Geoinformatica
https://doi.org/10.1007/s10707-019-00392-9

Enhancing local live tweet stream to detect news

Hong Wei[1] · **Jagan Sankaranarayanan**[2] · **Hanan Samet**[1]

Published online: 09 January 2020

Abstract
Twitter captures invaluable information about real-world news, spanning a wide scale from large national/international stories like a presidential election to small local stories such as a local farmers market. Detecting and extracting small news for a local place is a challenging problem and the focus of this work. The main challenge lies in identifying these small stories that correspond to a local area of interest, which are typically harder to detect compared to national stories in the sense that there may be just a handful of tweets about a local story. A system, called Firefly, is proposed that overcomes the data sparsity and captures thousands of local stories per day from a metropolitan area (e.g., Boston). The key idea lies in combining the enhancement of a local live tweet stream in Twitter, the identification of "locality-aware" keywords, and using these keywords to cluster tweets. Experiments show that the proposed system has a significantly higher recall over a set of representative local news agencies, and at the same time, outperforms the baseline approach TwitterStand. More importantly, the results also demonstrate that our system, by utilizing the enhanced local live tweet stream, discovers much more local news than the methods working only on geotagged tweets, i.e., those with embedded GPS coordinate values.

Keywords Twitter · Live tweet stream · News detection · Local news · Geotagging · Apache spark

1 Introduction

The popularity of Twitter arises from its capability of letting users promptly and conveniently contribute tweets on a wide variety of subjects such as news, stories, ideas, and

✉ Hong Wei
hyw@cs.umd.edu

Jagan Sankaranarayanan
jagan@umiacs.umd.edu

Hanan Samet
hjs@cs.umd.edu

[1] Department of Computer Science, University of Maryland, College Park, MD 20742, USA

[2] UMIACS, University of Maryland, College Park, MD 20742, USA

opinions. As a result, with people discussing what is happening outside in the real world by posting tweets, an invaluable amount of information on the real world news is hidden in Twitter. Therefore, many researchers have devoted remarkable efforts to discover this knowledge. For example, TwitterStand [1] is a news tweet processing system that aggregates tweets from a sparsely sampled tweet source to detect news. This is not a problem for major news stories since there are more than enough tweets to capture them.

However this approach is too brute-force for smaller-scale local news where every single tweet matters because such types of news may only span a very limited number of tweets. Figure 1 shows a news story about the "Westborough Education Foundation" that happened at around 6:30 PM on Oct 24, 2016 at Westborough, MA. We only found 6 tweets (8 if retweets are included) about this news by the time we captured the screenshot, and none of them is geotagged, i.e., containing a pair of geographical lat/lon coordinate values. No access to full tweets in Twitter makes data sparsity pervasive in Twitter's publicly accessible tweets, and further compromises the possibility of collecting all 6 tweets about this news. The challenge in capturing such news lies in being able to find these tweets, cluster them into a news story, and then subsequently displaying it on a map.

In this paper, we are interested in detecting news (a set of tweets) that are being discussed by local people from a given place (e.g., Boston city), and meanwhile emphasizing on finding local news. The term "local news" refers to a news event that happens at or is of great interest to the given place. For instance, the news story in Fig. 1 may only be of interest to the local community and not much further beyond. Local news can sometimes escalate to be of national/international interest such as when it is dramatic (e.g., Boston Marathon bombing in May 2013). We want to capture both these types. Other national and international stories that are discussed by local people (e.g., a presidential election) are also in by providing a local perspective to larger news stories. Our focus is primarily the former two classes of stories, and later in our experiments we evaluate how well we do with and without considering these national and international news stories.

Identifying the news stories that are of great interest to a place requires a combination of approaches. It requires first finding users that reside and tweet about our place of interest. To find such users, we implement an efficient online social network-based Twitter user geotagging approach, which is to approximate the location of a Twitter user by examining the publicly-known locations of his social friends (neighbors). The publicly-known location, termed the *profile-location*, is provided in a Twitter user's profile, but is only available for around 20% (in our case, 32%) of Twitter users [2]. This makes the procedure of geotagging Twitter users indispensable in our system. With the help of this scheme and its efficiency,

Fig. 1 A local news in Westborough, MA on Oct 24th, 2016

our system, Firefly, keeps trying to find as many as possible active Twitter users from a given area and putting their posting statuses (tweets) to a local live tweet stream to largely increase its number of local tweets.

Next, there is a larger problem of clustering these local tweets so that news can be captured. For example, some features like bursty words [3, 4] or TF-IDF [1, 5] that are commonly used to group tweets together might not work well with small local news because such news span over a very limited number of tweets, and thus words in them hardly bring about burstiness or yield distinguishing TF-IDF scores. Another category of methods that only exploit geotagged tweets such as [6, 7] would simply miss the news example in Fig. 1 because few of its tweets are geotagged.

In this paper, we utilize an idea of "locality-aware keywords" to capture the changes in word-usage patterns caused by a news of limited local interest from the perspective of individual people. Essentially, the locality-aware keywords in each tweet are a set of words that are used only recently by this tweet's publisher and also at the same time only appear in a limited number of other Twitter users' tweets. Such locality-aware keywords correspond to the aspects of a local news being "novel" as its nature of being new, as well as having a small spread span among Twitter users. Take the one in Fig. 1 for example, "Westborough", "Education", "Foundation", "Trivia" and "Bee" are considered as locality-aware because they are new words used by this set of people.

To capture news from the enhanced local live tweet stream, we keep identifying and updating locality-aware keywords from tweets that are in the latest 6-hour sliding time window (The choice of 6-hour window is in recognition that television media usually has four times of locally-oriented news broadcast in one day and thus is an appropriate lifetime of local news), and group tweets together that share at least a number of locality-aware keywords to form news clusters. Finally, in our system's UI, a Twitter timeline is created to post the news we detect from an area in real time. We also estimate the geographic focus of detected news (tweets clusters) to display them on maps.

The main contribution of this paper is summarized as follows:

- We implement an efficient online Twitter user geotagging procedure on Apache Spark, which takes less than 3 seconds to geotag Twitter users appearing in 1000 tweets. Such efficiency is essential to maintaining the liveness of the enhanced local tweet stream and furthermore the timeliness in news detection.
- Our enhanced local live tweet stream easily covers up a typical metropolitan area. For example, in Boston, we are tracking 176K Twitter users, which is considered sufficient since Boston has a population of 646K[1] and that one-fifth of the USA population are active Twitter users.[2]
- The design of locality-aware keywords emphasizes the word usage characteristics of small, local news from the view of Twitter users who are discussing them (e.g., only a small number of people talk about them and they use words they didn't use before).
- We evaluate our system against a set of representative local news agencies as well as a few baseline approaches. The results show we achieve the highest news coverage and at the same time, outperform the baseline approaches. More importantly, our method detects hundreds of more local news in comparison with the methods that solely utilize the existing Twitter's publicly available tweet stream.

[1] http://www.census.gov/popest/about/terms.html

[2] https://www.statista.com/statistics/274564/monthly-active-twitter-users-in-the-united-states/

The rest of this paper is organized as follows. Section 2 summarizes the related work. Section 3 details the design and implementation of our system. Section 4 describes the experimental evaluation of our methods. Section 5 contains concluding remarks as well as directions for future work.

2 Related work

There is a large body of related work that deals with extracting useful patterns (e.g., news, events) from social media, Twitter in particular. Two recent surveys Atefeh and Khreich [8], and Abdelhaq [9] provide an excellent description of different techniques. We review some of the related work that deals specifically with the problem of detecting local events. There are two broad categories of methods for taking location into consideration when performing detection tasks, namely: *location-anchored* and *event-anchored*. The essential difference is whether event or location is the primary clustering key. For example, event-anchored methods first detect an event and then determine its location, while location-anchored methods examine if an event happens at a certain location.

Location-Anchored Methods: Among the location-anchored methods are two popular approaches: *model dimension extension* and *geographical space tessellation*. Model dimension extension treats geographical information as an additional variable to the existing models. For example, in calculating similarity between documents while performing a clustering algorithm, geographical distance between tweets can be incorporated in the clustering algorithm [10] to form potential events [7, 11–13]. Hong et al. [14], Zhou and Chen [15] and Wei et al. [16] treat geographical regions as latent variables in their generative topic model.

Geographical space tessellation fills the map with small, non-overlapping cells. The motivation here is that local news or events, which usually have an limited geographical area impact, should fall in the same or nearby cell(s). *Grid* tessellation is the simplest yet most commonly used way of subdividing the geographical space into small equal-sized cells [17–21]. In reality however, the geographical distribution of social media documents is not homogeneous, frequently requiring the consideration of adjacent cells in the analysis. To alleviate this issue, a few strategies are proposed including re-sizing the cells, connecting nearby cells if they share similar features, or utilizing an adaptive hierarchical tessellation structure [22]. For example, Krumm and Horvitz [6] et al. discretize the space with a hierarchical triangular mesh. Magdy et al. [23, 24] describe a system called Mercury for querying top-k spatio-temporal queries on microblogs in real-time using a pyramid structure.

After tessellation, the social media documents or features are aggregated into small cells according to their inferred geographical information. Next, an intuitive way to detect the existence of any anomaly at a specific location is to count aggregated documents or other feature entities like keywords to see if their number exceeds a certain threshold. Counting, however, is easily plagued by distribution heterogeneity both temporally and spatially. Therefore, various anomaly detection techniques have been explored. For example, Xu et al. [25] employ a probabilistic model that recovers spatio-temporal signals using a Poisson point process estimation to deal with sample bias and data sparsity problems. Others exploit the usages of a discrepancy paradigm which compare between previous data (to build up a baseline) and the newly observed data [6, 18, 26, 27].

Nevertheless, such methods have heavy dependence on the availability of social media documents containing geographical information. Such geographical information, however, is very rare in Twitter, with geotagged tweets accounting for less than 1% [20, 28, 29]. Some works have proposed to estimate a geographical location for a non-geotagged tweet. The intuitive approach towards this problem is to geotag nominal locations (place names) embedded in the content of a microblog to get its possible longitude/latitude coordinates by aligning against existing gazetteer databases or services, e.g. GeoNames[3] [9, 20, 30, 31]. While another set of works try to assign a geographical location to a non-geotagged tweet by its poster's location [28, 32, 33], which might be initially estimated through a social network based procedure [2, 34–37] or tweets content-based methods [38–43].

Event-anchored Methods: This class of methods, after identifying events, leverages an additional step of spatial analysis to determine the locations where they are happening. For example, TwitterStand [1], after clustering tweets to identify events, estimates each news cluster's geographical focus by making use of both geographical information in the content of the tweet and by the source location of the users. This geographic focus is computed as a whole by ranking the geographic locations in the cluster. One basic measure of relevance used in their ranking is the frequency of occurrence of each geographic location in the cluster. The reasoning is that if a geographic location is important to the event at hand, the it would be mentioned in many tweets and linked articles belonging to the cluster. In addition, they also give a higher relevance score to groups of locations that are mutually proximate by considering that geographic locations that are nearby to each other lend evidence to each other. To infer and track the location of detected earthquake or typhoon events, Sakaki et al. [44] resort to Kalman filtering and particle filtering by treating each Twitter user as a sensor.

