Gradual Network Sparsification and Georeferencing for Location-Aware Event Detection in Microblogging Services

Diaoulé Diallo(✉) and Tobias Hecking

German Aerospace Center (DLR) Institute for Software Technology, Linder Höhe, Cologne 51147, Germany
diaoule.diallo@dlr.de

Abstract. Event detection in microblogging services such as Twitter has become a challenging research topic within the fields of social network analysis and natural language processing. Many works focus on the identification of general events with event types ranging from political news and soccer games to entertainment. However, in application contexts like crisis management, traffic planning, or monitoring people's mobility during pandemic scenarios, there is a high need for detecting localisable physical events. To address this need, this paper introduces an extension of an existing event detection framework by combining machine learning-based geo-localisation of tweets and network analysis to reveal events from Twitter distributed in time and space. Gradual network sparsification is introduced to improve the detection events of different granularity and to derive a hierarchical event structure. Results show that the proposed method is able to detect meaningful events including their geo-locations. This constitutes a step towards using social media data to inform, for example, traffic demand models, inform about infection risks in certain places, or the identification of points of interest.

Keywords: Event detection · Social media analysis · Twitter analysis · Data science · Georeferencing · Network clustering

1 Introduction

Microblogging has become an important medium to exchange information, messages, opinions, and upcoming events. Private individuals as well as politicians, celebrities and in general people with great influence use the platform to disseminate information. Recognizing that rich and user-generated content is of great utility to identify events happening in time and space, the challenging discipline of event detection has emerged. As one of the largest microblogging services, which also offers an easily accessible API,[1] Twitter has become a common data

[1] https://developer.twitter.com/en/docs/twitter-api.

H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2016 2022, SCI 1077, pp. 108–120, 2023.
https://doi.org/10.1007/978-3-031-21127-0_10

source for event detection in social media content. Current Twitter event detection methods differ not only in terms of the underlying methods, but also in terms of the type of events to be detected [1–3]. The obtaining of tweets from a specific country for the purpose of event location detection is usually connected with the necessity of a geo-tagged Twitter stream. The Twitter API can only provide geo-tagged tweets when a request is limited by a geographical area. As noted in previous work [4–6], a common issue is that only about 1–2% of all tweets are geo-tagged with exact coordinates. Therefore, a geo-tagged stream might reduce the overall quality of the detected events or even change the kind of detected events in general. With the qualitative improvements of geo-place name extractors and generally named entity recognition models in different languages, the possibilities to localize events have improved. Besides the focus on the localisation of events, the exhaustive collection of events of different granularity as well as hierarchical structures that can reveal sub-events are of great importance. To advance event detection towards serious applicability in real-world use cases we present the following contributions:

- We adapt an existing state-of-the-art event detection approach to find more events in the same time window using gradual sparsification of a co-occurence network of named entities, hashtags, and geo-locations. With these modification also the dependence on external information sources (in this case Wikipedia) is reduced.
- We analyse the resulting events in terms of the type, granularity and hierarchical structure revealing an higher diversity and sub-events.
- We investigate how well events can be localized using geo-information from tweet texts and the application of a microblog-specific place name extractor.

In the following, Sect. 2 presents related work in the area of event detection and local event detection. Subsequently, Sect. 3 presents the adaption of an existing event detection algorithm as well as the introduction of event localisation. We present our results in Sect. 4 and provide a brief conclusion in Sect. 5.

2 Related Work

There have been many attempts to define events in social media [1,2,7,8]. Following these explanations, events in online streams can be defined as the occurrence of topics and entities of substantial volume during a certain period.

We can distinguish between event detection using external information and event detection without any prior knowledge. Thus, in the literature two problem formulation exists, namely specified event detection and unspecified event detection [1,2]. The former uses external knowledge about the type of event and can thus, for example, selectively collect tweets that contain specific keywords. The more important and also more challenging discipline is that of unspecified event detection. In that case, events are detected without the use of further information about the event.

Event detection techniques include but are not limited to graph theory [9–11], burst detection in time series [12–14], and clustering [12,15,16], using different text representations, tweet meta data, hashtags and entities.

