IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 18, Issue 3, Ver. I (May-Jun.2016), PP 01-05
www.iosrjournals.org

Identifying Bursty Local Areas Related To Emergency Topics

S. S. More[1], Deepak Walkar[2], Parikshit Pishte[3], Saurabh Patil[4], Ritesh Kedari[5], Dhananjay Prabhawalkar[6]

S. S. More, Professor, Dept. Computer Science and Engineering, Sanjay Ghodawat Institute,
Deepak Walkar, Parikshit Pishte, Saurabh Patil, Ritesh Kedari, Dhananjay Prabhawalkar,
BE Scholars, Sanjay Ghodawat Institute, Atigre.

Abstract: *As the social media has gained more attention from users on the Internet, social media has been one of the most important information sources in the world. And, with the increasing popularity of social media, data which is posted on social media sites are rapidly becoming popular, which is a term used to refer to new media that is replacing traditional media. In this paper, we concentrate on geotagged tweets on the Twitter site. These geotagged tweets are known to as georeferenced documents because they include not only a short text message, but also have documents' which are posting time and location. Many researchers have been handling the development of new data mining techniques for georeferenced documents to recognize and analyze emergency topics, such as natural disasters, weather, diseases, and other incidents. In particular, the utilization of geotagged tweets to recognize and analyze natural disasters has received much attention from administrative agencies recently because some case studies have achieved compelling results. In this paper, we propose a novel real-time analysis application for identifying bursty local areas related to emergency topics. The aim of our application is to provide new platforms that can identify and analyze the localities of emergency topics. The proposed application is of three core computational intelligence techniques: the Naive Bayes classifier technique, the spatiotemporal clustering technique, and the burst detection technique. Also, we have implemented two types of application: a Web application interface and an android application. To evaluate the proposed application, we have implemented a real-time weather observation system embedded the proposed application. We used actual crawling geotagged tweets posted on the Twitter site. The weather detection system successfully detected bursty local areas related to observe emergency weather topics.*

Keywords : *Spatiotemporal clustering, Density-based clustering, Social media, Naive Bayes, Burst detection.*

I. Introduction

From recent years, social media has played a significant role as an another source of information. Now a days, people actively send and receive information about emergency topics, such as diseases, natural disasters, weather, and other incidents. Improvement of the utilization of social media for emergency topics management is one of the most important issues being debated in public and governmental institutions. Therefore, a large number of researchers have focused on the improvement of emergency topic and event detection through social media. This direction provides an opportunity for addressing new challenges in so many different application domains: how to detect where emergency occur and what they are going on. In this case, we mainly consternate on geotagged tweets posted on the Twitter site. These geotagged tweets are assumed to as georeferenced documents because they usually include not only a short text message, but also the documents' posting time and location. People on the Twitter site are referred to as a social sensors and geotagged tweets as a sensor data supervised by the social sensors. It is of value to people interested in a several topic to supervise dense areas where many georeferenced documents related to the topic are located. In this paper, these dense areas are referred to as bursty local areas related to the topic. In this paper, we propose a novel real-time analysis application for identifying bursty local areas related to emergency topics. The aim of our new application is to provide new platforms that can searching and analyze the localities of emergency topics. The proposed application is composed of three main computational intelligence techniques: the Naive Bayes classifier technique, the spatiotemporal clustering technique, and the burst detection technique. The density-based spatiotemporal clustering algorithm is a useful algorithm for extracting bursty local areas; however, two functional problems remain unresolved. One problem is that the density-based spatiotemporal clustering algorithm does not support real-time extraction. The second problem is that the proposed algorithm is based on keywords. Therefore, relevant georeferenced documents are extracted if they include an supervise keyword, not an observed topic; and this causes error extraction.

II. Density-Based Spatiotemporal Clustering

The density-based spatiotemporal clustering algorithm is based on the density-based spatial clustering algorithm Sander. In the density-based spatial clustering algorithm, spatial clusters are dense level areas that are divided from areas of low level density. In other words, areas with high level densities of data points can be supposed spatial clusters, whereas those with low level density can't. The main concept under the use of the density based spatial clustering algorithm shows that, for each data point within a spatial cluster, the neighborhood of a user-defined radius must contain at least a low number of points; that is, the density in the neighborhood must greater than some predefined threshold. Algorithm that has impacted the density-based spatial clustering algorithm is the DBSCAN algorithm, which was firstly presented by Ester et al. (1996). DBSCAN utilizes neighborhood density and identifies areas in which densities are greater than in other areas. However, it does not assume limited time changes. The density-based spatiotemporal clustering algorithm drawn out density-based spatiotemporal clusters that are both limited time and spatially-separated from other spatial clusters.