Even though all event related documents are exploited (not just the ones with location information) in event-anchored methods, their data sources still suffer from sparsity to detect small, local events. For example, TwitterStand's data source, which then claimed to sample around 10% of all tweets but now only 1%, is still too few for small-scale events that might only span $3 \sim 5$ tweets in total.

Therefore, realizing it is the local data sparsity that undermines the opportunities for researchers to discover small-scale events in Twitter, our system proposes to enhance the public local live tweet stream for an area by i) identifying as many Twitter users as possible that are from that area and then ii) tracking the tweets that they publish in real-time. Weng and Lee [45] similarly track a number of users in Singapore to detect news but only at a small scale, i.e., 1K Twitter users. In contrast, we identify and track 176K users in Boston. Our work is also different from Albakour et al. [46], which directly chooses several areas in London to collect tweet data, and tries to detect events for each of these areas separately. Their method doesn't solve the problem of local data sparsity by using Twitter's Streaming API, i.e., statuses/filter with parameter "locations", in our experiment, is still very sparse and thus makes a very limited contribution to local news detection.

3 System

In this section, we present the design and implementation of our event detection system, Firefly, as illustrated in Fig. 2. Including the User Interface, Firefly consists of 5 major modules, which are described below sequentially.

[3] http://geonames.org/

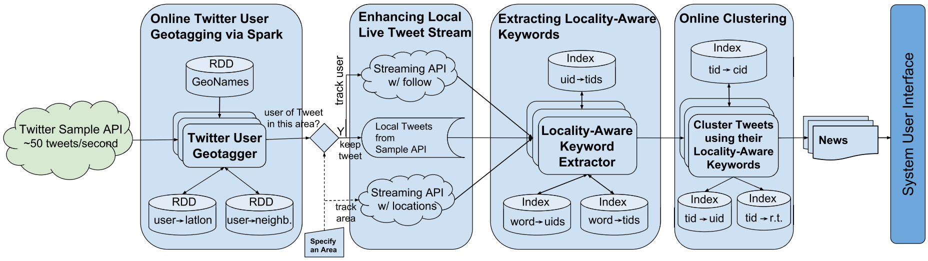


Fig. 2 System architecture of firefly

3.1 Online Twitter user geotagging via spark

The goal of this module is to keep estimating the geographical locations for more Twitter users, and thus to maintain a large pool of geotagged Twitter users. In so doing, for a given geographical area like the Boston Metropolitan area, our system can easily retrieve a large body of Twitter users in it. Tracking tweets posted by these users significantly enhances our local live tweet stream.

The motivation behind geotagging Twitter users is that the profile-location information for specifying where a Twitter user comes from is only sparsely available in public data. Therefore, inspired by studies [47, 48] that online social friendships are often formed over short geographic distances, a social network-based Twitter user geotagging method is proposed in [2], which approximates a user's location by examining the publicly-known locations of his online friends (neighbors). This method is reported to have the state-of-the-art city-level accuracy when geotagging a large-scale body of Twitter users and, more importantly, doesn't require sophisticated natural language processing in comparison with tweets content-based methods [38–42], thus making it more suitable for online geotagging.

To be specific, the social network-based geotagging problem is addressed from the point of view of solving an optimization problem, i.e., inferring user locations is solved by finding

$$min_f \|\nabla \boldsymbol{f}\| \;\text{ s.t. } f_i = l_i,\; \forall i \in L \tag{1}$$

where $\boldsymbol{f} = (f_1, f_2, f_3 ... f_n)$ represents location estimation for each user $1...n$, and L denotes the set of users who opt to make their locations l_i public. The total variation is formulated as $\|\nabla \boldsymbol{f}\| = \sum_{ij} w_{ij} * d(f_i, f_j)$, where $d(\cdot,\cdot)$ measures geographical distance. and w_{ij} weighs the friendship between user i to user j, which essentially reflects how many times user i reciprocally interacts with j such as retweeting, mentioning etc. Note that, an edge between i and j in the graph is bidirectional and only formed if both i and j have actively initiated at least one interaction with each other, and we use reciprocal neighbors or friends to term such edges.

The above minimization problem could be solved by calculating, for each user, the *L1-multivariate median* from his reciprocal neighbors' locations. The value of *L1-multivariate median* [49], which acts as a user's estimated (geotagged) location and is denoted by l^{L1mm}, essentially finds a point that minimizes the sum of its distances to the users' reciprocal neighbors. For a user j, its *L1-multivariate median* l_j^{L1mm} is mathematically defined as,

$$l_j^{L1mm} = \underset{l}{\operatorname{argmin}} \sum_{l_i \in L_j} w_{i,j} * d(l, l_i) \tag{2}$$

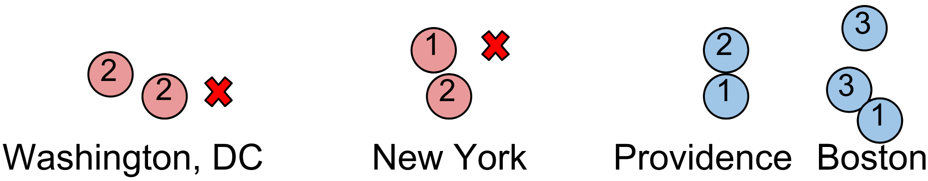


Fig. 3 An illustration of outliers in the locations of reciprocal friends

where, L_j contains the locations of j's reciprocal neighbors. In the implementation, Eq. 2 can be solved through a coordinate descent procedure.

Upon completing the calculation of location estimate, for a user j, how far l_j^{L1mm} deviates from his reciprocal neighbors determines whether he accepts l_j^{L1mm}. This deviation, called *Geographical Dispersion*, is defined as,

$$GD(L_j) = median_i \; w_{i,j} * d(l_j^{L1mm}, \; l_i) \quad \text{s.t. } l_i \in L_j \tag{3}$$

For example, user j will accept his estimated location if $GD(L_j)$ is less than a given threshold, γ. In our experiments, we set $\gamma = 100$ km, which is suggested as a suitable trade-off between geotagging coverage and accuracy for the city-level scenarios [2].

One drawback of [2] lies in indiscriminately utilizing all available location information from reciprocal friends to calculate a candidate location estimation in Eq. 2, while some of them might be noisy points as discussed in [35]. For example, as illustrated in Fig. 3 (where each circle represents a reciprocal friend and the number in each circle denotes the weight to that friend), a user from Boston has 9 reciprocal friends with available location information, 4 of them (red circles) are relatively far away from Boston and can be seen as noisy points or outliers because incorporating them into Eq. 2 is likely to yield a location estimation that does not satisfy the geographical dispersion constraints γ, and thereby fails to geotag this Twitter user.

Inspired by the observation in [35] that the location of a friend is usually more reliable if a user has multiple friends from that or nearby location, we propose a single-linkage-clustering based outlier removal procedure to get rid of potential noisy points. As presented in Algorithm 1, this procedure works as follows. Take the locations of a user j's reciprocal neighbors, L_j, as the input, we first perform the Single Linkage Clustering with geographical dispersion γ being the distance threshold. During the clustering, two location points in L_j that are within γ are grouped into the same cluster; and two clusters are merged if a pair of points from each of them are within γ. Next, we select the cluster with maximum sum of weights and use it as new L_j in Eq. 2 to calculate the location estimation.

Algorithm 1 Outlier Removal.

Input: The locations of user j's reciprocal neighbors $-L_j$; distance threshold $-\gamma$; cluster size threshold$-\lambda$

Output: A list of locations after removing outliers $-\; L'_j$

1: A set of clusters $C = \{C_1, C_2, C_3 ...\} \longleftarrow$ Single Linkage Clustering on L_j with γ;

2: $L'_j = \underset{C_k \in C}{\text{argmax}} \sum_{l_i \in C_k} w_{i,j}$

3: return L'_j if $|L'_j| \geq \lambda$; else $\emptyset$

Another improvement over [2] is a minimum size constraint for L'_j because too few location information might be considered as weak evidence [36]. In other words, we refuse to calculate l_j^{L1mm} for user j if $|L'_j|$ is less than a given threshold λ. The experimental results show that such a constraint for λ might effectively improve the accuracy of geotagging in the sparse social networks where users have only a few reciprocal friends, especially the ones with valid locations.

Algorithm 2 Online Twitter User Geotagging via Spark.

Input: Twitter's Public Live Tweet Stream – G; 1 year of tweets collected from Twitter Sample API – T

Output: Geotagged Twitter Users

1: **Boosting Phase**:
  a. Load location→latlon from GeoNames; and extract location→user, user→neighb.,
     and user→twGPS in T;
  b. user→latlon ⟵ location→user join location→latlon;
  c. Update user→latlon with users whose lat/lon can be calculated upon user→twGPS using Equation 2 and 3.

2: **Online Geotagging**:
  a. Init a Spark DStream D in G w/ a 23s time window;
  b. Update user→neighb. and user→twGPS with D;
  c. **for each** user u in D who is not in user→latlon and fails to align profile-location in location→latlon and fails to calculate a lat/lon in user→twGPS **then do**
    i). get u's reciprocal neighbors' coordinates L_u by joining u, user→neighb. and user→latlon;
    ii). $L'_u \leftarrow$ Outlier-Removal(L_u)
    iii). calculate l_u^{L1mm} by L'_u **if** $|L'_u| \geq \lambda$;
    iv). u accepts l_u^{L1mm} **if** $GD(L_u) \leq \gamma$;
  end for

Publicly-Known Locations of Twitter Users In Twitter, there are two sources to know a user's location: profile-location or the GPS coordinates embedded in his tweets. The profile-location is often in the form of place names like "College Park, MD" and can be aligned with databases like GeoNames to decode its geographical latitude/longitude coordinates. In order to assign a unique pair of latitude/longitude coordinates, for a user having multiple GPS points available in his tweets, we compute the *L1-multivariate median* for these points and similarly check the geographical dispersion to decide whether to use this median or not. At last, for a Twitter user who has a valid profile-location as well as a valid *L1-multivariate median* calculated from his tweets, we opt to use his profile-location if this location is within γ of the median; otherwise, his two sources of location information seem to be conflicting with each other and thus wouldn't be utilized. Algorithm 2 outlines our online Twitter user geotagging procedure, which utilizes a streaming computing platform Spark Stream by maintaining 4 RDD variables [50–53]. Resilient Distributed Dataset (RDD), is a distributed memory abstraction which gives Spark the ability to perform fast in-memory map-reduce operations. IndexedRDD extends *key-value* RDD by enforcing key uniqueness and pre-indexing the entries for efficient look-up operations. In practice, RDD could be seen as a table in the database. The IndexedRDD variable for GeoNames, location→latlon, is to align the profile-location, e.g., "Boston, MA", to decode its latitude/longitude coordinates, e.g., [42.3584, -71.0598]. The RDD variable, location→user keeps a reversed index from a user to his profile-location to perform join operation in Spark. The RDD variable, user→twGPS, stores for each user, the GPS coordinates embedded in his tweets. The RDD variable, user→neighb., stores the neighbor-ships between users. Finally, the IndexedRDD variable, user→latlon, caches the geotagged user to retrieve users in a given area.

