BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **15**, No 3

Sofia • 2015

Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.1515/cait-2015-0042

Efficient Emergency Event Tracking Using Features of Web Data

Qunhui Wu[1], *Jianghua Lv*[1], *Hao Wang*[2], *Shilong Ma*[1]

[1]*School of Computer Science and Engineering, Beijing University of Aeronautics and Astronautics, Beijing 100191, China*
[2]*Beijing BoWang China Science and Technology CO&LTD, Beijing 100053, China*
Emails: wuqunhui126com@126.com

Abstract*: Huge amount of information has become now available in web services due to their popularity. This web data contains user-contributed information for a variety of emergency events. However, tracking these emergency events is often limited by the lack of efficient tools to analyze the potential events or topics over time, since these events are inherently difficult to predict due to the interference of other unpredictable evolutions. In this paper we propose a two-phase approach, in which we first introduce a novel extraction algorithm to acquire relevant web data and then we utilize a limit theory to determine the periodical convergence time of a specific event, and an event tracking model is constructed using the extracted web data. Based on the significance of multiple features weights and clustering solutions, the interplay between the ordinary events and latent events is discovered to efficiently track the emergency events. Finally, we conduct extensive experiments to verify the effectiveness and efficiency of our approach.*

Keywords*: Emergency event, event tracking, web data extraction, clustering.*

1. Introduction

In recent years, with the prevailing usage of web data repository over emergency field, the rescue and relief activities in emergency situations can be enhanced by tracking a particular event. In the meantime, instead of using web search engines,

people are more willing to obtain information about an on-going event. The information trends can be helpful for the aid-team to carry out the rescue and relief activity management during such an event. Web data extraction establishes the fundamental assignment for real-time event tracking, this technique targets at extracting event related data from web sources [1, 2]. Web data extraction usually interacts with a web source and extracts the data stored in it: for instance, if the source is an HTML web page, the extracted content could consist of elements in the page, as well as the full-text of the page itself [3, 4]. If the web data extraction cannot work well in real-time organizing for event evolution, it would greatly affect the efficiency and quality of event tracking, resulting in a tremendous negative influence for further emergency disposition activities [5]. In addition, the availability and analysis of this web data is an important requirement to understand complex information. It is common for web data sources to discuss information intensely for a period of time [6]. Identifying the time periods with a burst of activities, related to a specific emergency event, has been an important problem in analyzing time-stamped web data. The topic structure is powerful enough to capture multiple web data, in which a topic may appear and logical relationships among the web data as well. We consider a complex topic, which is a logical expression of the event and drive events [7]. By means of tracking the topic distributions of an event, we can know what guides the task to find the latent events according to the web news.

The emergency-oriented web data may change frequently and be replaced with new data, or they stay active and the context surrounding the event update dynamically. Thus the importance of web data extraction depends on the fact that the web data is steadily produced, shared and diffused online: web data extraction needs to collect field oriented data with limited human supervision. Furthermore, the extracted data might be further processed, normalizing the data in the most convenient structured format and stored for further usage [8, 9]. The design and execution of web data extraction approaches has been studied from wide perspectives and it leverages on authoritative methods coming from variety of disciplines, including Machine Learning, Text Information Processing and Natural Language Processing. All these are more important if the user is making use of the existing methods with relevant data. In fact, the recent surveys [1] have summarized that the use of state-of-the-art technologies cannot handle the correlated uncertain and unreliable emergency events. Given the task of tracking an emergency event, one of the most important problems is how the event tracking model predicts the drive events to the specific event timely, or analyzes an event to identify a right trend and form the current summarization [10]. For the purpose of enhancing the predictability of an event, an efficient approach is to meet the demands in terms of semantic topic distribution and historical interrelation of events. By means of mining the TF-IDF (Term Frequency–Inverse Document Frequency) features and estimating the drive events over the current data, the latter programming technique being the most widespread currently, to direct the data processing platform to finding indication of the drive events. Here, TF-IDF is a numerical statistic that is intended to reflect how important a word is to a document in a collection.

Although the event tracking is a basic operator in analyzing the evolutionary process, it is not trivial to capture possible events over real-time data extraction, using existing techniques. Based on the example about the event and its drive events, Table 1 includes several natural hazard events and their drive events published from actually existing ones.

Table 1. Natural hazard events and their drive events published from actually existing

Natural hazard events	Table column head
6.5 Ludian County, Yunnan Region, 2014-08-03 16:30:00	2014-08-10 16:00:00, 1286 aftershocks;
	Virtual vibration net for Ludian County is rebuilt to finish;
	The supplies for rescue team are abundant...
4.3 Zhuolu County, Hebei Region, 2014-09-06 18:37:00	2014-09-07 06:30:00, 15 aftershocks, with the biggest being only M 1.5...
6.6 Gujing Country, Yunnan Region 2014-10-07 21:49:00	2014-10-13 14:00:00, 1199 aftershocks;
	2014-10-08 00:00:00, one death...
...	...

Any particular event and trends data in Table 1 contains entries that are chronologically ordered and each entry contains its evolutions and its corresponding rescue operations (if any). In the first and the third disaster events entries of Table 1, we can observe that the earthquake events have many related data, and the second one is relatively less discussed. Namely, the greater magnitude of the earthquake event, the more is the complexity for derivative events, the more details and context can provide web. If we use the existing new event mining techniques, such as New Event Detection (NED) models, to forecast the possible event, a simple solution is to consider each event and the corresponding data. However, for the online updateable data in real-time, NED models might not identify a new event timely because there is no incremental update action appearing in NED models [11].

To the best of our knowledge, no existing algorithms can be applied into our problem directly for three reasons. First, the algorithms cannot handle the correlated uncertain and unreliable emergency event. Secondly, web data extraction requires similar query strategies, which is different from our similarity calculation periodically between the web data and on-going features. Thirdly, if the existing event tracking systems are extended to our problem directly, we cannot efficiently track the drive events for an emergency event.

The key challenges we can encounter in the design of Event Tracking over Emergency-oriented Web Data method can be summarized as follows:

1. Web data extraction for a specific emergent event techniques implemented often requires the supervision of human experts. To identify a reasonable trade-off between the process of creating automated extraction algorithm and the requirement of achieving accurate performance is a related challenge.

2. Timeliness is a significant characteristic for web data of an emergency event. Therefore, approximate attempts to end the web data extraction should be confirmed.

3. Design an efficient model to track the event over emergency-oriented web data has two critical issues. The first is how to process the massive web data and determine the feature of the emergency event. The other one is how to establish an efficient event tracking approach, which can help analyze the current event and estimate the drive events in order to organize the data extraction adaptively.

The aforementioned challenges reflect the high demand of an advanced event tracking model which is tailored to meet the emergency-oriented web data form and large data size. The proposal in our study adopts the improved data extraction method, the original pseudo-codify of weighted tree matching algorithm recently presented in [12]. However, uncertainty has played an important role in data extraction of emergency-oriented, since if the event occurs, many induction factors of evolution must be taken into account and some of them are dependent on the specific event tracking process. We design and present the Emergency Event Tracking model (EET), which utilizes multifarious features clustering to discover the latent topic distributions from massive emergency-oriented web data with high relevancy. Moreover, in order to achieve convergence of the extraction process, Web Data extraction for Emergency Event (WDEE) integrates the limit theory into a stochastic evolution framework, in which a series of online web data updates leads to the staged point of the emergency event.

