Mining spatio-temporal information on microblogging streams using a density-based online clustering method

Chung-Hong Lee
Department of Electrical Engineering, National Kaohsiung University of Applied Sciences, Kaohsiung, Taiwan

Abstract
Social networks have been regarded as a timely and cost-effective source of spatio-temporal information for many fields of application. However, while some research groups have successfully developed topic detection methods from the text streams for a while, and even some popular microblogging services such as Twitter did provide information of top trending topics for selection, it is still unable to fully support users for picking up all of the real-time event topics with a comprehensive spatio-temporal viewpoint to satisfy their information needs. This paper aims to investigate how microblogging social networks (i.e. Twitter) can be used as a reliable information source of emerging events by extracting their spatio-temporal features from the messages to enhance event awareness. In this work, we applied a density-based online clustering method for mining microblogging text streams, in order to obtain temporal and geospatial features of real-world events. By analyzing the events detected by our system, the temporal and spatial impacts of the emerging events can be estimated, for achieving the goals of situational awareness and risk management.

1. Introduction

1.1. Motivations
Recently a growing number of internet users keep up with newest information by utilizing social media tools (e.g. Twitter), searching for a hot news topic about some emerging events. However, while some research groups have successfully developed topic detection methods from the text streams for a while, and even some popular microblogging services like Twitter did provide information of top trending topics for selection, it is still unable to fully support user for picking up all of the real-time event topics with a comprehensive spatio-temporal viewpoint to satisfy their information needs.

In this work, we take Twitter as our microblogging data source to identify the problem domain, for describing and realizing evolving events in depth. In this work, an “event” is taken to be something that happens at some specific time and place (e.g. an earthquake striking Japanese cities on March 2011). Twitter, regarded as one of powerful social media tools, allows users to post short messages (i.e. maximum 140 characters, also known as “tweet”) to communicate each other. In particular, once people suddenly suffer from some unexpected disasters, over thousands of Twitter users seized their mobile phones or computers to peck out tweets for communication in the form of text messages, web-based instant messages, or posts on Twitter’s site. The vast amount of tweets might cause Twitter’s datasets to be fairly difficult to process. Hence, the application platforms should be able to process large volumes of such textual data arriving over time in a stream.

On the other hand, severe natural disasters such as earthquakes, tsunami, etc., require new scientific methodologies for early warning of risk and real-time event awareness. Understanding their possible impacts and striving towards their timely detection and prevention can help protect lives and properties. Under such circumstances, the location from which a tweet was issued can be an enormous help. Messages from mobile phones with GPS receivers can contain location information. Recently Twitter also allowed such information to be attached to tweets, in order to enable people to apply this data to produce more relevant location based real-time results. Unfortunately, enabling the functions of latitude and longitude features on tweets is optional, up to users’ choices. Therefore, in this work we developed an online spatio-temporal information platform which can generate really useful results that are obviously impactful in real-world applications.

1.2. Problem statements
Given the fact that some perspectives of on-going and unknown events are still better observed by human eyes and, in some cases news reporters are unable to get the event information, messages in tweets about the events could prove to be useful. For example, in many cases, when some earthquake event occurred, a few
seconds and minutes after the quake, people eager to talk to their families by phones or other communication tools, and often they found the only live media for realizing the situation related to the new quake were – tweets. Under such a circumstance, an actual demand is to develop a way for discovering real-time event truth with temporal and geospatial information from microblogging messages to offer users insightful information in an efficient manner.

Recently Twitter released some part of location service that enables mobile users publish their tweets with geospatial data such as latitude and longitude. Such services encourage more research work involved in mining the spatio-temporal information on microblogs to get real-time and geospatial event information. Mining the spatio-temporal information related to critical events is a challenging task, which attempts to extract useful information from large volume of continuously arriving tweet streams. Hence, in this work we are keen to explore the potential of spatio-temporal information provided by Twitter messages, thus contributing to a satisfaction of the information need of ‘situational awareness’ for event control.

