Application (ICIDCA 2024)

EDT_{BERT}: Event Detection and Tracking in Twitter using Graph Clustering and Pre-trained Language Model

Abhaya Kumar Pradhan[a,*], Hrushikesha Mohanty[a,b], Rajendra Prasad Lal[a]

[a]*Artificial Intelligence Lab, SCIS, University of Hyderabad, Hyderabad-500046, India*
[b]*CVR College of Engineering,Hyderabad-500029, India*
[1,a]*abhaya08csc007@gmail.com,* [2,a]*hmcs@uohyd.ac.in,* [2,b]*hmcs@cvr.ac.in,* [3,a]*rlal77@gmail.com*

Abstract

The identification of events from social media platforms such as Twitter (now known as X) is a hot research problem. It has applications in diverse domains such as journalism, marketing, public safety, crisis management and disaster response. The process includes the identification, monitoring, and analysis of events or incidents while they are being discussed or reported on Twitter. When it comes to identifying events from tweets (i.e. feeds from Twitter), many of the currently available event detection methods mainly rely on keyword burstiness features or structural changes in the network. However, due to the intricate characteristics of tweets and the ever-changing nature of events, they frequently fail to recognise noteworthy occurrences before they become trending. Moreover, these methods face difficulties when it comes to capturing the evolving characteristics of events with limited or insufficient contextual information. In this paper, we propose a window-based tweet-processing method called EDT_{BERT} for detecting events and tracking the evolution of events over time. Our proposed method utilizes the structural and semantic affinities that exist among words in tweets. The method starts by generating graph of tweets, where tweets are represented as nodes, and edges are the similarities between tweets. The method utilizes overlapping hashtags and named entities to capture the structural relationship between tweets. Additionally, a pre-trained sentence transformer model, specifically BERT, is employed to collect contextual knowledge and find semantically similar tweets. Next, the graph clustering technique is employed to identify optimized event clusters. Subsequently, our method generates chain of event clusters for each event to track the evolving variation of the event over time by utilising the ”Maximum-Weight Bipartite Graph Matching” (MWBGM) algorithm. The effectiveness of our approach is assessed using standard Tweet Datasets. Our evaluation demonstrates that our approach outperforms the baseline approaches.

Keywords: Event Detection; Twitter Events; Graph Clustering; BERT Sentence Embeddings; Language Models; MWBGM

* Corresponding author. Tel.: +91-739-657-9057.
E-mail address: abhaya08csc007@gmail.com

10.1016/j.procs.2024.03.238

1. Introduction

The proliferation of the internet and other kinds of digital technology has made it possible for individuals, regardless of their physical location, to connect and communicate with one another through various social media platforms. Globally, social media platforms attract a substantial number of users. Social network statisticians predict that by 2025, the number of people actively using social media will have increased from 2.86 billion in 2017 to 4.41 billion [11]. Twitter, a social media platform launched in March 2006 and now known as X, has amassed enormous popularity and userbase[1]. Twitter is a microblogging platform that enables users to publish short and concise messages known as tweets. Twitterites (i.e., Twitter users) can publish tweets up to 280 characters long. The active participation of users on Twitter has significantly altered its role, making it a primary means of discovering, analysing, and investigating events. Additionally, it serves as a valuable source for obtaining real-time updates on crucial global and local incidents. Nowadays, researchers and data analysts interested in user-generated content and comprehending topics of interactions (called Event Detection) have found Twitter to be a significant resource because of the platform's real-time information propagation and active user base. Event detection on Twitter has a wide range of applications across various domains, including Breaking news detection in journalism [9], Crisis Management and Disaster Response [15], Public Health Monitoring[4, 17], Social and Political Analysis[18], Sports and Marketing[1], Financial Markets, Investment and research etc.[10, 25, 14]

However, tweet processing poses unique challenges due to the volume, velocity, and diversity of data generated on the platform. The dynamic and instantaneous nature of tweets, along with the frequent changes in subjects of interest, pose notable obstacles in the process of event detection. The challenges in accurately capturing an event's descriptive words arise from linguistic variances and the subjective nature of opinions surrounding it. Moreover, it is essential to acknowledge that in the majority of languages, numerous words exhibit various interpretations based on the particular situation in which they are used. Extensive research efforts have been dedicated to (I) identification of bursty keywords as event characteristics [5, 2], (II) applying clustering techniques to group tweets that pertain to the same event [16, 21], and (III) utilizing modular decomposition and community detection algorithms to harness network structures in order to identify event-related phrases [7]. Nevertheless, the majority of research neglects to take into account the contextual information contained inside tweets while grouping tweets that exhibit a significant semantic association. To address the aforementioned concerns, we exploit the contextual knowledge and structural linkages of the tweets to improve the efficiency of the event detection task. To capture the contextual knowledge, a pre-trained BERT sentence embedding is employed, and the cosine similarity between the embeddings of tweets is used to capture the semantic similarity. Similarly, the clique-forming behaviour of the tweets related to the same event can be represented by the structural information based on the overlapping hashtags and named entities present within the tweets. The incorporation of contextual knowledge significantly improves the proposed method's capacity to recognize word associations characterized by substantial lexical and semantic coherence. These associations are commonly employed by Twitter users to describe specific events. In this paper, our main contributions are as follows:

1. **Novel Approach**: The proposed EDT_{BERT} is a window-based tweet processing model for detecting events by exploiting structural information and contextual representations of tweets. It creates a graph of tweets to model the structural and contextual relationships among tweets. It leverages the graph clustering method to find semantically similar tweet clusters related to an event. A bipartite graph is generated as part of the Event Tracking task in order to record the relationship that exists between two events that take place within consecutive time windows. The ”Maximum Weighted Bipartite Graph Matching Algorithm” is applied to find the matched event pairs to create a chain of events.(See section 3)
2. **Improved Performance**: A new similarity measure is proposed to find semantically similar tweets. It combines both contextual knowledge and structural association. A pre-trained BERT sentence transformer is applied to tweets to find contextual embeddings. Structural association between tweets is computed based on the overlapping hashtags and named entities present within the tweets. The graph representation of tweets based on contextual knowledge and structural association improves the quality of clusters, enhancing the proposed method's efficiency.(See section 3.2)

[1] https://en.wikipedia.org/wiki/Twitter

3. **Validation and Evaluation**: EDT_{BERT} is evaluated on standard Tweet Datasets available in the literature and outperformed baseline approaches, which shows the efficiency of the proposed method.(See section 4)

The rest of this paper is structured as follows: In Section 2, a concise overview of the relevant literature is presented. In Section 3, we introduce our proposed approach for Event Detection and Tracking, which we refer to as ”EDT_{BERT}”, Section 4 presents the evaluation methodology employed to assess the effectiveness of the proposed approach. Finally, the work is concluded in Section 5 by discussing potential avenues for future research.

2. Related Work

Even Detection gained attention from the research community in the year 2002. As part of the TDT (Topic Detection and Tracking) Research program, Allan et al. [3] extensively studied the event detection problem. The process of event identification in microblogs is theoretically quite comparable to the task of clustering. Like the TDT Project, the detection task analyses a continuous stream of time-ordered documents and groups these documents into the most appropriate event clusters. The crucial difference between TDT and microblogging mining is the document type and the stream’s volume. The most significant event detection techniques have been surveyed by [10, 20]. These techniques can be broadly categorized as identifying unspecified, predetermined, and specific events. The first category of techniques focuses on identifying generic events without any prior description provided [19, 21]. In the second category, researchers endeavour to identify predetermined events belonging to a specific category, such as sports, politics, earthquakes, crime and disaster events [25]. The third category pertains to the specification of event details to identify events that precisely correspond to an explicit description of the relevant event categories [5]. Our proposed method falls within the first category. Additionally, we elucidate the relevant literature that has inspired our work in this domain.

GraphClus[16]: Manaskasemsak et al. proposed a graph-based method for real-time event detection from Twitter. In their method, tweets are vectorized using tf-idf formula. A tweet graph is constructed, where nodes represent tweet vectors and edges between nodes represent the cosine similarity between two tweet vectors. Then, the well-known MCL algorithm is applied to identify events. Their method has a few sensitive parameters that require fine-tuning for different datasets and applications. Furthermore, the model fails to take into account the contextual information and social characteristics inherent in tweets.

BOWED[21]: Pradhan et al. proposed a bag-of-words-based approach for detecting events and aspects (sub-stories) of an event on Twitter. Their work proposes a three-phase-based incremental clustering algorithm for grouping similar tweets. Then, from output clusters, events and aspects are identified based on a heuristic which utilizes ”cluster quality”, ”tweeter participation”, and ”word commonality” features. The methodology employed fails to incorporate spatiotemporal information and contextual knowledge in the computation of tweet similarity.

RTED[7]: Fedoryszak et al. introduced a real-time framework for detecting clusters of named entities associated with events. The researchers employed a vectorization technique to identify named entities (NEs) within tweets. The construction of the entity graph involves the representation of named entities as nodes, and edges between two nodes is determined by the cosine similarity of their respective named entity vectors. The Louvain community discovery algorithm is applied to split the entity graph into clusters of events. It is important to note that this study’s focus was exclusively on the lexical representation of named entities, with no incorporation of semantic information.

ESBLA[23]: Singh et al. introduced a real-time event detection method that utilises clustering. A novel dynamic weighting scheme, the CTF-AIWF (”Conditional Term Frequency-Average Inverse Window Frequency”), is suggested to extract emergent keywords from tweets, drawing inspiration from TF-IDF. Then, to group similar event keywords, a novel clustering algorithm called Edge significance-based Louvain Algorithm (ESBLA) is introduced. The approach exclusively considers the lexical similarity among tweets, disregarding semantic relationships.