In the rest of our paper we summarize the related work in Section 2. Section 3 is devoted to provide problem definitions and framework, which are helpful to understand the approaches proposed. In Section 4 we introduce web data extraction algorithm WDEE for collecting evolutionary data. In addition, EET model is presented to analyze the event and predict latent events in Section 5. In Section 6 we report our experimental results. Finally, we summarize the conclusions in Section 7.

2. Related works

Our work overlaps two critical areas of research: web data extraction and event tracking. In Section 2 we review the related existing work concerning the above aspects.

2.1. Web data extraction

Web data extraction is a long-existing technique and has been practiced for centuries. Web data extraction means extracting data from web, which has been designed to resolve specific issues and operates in corresponding domains. Many previous studies of the web data extraction indicate that the accurate extraction relevant information can help to locate eligible resources and allow the web user to obtain the autonomy of appropriately conditions [9, 13]. Good overviews of the advances in web data extraction are given in [14]. It presents a survey that offers a strict taxonomy to classify web data extraction systems. A set of criteria and a qualitative analysis of various web data extraction tools are introduced. However, these tools were not well resolved in extracting the information with redundancy of natural signals. He et al. [15] overcame similar deficiencies by compressing the

redundancies and noises. Moreover, taking benefits of randomness is quite useful for data extraction in both compressed sensing and sparse representation-based pattern recognition.

The theme of Web Data Extraction is a computer software technique of extracting information from a website, which is covered by a number of reviews. A tri-dimensional categorization of Web Data Extraction systems is introduced by C h a n g et al. [16]. This system based on task difficulties, techniques used and degree of automation. F i u m a r a [17] and F l e s c a et al. [18] applied the above criteria to classify four state-of-the-art Web Data Extraction systems. An unstructured data management system was proposed in [19], which analyzes raw web information, extracts from it some structure, integrates the structure and uses the integrated structure to build a database. To the best of our knowledge, the survey from E m i l i o and P a s q u a l e [1] is the most recently updated survey on the domain.

The methodology of web data extraction is now receiving countless success in many domains, such as sentiment analysis [20], taxonomies [21] and so on. All of these described approaches pointed out that our work must understand the dynamic web data at a planetary scale and in a real time-fashion [22].

2.2. Event tracking model

There have been extensive studies on event tracking, addressed in uncertain environment for a long time [23]. We will not survey the entire field here, but instead zoom-in specifically onto the work related to event tracking. Event tracking as a state automation model was proposed by K l e i n b e r g [24] to detect bursty activities from an information arrival stream, by assuming that the rates of messages are determined by underlying hidden states. G u o, H u a n g and D i n g [25] aimed at continuously detecting an emergency event forming object data streams, which referred to a series of clusters that increase or decrease rapidly for a period of time. T o n g, C a o and C h e n [26] proposed the TCS (Topic Crowd Service) model together with the BPE (Block Parameter Estimation) algorithm and the pSketch (pair Sketch) structure that resolved the problem of topic discovery and tracking over massive crowd-oriented service data. The above research work provides solutions with solid performance.

Web service in emergency application is getting increasingly important, comparable to its intense utilization in multifarious areas to communicate different situation, news and contextual information. Extensive research has been done on web resources in emergency events by studying the social media data [27, 28]. The study identifies and tracks the types of emergency messages related to emergency situations. The use of Flickr photographs features for event detection and tracking was discussed in [29]. Here, Flickr is a popular website for users to share and embed personal photographs, and an effective online community. Moreover, several researches worked on tracking an emergency event by clustering features, [30] used self-organizing maps for clustering the features into clusters, representing latent events, which was not suitable for real-time web data. To overcome this problem, TF-IDF weight vector and cosine similarity metric were utilized to generate features

dynamically, which detected the event by the incremental updatable framework [31].

However, the subsistent models are not good enough to track the real emergency event; it depends on the way of performing data processing scan web data repetitively. These approaches not only failed to decrease I/O resources consumption, but also add overhead to algorithm processing time. In brief, their algorithms would disable the advantages to a certain extent.

Thus, the difference between researches of the above works and our research lies in the research objective. The key contribution of this work is our novel proposal to overcome the lack of drive events tracking with real-time emergency-oriented web data. Moreover, our work is devoted to enhancing the efficiency of extracting data process and estimating the latent drive events through WDEE algorithm and EET model. In addition, our work is able to determine the periodically convergence time of web data extraction process.

3. Problem formulation and framework

3.1. Problem definitions

In this section we formally define the related concepts and the task of tracking an event over emergency-oriented web data. Let us begin with defining a few key concepts.

Definition 1 (Web Resource). Let $R=\{R_1,R_2,\ldots,R_m\}$ be a group of web resources. The web resource refers to a source of web data, which is composed of web pages.

Definition 2 (Document Sequence). Let $D=\{D^0,D^1,D^2,...,D^T\}$ be a sequence of web page collections, where D^k is the set of documents published until time t_k. We further denote $D^k=\{d_0^k,d_1^k,\ldots,d_n^k\}$, where d_i^k is the document associated with the event.

Definition 3 (Emergency Event Model). The emergency event model is defined as a 3-tuple $E=\{T,L,\Theta\}$. We call E^0 the primitive emergency event, which is independent on the web data. E^0 can either be specified by the users or automatically discovered by an event detection algorithm [32]. E^k corresponds to the version of E at time t_k, and E^k is dependent on the web data, which indicates the major aspects of the event.

Here:

T represents the time when the event is taking place;

L represents the location where the event is taking place;

$\Theta=\{\Theta_0,\Theta_1,\ldots,\Theta_j,\ldots\}$ is a sequence of topics; each topic is represented as $\Theta_j=\{\mathrm{HF}_j,\mathrm{NF}_j\}$, where $\mathrm{HF}_j=\{hf_1,hf_2,\ldots,hf_x\}$, each history feature is represented as $hf_a=\{hw,\ if\}$;

hw is a history word, if is the weight of this word;

$\mathrm{NF}_j = \{nf_1, nf_2, \ldots, nf_y\}$, each new feature is represented as $nf_a = \{nw, if\}$;

nw is a new word, if is the weight of this word.

In the Emergency Event Model, the weight of an arbitrary feature element is described as follows. If if is the weight of a history feature, if is the frequency of hw and SW, SW is a word set which has similar semantic words for *hw*. If if is the weight of a new feature, if is the frequency of *nw* and SW, SW is a word set which has similar semantic words for *nw*.

Based on the definitions above given, we can now formally define the major task in the problem of tracking an emergency event over web data as follows.

Tracking an emergency event over Web Data problem:

Given a specific emergency event, we are required to extract relevant data over time and track the latent events over this real data.

3.2. Framework

An overview of our proposed emergency event tracking over web data is illustrated in Fig. 1. We assume that the input to the system is a web data streaming that has been the acquired information from web data resources with a specific event. Here the web data streaming interface module can be easily implemented using APIs of the web networking site. This could be relevant web data, returned by computing the similarity between the information and the event. For each web data, a pre-processing step is performed to information standardization, which involves some basic theories and technologies. This step is important because it helps us to alleviate problems associated with the informal messages. Given an input web data, multifarious features in this input information are utilized to generate clusters. Matching topics are then assigned to for tracking the event.