1.3. Research objectives

In this work, we study the problem of event detection and awareness by monitoring on-line real-time messages (i.e. Twitter messages), and exploiting the geolocation data provided by the experimental social networking datasets for situational awareness. To our best knowledge, previous researches those try to extract hot topics from Twitter in real time focused on only temporal models. This research also extracts spatial information of each topic. This is a novel approach in this research area. Alone with the event detected from the Twitter messages, we attempt to reverse engineer the location of a tweeted event from text analysis because some location information cannot be acquired for free public Twitter data. The aim of this work is to study how microblogging social networks (i.e. Twitter) can be used as a reliable information source of emerging events by extracting their spatio-temporal features from the messages to enhance event awareness. We believe that Twitter could be used to detect events and notify users who are concerning event development, by applying Twitter information to establish a spatio-temporal model for event estimation, which is able to find the center and trajectory of event location. Hence, in this work we developed several algorithms for mining Twitter text streams to obtain real-time and geospatial event information. The significance of the work lies more in the application than in the modeling algorithms. The proposed solution is being described in the following sections.

2. System framework and approaches

In this chapter, we describe our system framework and approaches. First of all, we present some problem characteristics and difficulties in system development for acquiring spatio-temporal information associated with the discovered events as below.

- There are hundreds of thousands Twitter messages and spams performing data exchange on the internet incessantly. The source of Twitter messages is an open-ended data stream and the amount of the accumulated data is extremely large, so it is impossible to allow all data be loaded into the memory for computation. Thus, an effective incremental learning approach is essentially required for discovering knowledge from such text streams. In general, there are two fundamental data mining techniques that can be considered in conjunction with Twitter data: (a) graph mining using analysis of the links among messages, and (b) text mining based on analysis of the messages textual contents. In this work, we mainly focus on investigation of text mining methods for mining Twitter data.
  - Once dealing with Twitter streams, one important indication of change is the presence of bursts. A burst in the Twitter messages implies that the occurrence of a certain topic feature is unexpectedly frequent in a short period of time. Such situations normally indicate some real-world event has now drawing much attention by Twitter users. The burst detection method applied in our work is concerned with automatic identification of bursts from Twitter posted messages, providing useful insights into the unusual events and in turn facilitating timely event monitoring.
  - Real-time operation is a critical factor for the use of Twitter messages. User posts a tweet with some specific timestamp which indicates what someone says has happened at the specific time point and places. It is believed that the messages act as a useful lens into the social perception of an event in any region, at any point in time. For instance, people posted their tweets when earthquake occurred, the importance of these tweets are valuable just at the moment. To cope with temporal dynamics of tweets, in this work we developed a dynamic weighting scheme called Burst to adapt to such requirements, which is able to subtly reflect the changes over time and quickly assign proper weights for achieving accurate temporal evaluation of messages in such a dynamic environment.
  - According to recent investigation of most users’ location entries on Twitter, lots of users did not provide real location information, often incorporating fake location that can mislead most geographic event detection systems. Furthermore, even the tweet samples with the data of Twitter’s geo-tagging location service that enables mobile users publish their tweets with latitude and longitude data, it is still not precise enough for detecting accurate location associated with the occurrence of some specific event only by utilizing such information. This is due to the fact that lots of tweets were sent by mobile devices and, the mobility of Twitter users make it difficult to detect actual fixed location related to the ongoing events.
  - A difficult problem alone with online learning on the continuously incoming text streams is that the concept of interest behind content may change with time, depending on some hidden context, which is not given explicitly in the form of initial features. The phenomenon is known as concept drift. In the case of microblogs, this happens to produce a topic drift on messages. For microblogging services, it is obvious that the hot topics of some issue discussed by Twitter users often drift with time, depending on the newest development of the original event and some hidden context. For instance, in the case of Fukushima nuclear accidents, the hot topics on tweets starts with “earthquake” and “tsunami”, and then move to “nuclear and radiation accident”, and “supply chain risk”, etc. A challenge in handling concept drift is distinguishing between true concept drift and noise, since some algorithms may overreact to noise, mistakenly interpreting it as concept drift.
  - The topic transition related to evolving events is generally hard to be detected. If we have background knowledge such as “earthquake → tsunami → nukes”, we may be able to understand the obtained clustering result. But it would be quite difficult to analyze transitions of unknown events. Extraction of meaningful information from a clustering result is not an easy task.

To solve the aforementioned issues, we have proposed a framework for event detection by spatio-temporal information discovered from Twitter messages. Our proposed system framework mainly consists of two modules, say, content and temporal analysis
module and spatial analysis module. The content and temporal analysis module is developed for handling microblogging message streams, and categorized them into thematic topics. Subsequently, the module of spatial analysis performs allocating topic centric messages to appropriate locations in the real world map. The system framework is developed based on a density-based online clustering method. As mentioned above, the concepts that we attempt to learn from the text streams may drift with time. To flexibly react to concepts drift in the messages, we have developed algorithms in the density-based clustering method using the sliding window technique, which are able to detect context changes without being explicitly informed about them. Detailed description of the technique is being discussed in Sections 3 and 4.