Algorithm:

Algorithm 1 in detailed the batch algorithm for density-based spatiotemporal clustering. In algorithm 1, for each georeferenced document *gdp* in *GDOC,* the function **IsClustered** checks whether document *gdp* is already assigned to a spatiotemporal cluster. Then, the density-based neighborhood of document *gdp* is obtained using the function **GetNeighborhood**. If georeferenced document *gdp* is a main document according to Definition, it is assigned to a new spatiotemporal cluster, and all the neighbors are queued to *Q* for further processing. The processing and assignment of georeferenced documents to the current spatiotemporal cluster continues until the queue is empty. The next georeferenced document is dequeued from queue *Q*. If the dequeued georeferenced document is not already assigned to the current spatiotemporal cluster, it is assigned to the current spatiotemporal cluster. Then, if the dequeued document is a core document, the georeferenced documents in the (ϵ, τ)-density-based neighborhood of the dequeued georeferenced document are queued in queue *Q* using the function **EnNniqueQ**, which places the input georeferenced documents into queue *Q* if they are not already existing in queue *Q*.

Algorithm 1: (ϵ,τ)-density-based spatiotemporal clustering algorithm
input: *GDOC* - a set of georeferenced document, ϵ - neighborhood radius, τ - interarrival time, *MGDoc* is threshold value
output: *SSC* - set of spatiotemporal clusters

```
ctid ← 1;
SSC ← φ;
for i ← 1 to |GDOC| do
        pd ← gdi ∈ GDOC;
        if IsClustered(gdp) == false then
            N ←GetNeighborhood(gdp, ϵ, τ);
            if |N| ≥ MGDoc then
sscctid ←NewCluster(ctid, gdp);
        ctid ← ctid + 1;
        EnQ(Q,N);
        while Q is not empty do
            pq ←DeQ(Q);
    sscctid ← sscctid ∪ pq;
    N ←GetNeighborhood(gdq, ϵ, τ);
    if |N| ≥ MGDoc then
            EnUniqueQ(Q,N);
           end if
           end while
        SSC ← SSC ∪ sscctid;
            end if
            end if
end for
return SSC;
```

III. System overview

Figure shows an outline of the system for the planned application. Within the system, the application server has 3 main managers as Document Extraction Manager, Document Clustering Manager, and web Service Manager. We will observe bursty native areas of emergency topics through a Web application and an Android application. Georeferenced document database is constructed on the application server. Every georeferenced document proceeds step by step. Steps executed on the application server are shown as below.

1. Document Extraction Manager fetches a georeferenced document that is freshly inserted within the georeferenced document information database.
2. Document Extraction Manager classifies the fetched georeferenced document gdi employing a Naive Bayes classifier. If and on condition that gdi categoryified to "positive" class, which implies gdi is related to emergency topic, head to subsequent step.
   Document Clustering Manager executes the incremental algorithm program for extracting the density based spatiotemporal clustering, that there are 2 input data as gdi and a group of current extracted density-based spatiotemporal clusters.
3. For every density-based spatiotemporal cluster, the burstiness of the cluster is calculated.
4. Web Service Manager provides the Web based application interfaces to access info regarding extracted bursty native areas.

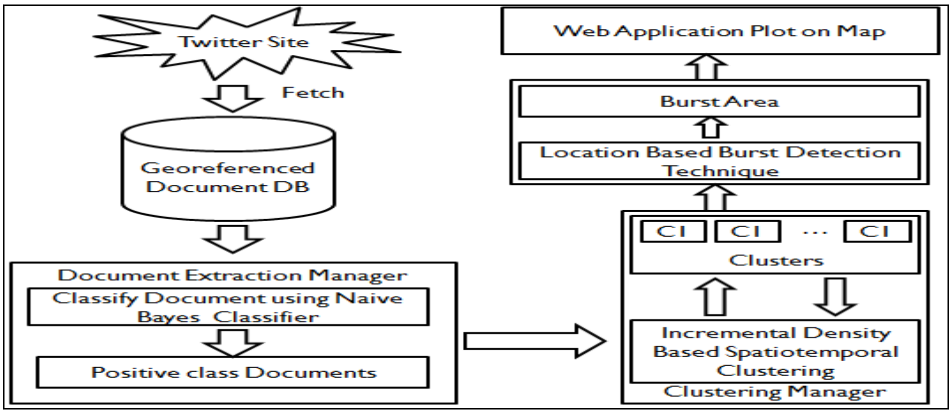

Figure 1 System overview of the proposed application.