To quickly start-up our online geotagging procedure, i.e., fill in the RDD variables, we boost our algorithm with one year of tweets data collected from the Twitter Sample API statuses/sample. We discretize this live tweet stream into 23-second intervals using DStream in Spark to perform the online Twitter user geotag. For an incoming user, we first look-up his geographical coordinates in user→latlon; if this fails, then we try to align his profile-location (if provided) to GeoNames; otherwise, we retrieve a list of his reciprocal neighbors' locations to estimate his location. In this way, we successfully determined that the location of the first user in the example of Fig. 1, i.e., "@kathyswanson73" is at Boston, and thereby help us find that exampled news in Fig. 1.

3.2 Enhancing local live tweet stream

Given a geographical area, this module tries to collect as many tweets as possible from three sources: two of Twitter's statuses/filter Streaming API – "follow" and "locations",[4] and tweets filtered from another Twitter Sample API statuses/sample,[5] which returns a small random sample (usually 1%) of all public tweets. The Statuses/filter "follow" real-time returns the postings of a list of specified Twitter users (5,000 at most) as they publish tweets; while "locations" tracks the tweets falling in a geographical area either according to tweet's embedded GPS coordinates or place names.

After specifying an area A, our system first retrieves a set of Twitter user who fall inside A using IndexedRDD variable user→latlon built in Section 3.1, and collects their live tweets via statuses/filter "follow". Our experiments in Section 4.2 show that doing so dramatically increases the number of local tweets and thereby boosting the number of detected local news in our system. Meanwhile, statuses/filter "locations" is also initiated to collect tweets with embedded GPS coordinates or place names falling inside A. Finally, we also keep one's tweets captured from Twitter Sample API if he is from A. Note that as the system runs, we also keep following the newly found Twitter users belonging to A to track their real-time tweets.

3.3 Extracting locality-aware keywords

"Hot" news or events in Twitter often cause, temporally or spatially, noticeable changes (e.g., word usage and increase in the number of related-tweets) in Twitter, thereby encouraging the exploitation of anomaly detection techniques such as the discrepancy paradigm [6, 18, 26] which makes a comparison between previous data (to build up a baseline) and the newly observed data to discover anomalies. These techniques are often addressed only from the perspective of detecting anomalies in the entire set of tweets (e.g., a set of tweets collected or aggregated together either geospatially or temporally), and in so doing might miss

[4] https://dev.twitter.com/streaming/reference/post/status/filter
[5] https://dev.twitter.com/streaming/reference/get/statuses/sample

small-scale local news. Again, the data sparsity might make the problem worse. For example, to detect the news in Fig. 1 is like finding a needle in a haystack from tweets because such a story, with only 6 tweets, hardly affects the word usage pattern in that evening at Westborough, MA.

However, if we look at the news story in Fig. 1 from the view of individual people involved, such a small news poses noticeable changes in their word-usage pattern. For example, "Westborough Education Foundation Trivia Bee" are recently used words for 3 the Twitter users in that afternoon.

Therefore, given the sparsity of local news tweets, we utilize the following observations to capture such news. First, instead of looking for bursty or frequently used words with respect to a corpus of tweets from different Twitter users, we focus on the newly-used words with respect to the tweets from a single Twitter user. In other words, for a Twitter user, we are only interested in the words recently used by him. Such newly-used words correspond to the aspect of local news being "novel" as its nature of being news. Second, to reflect the aspect of local news being discussed by a limited number of people, we look for the words that are only used by a limited number of Twitter users, instead of the ones intensively used by people. Therefore, for a given tweet, we identify the words exhibiting the above two properties and call them *locality-aware keywords* in the sense that they are aware of the characteristics of local news. For example, consider the tweets in Fig. 1 where "Westborough", "Education", "Foundation", "Trivia" and "Bee" are considered as locality-aware because they are new words used by this set of people.

Inspired by this, we recognize a word (only non-stopwords) in a tweet to be locality-aware by looking at 3 measures: how many times this tweet's publisher uses it, how many other users are using it and how many tweets contain it. To ensure the local news we detect are up to date, all these measures are computed in the latest 6-hour sliding time window from the enhanced local live tweet stream. If we treat a user's tweet as a sentence, then all his tweets in time order form a document, and all the tweets in the latest time window consist of a corpus. This is different from the idea of TF-IDF used in [1, 5] which treat each single tweet as a document.

We term the above 3 measures as *term frequency*, *document frequency* and *corpus frequency*, i.e., *TF*, *DF* and *CF*, respectively. Here we assume that a word appears at most once in a tweet (or counts only once if more), which is reasonable given the 140-character limit. For a given word w in the tweet posted by user u, these measures are computed as: $TF_w = |T_u \cap T_w|$, $DF_w = |U_w|$, and $CF_w = |T_w|$. T_u denotes the tweets of user u, T_w denotes the tweets containing word w, U_w denotes the users who recently used the word w. Our heuristic is that, in order for a word w to be locality-aware, it should have a smaller TF_w (i.e. how many times it has been used recently by a Twitter user), which indicates that word w might be newly used by this user and thereby captures a news's "novelty"; DF_w (i.e., how many Twitter users have been using word w recently) should have a limited range (like $[3, \frac{|U_S|}{20}]$ specified in parameter settings in Section 4.4.1, where U_S is the set of Twitter users), to reflect the local news's characteristic of having a limited spread among people; and also CF_w should be small to avoid commonly used words like "day" and "people" etc. In our implementation, to account for the heterogeneity of the rates of publishing tweets for different users and for the number of tweets collected at different times and different places, we also use the relative frequencies of TF_w and CF_w, i.e., $TF'_w = \frac{|T_u \cap T_w|}{|T_u|}$, $CF'_w = \frac{|T_w|}{|T_S|}$, where T_S represents all current tweets. The constraints for TF, TF', DF, CF and CF' — denoted by R_{TF}, R_{DF} and R_{CF} — are discussed in Section 4.4.1.

Algorithm 3 Online Extracting Locality-Aware Keywords and Online Clustering to Detect News.

Input: the latest 6-hour sliding window in enhanced local live tweet stream – S; the locality-aware constraints – R_{TF}, R_{DF} and R_{CF}; the threshold values m, n and r

Output: news, i.e., clusters of tweets

1: load hash variables uid→tids, word→uid, word→tids, tid→uid, tid→cid, tid→r.t.
in last time window;

2: **while** true **do**

a. pull a tweet from S, get its non-stopword tokens W, tweet id t, and user id u;

b. Locality-Aware Keywords $W_L \longleftarrow \emptyset$

c. **Extracting Locality-Aware Keywords Procedure:**
for each word $w \in W$ **do**

i). calculate TF_w, TF'_w, DF_w, CF_w, CF'_w from uid→tids, word→uid, and word→tids;

ii). $W_L \longleftarrow W_L \cup \{w\}$ **if** TF_w, TF'_w, DF_w, CF_w and CF'_w meet with the constraints of R_{TF}, R_{DF} and R_{CF};

iii). update word→uids, word→tids by inserting w and its corresponding u and t;

end for

d. update uid→tids by inserting u and t;

e. update tid→r.t. by inserting t and retweet number;

f. **Online Clustering Procedure:**
for each $W_L^m \in$ subsets of W_L with size m **do**

i). retrieve Q – the ids of tweets containing all words in W_L^m, from word→tids;

ii). retrieve the user set U in Q using tid→uid;

iii). **continue if** $|U| < n$;

iv). extract the largest group of tweets C from Q with the same cluster id c
(a null c means that the tweets in C haven't formed a cluster yet);

v). calculate RT_C, which is the sum of retweet number of each tweet in C, using tid→r.t..

vi). **if** $|C| \geq \lceil \frac{|Q|}{2} \rceil$ **and** $|U| \geq \lceil r * RT_C \rceil$ **then**
$c \longleftarrow$ generate a new id **if** c is null;
assign t to cluster c in hash tid→cid;
report t as a *news* tweet to system UI;
break;
end if

end for

g. Remove obsolete tweets from uid→tids, word→tids, tid→uid, tid→cid and update word→uids;

end while

3.4 Online clustering to detect news

As presented in Algorithm 3, we take into account the following two aspects to group tweets together. First, the tweets need to share at least a number m of locality-aware keywords to be grouped together. Second, at least n different Twitter users must exist in a cluster. Existing methods usually neglect the importance of these two aspects. For example, GeoBurst [7] measures the semantic similarity between two tweets by performing random walks on their keyword co-occurrence graph to calculate the average probability that one tweet reaches another. However, without requiring a minimum number of keywords in a tweet, two tweets containing and sharing very few keywords could be mistakenly considered semantically coherent even if they are not on the same topic. In addition, TwitterStand [1] groups tweets together as long as they are similar enough in the TF-IDF vector space and in so doing, might form noisy clusters out of a single Twitter user's repeated tweets.

Therefore, in our method, to cluster an incoming tweet, we first retrieve a set of tweets sharing at least m locality-aware keywords. If these tweets were contributed by less than n Twitter users, or the majority of the tweets don't locate in the same cluster, then we don't group this new tweet and try another set of m locality-aware words. We also require that a news spreads among more local people. In Twitter, the spread extent of a tweet is provided by its retweet number, i.e., how many other Twitter users retweet it. We now define, for a given news cluster C, its spread extent RT_C to be the sum of the retweet number of each tweet in it. And the local spread ratio $spread_{local}$ is computed by $\frac{|U|}{RT}$, where U is the users contributing to C. In our experiments, we set $spread_{local} \geq r = 0.4$ to account for the local tweets that we might not capture.

The details of calculating the above measures are presented in Algorithm 3. Generally, Firefly uses a one-shot process, meaning that once a tweet is added to a cluster, it remains there forever. We will never revisit or recluster the tweet, which is desirable for real-time detection of news from a local live tweet stream. We don't incorporate additional care of the aspects from geographical dimension or temporal dimension as they are implicitly reflected in the procedure enhancing the local live tweet stream and its 6-hour sliding window.