2.1. Assumptions and system framework

Prior to our discussion on our system architecture for mining spatio-temporal event information, we present several assumptions made for system development.

Assumption 1. The gathering messages tend to be a phenomenon of temporal locality.

In this work, “event” is regarded as a set of messages that are highly concentrated on some issues in a period of time. Such a phenomenon is also described as the characteristics of temporal locality among messages. The concept of temporal locality is used to present that an event that is discussed at one point in time will be discussed again sometime in the near future.

Assumption 2. The messages associated with a real-world event are of a nature of event lifecycle.

To process incoming texts with a chronological order, a fundamental issue we concerned is how to find the significant features in text streams. In classic text retrieval systems, the most common method for feature extraction is to deal with each document as a bag-of-words representation. Such an approach is not completely suitable for our dynamic system. The main technical issue of detecting events in text streams is to derive a set of features (words) to describe each message and a similarity measure between messages (Lee, Wu, & Chien, 2011). It has been observed that, in microblogging text streams, some words are “born” when they appear the first time, and then their intensity “grow” in a period of time till reach a peak. These words are called burst words. As time passes by, once the topics are no longer discussed by people, they “fade away” with power law and eventually the feature words become “death” (disappear), or change to a normal state. Such a phenomenon is regarded as a lifecycle of the selected features associated with a particular event under investigation.

Assumption 3. Most messages tend to be a phenomenon of spatial locality.

For some events, particularly the events related to some disasters (e.g. flooding), the affected areas were often being significantly migrated or expanded as time passes by. Initially the popular event topic largely represent the common interests of local users, and migrated or expanded as time passes by. Initially the popular event topics (e.g. flooding), the affected areas were often being significantly migrated or expanded as time passes by. Initially the popular event topics, which are highly densely located in a specific geographical area. Such a phenomenon is considered in our work to derive algorithms for estimating location of an event by using location feature vectors. These concepts will be further described in Section 3.2.

Assumption 4. Most related locations of events can be obtained by content-based learning methods.

Ideally, as mentioned previously, the most intuiting method to obtain spatial information of messages is to annotate them with geographic coordinates which are based on a precise form of location (i.e. latitude and longitude). Unfortunately, enabling the functions of latitude and longitude features on tweets is optional, up to users’ choices. Due to the consideration of privacy, the location information contained in a tweet structure is quite limited; only the time-zone geographic information on tweets can be obtained from user profiles. On the other hand, owing to the fact that some tweets are posted by the mobile devices (e.g. smart mobile phone), the event location and the location information extracted from tweet tags may not be exactly the same place. Under such a circumstance, we argue that, the strategy on tracking event location only by means of the feature of geographic coordinates on tweets is not essentially required, which may possibly mislead the results of the related studies. As a result, extracting location information (i.e. keywords) from text content seems to be a sensible solution. Although the methods of event tracking by monitoring specific strings might lose some hidden information due to some user-generated contents and critical messages may not contain any location keywords, in our system framework we develop effective mining approach to overcome such a limitation.

Accordingly, we develop a novel spatio-temporal topic detection system framework, as illustrated in Fig. 1. The system is designed as a two-pass process, consisted of two modules, say, content and temporal analysis module and spatial analysis module. As shown in Fig. 1, the content and temporal analysis module is developed for handling Twitter streams, and categorized them into thematic topics, which has been reported in our previous work (Lee, Chien, & Yang, 2010; Lee et al., 2011). Subsequently, the module of spatial analysis performs allocating topic centric messages to appropriate locations in the real world map Fig. 2 illustrates an overview of proposed microblogging topic detection system.