IV. Naive Bayes classifier

The proposed application needs that georeferenced documents related to a supervised emergency topic are extracted. Georeferenced documents containing the supervised emergency topic include many kinds of keyword. Therefore, a keyword-based search is not effective for extraction. For example, suppose that an supervised emergency topic is "earthquake". Sequences "It is r earthquakeing" and "It could earthquake this weekend" include the keyword "earthquake"; but, they have several topics. In this study, only "It is earthquakeing" is extracted as a relevant georeferenced document related to the topic "earthquake". To satisfy this requirement, in *Document Extraction Manager,* theNaive Bayes classifier technique is utilized in sequence to extract georeferenced documents. A Naive Bayes classifier is a sober probabilistic classifier based on applying Bayes' theorem, which is based Bayesian statistics with naive independence assumptions. *Document Extraction Manager* classifies geotagged tweets as either "positive" class or "negative" class manually, where "positive" class is related to the supervised emergency topic and "negative" class is not. Georeferenced documents in the "positive" class are the relevant georeferenced documents.

V. Incremental algorithm

In the incremental algorithm updates the states of the extracted spatiotemporal clusters and extracts new spatiotemporal clusters each time a georeferenced document is appended. Algorithm 2 in detailed the incremental $(\in, \tau)$ density-based spatiotemporal clustering algorithm, which extracts $(\in, \tau)$ density-based spatiotemporal clusters based on each georeferenced document that comes for real-time extraction. There are two features in the incremental $(\in, \tau)$ density-based spatiotemporal clustering algorithm: limited reclustering and merging. Whenever a georeferenced document is appended to the georeferenced documents, existing $(\in, \tau)$ density-based spatiotemporal clusters must be updated; but the appended georeferenced document affects only its $(\in, \tau)$ density-based neighborhood within τ directly. Function **GetRecentData**(*gdoc)* returns *gdoc*'s $(\in, \tau)$ density-based neighborhood within τ. After the $(\in, \tau)$ density-based neighborhood is extracted to generate seeds and these seed georeferenced documents are reclustered again. In the incremental algorithm, during reclustering, some $(\in, \tau)$ density-based spatiotemporal clusters need to be added to other $(\in, \tau)$ density-based spatiotemporal clusters. Assume that $(\in, \tau)$ density-based spatiotemporal cluster *ssc* is expanding. If a core georeferenced document in ssc includes a georeferenced document, which is clustered in ssc , ssc is appended to ssc. Function AppendClusters appends two spatiotemporal clusters and return a appended spatiotemporal cluster.

Algorithm
Algorithm 2: Incremental *(∈, τ)* - density-based spatiotemporal clustering algorithm
Input : *gdoc* - a newly input georeferenced document, *GDOC* - a set of georeferenced, *CSSC* - a set of extracted spatiotemporal clusters, ∈ - user specified value, *τ* – user specified value, *MGDoc* - the minimum number of georeferenced document
Output: *NSSC* - a set of updated patiotemporal clusters

```
NSSC ← CSSC;
RD ←GetRecentData(gdoc, τ ,GDOC);
for i ← 1 to |RD| do
  pd ← rdi ∈ RD;
  N ←GetNeighborhood(pd, ∈, τ);
  if |N| ≥ MGDoc then
          if IsClustered(pd) == false then
          ssc ←MakeNewCluster(cid, pd);
          end if
          else
            ssc ←GetCluster(pd, NSSC);
          end if
          EnQueue(Q,N);
          while Q is not empty do
            gdoc ←DeQueue(Q);
            if IsClustered(gdoc) == true then
               N ←GetNeighborhood(gdoc, ∈, τ);
      if |N| ≥ MGDoc then
          ssc ←
          GetCluster(gdoc, NSSC);
          ssc ←
          AppendClusters(ssc, ssc);
      end if
            end if
            else
      ssc ← ssc ∪ gdoc;
               N ←GetNeighborhood(gdoc, ∈, τ);
               if |N| ≥ MGDoc then
          EnNniqueQueue(Q,N);
               end if
            end if
          end while
          NSSC ← NSSC ∪ ssc;
  end if
end for
return NSSC;
```