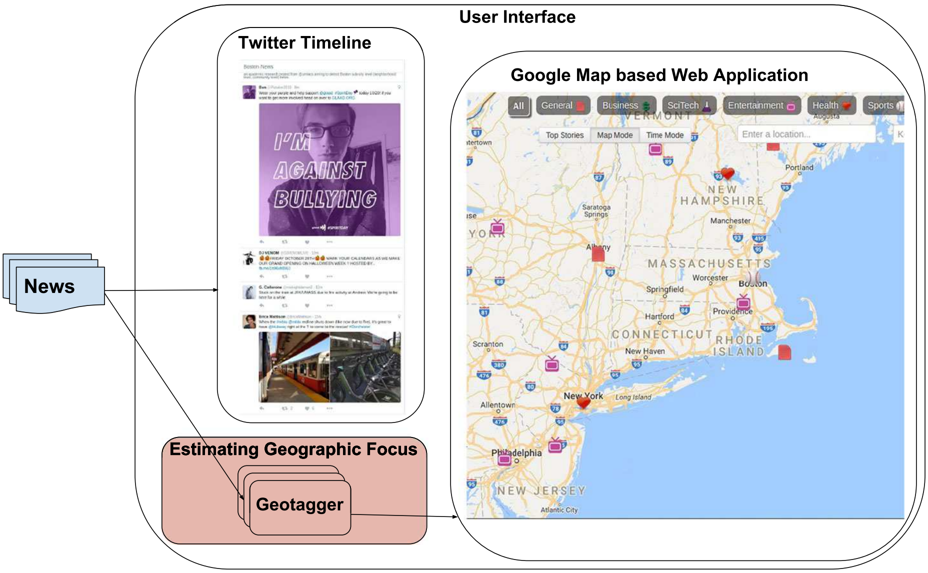


Fig. 4 System user interface

3.5 System user interface

As shown in Fig. 4, our user interface consists of two parts: a Twitter Timeline[6] and a Google Map based Web Application [1]. The Twitter Timeline allows a user to view a list of tweets collected for various purposes, such as real-time monitoring of a Twitter user's updates or searching for the latest tweets on a specific topic. Therefore, in order to demonstrate the latest news that we detect in real-time, a Twitter Timeline[7] is created via Twitter Collections API, which is very convenient for other Twitter users to view and even subscribe to. Note that the Collections API only allows for a user to retain a few thousand of tweets and automatically delete the oldest ones if it has too many tweets.

To display the events that we detect on the Google map-based web application, we utilize a procedure to estimate the geographical focus for a news cluster in [1]. This procedure, by making use of both the geographical information in tweet content and the source location of the users in an event cluster, computes a geographical focus as a whole by ranking the geographic locations mentioned in the cluster. After geotagging an event cluster, the Google map-based web application displays a marker for this event at its geographical coordinates.

4 Experiments

4.1 Online processing settings and efficiency

Our system adopts sliding time window techniques to meet the demand for online processing of a live tweet stream. The experiments are evaluated on a Spark cluster of 5 computing nodes where each node has two 6-core Intel Xeon E5-2620 v3 CPUs and 128GB of RAM.

For Online Geotagging, we utilize Spark Stream to discretize the live tweet stream from the Twitter statuses/sample API into intervals of 23 seconds, which is the average time to accumulate 1000 tweets. Similarly, a 6-hour sliding time window is applied on the enhanced local live stream for locality-aware keyword extraction and online clustering. The 6-hour window size is intuitively set in recognition of the fact that television media usually has four times of a locally-oriented news broadcast in one day. The day of Jan 16, 2017 is chosen to evaluate our system for news detection with respect to the Boston metropolitan area i.e., the rectangle area [42.008339, -71.803026, 42.732923, -70.577545].

In our experiments, we find the major overhead is the Boosting Phase in Algorithm 2, which takes around 76 minutes to finish. But this procedure runs only once to start up the system and does not affect the timeliness of subsequent procedures. After the Boosting Phase, the online geotagging procedure takes an average of 3 seconds to process 1,000 tweets from the Twitter statuses/sample API, and geotags an average of 47 unknown-location Twitter users per second. Afterwards, Algorithm 3 processes 70 tweets per second on average (which is also the approximate arriving rate of tweets in enhanced local live stream) and reports about 3 tweet clusters per minute.

[6] https://support.twitter.com/articles/164083

[7] https://twitter.com/bostonnewslocal/timelines/878280225074950144

4.2 Twitter user geotagging via spark

4.2.1 Boosting dataset

To boost the startup of geotagging Twitter users, we utilize a set of tweets collected between 09/2015 and 09/2016. This dataset consists of 2,876,822,081 tweets, 102,382,292 users and 824,303,126 pairs of neighbor-ships. Among these users, 31,250,047 have valid location source (successfully aligning profile-location to GeoNames or having embedded GPS coordinates) and are used to build-up the variable user→latlon. Accordingly, variable user→neighb. builds from the extracted neighbor-ships. Filtering down to only reciprocal neighbors, we have a reciprocal graph of 24,946,962 vertices (8,787,152 of them have lat/lon coordinates) and 54,550,871 bidirectional edges.

4.2.2 Effectiveness

In lack of a ground-truth for Twitter users' locations, we exploit the *boosting dataset* to evaluate the effectiveness on coverage and accuracy. Specifically, for the 8,787,152 Twitter users with lat/lon coordinates in the reciprocal graph built in Section 4.2.1, their lat/lon coordinates are obtained from their profile-location or GPS coordinates in their tweets, and are thus treated as ground-truth. We then perform a leave-p-out validation by randomly sampling 10% (i.e., 878,715) of these Twitter users to evaluate the coverage and accuracy. The coverage is to calculate how many Twitter users in the sampling set would get geotagged, while accuracy is to calculate the mean distance error between the ground-truth and their estimated location.

To geotag the 10% users, we again utilize the sub-procedure iii) in the Online Geotagging of Algorithm 2. Our experiment shows that with $\gamma = 100$ km, $\lambda = 2$, 13.6% (i.e., 119,505 out of 878,715) test users get geotagged with a mean error of 228.66 km and a median of 27.93 km, which as shown in [2], is accurate at city-level for majority of test users.

Effect of outlier removal To evaluate the effect of outlier removal, we now exclude the step of outlier-removal in Algorithm 2 to geotag the 10% test Twitter users with γ fixed at 100 km and λ at 2. This brings us a lower 7.1% coverage with a larger mean error of 279.81 km, showing that removing outliers significantly increases the chances for test users to get successfully geotagged while without compromising accuracy.

Effect of λ (the minimum number of reciprocal friends with valid locations) We first plot the distributions on the number of reciprocal friends of these 10% Twitter users in Fig. 5, as well as the ones with locations and the ones that have survived from the outlier removal

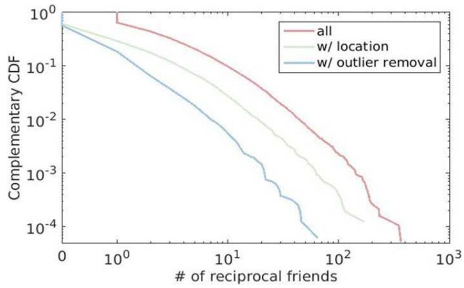


Fig. 5 CCDFs of reciprocal friends

Table 1 Effect of λ

λ	Coverage (%)	Mean error (km)
1	53.3%	7900.54
2	13.6%	228.66
3	6.5%	251.34
4	3.9%	213.62
5	2.5%	191.65
10	0.6%	234.72
20	0.1%	187.16

step. Figure 5 shows that lots of the Twitter users have very few reciprocal friends that have locations. In such a sparse reciprocal graph, it may not be fair to decide the location for a Twitter user only based on very few of his friends locations. To avoid generating noisy location estimations, a Twitter user is not going to be geotagged until the number of his reciprocal friends having locations exceeds the required minimum.

To investigate the sensitivity of the minimum constraint parameter λ in Algorithm 2, we fix $\gamma = 100$ km and use different values of $\lambda = \{1, 2, 3, 4, 5, 10, 20\}$ for the 10% sampling test users and list the corresponding coverage and mean errors in Table 1. The results show that although $\lambda = 1$ is able to geotag more than half of the test users, it brings about an acceptably large error; $\lambda = 2$ seems to reach the best trade-off point between coverage and accuracy; while larger λ values have similar accuracy, they have relatively low coverage.

4.3 Enhanced local live tweet stream

At the start of the day on Jan 16, 2017, 176,007 users are found in the input Boston bounding box. Among them, 101,409 provide valid location source (profile-location or GPS), and the remaining 74,598 are geotagged using Algorithm 2. Following these two sets of users to track their real-time postings comprises of the two sources of Streaming API w/ "follow" I and II as listed in Table 2, respectively.[8]

Table 2 first shows how many local tweets (i.e., the tweets that fall in the given area or are published by people there) as well as how many cluster tweets (i.e., the tweets that compose the detected clusters) each source contributes to our enhanced local live tweet stream. We collected a total of 4,800,345 tweets from the Boston area during a 24-hour period. Among which, Twitter Streaming API statuses/filter "follow" (I and II) contributes the most, by making up 98.27% of all the tweets in the enhanced local live tweets, while the other two sources only output a very small amount of local tweets. Similarly, regarding the tweets comprising the clusters, 93.66% come from source Streaming API statuses/filter "follow". For example, all the tweets related to the news in Fig. 15 are in source "follow". More importantly, Table 2 further shows that tracking Twitter users who don't have valid location sources also make significant contributions just like tracking the users with valid location sources. This reinforces the important role that the online Twitter user geotagging procedure plays in our system.

In addition, Table 2 also lists the number of "Involved" news (i.e., how many news a source's tweets have participated in forming) and the number of "Exclusive" news (i.e., how many news a source's tweets have exclusively formed, in other words, these news are

[8]Multiple API tokens are used because one only follows up to 5000 users.

Table 2 Contributions of different local live tweet sources

Source	# of tweets		# of news	
	Local tweets	Cluster tweets	Involved	Exclusive (acc. %)
Sample API	6,182	638	167	21 (35.5%)
Str. API w/ loc.	76,983	2,123	359	76 (52.9%)
Str. API w/ fol. I	2,986,291	23,120	2,489	1,241 (58.5%)
Str. API w/ fol. II	1,730,889	16,654	1,857	609 (52.7%)
Total	4,800,345	43,535	N/A	N/A

formed by tweets only from this source), along with its accuracy of positive local news (the accuracy evaluation method is detailed in Section 4.4.5). The result shows that the majority of news events are generated using the tweets in Streaming API statuses/filter w/ "follow", indicating that by tracking local Twitter users, our method is able to find much more news than solely using the Twitter's publicly available tweet streams.