2.2. Content and temporal analysis (1st pass)

In order to effectively detect emerging events, our work started with mining hot topic news from numerous contextual posts on microblogging messages. In this work, “event” is regarded as a set of messages that are highly concentrated on some issues in a period of time. Such a phenomenon is also described as the characteristics of temporal locality among messages. Suppose that we have a temporally-ordered message stream $M = \{m_1, m_2, \ldots, m_k, m_{k+1}, \ldots\}$, which arrived at time $T = \{t_1, t_2, \ldots, t_k, t_{k+1}, \ldots\}$, $\forall m_k, m_l \in M, t_k > t_l$. First, a language filter will filter out some incoming messages which contain non-ASCII characters (i.e. Chinese, Japanese, etc.), and then the system decompose the textual messages into a bag-of-words feature. Subsequently, our algorithm started with construction of a dynamic feature space which maintains messages with sliding window model to deal with the message streams. New incoming messages will be reserved in memory till they are out of the time window. This process model can prevent the memory limitation problem caused by continuously coming messages. Then we utilized a dynamic term weighting scheme (Lee et al., 2011) to assign dynamic weights to each word, by comparing historical records. The neighborhood generation algorithm is performed to quickly establish relations with messages, and carry out the operation of text stream clustering. In this work, we utilized IncrementalDBSCAN as our online clustering algorithm. Therefore, the system constantly groups messages into topics, and the shape of clusters would change over time. Finally, hot topic events on microblogs can be determined and ranked by analyzing the collected cluster records.

The Twitter messages continuously posted by users around the world; so it is almost impossible to store all messages at one time due to the restrictions of memory limitation and constantly time
Fig. 1. Framework of system architecture.

Fig. 2. Overview of proposed microblogging topic detection system.
lapses. A typical approach for dealing with the problem is based on the use of so-called sliding windows (Bifet, 2010). As a result, the sliding window method is adopted for tackling the issue in this work, as shown in Fig. 3. Briefly speaking, the steps of sliding window technique include: (i) the insertion operation in which a new index entry is built when a message comes in, (ii) the message is reserved until its lifetime exceeds the fixed length of time window \( tw \), and (iii) the deletion operation in which the message will be removed from memory.

### 2.3. Spatial analysis (2nd pass)

When event topics are detected, the next step is to analyze spatial distribution of them. In this work a location estimation method is utilized for estimating where the event occurs. The main idea behind our event detection approach is based on the characteristics of spatial locality. Spatial locality is described as a set of messages concerning some topic, which are highly densely located in a specific geographical area. Such a phenomenon is considered in our work to derive algorithms for estimating location of an event by using location feature vectors. A location feature vector records the location distribution of the topics at a specific time point. For example, the Christchurch earthquake in New Zealand took place on Feb 22, 2011. Once the news media broadcast the news, people in other places started to discuss about this event. The distribution of the users discussed the event was being expanded to other continents. Therefore the location feature vector is also changed over time as shown in Fig. 4. The algorithm for a spatial analysis model is shown in Table 1.

In Fig. 4, \( \text{occur}_{m,n} \) denotes the number of occurrence of the \( m \)th location-feature in the \( n \)th topic and, \( t \) denotes the time of event occurrence. The location feature vector can be used to distinguish whether the event is a local topic or global topic. First we formulate the probability of a topic \( \text{topic}_i \), which belongs to a specific location \( \text{loc}_j \), as Eq. (1).

\[
p(\text{loc}_j|\text{topic}_i) = \frac{\text{occur}_{i,j}}{N_i} * \frac{1}{|\text{loc}_j|}
\]

where the probability of the \( \text{topic}_i \), belonging to a location \( \text{loc}_j \) is obtained by dividing the number of messages which contain \( \text{loc}_j \) in \( \text{topic}_i \) (\( \text{occur}_{i,j} \)) by the total number of messages \( (N_i) \). The \( 1/|\text{loc}_j| \)

### Table 1

<table>
<thead>
<tr>
<th>Time Zone</th>
<th>Topic(_1)(t)</th>
<th>Topic(_2)(t)</th>
<th>Topic(_3)(t)</th>
<th>Topic(_4)(t)</th>
<th>Topic(_5)(t)</th>
<th>Topic(_6)(t)</th>
<th>Topic(_7)(t)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Athens</td>
<td>occur(_1),0</td>
<td>occur(_2),0</td>
<td>occur(_3),0</td>
<td>occur(_4),0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bangkok</td>
<td>occur(_1),0</td>
<td>occur(_2),0</td>
<td>occur(_3),0</td>
<td>occur(_4),0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Beijing</td>
<td>occur(_1),0</td>
<td>occur(_2),0</td>
<td>occur(_3),0</td>
<td>occur(_4),0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hawaii</td>
<td>occur(_1),0</td>
<td>occur(_2),0</td>
<td>occur(_3),0</td>
<td>occur(_4),0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Central Time</td>
<td>occur(_1),0</td>
<td>occur(_2),0</td>
<td>occur(_3),0</td>
<td>occur(_4),0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(US &amp; Canada)</td>
<td>occur(_1),0</td>
<td>occur(_2),0</td>
<td>occur(_3),0</td>
<td>occur(_4),0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hong Kong</td>
<td>occur(_1),0</td>
<td>occur(_2),0</td>
<td>occur(_3),0</td>
<td>occur(_4),0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Fig. 3.** A continuous message stream using sliding window model.