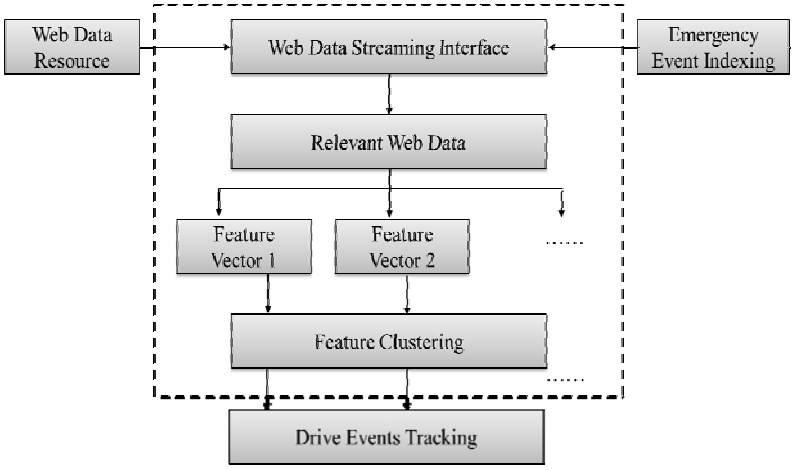

Fig. 1. Emergency event tracking of a Web data framework

1. *Relevant web data extraction*. As described earlier, our goal is to extract the closely related web data for a particular emergency event. The stream of relevant

web data can potentially be dynamically generated; we have decided to calculate the similarity from the existing techniques. For efficient processing of the web data, the information often makes use of standardized representation (e.g., title, date, keywords, context and so on). The use of the distance measure between the web data of the adjacent time-periods is dynamically generated given a specific event. Limit theory will be applied to estimate first the periodical convergence time of this process. In future work, the standardized web data will be analyzed at the next step, which is explained in detail in Section 4.

2. *Event tracking*. Having obtained the relevant web messages for further processing, the features are generated according to the significance of words, which is employed as a weight to adjust the contributions of each feature. The weight measurement adopts an improved TF-IDF and entropy theory to calculate. These features give the users a high-level overview of the nature and content of the relevant web data. This can then guide subsequent clustering to acquire feature clusters. We believe that this will help the users get better understanding of the possible drive events of the topic distribution. More details of the event tracking solution we propose will be given in Section 5.

4. Web data extraction approach

In this section we discuss the details of Web Data extraction for an Emergency Event algorithm (WDEE). Section 4.1 first defines the timeliness of the extracting data procedure. Section 4.2 shows the procedure of utilizing a timeliness definition to extract data for an emergency-oriented event.

4.1. Timeliness definition

The related web data of emergency events has a prominent characteristic of the lifecycle, namely the timeliness of emergency-oriented web data. It is necessary for the web data extraction procedure to define the timeliness and realize it in details.

The time interval of updating the web page over web source is Δt, the event occurs at t_0, the convergence time of extracting the web data is t_{end}. Assume that there exists N_0, N_0 is sufficiently large, such that t_{N_0} is the end of the emergency event, that would indicate the web data for an event almost never published in this period. When $t_{N_0} = t_{\text{end}}$, the convergence time should be confirmed. In addition, the emergency event tracking task should be completed during the period of Δt, namely D^{end} is the document set of the first convergent period for the emergency event.

In addition, the emergency event tracking task must be completed during the time interval Δt. The first periodically timeliness characterization is introduced as follows. Assume that the time point of updating the web page for a web source is represented as $t_0, t_1, \ldots, t_i, t_{i+1}, \ldots (t_{i+1} = t_i + \Delta t)$. The corresponding document set to the time point is $D^0, D^1, \ldots, D^i, D^{i+1}, \ldots$ The upper and lower limits of the sequence above are represented as:

$$\limsup_{i\to\infty} D^i = \bigcap_{i=1}^{\infty}\bigcup_{j=i}^{\infty} D^j,$$

$$\liminf_{i\to\infty} D^i = \bigcup_{i=1}^{\infty}\bigcap_{j=i}^{\infty} D^j.$$

If $\limsup_{i\to\infty} D^i = \liminf_{i\to\infty} D^i = \psi$, then there exists $\lim_{i\to\infty} D^i$ such that $\lim_{i\to\infty} D^i = \psi$, namely there exists t_{end} such that $\lim_{i\to\infty} D^{\text{end}} = \psi$.

Similarly to the definition of the condition with the web resources $R = \{R_1, R_2, ..., R_m\}$, the time interval of updating the web page over web sources *R* is $\Delta t_1, \Delta t_2, \ldots, \Delta t_m$, the convergence time of extracting the web data for the emergency event is $t_{\text{end}_1}, t_{\text{end}_2}, ..., t_{\text{end}_m}$. To facilitate the assessment and validation of the algorithm, the time interval of updating the web page over web sources *R* is assumed as $\Delta t = \min(\Delta t_1, \Delta t_2, \ldots, \Delta t_m)$, the convergence time of extracting the web data for an emergency event is $t_{\text{end}} = \max(t_{\text{end}_1}, t_{\text{end}_2}, ..., t_{\text{end}_m})$. The document, set at the time point t_i is $D^i = D_1^i \cup D_2^i \cup \ldots \cup D_m^i$, where D_k^i is denoted as the document set of the web sources R_k at time point t_i.

The timeliness discusses that tracking subsequent events of a specific emergency event increment within critical duration. This duration is a key period for acquiring development of the disaster, and directing the following rescue activities. Our work estimates the periodically coverage time after the event occurs, especially suitable for a current event in real-time tracking. We are committed to the first critical period of convergence.

4.2. Web data extraction for an emergency event algorithm

As discussed in Section 4.1, calculating the convergence time is an important problem. On this basis, we use the theory of convergence to get an approximate result. As the aforementioned definitions, each document is critical information for emergency event tracking. In order to mine the most relevant information, we must improve the existing algorithm for adapting the evolution of the emergency event. In this section we give the Web Data Extraction for Emergency event algorithm (WDEE). Then, the pseudo code of WDEE is shown in Algorithm 1.