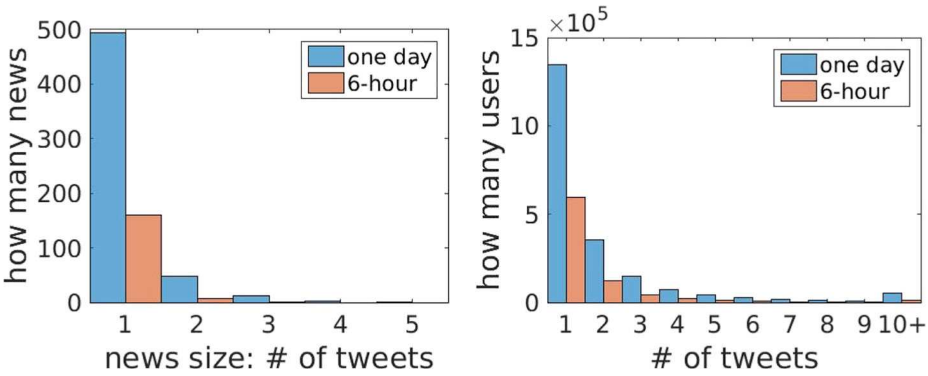
4.4 Local news detection

In this section, we evaluate the performance of our system on detecting news from the enhanced local live tweet stream. In the evaluation, we first present several positive and negative examples for illustrative case studies and then compare our system with baseline approaches using *mutual recall* and *precision*. Mutual recalls are evaluated between our system and a set of local news media agencies, together with a few baseline approaches. As for precision, we recruited 3 volunteers to individually judge the detected news and collect the results using the strategy of majority votes.

4.4.1 Parameter settings

Note that although some of the following parameter settings depend on the specific input city, they are simply statistics and easy to infer for other places. There are 3 constraints for a word to be locality-aware: R_{TF}, R_{DF} and R_{CF}. For R_{TF}, the main goal is to capture a local news's nature of being new and to reflect a person's word-usage anomaly, by requiring both TF and TF' to be small (of course, at least greater than 0). This means that, an upper boundary needs to be imposed on TF. To obtain an empirical value of this, we collect the tweets posted by the Twitter accounts listed in Table 4 (note that the Twitter account of @fox25news has changed to @boston25 in April 2017), and perform an analysis, for each individual agency, of how many of its tweets are about the same news. The results, presented in Fig. 6a, show that an agency usually tweets only 1 or 2 tweets (5 at most) about the same news. The situation is similar when the time period narrows down to a 6-hour (e.g., from 15:00 to 21:00). We therefore set the upper bound of TF to 5. Figure 6b reminds us that this value could work for most of Twitter users as they usually post less than 10 tweets, either in one day or in a 6-hour time window, This value, however, seems too strict for Twitter users who publish 10 or more tweets and perhaps keep posting updates on the same news event. We therefore turn to TF' to relax the constraint of TF, and set a threshold value of 0.3 for TF'. To summarize, we have $R_{TF} := (|T_u| < 10 \wedge TF \leq 5) \vee (|T_u| \geq 10 \wedge TF' \leq 0.3)$.

R_{TF} alone, however, is not enough because it would mark most of the words for most of Twitter users as locality-aware. We further utilize DF to explore another characteristic



(a) News size by news agency (b) Tweet number by Twitter user

Fig. 6 Histograms of # of tweets. **a** Histogram of # of tweets in a news by each individual news agency. **b** Histogram of # of tweets posted by each individual Twitter user

of local news: being "limited spread". Recalling the fact that one-fifth of the population are active Twitter users, we set $R_{DF} := 3 \leq DF \leq \frac{|U_S|}{20}$, where U_S are all the users in the time window S. Our argument is that when $DF = 3$, there might be an equal number of users reporting the same activity in Twitter. This further indicates that in reality, there might exist an ongoing news event that involves 15 people. Likewise, we set the upper boundary to 1% of the population, which is around $\frac{|U_S|}{20}$. The distribution of detected cluster size in Fig. 16a further validates our assumption.

Finally, there is an additional constraint R_{CF} to get rid of commonly used words. Our analysis on the CF's of most common non-stopwords in English shows that they have a min CF of 0.57% (max: 2.7%, mean: 1.6% and median: 1.8%) . Also considering that the average number of tweets published by a Twitter user is around 2 (e.g., in Fig. 6b, 2.30

Tweets in the news	Links
Four #Lawrence men arrested with fighting roosters in their car. Released on $500 bail « #wbz	https://twitter.com/chrisWBZ/status/820941754081939456
FROM THE NEWSROOM: STURBRIDGE, Mass. (AP) — Four Lawrence, Massacusetts men were arrested and three roosters were	https://twitter.com/WINYRadio/status/820963426876932096
Four Lawrence MA men arrested on Cockfighting Charges. These "Bad Hombres" had a garbage bag of Cocks stashed in their car. It's a Thing!!	https://twitter.com/carminelbo/status/820962866111115264
Four Lawrence men arrested on animal cruelty charges after police say injured roosters found inside car	https://twitter.com/AlbanMurtishi/status/820956214444040192
Four Lawrence men arrested on cockfighting charges	https://twitter.com/MyBostonNews/status/820884931974987776

Fig. 7 Positive example #1: a local event about local men being arrested for rooster fighting

Tweets in the news	Links
Worcester man charged with human trafficking sexual assault of 14yearold	https://twitter.com/ThatNigga_Nicee/status/821052208066260993
From the Worcester Police Department RT Police Arrest Male for Human Trafficking and Sexual Assault	https://twitter.com/WorcesterSun/status/821048899867725824
WNT Human Trafficking Arrest January 17th 2017	https://twitter.com/WorcNewsTonight/status/821207900467105796
Worcester man charged with human trafficking sexual assault of 14yearold	https://twitter.com/MyBostonNews/status/821051062614421504
Id say less sexual assault scandals Still plenty of scandals with the whole human rights bit	https://twitter.com/killersim00/status/821064522534440961

Fig. 8 Positive example #2: a local event about local men being arrested for human trafficking

and 1.82 for one day and 6-hour) and DF's upper bound, we set the upper bound of CF to $\frac{|U_S|}{10}$. Therefore, R_{CF} is set as $R_{CF} := CF \leq \frac{|U_S|}{10} \wedge CF' \leq 0.57\%$, which helps us to successfully recognize words like “trump”, “martin”, “luther”, “day” and “people” as not locality-aware.

We then have 3 more threshold values to set for online clustering in Algorithm 3. For the least number of overlapping words between two tweets to cluster together, we set $m = 5$ because it is usually large enough to cover a news's “who”, “what” and “where” information, e.g., the bold words in the example event of Fig. 1. In our experience, a larger m makes

Tweets in the news	Links
FYI subway and buses are running on a Saturday schedule Commuter rail and Silver Line on a weekday schedule	https://twitter.com/NVermaNBCBoston/status/820955164421091328
Holiday schedule Subways busses on a Saturday schedule Commuter Rail Ferries on a weekday schedule	https://twitter.com/KBBostonTraffic/status/820928209286008832
Holiday schedule Subways busses on a Saturday schedule Commuter Rail Ferries on a weekday schedule	https://twitter.com/dougmeehan/status/820929890379853825
The Commuter Rail is on a regular weekday schedule today	https://twitter.com/MBTA_CR/status/820976397472645120
today subways on wknd schedule weekday schedule Commuter rail and commuter boat on weekday schedule	https://twitter.com/NicholeDWBZ/status/820994250586411013

Fig. 9 Positive example #3: a local news about subway schedule

Tweets in the news	Links
Tie game Westborough vs Leominster 3 minutes left Go Rangers	https://twitter.com/NicoleDSullivan/status/821088306322698241
Westboro Varsity Boys hockey up 21 against Leominster end of the second	https://twitter.com/WestboroRangers/status/821084911738257408
Tie game Westborough vs Leominster 3 minutes left Go Rangers	https://twitter.com/WestboroRangers/status/821089922715107330
Westboro Varsity Boys hockey up 21 against Leominster end of the second	https://twitter.com/tgsports/status/821082878872289280
HOCKEY Westboro 2 Leominster 2 Tumenas scores late to help Blue Devils earn draw with Rangers	https://twitter.com/tgsports/status/821178474257874944

Fig. 10 Positive example #4: some updates on a local sports event

clustering tightly cohesive yet might split the same news story into several clusters; while a smaller m might not fully reveal a story's own trait and groups non-related things together. To be consistent with R_{DF}, we set the least number of people in a cluster $n = 3$. At last, although we require that a local news should have more local people talking about it, other than the people from outside world, we set the local spread ratio threshold $r = 0.4$ to deal with the tweets we might miss.

Tweets in the news	Links
a representative that boycotts the peaceful transition of power is not a representative of the people	https://twitter.com/BruSox/status/820978748069703680
Shameful partisan display Inauguration celebrates PEACEFUL TRANS of POWER big part of what makes USA what we a	https://twitter.com/poochieOFS/status/821057164848623616
a medal of freedom winner who does not respect the transition of power incredible	https://twitter.com/JerRobbins1/status/821068987761131520
remarks during what should be peaceful transition of power right before inauguration just to raise money along w	https://twitter.com/PoliticianBust/status/820961602166001664
Shameful partisan display Inauguration celebrates PEACEFUL TRANS of POWER big part of what makes USA what we a	https://twitter.com/bostonnews002/status/821061037101301762

Fig. 11 Negative example #1: a national news about presidential inauguration

4.4.2 Illustrative cases

We select several positive and negative examples of local news detected in our system and present them in Figs. 7, 8, 9, 10 and 11, 12, 13, 14, respectively. Each example is described by 5 representative tweets, along with their links in Twitter. The 5 tweets are selected by having the most non-stopwords, retweet numbers and overlapping words with each other.

Figures 7, 9, 10 and 8 report 4 cases of local news on the topic of local events, crimes, transportation and sports, accordingly. These topics could be beneficial for the daily life of local people. For example, local events let local people learn of what are happening in their local communities; local crime reports help people pay attention to their living surroundings and their own safety; local sports games are usually entertaining activities for the local people; and the updates on local transportation schedules and traffic bring convenience to people's life activities especially when they make outdoor plans.

These positive examples also provide demonstrations on the locality-aware keywords. For examples, the words "four", "lawrence", "men", "arrested" and "roosters" in the example of Fig. 7 and words "human", "trafficking", "sexual", "assault" and "worcester" in the example of Fig. 9. Figures 11, 12, 13 and 14 report 4 cases of falsely identified local news. Comparing to the tweet clusters representing positive local news, the topics on the negative local news are various. For example, Fig. 11 is reported to be a local news. It, however, actually refers to the national news about presidential inauguration. The major reason of this tweet cluster being picked up is that: although this news is a national one, it attracts relatively limited attention from local Boston people. This results in only a small number of local tweets covering this topic. In this case, it bears similar characteristics with local news by our definition and is thus falsely reported as a local news. It worth mentioning that such false positive scenarios are very common as shown in the evaluation results in Section 4.4.5, making it a typical error case of our system. In the future, we plan to build real-time national news database to help us filter out this type of error case.