EnrichEvent[6]: Esfahani et al. presented a framework for real-time event identification. This framework utilizes co-occurrence characteristics and ParsBert sentence embeddings to capture contextual knowledge of named entities. The hierarchical DBSCAN algorithm is utilized for the purpose of identifying semantically correlated tweets during their first emergence, with the objective of event detection. The tweet classification module inside the proposed architecture, responsible for identifying event-related tweets, incurs a significant computational cost. Additionally, the methodology employed fails to take into account the location information associated with the tweets. We have

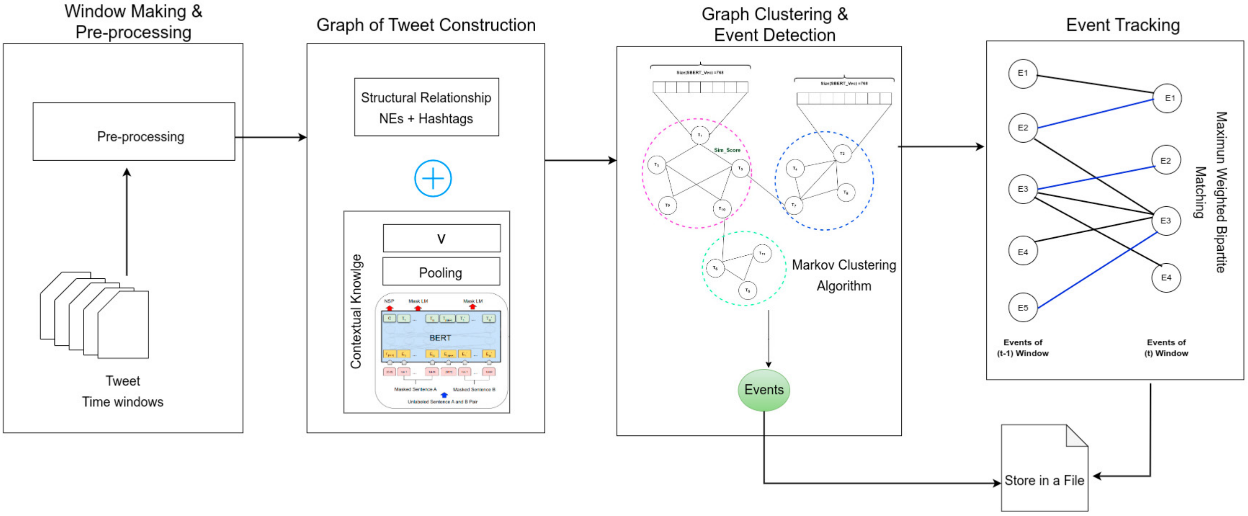


Fig. 1: Workflow diagram of the Proposed Method

conducted a comparative analysis of our proposed method with all the aforementioned baseline methodologies, as outlined in Section 4.

3. Event Detection & Tracking Method

Emerging tweets at various time-frames provide narratives of events. The issue at hand is locating a description of an event in the form of a collection of words extracted from various tweets. The word selection process is designed to enable the words to form groups based on their inherent affinity. A method for computing affinity based on which words form an association representing an event is proposed in this paper.

In this study, word affinity has been determined by examining temporal correlation, structural characteristics, and semantic similarity. To establish temporal coherence, the received tweets are divided into several time intervals according to their chronological order. Tweets within a specified time window are analyzed in order to detect events. The Event Detection Task starts with constructing a graph of tweets to model the structural and contextual relationships among tweets. Specifically, the BERT sentence encoder is utilized to find tweet embeddings and overlapping named entities, and hashtags are used to find structural associations among tweets. Then, the MCL graph clustering algorithm is employed to find semantically similar tweet clusters related to an event. For the Event Tracking task, the relationship between two events that take place within consecutive time frames is represented by a bipartite graph. The ”Maximum Weighted Bipartite Graph Matching” algorithm (MWBGM) is applied to find the matched event pairs to create chain of events. The experimental analysis indicates that our proposed method is more effective in capturing event descriptions improving the accuracy of event identification tasks than the baseline methods. In addition to event identification, our method tracks the evolution of detected events over time. The proposed method is referred to as ”EDT_{BERT}” and is schematically presented in Figure 1. In this work, the proposed method comprises four modules, as shown in Figure 1: Window Making and Preprocessing module, Graph of Tweet Construction module, Graph Clustering and Event Detection module, and Event Tracking module. The subsequent subsections provide a comprehensive description of each module.

3.1. Window Making & Preprocessing

Twitter API is used to collect tweets. Specifically, Tweepy[2], an open-source Python API, is used to gather tweets by passing unique tweet ids from the standard datasets. Then, based on the chronological order of generation, our

[2] https://docs.tweepy.org/en/stable/

window-making module divides tweets into fixed-sized time windows. As was previously mentioned, tweets are short and frequently contain noise. In order to improve the quality of input to our proposed method and the performance of subsequent phases, the Pre-processing module is specifically engineered to eliminate common terms that barely impart information about events. We have applied basic pre-processing steps such as removing stopwords and punctuations, web links, non-ASCII characters, retweets, and tweets with words less than 3, and then making the tweet to small case letters using the NLTK toolkit[3].

3.2. Graph of Tweet Construction

The Graph Construction module utilizes processed tweets and modelled them as a graph of tweets, in which the nodes represent tweets, and edges represent the similarity between tweets. The similarity computation plays a vital role in clustering event-related tweets. A novel similarity measure is proposed to find the structural association and contextual relationship between tweets. Events are clusters of tweets, i.e. event-representative terms that effectively explain various elements of an event, including the nature of the event and its specific occurrences. When and where did it occur? Furthermore, it is vital to ascertain the individuals actively participating in the event.