Algorithm 1. WDEE

Input. Emergency event $E = \{T, L, \Theta\}$, web resources $R = \{R_1, R_2, \ldots, R_m\}$, parameter estimation of end time N_0, the threshold of similarity δ,

Output. The document set of the convergence time D^{end},

Step 1. $D^0 \neq \varnothing$; *i*=0;

Step 2. if $t_i < t_{N_0}$ then

Step 3. for each web resource $R_k \in \{R_1, R_2, \ldots, R_m\}$ do

Step 4. if $\text{WeightedTreeMatching}\left(\Theta, d^i, t\right) > \delta$ then

Step 5. $D_k^{i+1} \leftarrow D_k^i \bigcup d^i$; $D_k^i \leftarrow D_k^{i+1}$;

Step 6. $D^i \leftarrow D_k^i \bigcup D^i$;

Step 7. else

Step 8. while $\text{Dis}\left(D^i, D^{i+1}\right) \neq 0$ do

Step 9. execute Steps 3-6;

During the web data extraction for a specific emergency event, each document provides assessment basis for a topic. Thus, it is crucial for extracting relevant data over time, i.e., it is necessary to calculate the similarity between document d^i and topics Θ. However, the existing web data extraction only handles dynamic data extraction rather than a specific event related. For the purpose of filtrating web data, the topics are utilized to calculate similarity. Moreover, the history and new features are updated in real time, which serve for WDEE algorithm. Inspired by the *WeightedTreeMatching* algorithm [1], which is one of the fastest extracting algorithms, we introduce topics of the event to obtain efficient web data. The advantage of the weighted tree matching is that it better reflects a measure of similarity between the topics and web data. In addition, we organize the documents according to time, which enhances the performance of the expired web data. The algorithm is called *WeightedTreeMatching* algorithm subroutine, including time constraint in line 4.

In order to estimate the convergence time of WDEE algorithm, the distance between the documents method [33] is introduced in our algorithm. Hence, the semantic extension is not considered in $\text{Dis}\left(D^i, D^{i+1}\right)$ method. In line 8 of [33], some subtle adjustment is taken, so that the frequency of the similar semantic word should be cumulated when calculating the similarity between sentences. If WDEE algorithm reaches convergence at time point t_i, EET model should be quitted before time point t_{i+1}, namely EET must analyze all the documents D^i during Δt.

5. Emergency event tracking model

In this section we introduce the details of the Emergency Event Tracking model (EET). Section 5.1 first shows the calculation formulas of weight. Section 5.2 gives the standardized representation of the web data. Section 5.3 gives the procedure of features clustering in order to detect topic distributions. Section 5.4 shows the procedure of tracking an event over emergency-oriented web data.

5.1. Weight formulation

In this section we formally define the calculation formulas of weight during the process of an event tracking model over emergency-oriented web data.

1. The weight of the feature vector F^i to document set D^i at time point t_i is:

(1) $$\text{weight}_{\text{fd}}\left(F^i\right)=\sum_{f_k^i\in F^i}\text{weight}_{\text{fd}}\left(f_k^i,D^i\right),$$

(2) $$\text{weight}_{\text{fd}}\left(f_k^i,D^i\right)=1-\frac{H\left(D^i\middle|f_k^i\right)}{H\left(D^i\right)}.$$

In formula (2) $H\left(D^i\middle|f_k^i\right)$ denotes the entropy value of documents D^i to feature f_k^i, $H\left(D^i\right)$ means the entropy value of documents D^i,

$$H\left(D^i\middle|f_k^i\right)=-\sum_{d_j^i\in D^i}p\left(d_j^i\middle|f_k^i\right)\times\log p\left(d_j^i\middle|f_k^i\right),$$

$$H\left(D^i\right)=\log\text{count}\left(D^i\right),$$

$p\left(d_j^i\middle|f_k^i\right)$ denotes the probability of the feature f_k^i in document d_j^i, $\text{count}\left(D^i\right)$ denotes the number of document in documents D^i. By $p\left(d_j^i\middle|f_k^i\right)$ is denoted the ratio of $\text{fre}\left(d_j^i\middle|f_k^i\right)$ to $\text{fre}\left(f_k^i\right)$, where $\text{fre}\left(d_j^i\middle|f_k^i\right)$ is the frequency of f_k^i in d_j^i, $\text{fre}\left(f_k^i\right)$ is the sum of frequencies of the feature f_k^i in document set D^i.

2. The weight of the document D^i to the feature vector F^i at time point t_i is:

(3) $$\text{weight}_{\text{df}}\left(D^i\right)=\sum_{d_j^i\in D^i}\text{weight}_{\text{df}}\left(d_j^i,F^i\right),$$

(4) $$\text{weight}_{\text{df}}\left(d_j^i,F^i\right)=1-\frac{H\left(F^i\middle|d_j^i\right)}{H\left(F^i\right)}.$$

Here $H\left(F^i\middle|d_j^i\right)$ denotes the entropy value of the features F^i to document d_j^i, $H\left(F^i\right)$ denotes the entropy value of the features F^i.

$$H\left(F^i\middle|d_j^i\right)=-\sum_{f_k^i\in F^i}p\left(f_k^i\middle|d_j^i\right)\times\log p\left(f_k^i\middle|d_j^i\right),$$

$$H\left(F^i\right)=-\sum_{f_k^i\in F^i}p\left(f_k^i\right)\times\log p\left(f_k^i\right),$$

$p\left(f_k^i\middle|d_j^i\right)$ is defined as the ratio of $\text{fre}\left(f_k^i\middle|d_j^i\right)$ to $\text{len}\left(d_j^i\right)$; $p\left(f_k^i\right)$ is defined as the ratio of $\text{fre}\left(f_k^i\right)$ to $\text{fre}\left(F^i\right)$, where $\text{len}\left(d_j^i\right)$ is the length of d_j^i, $\text{fre}\left(F^i\right)$ is the sum of frequencies of all features F^i with t_i in documents D^i.

3. The weight of F^i to the capability of identifying the drive event with t_i is

(5) $$\text{weight}_{\text{d}}\left(F^i\right)=1-\sum_{f_k^i\in F^i}p\left(f_k^i\right)\times\log p\left(f_k^i\right),$$

where $p\left(f_k^i\right)$ is the ratio of $\text{dcount}\left(f_k^i\right)$ to $\text{dcount}\left(F^i\right)$; $\text{dcount}\left(f_k^i\right)$ is the sum of frequencies of the f_k^i in D^i, $\text{dcount}\left(F^i\right)$ is the sum of frequencies of F^i in D^i.

5.2. Standardized representation of web data

In this section the standardized representation of the web data is given. Then we formally define the normalization of document *d* during the process of event tracking over emergency-oriented web data.

To efficiently represent document *d*, a Vector Space Model (VSM) [34] is utilized to normalize document *d*.

In Equations (6) and (7), T is time (when the event is taking place); L is location (where the event is taking place); Θ is a sequence of topics; F_T^{AT} denotes the frequency value, and F_T^{AT} is the term frequency of item T in the information title AT. The other items (F_L^{AT}, F_Θ^{AT} and so on) have similar explanations.