**Fig. 4.** Location feature vector.
is the penalty factor to penalize a topic which is widely discussed in many places. In addition, a candidate location (as shown in Eq. (2)) can be determined by the maximum probability for topic\(_i\).

\[
candiLoc(topic_i) = \arg \max_{loc} \{p(loc|topic_i)\} \tag{2}
\]

In Eq. (2), a candidate location of the topic is calculated. Subsequently the rule for determining whether the topic is a local topic or global topic can be formulated as Eq. (3).

\[
Loc_i = \begin{cases} 
    candiLoc(topic_i), \text{ if } p(candiLoc(topic_i)|topic_i) > \theta \\
    "globalTopic", \text{ otherwise.} 
\end{cases} \tag{3}
\]

In Eq. (3), a topic would be regarded as a local topic if the probability of candidate location exceeds the threshold \(\theta\). The cut-off point \(\theta\) represents a tradeoff between the level of sparsity and concentricity for a given topic. For setting cut-off point \(\theta\), the higher the threshold value is chosen, the more concentricity of the topic is required. Hence, an event is considered as a local event if the distribution of messages is sparse, otherwise it is a global event.

Finally, the geospatial distribution of the topic can be mapped to a real world map for visualization. The spatial analysis model is designed to support identifying where the event happened. Even once a local event has become a global event, we can still trace back to its early state.

3. The proposed density-based online clustering method

In this section, we describe the technical details of the proposed density-based online clustering method. Our work starts with investigating the solution for detecting topics and tracking events about the interests, hot news topics, and preferences of people from text information sources of microblogging services. In this work, an algorithm using a density-based method is developed for mining microblogging message streams. The purpose of our approach is to effectively detecting and grouping emerging topics from the user-generated content in a real-time or specified time slot. On the other hand, for tackling a key challenging issue in mining the microblogging messages, we attempt to analyze the real-time distributed messages and extract significant features of them in a dynamic environment. We propose a novel term weighting method, called BursT, using a sliding window technique for weighting message streams. This method was proven to be capable of dealing with concept drift problem, being able to detect context changes without being explicitly informed about them.

3.1. The density-based clustering approach

As the temporally-ordered messages streaming into the system, the next step is to incrementally gather messages into thematically topics. For such an information gathering process, one of the main difficulties is figuring out the meaning and value of those fleeting bits of information for mining the text streams. The challenge goes beyond filtering out spam, though that’s an important part of it. Microblogging messages may lose their value within minutes of being written. Therefore, the system should be able to quickly group them into clusters which are evolving over time. Meanwhile, the continuous evolution of clusters makes it essential to be able to swiftly identify new clusters in the data. That is, the algorithm has to deal with lots of external dynamic changes, i.e. various updates occur and topic shift (i.e. concept drift) issues, etc. In order to achieve this goal, we have to provide an effective solution in which online clustering operation can be well performed in mining the microblogging text streams.

3.1.1. The considerations for utilizing density-based clustering approach

The reasons of adopting a density-based clustering approach in this work are described as follows:

1. Messages collected from microblogs normally contain lots of noises. Once mining microblogging messages, the clustering algorithm should perform its best to filter out noises in processing the contents. Density based clustering groups data based on their density connectivity and treats noises as outliers which would not be involved in any cluster.
2. Density-based clustering techniques are capable of detecting arbitrary-shaped clusters.
3. There is no assumption about the number of clusters with fixed or flexible parameter of \(k\) (i.e. topic), and it is thus unsuitable for some real world applications in the problem domain, especially in dealing with the topic detection task with dynamic topics around the world.

Due to the dynamic natures mentioned above, it is highly desirable to perform data updates incrementally. Thus, in this work a density-based clustering based on the algorithm of IncrementalDBSCAN (Ester, Kriegel, Sander, Wimmer, & Xu, 1998) was used for our system development. IncrementalDBSCAN is an efficient algorithm which is based on DBSCAN for mining data with density-based connectivity (Nguyen-Hoang, Hoang, Bui-Thi, & Nguyen, 2009; Sun and Hu, 2011; Wen, Nie, & Zhang, 2002). The technique and operational application of IncrementalDBSCAN are described in details in the following subsections.