SEDTWik [12] combines a burstiness score with clustering and Wikipedia in order to detect unspecified events in an unsupervised manner. The algorithm takes into account not only the tweet texts but also various meta data and can be easily adapted for different time windows. It is still referred to SEDTWik as state-of-the-art for comparison [17,18]. As the algorithm is ideally suited to be extended in the direction of real-world event detection including event localisation and sub-events, it serves as the basis in this work. We provide a detailed description of SEDTWik in Sect. 3.

Many event detection techniques which attempt to detect event locations use geo-tagged tweets exclusively [19–22]. However, since geo-tagged tweets only account for 1% of the data [23], the quality of the resulting events will suffer. Consequently geo-tagged twitter streams are not suited as the only source for detecting and localising events. Recent changes in the Twitter policies [4] further support the claim that local event detection techniques need to take account of named entity recognition (NER) models to identify place names in microblogs used for location inference. Unakard et al. [6] use the reported user location as well as location entities extracted with the Stanford Named Entity Recognizer [24]. More recently location extraction has further improved with named entity recognition models (NER) specifically created for detecting place names in microblog entries [25].

3 Method

The basic intuition behind SEDTWik [12] is to use tweet texts, hashtags, as well as user mentions and other metadata to extract entities (or segments) that occur more often than expected in a certain period and relate them based on similarity scores. Events are then identified by clustering the resulting entity network and ranked using auxiliary information from Wikipedia. In this work an adaptation of SEDTWik is presented that aims to increase the number of meaningful events detected without requiring the time-consuming computation of news values from Wikipedia by using a gradual network sparsification algorithm. Furthermore, different NER approaches are used to extract geo-locations from tweet texts and associate events with locations. These basic steps are described below.

Tweet segmentation In the tweet segmentation step, tweet texts are pre-processed[2] and segments are extracted. Segments are defined as all uni- and multigrams where at least one word exists as Wikipedia title. Since hashtags and user mentions are commonly used to thematically label tweets, these are always considered and not depending being matched with a Wikipedia title. Hashtags are split for uppercase letters so that hashtags i.e. *#SpaceShuttleEndeavour* result in *space shuttle endeavour*.

[2] https://github.com/kevalmorabia97/pyTweetCleaner.

In addition to the original SEDTWik approach the extracted segments are associated with geo-locations based on co-occurrence as a prerequisite to location-aware events detection. To do so, tweets are searched for locations using GazPNE2 [25] and spaCy [26]. GazPNE2 combines global gazetteers, deep learning and pretrained transformer models to extract geo place names from microblogs. Since the model achieves state of the art results on a variety of twitter datasets, it is perfectly suited for our case. In addition we use the well-known NER model provided by spaCy [26]. For a segment s, we store all locations L_s that occur together with segment s in a tweet, as well as the associated location frequencies.

Bursty segment extraction Segments occurring more often than expected in a time window and belong to tweets that gain attention (measured by retweets and follower count) are considered as "bursty". For details on calculating the burstiness score we refer to [12]. In total the top $K = \sqrt{N_w}$ bursty segments are further processed, with N_w being the total number of tweets in time window w.

Event identification The identified bursty segments and their similarity scores constitute a weighted network G. In SEDTWik this network is reduced to a mutual $k-$nearest neighbour network by retaining only edges (s_i, s_j) if s_i belongs to the $k-$nearest neighbours of s_j and vice versa. The components of the resulting disconnected network are considered as event candidates. The events are ranked using Wikipedia "keyphraseness" values [27]. Keyphraseness is calculated as the probability that the entity appears as anchor text in Wikipedia articles containing the entity [12,27]. The "newsworthiness" of an event candidate is calculated from the keyphraseness of contained entities and their similarities. By discarding event candidates with newsworthiness below a threshold T, clusters of generic terms that emerge randomly are filtered out.

While the approach described above works well for identifying major events on Twitter, it has several shortcomings that are addressed by our alternative method using *gradual network sparsification*.

In the original approach segments are only considered if they are in any $k-$nearest neighbour list of another segment. However, since the similarity value could be high for several segments, important information is dropped here. On the right-hand side of Fig. 1 one can see an example where a segment (dark node) is removed from the graph although is has high similarities to other segments that may be a candidate for an event. Contrarily, parts of the similarity graph with overall low similarity scores can accidentally be detected as an event candidate as it can be seen in the left example in Fig. 1.