3.2.1. Structural Relationship

The structural information derived from overlapping hashtags and named entities (NEs) inside tweets indicates the establishment of cliques, where tweets are interconnected because of their association with a common event. In order to ascertain the structural association between nodes (i.e. tweets) in the graph of tweets, it is necessary to calculate the degree of overlap in hashtags and named entities between tweets using the Jaccard Similarity measure. The metric used to quantify the degree of structural similarity between two tweets, designated as ”t_i” and ”t_j”, is represented as O_{HT}. This metric is based on the degree of overlap in the hashtags used in the tweets. It is determined using eq-1, where HT(t_i) represents the set of hashtags present in the tweet t_i, and HT(t_j) represents the set of hashtags present in the tweet t_j. Similarly, the measure of structural similarity, which is based on the presence of overlapping named entities, is denoted as O_{NE}. It is computed using eq-2, where NE(t_i) represents the set of NEs present in the tweet t_i, and NE(t_j) represents the set of NEs present in the tweet t_j. The combined Structural Relationship (SR) between tweet t_i and t_j is computed using eq-3.

$$O_{HT}(t_i, t_j) = \frac{|HT(t_i) \cap HT(t_j)|}{|HT(t_i) \cup HT(t_j)|} \quad (1)$$

$$O_{NE}(t_i, t_j) = \frac{|NE(t_i) \cap NE(t_j)|}{|NE(t_i) \cup NE(t_j)|} \quad (2)$$

$$SR(t_i, t_j) = \alpha * O_{HT}(t_i, t_j) + \beta * O_{NE}(t_i, t_j) \quad (3)$$

3.2.2. Contextual Knowledge

In order to capture contextual knowledge, it is essential to convert textual input into vectors, which can then be employed for calculating the similarity between two tweets. Several embedding techniques can be employed for this particular approach, including One-Hot Encoding and Term Frequency Inverse Document Frequency (TFIDF). Not all techniques have the capacity to capture the contextual nuances that exist between tweets adequately. A pre-trained BERT sentence transformer [22] was utilised to construct the encoding of pre-processed tweets. In a given time window, the collection of preprocessed tweets can be denoted as $t_1, t_2, \ldots, t_p$. Each preprocessed tweet, ”t_i” is passed to the pre-trained BERT Sentence Encoder for the computation of token-level hidden representation. We can represent it as:

$$[B_{i,0}; \ldots; B_{i,k}; \ldots; B_{i,l}] = Sentence\_encoder(t_i), where \;\; B_{i,k} \in \mathbb{R}^{len(t_i) \times d} \quad (4)$$

In the above equation, ”l” represents the number of hidden layers such that $0 \leq k \leq l$, ”d” represents the size of the hidden representation, and the length of the tokenized sentence is represented by ”$len(t_i)$”. Then, the pooling function

[3] https://www.nltk.org/

p is applied to $B_{i,k}$ for deriving diverse sentence level views $b_{i,k} \in \mathbb{R}^d$ from all layers(which means it computes $b_{i,k} = p(B_{i,k})$). Finally, the sampling function $\sigma : R^l = \{b_{i,k} \mid 0 \leq k \leq l \mid\}$ is applied to find the embedding vector. The BERT sentence transformer uses the mean pooling function to compute the output layers' mean. Each preprocessed tweet is passed to the BERT embedding module, generating a 768-dimensional dense embedding vector. Next, the calculation of contextual similarity between tweet vectors is performed using the cosine similarity measure, as presented in eq-5. In the graph of tweets, each node (tweet) is represented with a BERT embedding vector of 768 dimensions, with edges connecting two tweets if their Semantic Similarity (SSIM) exceeds a predetermined threshold (in the experiment, the threshold is set to 0.3). The Structural Relation (SR) and Contextual Similarity (CS) are given equal weight while calculating semantic similarity using eq-6.

$$CS(V_{t_i}, V_{t_j}) = \frac{\sum_{k=1}^{768}(V_{t_{ik}} \times V_{t_{jk}})}{\sqrt{\sum_{k=1}^{768} V_{t_{ik}}^2} \times \sqrt{\sum_{k=1}^{768} V_{t_{jk}}^2}} \quad (5)$$

$$SSIM(t_i, t_j) = 0.5 \times SR(t_i, t_j) + 0.5 \times CS(V_{t_i}, V_{t_j}) \quad (6)$$

3.3. Graph Clustering & Event Detection

Using the structural relationship and contextual knowledge, our Graph of Tweet Construction module generates graphs from preprocessed tweets. And the tweet graph is passed to the next module for the event detection task. Fortunately, a wide range of network clustering and community identification algorithms are available to assist in accomplishing our objective. This study uses the Markov clustering (MCL) method [24] to analyse the graph representation of tweets, aiming to detect clusters of related tweets. The MCL algorithm has the capability to split the graph into clusters without requiring the number of clusters to be specified as an input parameter. The methodology employed in this technique is based on the idea that clusters are present inside a given graph. It utilizes a random walk approach to aid the process of clustering.

The MCL method has a tendency to generate numerous clusters that are modest in size and widely scattered, which produce events clusters that do not mean anything. In order to mitigate this issue, we propose a scoring mechanism to filter out these meaningless events effectively. We hypothesise that an event will be significant due to the fact that it is a concurrent topic and has been the subject of extensive community discourse. Event clusters are selected by employing a scoring measure known as the "User Diversity Score," which is based on Shannon's entropy. This measure was introduced in the work by Kumar et al. [13]. For a tweet cluster c, its' User Diversity score H(c) is defined as:

$$H(c) = -\sum_i \frac{N_{u_i}}{N} \log \frac{N_{u_i}}{N} \quad (7)$$

Where "u_i" represents the "i^{th}" user who has made contributions to the cluster "c". "N_{u_i}" denotes the count of tweets posted by the user "u_i" within cluster "c", while "N" represents the overall number of tweets present in cluster "c". Next, a set of most frequent terms (i.e. keywords) are extracted from event clusters to represent events.