Document *d* is defined as *d*={AT, AK, AF, AD}. We call AT the title of document *d*, AK are the keywords, AF and AD are the first paragraph and description, respectively. The standardized representation of the document is

$$d=\begin{cases}\text{AT}=\left\{\left\langle T^{\text{AT}},F_T^{\text{AT}}\right\rangle,\left\langle L^{\text{AT}},F_L^{\text{AT}}\right\rangle,\left\langle \Theta^{\text{AT}},F_\Theta^{\text{AT}}\right\rangle\right\}\\ \text{AK}=\left\{\left\langle T^{\text{AK}},F_T^{\text{AK}}\right\rangle,\left\langle L^{\text{AK}},F_L^{\text{AK}}\right\rangle,\left\langle \Theta^{\text{AK}},F_\Theta^{\text{AK}}\right\rangle\right\}\\ \text{AF}=\left\{\left\langle T^{\text{AF}},F_T^{\text{AF}}\right\rangle,\left\langle L^{\text{AF}},F_L^{\text{AF}}\right\rangle,\left\langle \Theta^{\text{AF}},F_\Theta^{\text{AF}}\right\rangle\right\}\\ \text{AD}=\left\{\left\langle T^{\text{AD}},F_T^{\text{AD}}\right\rangle,\left\langle L^{\text{AD}},F_L^{\text{AD}}\right\rangle,\left\langle \Theta^{\text{AD}},F_\Theta^{\text{AD}}\right\rangle\right\}\end{cases}. \tag{6}$$

In the attributes AT, AK, AF, AD of messages, the features in the title are relatively more crucial, the importance of the features in other attributes is taken into consideration as well. Then the features in the title are multiplied by the weight coefficient $\gamma\left(\gamma\geq 1\right)$. We obtain

$$d=\begin{cases}\text{AT}=\left\{\left\langle T^{\text{AT}},F_T^{\text{AT}}\times\gamma\right\rangle,\left\langle L^{\text{AT}},F_L^{\text{AT}}\times\gamma\right\rangle,\left\langle \Theta^{\text{AT}},F_\Theta^{\text{AT}}\times\gamma\right\rangle\right\}\\ \text{AK}=\left\{\left\langle T^{\text{AK}},F_T^{\text{AK}}\right\rangle,\left\langle L^{\text{AK}},F_L^{\text{AK}}\right\rangle,\left\langle \Theta^{\text{AK}},F_\Theta^{\text{AK}}\right\rangle\right\}\\ \text{AF}=\left\{\left\langle T^{\text{AF}},F_T^{\text{AF}}\right\rangle,\left\langle L^{\text{AF}},F_L^{\text{AF}}\right\rangle,\left\langle \Theta^{\text{AF}},F_\Theta^{\text{AF}}\right\rangle\right\}\\ \text{AD}=\left\{\left\langle T^{\text{AD}},F_T^{\text{AD}}\right\rangle,\left\langle L^{\text{AD}},F_L^{\text{AD}}\right\rangle,\left\langle \Theta^{\text{AD}},F_\Theta^{\text{AD}}\right\rangle\right\}\end{cases}. \tag{7}$$

From the web data related to a particular emergency event posted on various web resources, the goal is to track the event associated with the emergency event. The problem of the standardizing document is notably important for discovering and organizing the latent drive events in a continuous document stream. Using

standardized representation of the document is suitable for a new-centric report in real-time environment.

5.3. Features clustering of topic distributions

Using the feature vectors of the relevant documents, the similarity measures between these different features can be generated. Each cluster is the representation of one topic for an event or drive event. Let us say that $F_1,\ldots,F_k$ are the feature vectors of documents $D_1,\ldots,D_k$, using the appropriate similarity measures, different topics $\Theta_1,\ldots,\Theta_c$ can be formed. Here centroid similarity technique [29] is used.

1. Each document D_i is considered to compute the similarity against each cluster $\Theta_1,\ldots,\Theta_c$.

2. A new cluster is formed containing this data $D_i, 1\le i\le k,$ and with the centroid value as the value itself, for one reason, no cluster is found whose similarity to document D_i is greater than threshold λ.

3. Otherwise, document D_i is assigned to the cluster $\Theta_j, 1\le j\le c,$ which gives the maximum similarity value and after adding this document D_i to the cluster Θ_j, a new centroid value of cluster Θ_j is computed. Depending on the feature of the document being considered, the centroid for the cluster is the average of the improved TF-IDF score per feature $F_m, 1\le m\le k,$ according to Equation (2).

Each cluster is about a topic using the data with the event or drive event. In the case under consideration it is useful to measure the topic distributions according to features clustering.

5.4. Event tracking model over emergency-oriented web data

To meet the demands of the emergency event that could occur at any time, it is important to complete parallel execution of WDEE and EET. When the emergency event occurs, the web data extraction procedure must be started, meanwhile, the event tracking over related web data must be started as well [35]. If the latent topics are discovered, this must be increased for the subsequent extraction process. In addition, the new topics provide the foundation for generating a drive event. In this section, the Event Tracking model over Emergency-oriented web data (EET) is proposed. Then, the pseudo code of EET is shown in Algorithm 2.

Algorithm 2. EET

Input. Emergency event $E=\{T, L, \Theta\}$, The document set D^i, the threshold of the new word φ, the threshold of the new word feature ρ, the threshold of the generating drive event μ,

Output. The topics of the emergency event Θ; the drive event $E^{'}$.

Step 1. $i=0$; $\mathrm{NF}^i=\mathrm{NF}_0^i\bigcup\mathrm{NF}_1^i\bigcup\ldots\bigcup\mathrm{NF}_j^i\ldots\leftarrow\varnothing$;

Step 2. for each document $d_k^i\in D^i$ do

Step 3. utilize a word segmentation tool to segment the word document d_k^i, $W^i \leftarrow \text{Segmentation}\left(d_k^i\right)$;

Step 4. calculate the weight of the history feature vector HF^i to document d_k^i according to formula (2),

$$\text{weight}_{\text{fd}}\left(\text{HF}^{i+1}\right) \leftarrow \text{weight}_{\text{fd}}\left(\text{HF}^i\right) \bigcup \text{weight}_{\text{fd}}\left(\text{HF}^i, d_k^i\right);$$

Step 5. if $\text{weight}_{\text{fd}}\left(W^i, d_k^i\right) > \varphi$ and $\text{weight}_{\text{df}}\left(d_k^i, W^i\right) > \varphi$ then

Step 6. $NF_a^{i+1} \leftarrow w_c^i$;

Step 7. if $\text{weight}_{\text{df}}\left(D^{i+1}, \text{subset}\left(\text{HF}^{i+1}\right)\right) > \rho$ and $\text{weight}_{\text{df}}\left(D^{i+1}, \text{subset}\left(\text{NF}^{i+1}\right)\right) > \rho$ then

Step 8. if $\text{weight}_d\left(\text{subset}\left(\text{NF}^{i+1}\right)\right) > \mu$ then

Step 9. generate drive event $E^{'}$; $T_D \leftarrow t_{i+1}$; $L_D \leftarrow L$;

Step 10. $$\Theta_D \leftarrow \left(\bigcup_{\theta_b^{i+1} \in \text{subset}\left(\text{HF}^{i+1}\right)} \theta_b^{i+1}\right) \bigcup \left(\bigcup_{\theta_a^{i+1} \in \text{NF}^{i+1}} \theta_a^{i+1}\right);$$

Call a feature clustering procedure to determine the topic distribution for each event;

Step 11. else

Step 12. $\text{NF}^{i+1} \leftarrow \text{NF}^i \bigcup w_c^i$.

Due to the high volume of online web data, it is infeasible to extract all the data to be analyzed because of the high cost. Thus, WDEE algorithm extracts the related web data for the emergency event over time. Moreover, semi-structured data is normalized before being stored in a database, which is known as documents. Applications of natural language processing [36] and word segmentation [37] are utilized in Algorithm 2. Fortunately, the word features with significant weights provide the most important information for discovering the latent topics. So we propose to give priority to capture the contents of these word features. Since finding significant word features is the foundation of generating drive events, it efficiently selects the word features of symbolic meaning from documents. Since each document is usually topically coherent, we impose the constraint that the word features in each document must share the same topic, so that to capture the semantic coherency. Furthermore, the latent topics are influenced by the corresponding word features with similar semantic, which are utilized to determine the derived event.