A less restrictive method for event identification that does not delete nodes based on a fixed parameter k regardless of whether there are highly similar but not mutually top-k neighbours is presented in the following. Starting with G where initially almost all nodes are connected to each other by edges weighted based on their similarity of their endpoints. An initial value for an edge weight threshold $t_{sim}^{(1)}$ is chosen. All edges with weight below $t_{sim}^{(1)}$ are deleted. Next, the similarity threshold $t_{sim}^{(i+1)} = t_{sim}^{(i)} + \Delta_{t_{sim}}$ is increased by a defined value $\Delta_{t_{sim}}$, and the edge deletion procedure is repeated. After each edge removal step all

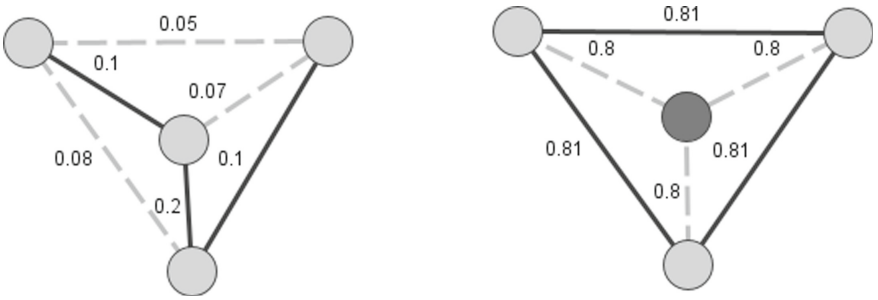

Fig. 1. Dashed edges do not belong to the mutual k-nearest neighbour graph. Left: Segment cluster with overall low similarity scores detected as event candidate. Right: The darker node will be removed from the graph although it probably belongs to an event candidate. For simplicity, k is set to 2 in this example.

isolated nodes are removed from the graph as well. In each iteration connected components of $size_{cc}$ nodes for which $size_{cc}^{min} \leq size_{cc} \leq size_{cc}^{max}$ are stored as events. This gradual network sparsification is repeated until the graph contains no more nodes. At that point, all nodes were either removed because they had no neighbors left or because they were assigned to an event. The algorithm is summarized in pseudo-code below:

Algorithm 1 Gradual network sparsification

```
t_sim^(1) ← 0.1
while G is not empty do
    for e in G do
        if sim_e ≤ t_sim then
            - Remove e from G
        end if
    end for
    for cc in connected_components(G) do
        if size_cc^min ≤ size_cc ≤ size_cc^max then
            - Store cc in event list
            - Remove nodes and edges of cc
        end if
    end for
    t_sim^(i+1) = t_sim^(i) + Δ_t_sim
end while
```

As the basic SEDTWik algorithm tends to cluster similar segments such as *wan na hear, wan na talk, wan na know, know takes, wa na see*[3] and *good day, today day, today good day, good morning, early morning*, it is reasonable

[3] Some of these segments result from the resolving of hashtags by capital letters.

to apply some method to detect and remove such *star clusters*.[4] Therefore, as a final step, sequence matching is applied, where we evaluate the term-based similarity of the different segments in a segment cluster. A segment cluster with the segments *this day, today day, good day* has a term-based similarity of 0.5, since each segment shares half of the containing terms with every other segment from the segment cluster. Previous experiments have shown that a term-based threshold of 0.3 detects most star events without removing significant segment clusters. The overall approach does not suffer from the issues of event detection based on a k-nearest neighbours graph depicted in Fig. 1. It is also possible to flexibly define how large or small segment cluster should be, and thus, allows for tuning granularity. Gradual network sparsification allows for subdivide larger segment clusters into smaller segment clusters representing sub-events. Another advantage over the original SEDTWik approach is that event identification does not rely on the costly calculation newsworthiness values from Wikipedia. Our method has certain similarities to the Girvan-Newman algorithm [28] for community in networks where edges are iteratively removed and re-evaluated based on their edge-betweenness centrality.

Event localisation The presented approach can generally detect all types of events. However, since we are interested in localisable real-world events the previously determined co-occurences of tweet segments and identified place names are especially considered. In large datasets almost all segments co-occur with some locations just by chance. However, in the case of localisable events we observe that the defining segments co-occur way more often with locations where the event takes place, while the frequency of other locations is almost uniformly distributed. Consequently, localisable events can be classified using the Gini coefficient of the association strength (number of co-occurences) of tweet segments that define an event and location names. Since the Gini coefficient is a measure how skewed a distribution is the value is high when one or a few place names stand out and almost 0 for non-localisable events.