Tweets in the news	Links
Former pro wrestler Jimmy Superfly Snuka dies at 73	https://twitter.com/martyx56/status/820971709796876289
Former Wrestler Jimmy Superfly Snuka Passes Away At 73 Dwayne The Rock Johnson Pays Touching Tribute	https://twitter.com/I_AM_Finance/status/820997217821474816
The world lost a great one Superfly Jimmy Snuka I will never forget Roddy Piper smashing a coconut on your dome	https://twitter.com/RockingRoger/status/820944522964246528
THEY REALLY DID NOT DO ANYTHING FOR JIMMY SUPERFLY SNUKA NO TRIBUTE WOW THATS RUDE	https://twitter.com/Tommy516Tommy/status/821161873861120000
RIP Jimmy Superfly Snuka	https://twitter.com/TheReal_JDavis/status/821036370391076865

Fig. 12 Negative example #2: a specific news about a former wrestler passing away

Tweets in the news	Links
Its been exactly 98 years since a giant wave of molasses killed 21 people in Boston	https://twitter.com/OAHSTigers/status/820968599502327810
been exactly 98 years since a giant wave of molasses killed 21 people in	https://twitter.com/MrBsmith617/status/820957637370056705
Its been 98 years since a giant wave of molasses killed 21 people in Boston	https://twitter.com/GregCaseyMA/status/821027746520453120
Its been exactly 98 years since a giant wave of molasses killed 21 people in Boston	https://twitter.com/dcd728/status/820947861772894208
been exactly 98 years since a giant wave of molasses killed 21 people in	https://twitter.com/MaryJoKurtz/status/820937346904428544

Fig. 13 Negative example #3: some tweets mentioning about a historical disaster in Boston

Similar to Fig. 11, the example in Fig. 12 can also been as a national news to some extent because it tells about the passing away of a famous former wrestler. The difference is that Fig. 12 has a specific topic focus on wrestling and thereby is more likely to limit its spread among people from or interested in wrestling.

Different from Figs. 11 and 12 which are about national news, the negative example in Fig. 13 is not about any current news or events in Boston but commemorating a historical event happened 98 years ago in Boston.

Tweets in the news	Links
Todays hot soups 116 Turkey Chili Lentil Grilled Chicken Corn Chowder	https://twitter.com/atkinsfarms/status/821024347653599232
RT special turkey Brie melt on wheat wrap Soups chowder beef stew apple butternut squash	https://twitter.com/mvtweets/status/821035542552911872
Serious Eats How to Make Chicken Scarpariello Italian SweetandSour Chicken With Sausage	https://twitter.com/Cakescupcakes/status/821025007107117056
RT special turkey Brie melt on wheat wrap Soups chowder beef stew apple butternut squash	https://twitter.com/bostonnews002/status/821039069237051393
Todays soups are Beef Stew Sweet Italian Sausage Hungarian Mushroom and Chicken Tom Yum	https://twitter.com/idylwildefarms/status/820985930559451136

Fig. 14 Negative example #4: some tweets talking about food

At last, although the tweets in Fig. 14 seem to be coherent with respect to its topic, it does not form a local news because it simply talks about food specials whose recipes happen to share certain words.

4.4.3 Local news media agencies and baseline approaches

Reputable local news media agencies We select 9 Boston local news agencies, as listed in Table 3 in the form of "@ScreenName", to collect their news tweets as a groud-truth dataset to compare with. The news stories in the news agencies come from two parts: tweets posted by their accounts and articles published in their websites. The articles are collected by crawling their websites every 5 minutes listed in Table 3. Due to copyright constraint, we only keep an article's *title, url* and *publish datetime*.

As most of the tweets posted by these agencies are of good quality, we perform a simple cluster algorithm to extract news from them. That is, for a single news agency, as long as any two of his tweets share 5 non-stopwords, we group them together. The value of 5 is heuristic, by accounting for the number of words to specify a story's "who, what and where". As presented in Table 3, the amount of tweets and the amount of news these agency cover are various, with "@BostonGlobe" being the most active and "@metroBOS" the least active.

Baseline approaches We also compare our method with the following four baseline approaches listed.

- *TwitterStand*: TwitterStand [1] groups news tweets into cluster of tweets to form news stories using a TF-IDF based similarity metric. In the experiments, the clustering similarity threshold ϵ is set to 0.8. It is worth mentioning that their concepts of TF and DF are different from ours in the sense that they treat each single tweet as a document while we treat all of a user's tweets as a document and each of his tweets as a sentence.
- *TwitterStand-3*: By default, TwitterStand only reports a cluster as a news story if it has more than 10 tweets. In this setting, we relax the minimum number of tweets to 3, out of the consideration of fairness for TwitterStand to be able to detect news of small scale.
- *EvenTweet*: EvenTweet [18] first identifies temporal bursty keywords (using a Gaussian distribution based discrepancy paradigm) and spatial local keywords (using the entropy of a word's spatial distribution) and then clusters them together according to their spatial density distribution. The spatial density distribution is calculated based on a $N \times N$ grid tessellation. We set $N = 50$ in our experiments. The temporal bursty keywords are identified using a Gaussian distribution based discrepancy paradigm, while spatial local keywords identified using the entropy of a word's spatial distribution on a regular grid tessellation.
- *GeoBurst*: GeoBurst [7] first generates candidate events by identifying pivot tweets based on geographical and semantic similarities and then ranks the candidates according to their spatiotemporal burstiness to filter out noisy ones. Geographical similarities between tweets are calculated by a kernel function on their spatial distance, while the semantic similarities are calculated by performing a random walk procedure with restarts on tweets' keyword co-occurrence graph. In our experiments, we adopt the default settings in their method, i.e., the spatial distance kernel bandwidth is set to 0.01, the random walk restart probability and similarity threshold are set to 0.2 and 0.02, respectively.

As summarized in Section 2, TwitterStand (or TwitterStand-3) is an event-anchored method and therefore is fed with the same enhanced local live tweet stream in Firefly to

Table 3 The 9 reputable Boston local news agencies

Agency name	Twitter			Newspaper	
	Screen name	Tweet num.	News num.	Website URL	Article num.
7News Boston	@7News	73	61	http://whdh.com/news/local/	15
boston.com	@BostonDotCom	29	21	https://www.boston.com/tag/local-news	4
The Boston Globe	@BostonGlobe	156	128	http://www.bostonglobe.com/?refresh=true	22
Boston Herald	@bostonherald	106	95	http://www.bostonherald.com/news/local_coverage	21
CBS Boston	@CBSboston	64	52	http://boston.cbslocal.com/category/news/	83
FOX25	@fox25news	85	64	http://www.myfoxboston.com/news/local	13
Globe Metro	@GlobeMetro	12	11	http://www.bostonglobe.com/metro	12
Metro Boston	@metroBOS	26	15	http://www.metro.us/boston/news/	22
WCVB 5	@WCVB	143	110	http://www.wcvb.com/local-news	27

detect news, while the last two are location-anchored methods which only take geotagged tweets (with embedded GPS coordinates) as input. In the 4,800,345 tweets we collected, 33,966 tweets are geotagged (Streaming API w/ follow: 23,810; Streaming API w/location: 10,101; Sample API: 55) and chosen as the input to EvenTweet and GeoBurst. Note that the geotagged tweets are less than the total tweets obtained from Streaming API w/location because this API also returns non-geotagged tweets containing place names that fall in the given query area.

By default, EvenTweet represents each cluster as a group of keywords, and we retrieve the tweets from which the keywords are extracted to represent its clusters to be consistent with other methods. To maximize the number of potential news detected in EvenTweet and GeoBurst, a cluster in them is selected to be output as long as it has at least 3 tweets.

4.4.4 Mutual recalls

The mutual recalls are computed by examining how many news in the news agencies or baseline approaches have been found by our system and *vice versa*. We claim a news cluster $c_{\mathbf{X}}$ in agency $\mathbf{X}$ recalls a news cluster $c_{\mathbf{Y}}$ in agency $\mathbf{Y}$ if there is a tweet in $c_{\mathbf{X}}$ and another tweet in $c_{\mathbf{Y}}$ that share at least 5 non-stopwords. The results are summarized in Table 4, in which a news agency's "@Screen Name" is to represent its tweets news. Also, to make the table compact, we give each agency an order denotation in the column headers. Below the column headers are the number of news found in an agency or our system Firefly. So for a cell, it shows how many news row $\mathbf{X}$ covers over column $\mathbf{Y}$.

Table 4 shows that Firefly achieves high recalls against most of news agencies. For example, we successfully detect news like "Stabbing Reported at Stoughton Home of UMass Boston Chancellor", "Dog killed by coyote in Gloucester, police issue warning" and "A woman caught in the line of fire in Lyn" etc which are also reported by "@7News". In contrast, a very large portion of news in Firefly don't receive coverage from any of the listed news agencies, e.g., "There is a growing collection of lonely hand warmers at Fallon Field in #Roslindale", "Hockey star Kacey Bellamy took a break from prepping for the 2018 Winter Olympics to chat with @BrooksSchool girls hockey team today!" and "Just a portion of the many people that volunteered today to build STEM kits for Boston schools" etc. This confirms the effectiveness of our design of enhancing local live tweet stream and extracting locality-aware keywords.

The result is in accordance with our observation that there would be lot more news happening in an area than reported locally [54], and is consistent with our expectation because we try to identify various kinds of news, activities and news like missing pets, sales events, concerts and farmer's market etc., while local news agencies usually publish news of greater public interest.

In contrast, the default settings of TwitterStand have much lower recalls across the 9 local news agencies. Although relaxing its cluster size to have minimum of 3 tweets yields many more clusters, it doesn't yield clearly higher recalls. We conjecture that in doing so, TwitterStand-3 is reporting many small clusters for the same news due to the fragmentation problem in its online clustering [1]. For example, the 409 news of TwitterStand are covering 1,607 news of TwitterStand-3. This also explains TwitterStand-3's extremely asymmetric mutual recalls over the local news agencies. In contrast, Firefly's locality-aware keywords based clustering is more reliable by finding word-usage anomaly from the perspective of a Twitter user instead of a tweet itself.