3.4. Event Tracking

Identifying significant tweet clusters for each time window allows us to uncover various events. However, it is essential to remember that even while events may take place in different periods, they may still be interconnected. In this part, we want to analyse and monitor the developments occurring across several temporal contexts in relation to the events under consideration. In this context, the event tracking task is conceptualised and framed as a problem of bipartite graph matching. A bipartite graph is constructed to represent the event clusters occurring in two consecutive time windows. If two clusters of events are connected to one another to form an event chain, then an edge will be assigned between the two clusters of events. Formally, a bipartite graph G can be represented as $(V_1 \cup V_2, E)$, where V_1 and V_2 denote two sets of event clusters in consecutive time windows (vertices of the bipartite graph), and the set of edges, E = $\{(v_i, v_j) \mid v_i \in V_1, v_j \in V_2\}$ in which each (v_i, v_j) represents two events which are related to each other, and should be linked together. Furthermore, the bipartite graph G is extended to G' by assigning edge weights. The weight of an edge indicated as $w_{i,j}$, represents the probability of two event clusters being connected, where v_i belongs to the

```
Algorithm 1: Event_Detection_Tracking($W$)
  Input: $W_t$: Tweet time windows, $t = 0, \dots, n$, $W_i$ = Set of tweets in each time-window ($W_i = t_1, t_2, \dots, t_p$),
         $W_{cur}$: Set of tweets in current time-window, $W_{prev}$: Set of tweets in previous time-window
  Output: E: Set of Events, CE: Chain of Events
  begin
      for $W_0$ to $W_n$ do
          foreach $t_i \in W_{cur}$ do
              $t'_i \longleftarrow Prepocess(t_i)$; // Apply standard pre-processing method on tweet
              $V_{t'_i} \longleftarrow Sentence\_BERT\_Embedding(t'_i)$; // Find BERT embedding vector
              $W'_{cur} \longleftarrow W'_{cur} \cup t'_i$; // Set of pre-processed tweets
          $G_{cur} \longleftarrow Graph\_of\_tweets\_Construction(W'_{cur})$ ; // Construct weighted undirected graph
           from tweets, where Edge weights are computed using eq-6
          $CLUS \longleftarrow Markov\_Clustering(G_{cur})$ ; // Extract tweet clusters from Graph-of-tweets
          foreach $c \in CLUS$ do
              if $H(c) \geq \delta$ then
                  $E_{cur} \longleftarrow E_{cur} \cup c$ ; // Event cluster is added to the E using eq-7
          $E_{prev} \longleftarrow E_{prev} \cup c$ ; // Event clusters of previous time-window
          $CE \longleftarrow Max\_Weighted\_Bipartite\_Matching(E_{prev}, E_{cur})$; // Creating Chain of Events using
           Hungarian method
      return $E$, $CE$;
```

set V_1 and v_j belongs to the set V_2. The weight is assigned between two event clusters if the overlapping Hashtags and Named Entities exceed a specified threshold (empirically chosen as 0.4). The use of ”Maximum Weight Bipartite Graph Matching” (MWBGM) is employed on the graph G’. The objective of the MWBGM algorithm is to maximise the product of the probabilities of the event cluster pairings that are ultimately linked. It involves the estimation of the maximum likelihood, which can be decreased in order to maximise the sum of the log probabilities associated with the final links. The Kuhn-Munkres algorithm [12] is utilised in order to find the matching. Finally, related event clusters are linked together to form a chain of events and stored in a file. The detailed Event Detection and Tracking algorithm is presented above (see algorithm 1).

4. Experiment and Analysis

4.0.1. Datasets:

The suggested ED approach was assessed on widely recognized benchmarks, including the ”FA Cup,” ”Super Tuesday,” and ”US Election” datasets [2]. The dataset also includes corresponding ground truths, which are outlined in Table 1. The dataset pertaining to the FA Cup consists of tweets that were published during the final match of the ”Football Association Challenge Cup” on May 5th, 2012. The FA Cup is commonly acknowledged as one of the oldest and most esteemed football tournaments, characterized by a significant and devoted fan base. The football match was held between the squads of Chelsea and Liverpool. Chelsea emerged as the triumphant team in the encounter, securing a 2-1 victory. Ramirez and Drogba successfully scored the goals, both of whom were victorious team members. Carrol scored the only goal for Liverpool. The FA Cup dataset’s ground truth encompasses a total of 13 distinct topics, which consist of ”kick-off,” ”goals,” ”half-time,” ”yellow card,” ”bookings,” and ”the end of the match” etc. The Twitter traffic during the final match and several crucial moments of the FA CUP football match is illustrated in Figure 2. The Super Tuesday dataset encompasses tweets disseminated throughout the primary elections in the United States. The aforementioned elections were held on the first Tuesday of March in 2012, comprising a total of 10 states within the nation. The ground truth dataset comprises 22 different topics that encompass the crucial events of the elections and include forecasts for the voting results in several states. The dataset pertaining to the US Election encompasses a collection of tweets that were disseminated during the presidential election of the United States in 2012, specifically on November 6 of that particular year. The election was won by President Barack Obama and Vice President Joe