4 Results

In the following, the results for event detection using gradual network sparsification are compared with the original version of the SEDTWik algorithm.

4.1 Experimental Setting

Dataset and Metrics. For our experiments, we used the same dataset as the authors of SEDTWik. They used seven days (October 11, 2012 to October 17, 2012) from the event detection dataset *Event2012* created by McMinn et al. [29]. Regarding Twitter data, there are many difficulties to face as for example in the creation, dissemination and comparability of Twitter datasets (for more detailed information see [3,4,30,31]). Hettiarachchi et al. [17] showed that only

[4] As the connected components often look like stars with one central node.

65.8% of the provided tweet ids from the *Event2012* dataset could be received. Nevertheless, the sample from the *Event2012* dataset used here is sufficient to evaluate the improvements of the adapted algorithm. The majority of all publications create their own tweet collections, which makes comparisons generally difficult [1].

According to the authors' description, the streaming API was used without further filtering and then non-English tweets and spam were removed [29]. As noted by SEDTWik's authors, the dataset lacks an exhaustive event list associated with the tweets [12]. Since there are no predefined labels, we evaluated all resulting events manually. This means that for each event consisting of a set of segments and a tweet summary, it is verified whether it is an actual event via web search. Only the events that were detected on the correct day are counted as valid events. For example, if the release of an album is detected as an event within the tweets of 15 October 2012, but the album was already released on 14 October 2012, this event will not be counted as a correctly detected event.

The number of detected valid events will serve as the main metric as in [12]. The authors also use a precision and a DERate as metrics. However, since a valid event is defined differently in this work (see evaluation subsection below) and due to subjective components in the verification of the events, we limit the evaluation metric to the number of detected actual events.

Parameter Setting As long as t_{sim} is initiated with a low value and $\Delta_{t_{sim}}$ is chosen sufficiently small in the network sparsification step, no events are lost. We found that setting an initial t^1_{sim} of 0.05 and $\Delta_{t_{sim}}$ to 0.01 is small enough while the computation time is acceptable. Setting $\Delta_{t_{sim}}$ too high could cause several segments to lose their edges within one step, causing them to be discarded instead of being recognized as an event. *lower_bound* is set to 3 and *upper_bound* to 10.

4.2 Effect on Gradual Network Sparsification on Event Detection

In the following, the variation of the SEDTWik approach including gradual network sparsification described in Sect. 3 is compared to the original formulation of the method in terms of the number and nature of detected events. For a fair comparison the events found by SEDTWik and reported by [12] (available here[5]) were evaluated. Taking into account that (1) no events are valid that were detected for the wrong day and that (2) simple discussions in tweets about past events are also not valid (for example, last weekend's football game), we have to reduce the number of detected events reported by SEDTWik from 79 to 71. Both SEDTWik and our approach detect the *breast cancer awareness month*, which we exclude based on the criteria described in Sect. 4.1.

In total, we were able to detect 90 actual events that were manually verified through a web search, which is 21.1% more than the events detected by SEDTWik. This can be attributed to the applied gradual networks sparsification algorithm for event detection described in Sect. 3, that is less restrictive

[5] https://github.com/kevalmorabia97/SEDTWik-Event-Detection-from-Tweets.

regarding which tweet segments are considered to belong to an event and does not need to filter events based on their newsworthiness score calculated using Wikipedia as external knowledge base.

For a more detailed comparison Table 1 shows the number of events detected per day for each of the two methods. It can be seen that the proposed method never detects fewer actual (verifiable) events on each day than SEDTWik. Surprisingly although our approach yields more events not all SEDTWik events are included in the result set. One explanation can be that gradual network sparsification tends to discard loosely coupled subgraphs of the similarity graph (i.e. segments that make up an event but having low overall similarity) as shown on the left side of Fig. 1. While this property is helpful to filter in many cases there are also sometimes events that are described by very general segments that do not have a strong temporal correlation. As mentioned in Sect. 4.1, we will not focus on precision, recall or similar measures, which depend highly on subjective interpretation. However, approximately 70% of all detected events could be verified as actual events following the definition of Sect. 4.1. Considering the precision values published by Morabia et al. [12], our approach shows a slightly increased false positive rate. Nevertheless, it must be emphasized that a comparability is not given for the reasons already explained above.