It comes as no surprise that EvenTweet and GeoBurst, both of which only run on sparsely available geotagged tweets, have low recalls across the local news agencies too. This is

Table 4 The mutual recalls between firefly, baseline approaches and the 9 reputable Boston local news agencies

	Order	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
# of news		3364	409	2331	184	179	61	21	128	95	52	64	11	15	110	15	4	22	21	83	13	12	22	27
Firefly	A	3364	305	1213	135	164	48	21	85	32	41	49	6	10	69	11	4	16	6	20	9	7	4	10
TwitterStand	B	200	409	1607	71	46	7	5	13	15	4	4	0	2	12	1	0	5	2	11	1	0	1	3
TwitterStand-3	C	215	395	2331	51	66	8	6	16	19	4	5	0	5	15	2	0	6	2	13	1	1	2	5
Eyewitness	D	236	218	292	184	151	6	6	3	3	15	2	0	3	15	1	0	1	1	6	1	2	3	1
GeoBurst	E	132	64	202	126	179	7	1	3	3	7	5	0	3	5	0	0	0	2	3	1	0	1	3
@7News	F	49	73	212	2	13	61	3	4	7	2	6	0	1	9	13	0	2	1	3	0	1	2	1
@BostonDotCom	G	21	22	47	1	1	3	21	6	5	0	2	1	0	3	0	3	3	1	2	1	1	1	1
@BostonGlobe	H	85	83	210	2	6	4	6	128	4	3	1	2	0	5	0	2	12	0	4	0	6	3	0
@bostonherald	I	38	82	179	1	3	7	5	4	95	0	7	1	1	5	2	2	2	9	1	1	0	0	1
@CBSboston	J	41	64	149	2	8	2	0	3	0	52	1	0	0	4	0	0	1	0	7	0	1	0	0
@fox25news	K	49	23	64	2	5	6	2	1	7	1	64	0	2	2	4	1	0	2	0	9	0	0	1
@GlobeMetro	L	6	0	0	0	0	0	1	2	1	0	0	11	0	0	0	0	4	0	0	0	3	0	0
@metroBOS	M	11	27	62	2	5	1	0	0	1	0	2	0	15	0	1	0	0	0	0	1	0	3	2
@WCVB	N	69	95	217	7	13	9	3	5	5	4	2	0	0	110	1	0	1	0	4	0	0	1	13
7News Boston	O	12	2	2	1	0	13	0	0	2	0	4	0	1	1	15	0	1	0	0	0	1	1	1
boston.com	P	4	0	0	0	0	0	3	2	2	0	1	0	0	0	0	4	1	1	0	0	1	0	0
The Boston Globe	Q	17	30	72	1	0	2	3	12	2	1	0	4	0	1	1	1	22	0	2	0	4	1	1
Boston Herald	R	7	6	15	1	3	1	1	0	9	0	2	0	0	0	0	1	0	21	1	1	0	1	1
CBS Boston	S	20	147	352	6	3	3	2	4	1	7	0	0	0	4	0	0	2	1	83	0	0	1	0
FOX25	T	13	1	8	1	2	0	1	0	1	0	9	0	1	0	0	0	0	1	0	13	0	0	0
Globe Metro	U	9	0	1	0	0	1	1	6	0	1	0	3	0	0	1	1	4	0	0	0	12	1	1
Metro Boston	V	4	13	30	6	3	2	1	3	0	0	0	0	3	1	1	0	1	1	1	0	1	22	2
WCVB 5	W	10	25	43	1	3	1	1	0	1	0	1	0	2	13	1	0	1	1	0	0	1	2	27

essentially because geotagged tweets cover very limited news in our dataset. For example, none of the news tweets posted by local news agencies contain geotagged tweets. Similarly, in all the tweets clusters generated by our system Firefly, only 633 of them contain geotagged tweets and only 107 of tweets clusters are formed by only geotagged tweets. This shows that by utilizing non-geotagged tweets, we are able to detect much more local news than methods EvenTweet and GeoBurst and further reinforces the importance of enhancing local live tweet stream by finding and tracking local Twitter users.

Another factor contributing to the low recalls of TwitterStand-3 might be its classifier step which throws away more than half of the tweets (68.9%). To verify this, we omit the classifier in TwitterStand-3 and the resulting clusters are able to recall 1,135 ones detected in Firefly. It, however, outputs a total of as high as 9,314 clusters but 5,713 of them are covered by Firefly, indicating that in so doing, TwitterStand-3's clustering is working very poorly without effectively merging similar groups of tweets. In contrast, Firefly's locality-aware keywords based clustering doesn't rely on a pre-trained classifier and is more reliable by finding word-usage anomaly from the perspective of a Twitter user instead of a tweet itself.

Table 4 also shows that our system Firefly misses quite a few of the news for some agencies such as "@bostonherald" and its newspaper "Boston Herald". To dig out the reasons behind this, we collect the 63 news of "@bostonherald" that we missed but only identified 5 of them as relevant. The major reason we missed these 5 news is because there are extremly few tweets covering them. For example, the news 'Good Samaritan rescues trapped dog from inferno" seemed contributed only by "@BostonHerald" as we only find this single 1 related tweet. The situation is similar regarding the website articles we missed in "Boston Herald", except that some of these articles don't appear in the tweets of its official news agency Twitter account.

It is also not unusual to find that some website articles relate to no tweets as we find local news agencies were not always publishing tweets about their website articles. One example is "@CBSboston" v.s. "CBS Boston": "@CBSboston" didn't post a single tweet about an accident of "Driver Suffered Serious Injuries When Car Crashed Into Pole In Carver" published in its website. This might give more explanations for website articles that our system didn't capture, and also inspire us to integrate cross-domain news source [55] to further mitigate the tweet data sparsity in the future. Another interesting observation from Table 4 is that different news agencies tend to cover different stories, with very few overlapping ones. This makes platforms like ours more valuable as a user doesn't have to browse different news agencies to learn about what is happening out there.

4.4.5 Precision

We asked 3 human judges to independently examine the 3,364 clusters of tweets detected in Firefly. As shown in Fig. 15, each candidate news is a set of tweets with their urls. The set of tweets are selected by having the most non-stopwords, retweet numbers and overlapping words with each other and no more than 5 tweets. The drop-down list provides 3 available options: "Positive", "Neutral" and "Negative", which are used by the judges to answer the question: "Are the three or more tweets describing the same local news?". The instructions given to the judges are summarized as follows:

> Each candidate news has a set of tweets, followed by their urls. Please read the tweets and answer if they are talking about the same news. A local news, here, refers to an event that happens in Boston Metropolitan area. For example, local news can be about

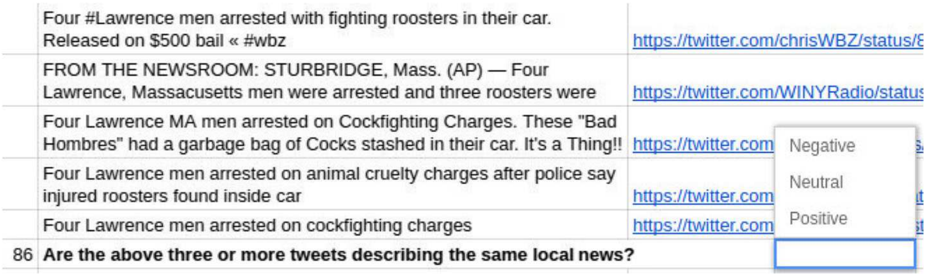
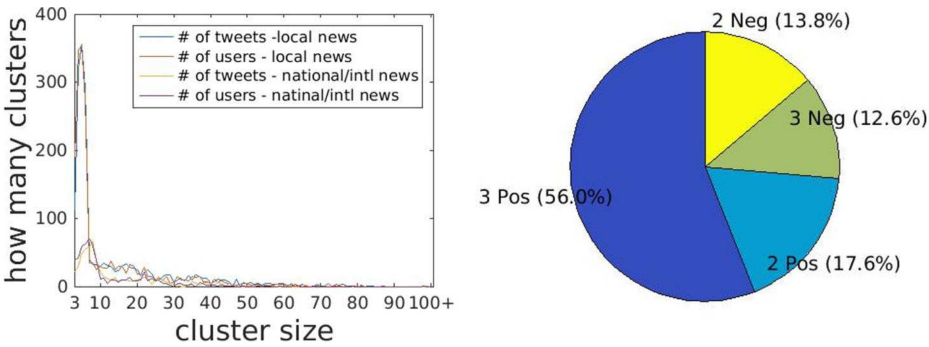


Fig. 15 Example of Human Judging UI

traffic, weather, missing persons/pets, farmer's market, yard-selling and book-selling, happy hour of bars and restaurants, crimes, protests, gatherings, award-nominations, and parties, meetings, celebrations, conferences, sports games etc. You can utilize the tweets' urls to get more information such as where the news happened. If you can't determine where it happened, choose "Negative". National/international news are recognized as "Neutral". News that happened in another place, like sports held in another city, should be "Negative". Also if you don't think the presented tweets are representing a news, select "Negative".

Figure 16b presents the distribution of judges' answers of the 2,574 events out of 3,364 that received a majority of "Positive"s or "Negative"s. Among the 2,574 events, 73.6% had 2 or more "Positive"s and were consented to be local news. The median number of tweets and median number of users in these local news are only 7 and 6, respectively, as shown in Fig. 16a. We also discovered that most of the clusters with a majority of "Negative" were formed by a set of people tweeting like "My fitbit for 1152017 6145 steps and 29 miles traveled". This surprised us because this crowd behavior meets our constraint for local news. In addition, out of 3,3364 news we detect, 649 received 2 or more "Neutral"s and were considered to be national or international news.

Next, we evaluate the clusters in TwitterStand, TwitterStand-3, EvenTweet and GeoBurst in the same way and list their proportions of news receiving more than 2 "Positives", 2 "Neutrals" and 2 "Negatives" respectively in Table 5. In comparison, among the 409 clusters



(a) Distribution of tweets and users.

(b) Distribution of human evaluation.

Fig. 16 Distribution of news cluster sizes and human evaluation

Table 5 Proportions of different types of tweet clusters

	≥2 positives	≥2 neutrals	≥2 negatives
Firefly	1,894 (56.3%)	649 (19.3%)	680 (20.2%)
TwitterStand	14 (3.4%)	306 (74.8%)	76 (18.6%)
TwitterStand-3	123 (5.28%)	1302 (55.9%)	816 (35.0%)
EvenTweet	44 (23.9%)	27 (14.7%)	90 (48.9%)
GeoBurst	52 (29.1%)	21 (11.7%)	70 (39.1%)

in TwitterStand, only 14 are identified as local news. The low proportion of local news in the default settings of TwitterStand is caused by its constraint that at least 10 tweets to form a cluster. Although relaxing this limit to 3 tweets in TwitterStand-3 captures more local news, its non-news proportion increases much more by falsely recognizing some repeating tweets from Twitter users as news, e.g. "@healylike". In contrast, by only exploiting the sparsely available geotagged tweets, EvenTweet and GeoBurst are only able to detect a small number of positive local news. Similarly, in Firefly, out of the 107 clusters that are formed by only geotagged tweets, 47 of them receive $\geq$ 2 "Positives" and are considered positive local news. This further illustrates that making only use of geotagged tweets will miss the majority of local news reported in non-geotagged tweets.