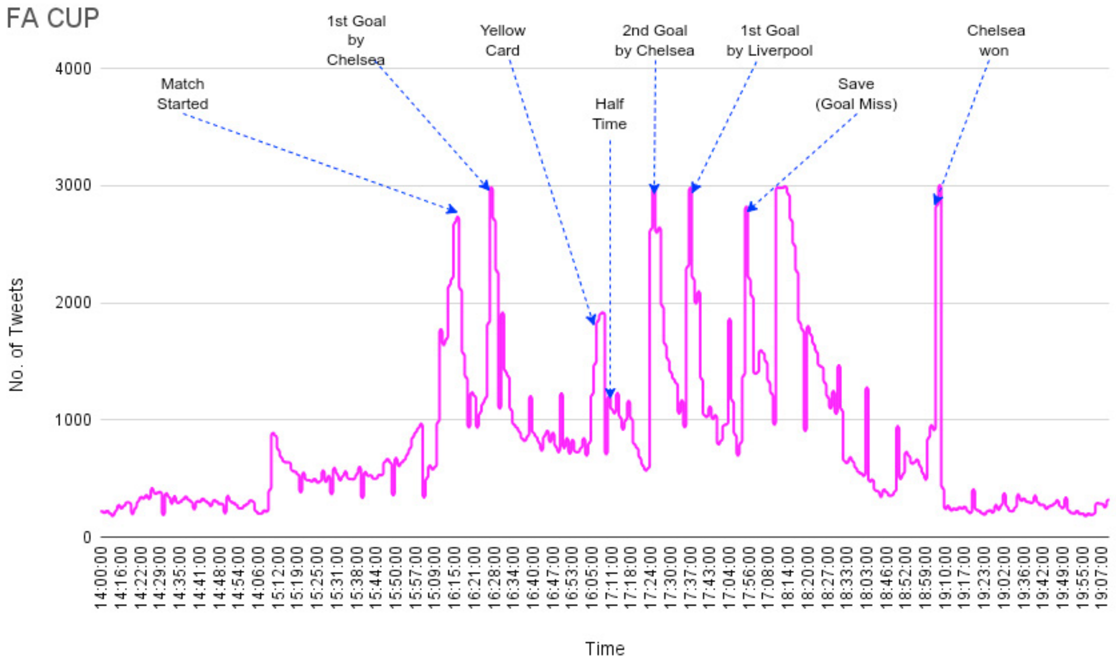


Fig. 2: Tweets traffic during FA Cup-2012 Final Match (Chelsea v/s Liverpool) and key moments of the match [2].

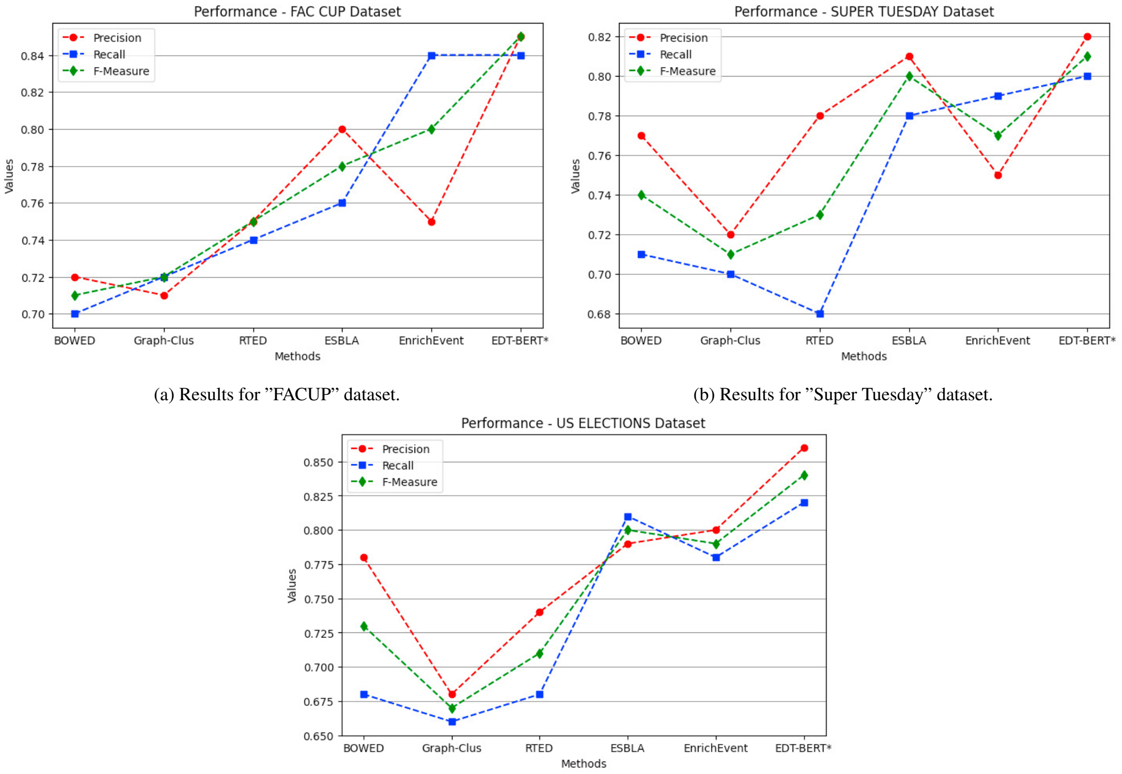
Table 1: The details of the datasets used and their temporal coverage.