Table 1. Comparison of number of detected events per date.

	Detected actual events		
Date	SEDTWik	Our approach	In common
2012/10/11	13	15	7
2012/10/12	8	8	7
2012/10/13	9	9	1
2012/10/14	10	13	7
2012/10/15	10	12	8
2012/10/16	14	19	5
2012/10/17	8	14	4

Table 2 shows some examples of actual events that could not be detected by SEDTWik but by our approach. As can be seen, the types of events range from announcements, current political world events, content about famous people, and sports to daily news and election reminders. The type of events discovered by SEDTWik are to some degree comparable, nevertheless the majority of them are sports-related events. The events detected by our approach are more diverse and not as biased towards sport events. This can be explained by the fact that SEDTWik discards events whose newsworthiness is below a threshold, to exclude non significant segment clusters. The drawback of this approach is that at the same time meaningful segment clusters with low newsworthiness are also

removed. Also, our observation does not confirm that events with high newsworthiness are necessarily more important events. In contrast to SEDTWik, we remove such segment clusters using the term similarity of the segments in one segment cluster, as explained in Sect. 3.

Table 2. Some events which were not detected by SEDTWik but our approach.

Date	Event Info
2012/10/11	– Samsung confirms Galaxy S III Mini
2012/10/12	– NASA space shuttle Endeavour begins 12-mile trip by road to new home – President Obama's campaign said Springsteen will appear along Clinton a pro-Obama rally
2012/10/13	– Jonathan Ross show with David Walliams, Ed Sheeran and more
2012/10/14	– Start of new seaseon Walking Dead
2012/10/15	– Man seriously hurt after being stabbed near Candlestick Park after San Francisco 49er game – Patriots versus Seahawks game
2012/10/16	– Last day to register vote in Maryland – Microsoft announces new surface details and starts pre-order
2012/10/17	– Police Taser blind man mistaking his white stick for a samurai – Google throws open doors to its top-secret data center

4.3 Localising Events

We categorised the detected verified events as localisable or non-localisable using the Gini coefficient as explained in Sect. 3. Out of 90 actual events, 32 could be associated with a specific location. Since it is noticeable that the granularity of the locations varies greatly and to a certain degree also depends on the nature of the event, it is difficult to make an absolute statement about the quality of the location extraction. However, to make a simple evaluation of the location determination procedures, we examined the two locations with the highest frequencies for each localisable event to see if they match the actual event location. In 28 of the 32 cases, the list of the two top locations contains at least one correct event location. The identified event locations are at state level 7 times, at state level 2 times, at city level 18 times, and at place level once. For most sport events the event locations can be found on city level. The extracted locations are greatly influenced by the team names, which in almost all cases already have locations in their names. Events like the *vice presidential debate* and *Ella Enderson and James Arthur performing at X Factor* could only be located on country-level. *Dancing with the Stars: Paula Abdul Guest Star Edition* is an example for a

localisable event which was incorrectly assigned by our approach. However, as the focus of TV shows is often not on the shooting location it is questionable in how far these kinds of events are localisable. We observe that the granularity of the detected locations indicate the 'spatial extent' of the event. For example, the detected location for the *presidential debate* was *America* while the sports event involving the *New York Yankees* was associated with *new york*. In these cases, the detected locations rather.

4.4 Hierachical Sub-event Structures

A closer look at the events reveals that they partly overlap thematically, but show a different focus. For example for the vice presidential debate on 2012/10/12 as well as for the presidential debate on 2012/10/17, sub-events emerged. Certain debate topics can be found in different segment clusters which indicates the potential to generate hierarchical sub-event structures by using gradual network sparsification. Topics around the presidential debate and therefore the different detected events were ranging from taxes over parenthood to gun violence (see Fig. 2). Figure 2 shows the state of the segment network during gradual network sparsification. As we can see, the cluster around *energy independence* has just been identified as an event as it becomes an isolated connected component. In the step before, the segments of that connected component were still part of the larger segment network.