VI. Kleinberg's Burst Detection Algorithm

Kleinberg defined a model with an continue state automaton in which bursts are represented as state transitions. Suppose that there are m states in the automaton, every interarrival time is a probabilistic output that depending up on the internal states of the infinite state automaton. In the model, a state is associated with a burstiness state and a higher state indicates higher burstiness.

Algorithm: Location-Based Burst Detection
input : cutoff distance dist, user position up, word time-series data wi, and parameter list for burst detection params
output: optimal state-transition sequence S

```
IAT′CTDi   () /* make a empty sequence */
for j   1 to |w−> CTDi| do
if j = 1 then
pctd  ←  stime
else
pctd  ← w−> CTDi[j − 1]
iat ←  w−> CTDi[j] − pctd + _(wi−>
```

CTPi[j], up)
IAT′CTDi append_sequence
(IAT′CTDi ,iat)
s KBD (IAT′
CTDi ,params)
return s

VII. Conclusion

So in this paper, we have developed web application and android application to identifying the bursty local areas related to emergency topics. In additional we are also providing today's hot topics from tweets are fetched in our application for the development of our application. We have used naïve bayes classifier technique, density-based spatiotemporal clustering algorithm and burst detection technique. We mainly focus on geo tagged tweets and by extracting that tweets we plot the bursty local area on map, for which we have used streaming API's of twitter. In future we are planning to develop application in multi language.

VIII. References

[1]. Kaneko T, Yanai K (2013) Visual event mining from geo-tweet photos. In: Multimedia and Expo Workshops (ICMEW) 2013 IEEE International Conference On. IEEE, San Jose, CA, USA. pp 1–6.
[2]. Tamura K, Kitakami H (2013) Detecting location-based enumerating bursts in georeferenced micro-posts. In: roceedings of the 2013 Second IIAI International Conference on Advanced Applied Informatics. IIAI-AAI '13. IEEE Computer Society, Los Alamitos, CA, USA. pp 389–394.
[3]. Tamura K, Ichimura T (2013) Density-based spatiotemporal clustering algorithm for extracting bursty areas from georeferenced documents. In: Proceedings of The 2013 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2013. IEEE Computer Society, Los Alamitos, CA, USA. pp 2079–2084.
[4]. Abdelhaq H, Sengstock C, Gertz M (2013) Eventweet: Online localized event detection from twitter. Proc VLDB Endow 6(12):1326–1329.
[5]. Musleh M (2014) Spatio-temporal visual analysis for event-specific tweets. In: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data. SIGMOD '14. ACM, New York, NY, USA. pp 1611–1612.
[6]. Hiruta S, Yonezawa T, Jurmu M, Tokuda H (2012) Detection, classification and visualization of place-triggered geotagged tweets. In: Proceedings of the 2012 ACM Conference on Ubiquitous Computing UbiComp. '12. ACM, New York, NY, USA. pp 956–963.
[7]. Hong L, Ahmed A, Gurumurthy S, Smola AJ, Tsioutsiouliklis K (2012) Discovering geographical topics in the twitter stream. In: Proceedings of the 21st International Conference on World Wide Web. WWW '12. ACM, New York, NY, USA. pp 769–778.
[8]. Hwang M-H, Wang S, Cao G, Padmanabhan A, Zhang Z (2013) Spatiotemporal transformation of social media geostreams: a case study of twitter for flu risk analysis. In: Proceedings of the 4th ACM SIGSPATIAL International Workshop on GeoStreaming. IWGS '13. ACM, New York, NY, USA. pp 12–21.
[9]. Mandel B, Culotta A, Boulahanis J, Stark D, Lewis B, Rodrigue J (2012) A demographic analysis of online sentiment during hurricane irene. In: Proceedings of the Second Workshop on Language in Social Media. LSM '12. Association for Computational Linguistics, Stroudsburg, PA, USA. pp 27–36