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

Biden, who emerged victorious by defeating their respective opponents, Mitt Romney and Paul Ryan. The ground truth encompasses a total of 64 topics. The subjects discussed pertained to the results of the presidential election, as sourced from mainstream media outlets. We used standard classification metrics: precision, recall and F-Measure (followed by Goyal et al.[8]) to find the efficiency of our method.

4.0.2. Results & Discussion:

The empirical results indicate that the proposed methodology achieves a significant level of Precision and Recall, leading to a suitably high F-measure in comparison to the baseline methodologies (see Figure 3). The methodology presented in this study exhibits enhanced performance in comparison to the baseline approaches on all three benchmark datasets. This superiority can be attributed primarily to its improved capacity to capture both the structural relationships and contextual knowledge that are embedded within tweets. Additionally, it asserts that pre-trained BERT encoders are well-suited for the event detection task.



(a) Results for ”FACUP” dataset.

(b) Results for ”Super Tuesday” dataset.

(c) Results for ”US Elections” dataset.

Fig. 3: Performance of EDT_{BERT} w.r.t. baseline methods, *** is used to highlight proposed method**.

5. Conclusion & Future Research Work

Event detection on social media platforms is a vibrant and continually evolving research area with significant implications across various fields. Identifying events from tweets can yield valuable insights and awareness that can assist critical decision-making processes. This paper proposes a novel event detection and tracking method called ”EDT_{BERT}” to identify unspecified events in the Twitter stream. The solution we propose leverages the structural relationships and contextual representations of tweets. The process involves generating a graphical representation of tweets in order to depict the interconnectedness and contextual associations between them. Contextual embeddings are obtained by applying a pre-trained BERT sentence transformer to tweets. The calculation of structural association among tweets is performed by considering the presence of overlapping hashtags and named entities in the tweets. Our proposed method leverages the graph clustering method to find semantically similar tweet clusters related to an event. The goal of event monitoring is framed as a graph-matching problem on a bipartite graph. Through the utilization of the MWBGM algorithm, namely the Hungarian approach, it is possible to achieve global optimization while tracking events that take place in contiguous time intervals. This technique allows us to effectively group events that are connected to each other, resulting in the formation of event chains. The available empirical evidence indicates that the proposed strategy effectively recognises events and monitors their development within a Twitter stream. A promising area for further investigation involves the integration of location data into the event detection procedure, as well as the inclusion of an event summarising component within our framework. This would result in developing a coherent summary of events, which would hold significant value for organisations and individuals seeking relevant information.

References

[1] Adedoyin-Olowe, M., Gaber, M.M., Dancausa, C.M., Stahl, F., Gomes, J.B., 2016. A rule dynamics approach to event detection in twitter with its application to sports and politics. Expert Systems with Applications 55, 351–360. URL: https://www.sciencedirect.com/science/article/pii/S0957417416300598, doi:https://doi.org/10.1016/j.eswa.2016.02.028.

[2] Aiello, L.M., Petkos, G., Martin, C., Corney, D., Papadopoulos, S., Skraba, R., Göker, A., Kompatsiaris, I., Jaimes, A., 2013. Sensing trending topics in twitter. IEEE Transactions on Multimedia 15, 1268–1282. doi:10.1109/TMM.2013.2265080.

[3] Allan, J., 2002. Topic detection and tracking, Kluwer Academic Publishers, Norwell, MA, USA. chapter Introduction to Topic Detection and Tracking, pp. 1–16. URL: http://dl.acm.org/citation.cfm?id=772260.772262.

[4] Alomari, E., Katib, I., Albeshri, A., Mehmood, R., 2021. Covid-19: Detecting government pandemic measures and public concerns from twitter arabic data using distributed machine learning. International Journal of Environmental Research and Public Health 18. URL: https://www.mdpi.com/1660-4601/18/1/282, doi:10.3390/ijerph18010282.

[5] Becker, H., Chen, F., Iter, D., Naaman, M., Gravano, L., 2011. Automatic identification and presentation of twitter content for planned events, in: Adamic, L.A., Baeza-Yates, R., Counts, S. (Eds.), Proceedings of the Fifth International Conference on Weblogs and Social Media, Barcelona, Catalonia, Spain, July 17-21, 2011, The AAAI Press. URL: http://www.aaai.org/ocs/index.php/ICWSM/ICWSM11/paper/view/2743.

[6] Esfahani, M.S., Akbari, M., 2023. Enrichevent: Enriching social data with contextual information for emerging event extraction. arXiv:2307.16082.

[7] Fedoryszak, M., Frederick, B., Rajaram, V., Zhong, C., 2019. Real-time event detection on social data streams, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, ACM, New York, NY, USA. pp. 2774–2782. URL: http://doi.acm.org/10.1145/3292500.3330689, doi:10.1145/3292500.3330689.