3.2. The IncrementalDBSCAN algorithm

In this work, we adopted a density-based clustering approach called IncrementalDBSCAN to against noises, instead of using spam classifier to determine uninformative message subjectively. The key idea of density-based clustering is that for each object of a cluster, the neighborhood of a given radius (Eps) has to contain at least a minimum number of objects (MinPts) to form as a basic unit of density region. The dynamic operations for insertion can be divided into Noise, Creation, Absorption and Merge conditions and deletion has three different cases of Removal, Reduction and Potential Split. Each case is judged from the properties of connectivity, and enables us to maintain the status of the cluster dynamically.

According to the theory of IncrementalDBSCAN clustering method, the shape of clusters will change over time when a message being inserted or a victim message being deleted from sliding window with its message density properties. Certainly the less density area would not be a topic, because of the distances between messages are long according to the calculations of temporal text similarity. Meanwhile, text stream cluster algorithm will generate several clusters at each time, due to its natural dynamics.

Essentially, the key idea of density-based clustering is that for each object of a cluster the neighborhood of a given radius (Eps) has to contain at least a minimum number of objects (MinPts), i.e. the cardinality of the neighborhood has to exceed some threshold. The core objects will be established as they reach the state of insertion, and then only the part of affected objects (see Fig. 5) will be updated with the incremental version. The core objects in Ne(p) can be defined as UpdSeedIns and UpdSeedDel for operations of insert and delete, where Ne(p) is the set of Eps-neighborhood of \(p\).

3.3. Temporal text similarity

When a message arrived, the similarity should be calculated with its neighbors. In our approach, we proposed a formula called
temporal text similarity which combines content and temporal dimensions. Supposed that we have an incoming message \( m_a \) and a neighbor massage \( m_b \), the similarity between \( m_a \) and \( m_b \) can be denoted as follows:

\[
\text{Sim}(m_a, m_b) = \cos(m_a, m_b) + tp(m_a, m_b)
\]  

(4)

\[
\cos(m_a, mb) = \sum_{t} \frac{m_{at} \times m_{bt}}{|m_a||m_b|}
\]  

(5)

\[
\text{tp}(m_a, mb) = e^{-\frac{|ta - tb|}{\zeta}}
\]  

(6)

In similarity function \( \text{Sim}(m_a, m_b) \), the cos similarity is used for content-based similarity measurement. In order to make a consideration of temporal information, the temporal penalty \( \text{tp}(m_a, m_b) \) is an exponential distribution which can reduce the similarity if two document with a long time distance, the temporal penalty is high when the distance \( |ta - tb| \) is close and vice versa. The \( \zeta \) is a parameter which can adjust the temporal decay rate.

### 4. Design of a dynamic term weighting scheme for adapting to changes in messages

As mentioned previously, in this work our solution to tackle the issues is based on the utilization of a sliding window. As the context is known to vary in time, the learner trusts only the latest examples – this set is referred to as the window. Data samples are added to the window as they arrive, the oldest samples are deleted from it. In our solution, the window is being of fixed size, and the oldest sample will be dropped whenever a new one comes in. Meanwhile, once changes of a concept have been detected, the system should be able to discard out-of-date examples and clusters (e.g. time window). In this work, we utilize a dynamic weighting scheme to discriminate the event messages, and cope with the concept-drift issues (Khalilian & Mustapha, 2010; Widmer & Kubat, 1996) in a dynamic environment. Once the concepts behind the messages evolve with time, the underlying cluster structures in our system will also significantly change with time. Under such a circumstance, the system should be able to be adapted itself to supporting topic and concept drifting in the microblogging text streams. We developed a novel term weighting method, called BursT, using sliding window techniques for weighting message streams (Lee et al., 2011). The experimental results show that our weighting technique has an outstanding performance to reflect the occurrence of concept drifts in tweets.

### 4.1. The dynamic term weighting scheme

For microblogs, the corpus tends to be dynamic as new items always being added and old items being changed or deleted. Therefore, an ideal term weighting scheme for mining microblogging messages should subtly reflect the changes over time and quickly assign proper weights in such a dynamic environment. In this section, we firstly describe the sliding window model, which is associated with the development of our weighting method, and then explain the weighting mechanism.