Especially, topic distribution is determined by the document itself, an emergency event often has original topics and a new topic. New topics often indicate the trend of the emergency event, if these new topics constitute the condition for generating a latent event, we combine the new topics and the original topics to determine the drive event. In other words, the new topics and the original topics induce a new event to guide the data extraction.

6. Experimental study

In this section we describe experiments conducted to evaluate the performance of WDEE and EET with real-time data exaction. We choose state-of-the-art new event detection methods as baseline methods [11]. It should be noted that the NED (New Event Detection) was chosen as the baseline method. The experiments were organized as follows. We first employed standard improved Accuracy, Recall Rate and *F*-measure to verify the first periodical convergence by introducing the theory of a limit superior and a limit inferior. This theory was not only applied to the process of web data extraction, but also employed by drive events discovery and tracking. Then, we conducted experiments to compare the performance of our EET model with the performance NED models in the aspect of tracking emergency events. Finally, the parameter settings of EET were investigated. All the experiments are executed on a PC with CPU Inter(R) Core(TM) i7-2600, frequency 2.9 GHz, memory 8.00 GB. The operation system is Microsoft Windows 7 Enterprise Edition. The development software is JRE 1.6.0 14, using Java language.

In our study we collected a real-world dataset from a popular search engine, Google and BaiDu. We will explain how the dataset was built up. We selected emergency events from an events trending list and crawled a collection of information related to these events using the search engine APIs. The real-world dataset consisted of emergency events observed at different focuses including drive information during the whole year of 2014.

The parameters are selected according to the experiment many times. The timespan is set to 10 minutes; the parameter estimation of the end time N_0 is set to 1440 (about ten days); the threshold of similarity δ is set to 0.6; the threshold of the new word φ is set to 0.3; the threshold of the new word feature ρ is set to 0.34; the threshold of the generating drive event μ is set to 0.29; the features in the title are multiplied by the weight coefficient γ set to 2.

We first demonstrated WDEE performance concerning the timeliness aspects. Specifically, we implemented WDEE with several emergency events, the events include the earthquake in Ludian County that occurred at 2014-08-10 16:00:00 as E-L, the earthquake of Zhuolu County that occurred at 2014-09-07 06:30:00 as E-Z, the earthquake of Gujing Country that occurred at 2014-10-13 14:00:00 as E-G and the earthquake of Taiwan that occurred at 2014-05-21 08:21:00 as E-T. In the following experimental results, "*vs*" is the abbreviation of "*versus*".

Fig. 2 shows WDEE convergence of the number of extracting web data. From the results we can observe that with the increase of time, the numbers of extracting web data for the events converge gradually. The convergent time of extracting web data approximates one week. In addition, we observe that the more damage the emergency event does, the more convergent-time will be needed. We then present Fig. 3 to show the WDEE convergence measurements of *F*-measure over time. The above experimental result shows that convergence with limit theory is correct. *F*-measure for these fluctuating values are verified in limits to WDEE method, in reality the assumptions themselves have co-evolutionary mechanisms that flux over time.

Then, we aim to measure *F*-measure *vs* web source size in different emergency events, the result is shown in Fig. 4. It can be seen that with the increase of the web source numbers, the *F*-measure is basically stable. The *F*-measure of four events is almost the same. The phenomenon above indicates that as the web source size increases, the efficiency of WDEE will not be influenced severely.

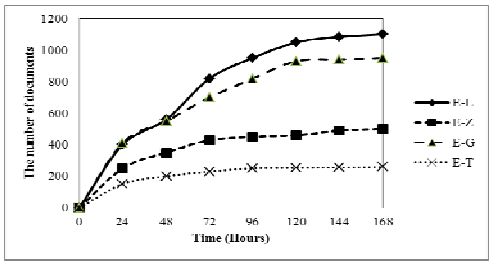

Fig. 2. Convergence of the number of documents

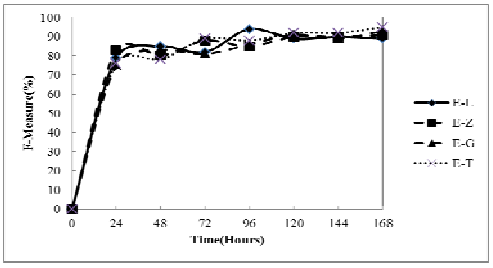

Fig. 3. Convergence of *F*-Measure

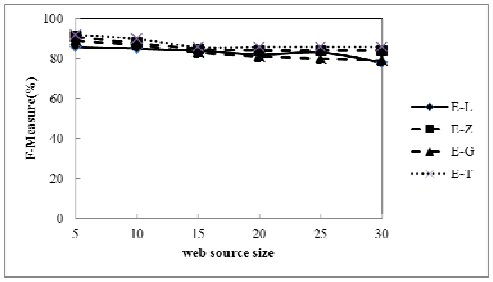

Fig. 4. *F*-measure *vs* web source size

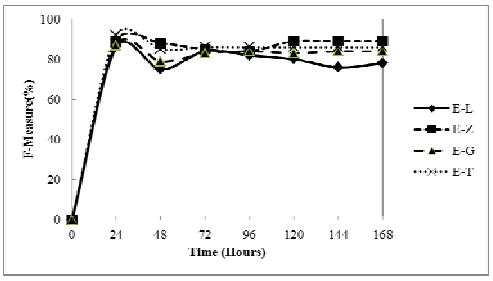

Fig. 5. EET convergence measurements of *F*-measure

Moreover, we wish to measure how efficient the proposed EET model is in terms of predicting a future event based on the portion of available tasks and drive events. We are interested to investigate whether the timeliness of WDEE has an effect on the quality of event tracking, thereby affecting tracking convergence performance. We then present Fig. 5 to show EET convergence measurements of *F*-measure over time. The above experimental results show that EET convergence has similar results, the limit theory is also applied to analysis and tracking research. With the development of the event, the related web data obtained could become less and less, until no more related web data are published in the first convergent period. With this in mind, we performed additional experiments to study the number of drive events for these four emergency events. The phenomena of the number of drive events *vs* time is shown in Fig. 6. With the development of the event, the drive events stabilize gradually after the extraction. It can be seen that the number of drive events for Ludian earthquake event is more than in the other three events, because this event was with the biggest magnitude.

We use the existing new event mining techniques, such as NED model with different feature weight calculation measures, to forecast the possible event; a simple solution is to consider each event and the corresponding data. NED-1 adopts a basic feature-weight calculation for NED model; NED-2 and NED-3 utilize the improved feature-weight calculation method. Fig. 7 shows the Accuracy *vs* time increase on EET model and NED models. It can be observed that EET model

always outperforms NED models in Accuracy. For the online updateable data in real-time, NED model with different feature weight calculations might not identify a new event timely because there is no incremental update action appearing in NED models [11]. This observation reveals that NEDs are not used with incremental update similarity measures. Thus, the experimental result is shown to verify that EET model has an obvious advantage in online updateable data.