[8] Goyal, P., Kaushik, P., Gupta, P., Vashisth, D., Agarwal, S., Goyal, N., 2020. Multilevel event detection, storyline generation, and summarization for tweet streams. IEEE Transactions on Computational Social Systems 7, 8–23. doi:10.1109/TCSS.2019.2954116.

[9] Hasan, M., Orgun, M.A., Schwitter, R., 2016. TwitterNews+: A Framework for Real Time Event Detection from the Twitter Data Stream. Springer International Publishing, Cham. pp. 224–239. URL: http://dx.doi.org/10.1007/978-3-319-47880-7_14, doi:10.1007/978-3-319-47880-7_14.

[10] Hasan, M., Orgun, M.A., Schwitter, R., 2018. A survey on real-time event detection from the twitter data stream. Journal of Information Science 44, 443–463. URL: https://doi.org/10.1177/0165551517698564, doi:10.1177/0165551517698564, arXiv:https://doi.org/10.1177/0165551517698564.

[11] Knupfer, H., Neureiter, A., Jörg Matthes, 2023. From social media diet to public riot? engagement with "greenfluencers" and young social media users' environmental activism. Computers in Human Behavior 139, 107527. URL: https://www.sciencedirect.com/science/article/pii/S0747563222003478, doi:https://doi.org/10.1016/j.chb.2022.107527.

[12] Kuhn, H.W., 1955. The hungarian method for the assignment problem. Naval Research Logistics Quarterly 2, 83–97. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/nav.3800020109, doi:https://doi.org/10.1002/nav.3800020109, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/nav.3800020109.

[13] Kumar, S., Liu, H., Mehta, S., Subramaniam, L.V., 2015. Exploring a scalable solution to identifying events in noisy twitter streams, in: Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015, ACM, New York, NY, USA. pp. 496–499. URL: http://doi.acm.org/10.1145/2808797.2809389, doi:10.1145/2808797.2809389.

[14] Li, X., Xu, M., Zeng, W., Tse, Y.K., Chan, H.K., 2023. Exploring customer concerns on service quality under the covid-19 crisis: A social media analytics study from the retail industry. Journal of Retailing and Consumer Services 70, 103157. URL: https://www.sciencedirect.com/science/article/pii/S0969698922002508, doi:https://doi.org/10.1016/j.jretconser.2022.103157.

[15] Loynes, C., Ouenniche, J., Smedt, J.D., 2020. The detection and location estimation of disasters using twitter and the identification of non-governmental organisations using crowdsourcing. Annals of Operations Research 308, 339 – 371. URL: https://api.semanticscholar.org/CorpusID:220508742.

[16] Manaskasemsak, B., Chinthanet, B., Rungsawang, A., 2016. Graph clustering-based emerging event detection from twitter data stream, in: Proceedings of the Fifth International Conference on Network, Communication and Computing, ACM, New York, NY, USA. pp. 37–41. URL: http://doi.acm.org/10.1145/3033288.3033312, doi:10.1145/3033288.3033312.

[17] Owuor, I., Hochmair, H.H., 2023. Temporal relationship between daily reports of covid-19 infections and related gdelt and tweet mentions. Geographies 3, 584–609. URL: https://www.mdpi.com/2673-7086/3/3/31, doi:10.3390/geographies3030031.

[18] Peng, X., Zhou, Z., Zhang, C., Xu, K., 2023. Detecting political opinions in tweets through bipartite graph analysis: A skip aggregation graph convolution approach. arXiv:2304.11367.

[19] Pradhan, A.K., Mohanty, H., 2015. Article: Finding tweet events. IJCA Proceedings on International Conference on Distributed Computing and Internet Technology ICDCIT 2015, 5–12. Full text available.

[20] Pradhan, A.K., Mohanty, H., Lal, P.R., 2022. Events in tweets: Graph-based techniques. Recent Advances in Computer Science and Communications 15, 155–169. URL: http://www.eurekaselect.com/article/109693.

[21] Pradhan, A.K., Mohanty, H., Lal, R.P., 2019. Event detection and aspects in twitter: A bow approach, in: Fahrnberger, G., Gopinathan, S., Parida, L. (Eds.), Distributed Computing and Internet Technology, Springer International Publishing, Cham. pp. 194–211.

[22] Reimers, N., Gurevych, I., 2019. Sentence-bert: Sentence embeddings using siamese bert-networks, in: Conference on Empirical Methods in Natural Language Processing. URL: https://api.semanticscholar.org/CorpusID:201646309.

[23] Singh, J., Pandey, D., Singh, A.K., 2023. Event detection from real-time twitter streaming data using community detection algorithm. Multimedia Tools and Applications URL: https://doi.org/10.1007/s11042-023-16263-3, doi:10.1007/s11042-023-16263-3.

[24] Van Dongen, S., 2008. Graph clustering via a discrete uncoupling process. SIAM J. Matrix Anal. Appl. 30, 121–141. URL: http://dx.doi.org/10.1137/040608635, doi:10.1137/040608635.

[25] Wu, J., Wang, Y., 2021. A text correlation algorithm for stock market news event extraction, in: Zeng, J., Qin, P., Jing, W., Song, X., Lu, Z. (Eds.), Data Science, Springer Singapore, Singapore. pp. 55–68.