#### 4.1.1. The design strategy for term weighting

To process texts with a chronological order, a fundamental problem we concerned is how to find the significant features in text streams. Specifically, the trends of concept are often not stable but change with time, which is also known as concept drift. Under such a circumstance, the design of weighting scheme for microblogging message should be constantly updated. Here we apply the term weighting scheme BursT which was proposed in our previous work (Lee et al., 2011). The experimental results indicate that has a better performance in weighting words of microblogging messages than incremental TFIDF (Brants, Chen, & Farahat, 2003) and TFPDF (Run & Ishizuka, 2002) techniques.

The word burst is defined as an unusual number of frequently posted messages happened in a short time. Fig. 6 shows a three-phase of word categories occurred in microblogging messages. In Fig. 6, uninformative word means that a word rarely occurred in the sliding window, such as an oral word. The axis df shown in Fig. 6 represents the value of document frequency. If the word was occurring very frequently but with lower burstiness, they could be recognized as common word or social word, such as “haha” and “lol”. If a word has a higher burst than expectation within a certain range of document frequency, we will highlight its importance for weight design in the sliding window.

Accordingly, our strategy in determining BursT value is that a heavier weight is achieved by a higher burstiness, in which some word occurs frequently in the window. Thus, the BursT weighting formula is shown in Eq. (7):

\[
\text{weight}_{wi} = \text{BS}_{w1} + \text{TOP}_{w1}
\]  

(7)

where the weight of the word \( w \) at time \( t \) will be constituted by two factors: \( \text{BS} \) (Burst Score) and \( \text{TOP} \) (Term Occurrence Probability). For calculating BursT weights of single words, each word \( w \) is recorded as a quartet \( \langle w, \text{ar}_{w,t-1}, n_{w,t}, \text{E}(\text{ar}_{w,t}) \rangle \), \( \text{ar}_{w,t-1} \) represents the last time word \( w \) arrived, \( n_{w,t} \) counts the total number of word \( w \) appeared in our system, and \( \text{E}(\text{ar}_{w,t}) \) is a long time cumulative expectation of arrival rate to the word \( w \). The detailed description of the weighting factors will be discussed in the following subsections.

![Fig. 6. Word categories in microblogging texts.](image-url)
4.1.2. The BS weighting factor

The interval of arrival time between messages can be transformed into arrival rate for many streaming data applications. If the feature of some message arrives with short intervals incessantly, the feature representing the importance of a message may be more useful. Suppose there is a feature word \( w \) occurs in message sequence \( \{m_{w,1}, m_{w,2}, m_{w,3}, \ldots m_{w,t}\} \), and each message has a specified arrival time \( at_{w,t} \). We can then define the arrival rate \( ar_{w,t} \) for current message \( m_{w,t} \) by the formula shown in Eq. (8):

\[
ar_{w,t} = \frac{1}{ar_{w,t} - ar_{w,t-1} + 1}
\]

In Eq. (8), if \( t = 1 \), the interval value becomes zero because \( w \) is a brand new word in the system. The arrival rate \( ar_{w,t} \) represents the reciprocal type of arrival gap \( (ar_{w,t} - ar_{w,t-1}) \) which could be normalized between 0 to 1. In order to reflect long time expectation of arrival rate of the word, the mean value for each word is calculated in an incremental manner.

\[
\mu_n = \mu_{n-1} + \left( \frac{1}{n} \right) (x_n - \mu_{n-1}) \quad (9)
\]

\[
E(ar_{w,t}) = \mu_{w,t} = \mu_{w,t-1} + \left( \frac{1}{n_{w,t}} \right) (ar_{w,t} - \mu_{w,t-1}) \quad (10)
\]

In this work we apply incremental mean (Finch, 2009) (i.e. Eq. (9)) in our weighting scheme to formulate equations of insertion (i.e. Eq. (10)), where \( ar_{w,t} \) is the new arrival rate of the word.

\[
RMOar_{w,t} = \sum_{n=1}^{k} \frac{ar_{w,t-n}}{k} \quad (11)
\]

Then the burst score is calculated as below:

\[
BS_{w,t} = \max \left\{ \frac{RMOar_{w,t} - E(ar_{w,t})}{E(ar_{w,t})}, 0 \right\} \quad (12)
\]

Therefore, we regard \( ar_{w,t} \) as the current observation result, to compare with expected value \( E(ar_{w,t}) \) of the word \( w \) at 7th arrival. In addition, we derive a formula Eq. (11) in which residual is the deviation between observation and expectation values. It should be noted that the result of Eq. (12) would not always be positive if the observation result is less than expectation value. In such a case, we define the word as a “falling word” at that time, and enable BS factor to be zero.