Note that TwitterStand captures national or international news (≥2 Neutrals) at a very high accuracy by setting the cluster size to be $\geq$ 10 tweets. And the mean and median number of tweets in such clusters in TwitterStand are 127 and 49, respectively, much larger than 16 and 11 in Firefly. This is because Firefly has a different strategy of finding such news in the sense that, from the perspective of an individual Twitter user, Firefly is only interested in some of his latest tweets that are discussing different content from his old ones, while TwitterStand might take all his tweets as news-related. This difference becomes more significant when it comes to columnists or sports reporters who might post many updates on the same news event. In addition, our 6-hour sliding window and the constraint for locality-aware keywords to be used by a limited number of people might also contribute to the relatively smaller number of tweets in national or international news clusters detected in Firefly.

5 Conclusions and future work

In this paper, we presented a system called Firefly to detect news for a given geographical area. In order to deal with the infamous sparsity problem in publicly available Twitter data, Firefly first enhances the local live tweet stream by identifying a large body of Twitter users in an area to follow via an online geotagging procedure and thereby significantly increases the amount of tweets generated from that area. With the enhanced local live tweet stream, we propose a method to identify locality-aware keywords and further use them to cluster tweets together to detect news. Comparing with news extracted from a set of local news agencies' tweets, our system achieves the highest recalls, and at the same time, outperforms the baseline approach TwitterStand regarding both recall and precision in detecting local news, and more importantly, is able to detect much more local news than the approaches that only use geotagged tweets.

A small portion of news might be present in two or more clusters if these news don't get updates in a time period that exceeds 6-hour, which is the main reason whey Table 4 is not symmetric for Firefly. A remedy to this problem in the future might be to simply lengthen the time window or to keep a pool of news clusters before the current sliding time window and keep them active if they receive updating tweets. In addition, the importance of various users should be addressed differentially. For example, reporter or news agencies should be more trustworthy to publish news. Additionally, as the human verification yields a ground-truth of local news, a learning procedure might be explored to help determine the parameter values in extracting locality-aware keywords and online clustering. At last, although the proposed system performs well in the Twitter domain, it may face challenges with respect to the generality to other platforms of social medias. This is because certain modules (especially data collection) have ad-hoc designs in Twitter platform. For example, in order to get local tweets from a place, we can create Twitter accounts to follow local people and thereby collect their real-time posts. This, however, may not be viable in other social media like Facebook because of the limitation of its public APIs and privacy policy. Therefore, different strategies of collecting local information from other social media need to explore. We leave these questions for our future work.

References

1. Sankaranarayanan J, Samet H, Teitler BE et al TwitterStand: news in tweets. SIGSPATIAL '09
2. Compton R, Jurgens D, Allen D Geotagging one hundred million Twitter accounts with total variation minimization. volume abs/1404.7152
3. Kwan E, Hsu P-L, Liang J-H et al Event identification for social streams using keyword-based evolving graph sequences
4. Mathioudakis M, Koudas N TwitterMonitor: trend detection over the Twitter stream. SIGMOD '10
5. Wei H, Sankaranarayanan J, Samet H Finding and tracking local Twitter users for news detection. SIGSPATIAL '17
6. Krumm J, Horvitz E Eyewitness: identifying local events via space-time signals in Twitter feeds. SIGSPATIAL '15
7. Zhang C, Zhou G, Yuan Q et al GeoBurst: real-time local event detection in geo-tagged tweet streams. SIGIR '16
8. Atefeh F, Khreich W (2015) A survey of techniques for event detection in Twitter. Comput Intell 31(1):132–164
9. Abdelhaq H (2015) Localized events in social media streams: detection, tracking, and recommendation, Heidelberg University, PhD thesis
10. Li Q, Nourbakhsh A, Shah S et al Real-time novel event detection from social media. ICDE '17
11. Zhang C, Liu L, Lei D et al TrioVecEvent: embedding-based online local event detection in geo-tagged tweet streams. KDD '17
12. Walther M, Kaisser M Geo-spatial event detection in the Twitter stream. ECIR '13
13. Boettcher A, Lee D EventRadar: a real-time local event detection scheme using Twitter stream. GreenCom '12
14. Hong L, Ahmed A, Gurumurthy S et al Discovering geographical topics in the Twitter stream. WWW '12
15. Zhou X, Chen L Event detection over Twitter social media streams, vol 23
16. Wei W, Joseph K, Lo W et al A Bayesian graphical model to discover latent events from Twitter. ICWSM '15
17. Skovsgaard A, Sidlauskas D, Jensen CS Scalable top-k spatio-temporal term querying. ICDE '14
18. Abdelhaq H, Sengstock C, Gertz M EvenTweet: online localized event detection from Twitter. Volume 6 of PVLDB '13
19. Kamath KY, Caverlee J, Lee K et al Spatio-temporal dynamics of online memes: a study of geo-tagged tweets. WWW '13
20. Watanabe K, Ochi M, Okabe M et al Jasmine: a real-time local-event detection system based on geolocation information propagated to microblogs. CIKM '11

21. Jonathan C, Magdy A, Mokbel MF et al GARNET: a holistic system approach for trending queries in microblogs. ICDE ’16
22. Kang W, Tung AKH, Zhao F et al Interactive hierarchical tag clouds for summarizing spatiotemporal social contents. ICDE ’14
23. Magdy A, Mokbel MF, Elnikety S et al Mercury: a memory-constrained spatio-temporal real-time search on microblogs. ICDE ’14
24. Magdy A, Aly AM, Mokbel MF et al GeoTrend: spatial trending queries on real-time microblogs. SIGSPATIAL ’16
25. Xu J-M, Bhargava A, Nowak R et al Socioscope: spatio-temporal signal recovery from social media. ECML PKDD ’12
26. Lappas T, Vieira MR, Gunopulos D et al On the spatiotemporal burstiness of terms. PVLDB ’12. VLDB endowment
27. He Q, Chang K, Lim E-P Analyzing feature trajectories for event detection. SIGIR ’07
28. Budak C, Georgiou T, Agrawal D et al GeoScope: online detection of geo-correlated information trends in social networks. PVLDB ’13. VLDB endowment
29. Liu Z, Huang Y, Trampier JR LEDS: local event discovery and summarization from tweets. SIGSPATIAL ’16
30. Valkanas G, Gunopulos D How the live web feels about events. CIKM ’13
31. Roller S, Speriosu M, Rallapalli S et al Supervised text-based geolocation using language models on an adaptive grid. EMNLP-CoNLL ’12
32. Marcus A, Bernstein MS, Badar O et al Twitinfo: aggregating and visualizing microblogs for event exploration. CHI ’11
33. Quezada M, Peña Araya V, Poblete B Location-aware model for news events in social media. SIGIR ’15
34. Li R, Lei KH, Khadiwala R et al TEDAS: a Twitter-based event detection and analysis system. ICDE ’12
35. Yamaguchi Y, Amagasa T, Kitagawa H Landmark-based user location inference in social media. COSN ’13
36. Davis CA Jr, Pappa GL, de Oliveira DRR et al (2011) Inferring the location of Twitter messages based on user relationships. Trans GIS 15(6):735–751
37. Sadilek A, Kautz H, Bigham JP Finding your friends and following them to where you are. WSDM ’12
38. Chen Y, Zhao J, Hu X et al From interest to function: location estimation in social media. AAAI ’13. AAAI Press
39. Cheng Z, Caverlee J, Lee K You are where you tweet: a content-based approach to geo-locating Twitter users. CIKM ’10
40. Mahmud J, Nichols J, Drews C (2014) Home location identification of Twitter users. ACM TIST 5(3):47,1–47,21
41. Han B, Cook P, Baldwin T (2014) Text-based Twitter user geolocation prediction. J AIR 49(1):451–500
42. Dalvi N, Kumar R, Pang B Object matching in tweets with spatial models. WSDM ’12
43. Li G, Hu J, Feng J et al Effective location identification from microblogs. ICDE ’14
44. Sakaki T, Okazaki M, Matsuo Y Earthquake shakes Twitter users: real-time event detection by social sensors
45. Weng J, Lee B-S Event detection in Twitter. ICWSM ’11
46. Albakour M-D, Macdonald C, Ounis I Identifying local events by using microblogs as social sensors
47. Takhteyev Y, Gruzd A, Wellman B (2012) Geography of Twitter networks. Social Networks 34(1):73–81. Capturing context: integrating spatial and social network analyses
48. Mok D, Wellman B, Carrasco J (2010) Does distance matter in the age of the internet? Urban Stud 47(13):2747–2783
49. Vardi Y, Zhang C-H (2000) The multivariate L1-median and associated data depth. Proc NAS 97(4):1423–1426
50. Zaharia M, Chowdhury M, Das T et al Resilient distributed datasets: a fault-tolerant abstraction for in-memory cluster computing. NSDI ’12
51. Dave A IndexedRDD: efficient fine-grained updates for RDDs
52. Zaharia M, Chowdhury M, Franklin MJ et al Spark: cluster computing with working sets. HotCloud ’10
53. Zaharia M, Das T, Li H et al Discretized streams: fault-tolerant streaming computation at scale. SOSP ’13
54. McMinn AJ, Moshfeghi Y, Jose JM Building a large-scale corpus for evaluating event detection on Twitter
55. Zheng Y, Zhang H, Yu Y Detecting collective anomalies from multiple spatio-temporal datasets across different domains

Publisher’s note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Hong Wei is now a Ph.D. student working with Prof. Hanan Samet in the Department of Computer Science at University of Maryland. His research focuses on spatial data mining and computing with spatial trajectories. Before coming to UMD, he finished his master degree at Department of Computer Science & Engineering at Shanghai Jiao Tong University. He got his Bachelor's degree from the School of Software majoring in Software Engineering at Huazhong University of Science and Technology in 2011.

Jagan Sankaranarayanan works at Google on its data infrastructure team. In his previous life, he was the department head of the data management group at NEC Labs America. He got his doctoral degree in Computer Science from the University of Maryland in 2008. His Ph.D. advisor was Prof. Hanan Samet. He got the bachelor's degree from RVCE, Bangalore. He mainly works on database management problems related to GIS, big data, analytics, and mobility.

Hanan Samet is a Distinguished University Professor of Computer Science at the University of Maryland where he researches the use of hierarchical data structures for GIS and spatial and image databases. He wrote "Foundations of Multidimensional and Metric Data Structures," Morgan-Kaufmann, 2006, and also the first two books in the field. His group devised the NewsStand system for reading news with maps by exploiting spatial synonyms (see video https://vimeo.com/106352925). His Ph.D is from Stanford. He's the founding chair of ACM SIGSPATIAL, the founding editor-in-chief of the ACM Transactions on Spatial Algorithms and Systems (ACM TSAS), recipient of the 2011 ACM Kanellakis Theory and Practice, and the 2014 IEEE Computer Society McDowell Awards.