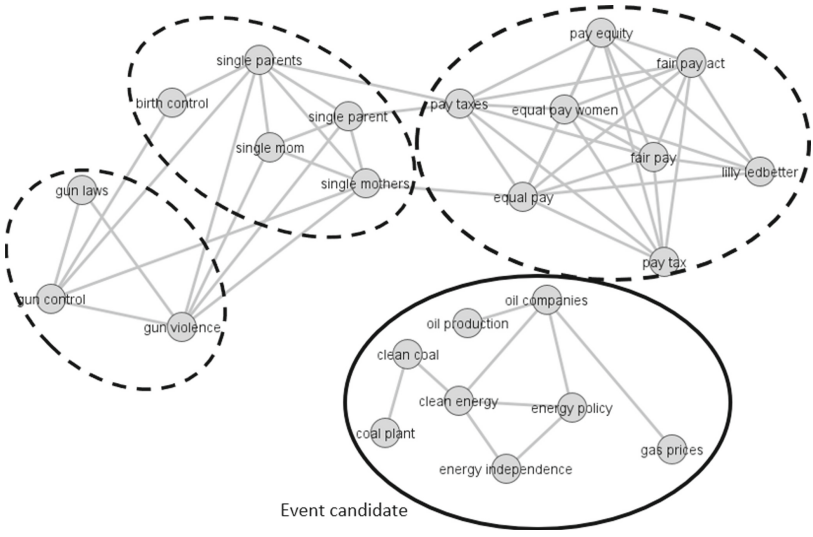

Fig. 2. An example of an intermediate step of the gradual network sparsification algorithm. Aspects of the presidential debate are first related but become event candidates after a sufficient number of iterations.

5 Conclusion

This paper described a method for localisable event detection in Twitter streams. It adapted the state-of-the-art method proposed by Morabia et al. [12] by incorporating place name extraction as well as a novel gradual network sparsification algorithm that reduces a co-occurrence network of tweet segments to meaningful components representing events. It could be shown that this approach leads to the detection of a significantly higher number of verifiable events along with physical locations.

In future works it will be investigated how to tune the parameters of the event detection algorithm according to spatial granularity (i.e. city, state, country level). Further improvements can also be made regarding the precision of the results since some detected events could not be verified.

References

1. Saeed, Z., Abbasi, R.A., Maqbool, O., Sadaf, A., Razzak, I., Daud, A., et al.: What's happening around the world? a survey and framework on event detection techniques on twitter. J. Grid Comput. **17**(2), 279–312 (2019)
2. Hasan, M., Orgun, M.A., Schwitter, R.: A survey on real-time event detection from the twitter data stream. J. Inf. Sci. **44**(4), 443–463 (2018)
3. Weiler, A., Grossniklaus, M., Scholl, M.H.: Survey and experimental analysis of event detection techniques for twitter. Comput. J. **60**(3), 329–346 (2017)
4. Kruspe, A., Häberle, M., Hoffmann, E. J., Rode-Hasinger, S., Abdulahhad, K., Zhu, X. X.: Changes in Twitter Geolocations: Insights and Suggestions for Future Usage. arXiv:2108.12251 (2021)
5. Choi, D., Park, S., Ham, D., Lim, H., Bok, K., Yoo, J.: Local event detection scheme by analyzing relevant documents in social networks. Appl. Sci. **11**(2) (2021)
6. Unankard, S., Li, X., Sharaf, M.A.: Emerging event detection in social networks with location sensitivity. In: World Wide Web, pp. 1393–1417 (2015)
7. Becker, H., Naaman, M., Graano, L.; Beyond trending topics: Real-world event identification on twitter. In: AAAI Conference on Web and Social Media, vol. 5, no. 1 (2011)
8. Dou, W., Wang, X., Ribarsky, W., Zhou, M.: Event detection in social media data. In: IEEE VisWeek Workshop on Interactive Visual Text Analytics-task Driven Analytics of Social Media Content, pp. 971–980 (2012)
9. Sayyadi, H., Hurst, M., Maykov, A.; Event detection and tracking in social streams. In: AAAI Conference on Web and Social Media, vol. 3, no. 1, pp. 311–314 (2009)
10. Schinas, M., Papadopoulos, S., Petkos, G., Kompatsiaris, Y., Mitkas, P. A.: Multimodal graph-based event detection and summarization in social media streams. In: 23rd ACM International Conference on Multimedia, pp. 189–192 (2015)
11. Edouard, A., Cabrio, E., Tonelli, S., Le Thanh, N.: Graph-based event extraction from twitter. In: Recent Advances in Natural Language Processing (2017)
12. Morabia, K., Murthy, N., Malapati, A., Samant, S.: SEDTWik: Segmentation-based event detection from tweets using Wikipedia. In: Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop, pp. 77–85 (2019)