4.1.3. The TOP weighting factor

The second consideration in BursT weighting scheme is TOP (term occurrence probability) factor, which is formulized by the proportion of the term in the sliding window. For the operation of mining hot news topics from messages, if a word occurs in more messages, it is more likely to be a trending topic. Thus, the term occurrence probability corresponding to the word \( w \) at 7th arrival is formulized as below:

\[
TOP_{w,t} = P(w_t | c_t) = \frac{[m : w_t \in C_t]}{|C_t|} \quad (13)
\]

where TOP represents the probability of the word occurrence in the sliding window, and \( C_t \) denotes the message collection in the corpus collected from the time \( t-tw \) to current time. This factor would enable the weight of the word to grow with its occurrence frequency in messages, for identification of trending topics.

5. Experimental results

We designed two sets of experiments to evaluate (i) the effectiveness of BursT weighting method, and (ii) the validity of our event detection system, by means of analyzing the spatio-temporal impacts of some selected events detected by our system as our case studies. The two sets of experiments are described in the following subsections.

5.1. Experiments with BursT weighting method for event detection

In order to examine the system performance in reflecting the concept drift of words, we selected “Chile’s Rescued Miners” event as our case study. Our experiment started with the first miner was rescued at 2010/10/13 11:11 (GMT +8:00), until all miners were rescued at 2010/10/20 20:56 (GMT +8:00). Fig. 7 indicates that the intensity of inter-arrival gap the feature word “chile”, and it suddenly dropped at Oct 13 (GMT +8:00) when the event was happening.

Subsequently, we compared the weighting values of BursT and TFIDF methods, as shown in Fig. 8. We found that the incremental TFIDF can’t reflect the actual trends in sliding window algorithm, but TFPDF and our approach performed well in topic words. However, in the outcome of oral word analysis, we demonstrate the word “lol”, which both has a high density of collection and arrival rate, as an example, and obviously it might not be suitable to define “lol” as a valid feature. It is worth mentioning that some popular oral words might be easily over weighted in TFPDF because it places too much emphasis on document frequency. As shown in Fig. 8(d) and (e), the weighted number of “lol” in TFPDF is still higher than in incremental TFIDF even the event is still on the fly.

![Fig. 7. Inter-arrival gaps using the feature word “chile”](image-url)
messages, in order to enhance the understanding of the evolving events. This suggests that microblogs can be a deployable tool for situational awareness of unexpected sudden events.

- The aim of our approach is to offer a way to organize real-time event topics, allowing users to quickly figure out emerging events in the world. Compared with most commercial real-time search services, our method does not require any form of queries from users to fulfill information acquisition.

- Once the concepts behind the messages evolve with time, the underlying cluster structures may also significantly change with time. In the experiments, we have demonstrated that our system is able to adapt itself to support “concept drifting” in the microblogging text streams.

- The length of each message leads to a problem with the lack of semantic integrality in tweets. This makes it more difficult to design a reliable weighting and clustering algorithms. In this work, we overcome such challenges, by utilizing our developed dynamic weighting method. The preliminary results show that our algorithmic model has the potential for event detection and awareness.

- In addition to content-based techniques and our location approximation methods, part of the analysis of spatial areas in this work is based on time zone. In the experiments, the examples and the results we have demonstrated are based on large-scale events. Going further, we will improve our method to analyze events which are happened in small spatial areas in future work. In our experiments, we found that the time-zone-based approach appears to be insufficient to deal with such cases.

In our future work, we will mainly focus on three tasks on the topic. The first task is to conduct a detailed study on evaluating other candidate on-line clustering methods for fulfilling microblogging text mining, compared their performance with our developed density-based methods. On the other hand, some tweets do contain time zone and latitude/longitude data, which could be used to build some ground truth data against which the experimental results from text mining could have been compared in order to evaluate the performance (mainly in terms of accuracy) of the proposed method. However, so far in our work no attempt to evaluate the method beyond face validity was done, even though it would have been possible. Thus, the second task is to study on better ways for mining event locations and developing evaluation methods for geospatial prediction of events. In the third task, the utilization of geospatial name-entity recognition (NER) techniques would be helpful for location estimation, and could be incorporated with our approach for identifying geo-location information in the microblogging text mining process.

References