We test EET and NED models with different event size to the estimators to show the Accuracy of different algorithms. In Fig. 8, although the emergency events are random selected from a database, the Accuracy of our model always outperforms NED-1, NED-2 and NED-3. That is because the topics of the event can be updated adaptively in EET model. Our model collects drive event relevant web data dynamically, in particular, since the updatable features and their weights tend to generate accurately, most feature vectors are clustered to determine the topic distribution of the drive event. Accordingly, many topic distributions belonging to a specific event cannot be determined simply by weights calculation of NEDs. In contrast, the regrouped feature vectors can provide contexts with richer co-occurrence features for each cluster; moreover, each cluster is the representation of an event topic. We further demonstrate in Fig. 9 to show the memory cost measurements while increasing the event size. The emergency events that have happened are stored in a database. In Fig. 9 the emergency events are chosen randomly. We can observe that with the increase of the event size, the memory cost of algorithms increases. The memory cost of our algorithm is obviously superior. Our EET model only analyzes the relevant document and scans this data in a single pass, this helps the execution of tracking to critical features. In summary, the tracking processing reduces the consumption of memory, and thus better extracts relevant web data, analyzes and regroups them.

A larger λ means a stronger relevance of the feature vectors to the clusters. $\lambda = 0$ means that the current feature vector cannot belongs to the topics, this feature vector should be as a new topic for event. $\lambda = 1$ means that the feature vector with a nearly limitless supply is evolutionary to the topic. For this evaluation, the Normalized Mutual Information (NMI) score and Purity results are given in Figs 10 and 11, respectively. This illustrates that the features relevance threshold of clusters can improve the performance of EET. However, the performance degrades slightly when $\lambda = 1$. The reason seems to be that the strong constraints make difficult the correction of noise in the clustering quality. Thus, according to the results in Figs 10 and 11, the value of λ is selected as 0.6.

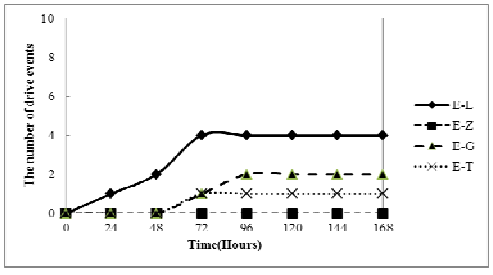

Fig. 6. The number of drive events *vs* time

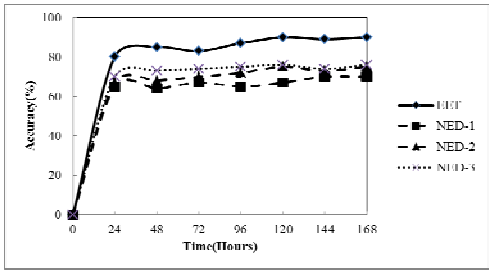

Fig. 7. Time *vs* Accuracy

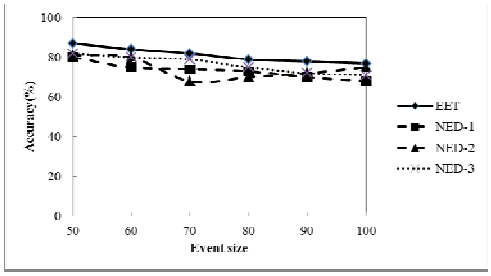

Fig. 8. Event size *vs* Accuracy

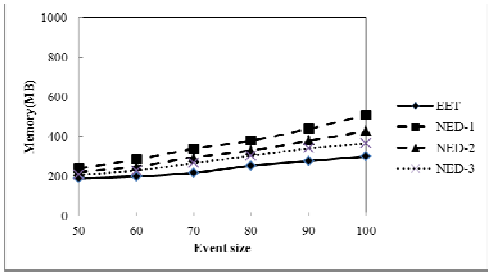

Fig. 9. Event size *vs* Memory

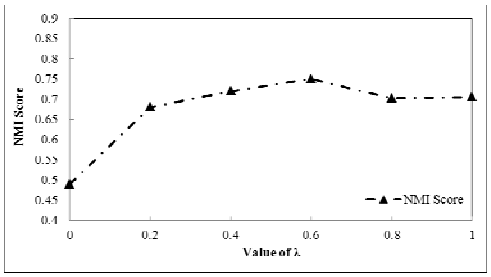

Fig. 10. Evaluation results in NMI

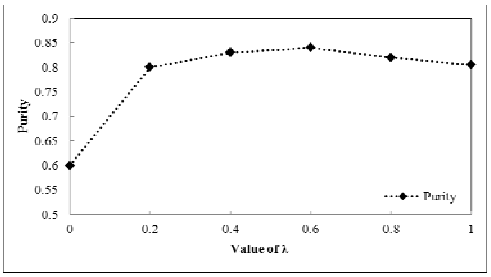

Fig. 11. Evaluation results in Purity

To sum up, to the best of our knowledge, EET model together with WDEE algorithm is the advanced technique that systematically investigates the problem of emergency event tracking over massive web data and provides solutions with solid performance.

7. Conclusion

In this paper we propose a novel approach for events tracking in emergency-oriented web data. For the purpose of a high degree of automation by reducing human supervision, extracting of web data is dynamically organized in real time for an emergency event, namely our approach is a real-time data extraction which discovers and analyzes latent events in web environment. Furthermore, we introduce a limit theory to approximately compute the periodical convergence of the data extraction process. We propose a model, Emergency Event Tracking model (EET), to guarantee the efficiency of the tracking process. Moreover, we perform comprehensive experiments. The experimental results not only outperform the existing methods on real data, but also verify the efficiency and accuracy of our algorithms.

***Acknowledgements*:** The authors would like to thank the anonymous reviewers for their insightful and constructive comments. This research work is supported in part by the Self-Conducted Exploratory Research Program by the State Key Laboratory for Software Development Environment in China (NO.SKLSDE-2013ZX-11), the Social Service Project in the National Earthquake Response Support Service "International Rescue and Disposition System Research against Strong Earthquakes" (NO SJZX-B11).