13. Xie, W., Zhu, F., Jiang, J., Lim, E. P., Wang, K.: Topicsketch: Real-time bursty topic detection from twitter. IEEE Trans. Knowl. Data Eng. **28**(8), 2216–2229 (2016)
14. Corney, D., Martin, C., Göker, A.: Spot the ball: Detecting sports events on Twitter. In: European Conference on Information Retrieval, pp. 449–454 (2014)
15. Hasan, M., Orgun, M.A., Schwitter, R.: Real-time event detection from the Twitter data stream using the TwitterNews+ framework. Inf. Process. Manage. **56**(3), 1146–1165 (2019)
16. Nguyen, S., Ngo, B., Vo, C., Cao, T.: Hot topic detection on twitter data streams with incremental clustering using named entities and central centroids. In: IEEE-RIVF International Conference on Computing and Communication Technologies, pp. 1–6 (2019)
17. Hettiarachchi, H., Adedoyin-Olowe, M., Bhogal, J., Gaber, M.M.: Embed2Detect: temporally clustered embedded words for event detection in social media. Mach. Learn. **111**(1), 49–87 (2022)
18. Ahmad, F., Abbasi, A., Kitchens, B., Adjeroh, D.A., Zeng, D.: Deep learning for adverse event detection from web search. IEEE Trans. Knowl. Data Eng. **34**(06), 2681–2695 (2022)
19. Zhang, C., Lei, D., Yuan, Q., Zhuang, H., Kaplan, L., Wang, S., Han, J.: GeoBurst+ effective and real-time local event detection in geo-tagged tweet streams. ACM Trans. Intell. Syst. Technol. **9**(3), 1–24 (2018)
20. He, J., Liu, Y., Jia, Y.: EventGraph based events detection in social media. In: International Conference of Pioneering Computer Scientists, Engineers and Educators, pp. 150–160 (2018)
21. George, Y., Karunasekera, S., Harwood, A., Li, K. H.: Spatio-temporal event detection using poisson model and quad-tree on geo-tagged social media. In: IEEE International Conference on Big Data, pp. 2247–2256 (2019)
22. Wei, H., Zhou, H., Sankaranarayanan, J., Sengupta, S., Samet, H.: Delle: detecting latest local events from geo-tagged tweets. In: 3rd ACM International Workshop on Analytics for Local Events and News, pp. 1–10 (2019)
23. Abdelhaq, H., Sengstock, C., Gertz, M.: Eventweet: online localized event detection from twitter. VLDB Endow. **6**(12), 1326–1329 (2013)
24. Jurafsky, D., Martin, J.H.: Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition (2000)
25. Hu, X., Zhou, Z., Sun, Y., Kersten, J., Klan, F., Fan, H., Wiegmann, M.: GazPNE2: a general place name extractor for microblogs fusing gazetteers and pretrained transformer models. IEEE Internet Things J. (2022)
26. Honnibal, M., Montani, I., Van Landeghem, S., Boyd, A.: spaCy: Industrial-Strength Natural Language Processing in Python (2020). https://doi.org/10.5281/zenodo.1212303
27. Li, C., Sun, A.,Datta, A.,. Twevent: segment-based event detection from tweets. In: 21st ACM International Conference on Information and Knowledge Management, pp. 155–164 (2012)
28. Girvan, M., Newman, M.E.: Community structure in social and biological networks. PNAS **99**(12), 7821–7826 (2002)
29. McMinn, A. J., Moshfeghi, Y., Jose, J. M.: Building a large-scale corpus for evaluating event detection on twitter. In: Proceedings of the 22nd ACM International Conference on Information & Knowledge Management, pp. 409–418 (2013)

30. Tromble, R., Storz, A., Stockmann, D.: We don't know what we don't know: when and how the use of Twitter's public APIs biases scientific inference. Available at SSRN 3079927 (2017)
31. Campan, A., Atnafu, T., Truta, T.M., Nolan, J.: Is data collection through twitter streaming API useful for academic research? In: 2018 IEEE International Conference on Big Data, pp. 3638–3643 (2018)