References

1. Emilio, F., D. M. Pasquale, F. Giacomo, B. Robert. Web Data Extraction, Applications and Techniques: A Survey. – Knowledge-Based Systems, Vol. **70**, 2014, pp. 301-323.
2. Gottlob, G., C. Koch. Monadic Datalog and the Expressive Power of Languages for Web Information Extraction. – Journal of the ACM, Vol. **51**, 2004, No 1, pp. 74-113.
3. Doan, A., J. Naughton, R. Ramakrishnan, A. Baid, X. Chai et al. Information Extraction Challenges in Managing Unstructured Data. – ACM SIGMOD Record, Vol. **37**, 2009, No 4, pp. 14-20.
4. Chen, H., M. Chau, D. Zeng. Ci Spider: A Tool for Competitive Intelligence on the Web. – Decision Support Systems, Vol. **34**, 2002, No 1, pp. 1-17.
5. Song, X., Q. S. Zhang, Y. Sekimoto, R. Shibasaki. Prediction of Human Emergency Behavior and Their Mobility Following Large-Scale Disaster. – In: Proc. of 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 24-27 August 2014, pp. 5-14.
6. Crescenzi, V., G. Mecca. Automatic Information Extraction from Large Websites. – Journal of the ACM, Vol. **51**, 2004, No 5, pp. 731-779.
7. Chen, W., C. Chen, L. Zhang, C. Wang, J. Bu. Online Detection of Bursty Events and Their Evolution in News Streams. – Journal of Zhejiang University, Vol. **11**, 2010, No 5, pp. 340-355.
8. Baumgartner, R., G. Gottlob, M. Herzog. Scalable Web Data Extraction for Online Market Intelligence. – In: Proc. of 35th International Conference on Very Large Data Bases (VLDB), 24-28 August 2009, pp. 1512-1523.
9. Juan, D. V. Web Mining and Privacy Concerns: Some Important Legal Issues to be Consider before Applying Any Data and Information Extraction Technique in Web-Based Environments. – Expert Systems with Applications, Vol. **40**, 2013, Issue 13, pp. 5228-5239.
10. Chen, W., P. Chundi. Extracting Hot Spots of Topics from Time-Stamped Documents. – Data & Knowledge Engineering, Vol. **70**, 2011, Issue 7, pp. 642-660.
11. Zhang, K., J. Z. Li, G. Wu, K. S. Wang. Term Committee-Based Event Identification within Topics. – Journal of Computer Research and Development, Vol. **19**, 2009, No 4, pp. 817-828.
12. Ferrara, E., R. Baumgartner. Automatic Wrapper Adaptation by Tree Edit Distance Matching. – Combinations of Intelligent Methods and Applications, Vol. **8**, 2011, pp. 41-54.
13. Krüpl-Sypien, B., R. R. Fayzrakhmanov, W. Holzinger, M. Panzenböck, R. Baumgartner. A Versatile Model for Web Page Representation, Information Extraction and Content Repackaging. – In: Proc. of 2011 International Conference on ACM Symposium on Document Engineering, 19-22 September 2011, pp. 129-138.
14. Laender, A. H. F., B. A. Ribeiro-Neto, A. S. D. Silva, J. S. Teixeira. A Brief Survey of Web Data Extraction Tools. – SIGMOD Record, Vol. **31**, 2002, No 2, pp. 84-93.
15. He, Z. X., X. Y. Zhao, S. Y. Zhang, T. Ogawa, M. Haseyama. Random Combination for Information Extraction in Compressed Sensing and Sparse Representation-Based Pattern Recognition. – Neurocomputing, Vol. **145**, 2014, pp. 160-173.
16. Chang, C., M. Kayed, M. Girgis, K. Shaalan. A Survey of Web Information Extraction Systems. – IEEE Transactions on Knowledge and Data Engineering, Vol. **18**, 2006, No 10, pp. 1411-1428.
17. Fiumara, G. Automated Information Extraction from Web Sources: A Survey. – In: Proc. of Workshop on between Ontologies and Folksonomies: Tools and Architectures for Managing and Retrieving Emerging Knowledge in Communities (BOF), 28 June 2007, pp. 1-9.
18. Flesca, S., G. Manco, E. Masciari, E. Rende, A. Tagarelli. Web Wrapper Induction: A Brief Survey. – AI Communications, Vol. **17**, 2004, No 2, pp. 57-61.
19. Sarawagi, S. Information Extraction, Found. – Trends Databases, Vol. **1**, 2008, No 3, pp. 261-377.

20. M a r t i n e z, D., G. P i t s o n, A. M a c K i n l a y, L. C a v e d o n. Cross-Hospital Portability of Information Extraction of Cancer Staging Information. – Artificial Intelligence in Medicine, Vol. **62**, 2014, No 1, pp. 11-21.
21. I t t o o, A., G. B o u m a. Minimally-Supervised Extraction of Domain-Specific Part-Whole Relations Using Wikipedia as Knowledge-Base. – Data & Knowledge Engineering, Vol. **85**, 2013, pp. 57-79.
22. F e r r a r a, E. A Large-Scale Community Structure Analysis in Facebook. – EPJ Data Science, Vol. **1**, 2012, No 9, pp. 1-30.
23. J i a n g, W., C. L. Z h a o, S. H. L i, C. L a w s o n. A New Learning Automata Based Approach for Online Tracking of Event Patterns. – Neurocomputing, Vol. **137**, 2014, pp. 205-211.
24. K l e i n b e r g, J. M. Bursty and Hierarchical Structure in Streams. – Data Mining and Knowledge Discovery, Vol. **7**, 2003, No 4, pp. 373-397.
25. G u o, L. M., G. Y. H u a n g, Z. M. D i n g. Efficient Detection of Emergency Event From Moving Object Data Streams. – In: Proc. of 19th International Conference on Database Systems for Advanced Applications (DASFAA), 21-24 April 2014, pp. 422-437.
26. T o n g, Y. X., C. C. C a o, L. C h e n. TCS: Efficient Topic Discovery over Crowd-Oriented Service Data. – In: Proc. of 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 24-27 August 2014, pp. 861-870.
27. V i e w e g, S., A. L. H u g h e s, K. S t a r b i r d, L. P a l e n. Microblogging During Two Natural Hazards Events: What Twitter May Contribute to Situational Awareness. – In: Proc. of 28th International Conference on Human Factors in Computing Systems (CHI), 10-15 April 2010, pp. 1079-1088.
28. C h e n, Y., X. M. Z h a n g, Z. J. L i, N g. J u n-P i n g. Search Engine Reinforced Semi-Supervised Classification and Graph-Based Summarization of Microblogs. – Neurocomputing, Vol. **152**, 2015, pp. 274-286.
29. B e c k e r, H., M. N a a m a n, L. G r a v a n o. Event Identification in Social Media. – In: Proc. of 12th International Workshop on the Web and Databases (WebDB), 28 June 2009, pp. 107-111.
30. P o h l, D., A. B o u c h a c h i a, H. H e l l w a g n e r. Automatic SubEvent Detection in Emergency Management Using Social Media. – In: Proc. of 21st International Conference Companion on World Wide Web (WWW), 16-20 April 2012, pp. 683-686.
31. S t r e h l, J. G., C. C a r d i e. Cluster Ensembles – Knowledge Reuse Framework for Combining Multiple Partitions. – Journal of Machine Learning Research, Vol. **3**, 2002, pp. 583-617.
32. F u n g, G. P. C., J. X. Y u, P. S. Y u, H. L u. Parameter Free Bursty Events Detection in Text Streams. – In: Proc. of 31st International Conference on Very Large Data Bases (VLDB), 30 August-2 September 2005, pp. 181-192.
33. L i u, M. L., D. Q. Z h e n g, T. J. Z h a o, Y. Y u. Dynamic Multi-Document Summarization Model. – Journal of Software, Vol. **23**, 2012, No 2, pp. 289-298.
34. Z h o n g, Z. M., C. H. L i, Z. T. L i u, H. W. D a i. Web News Oriented Event Multi-Elements Retrieval. – Journal of Software, Vol. **24**, 2013, No 10, pp. 2366-2378.
35. W u, Q. H., J. H. L v. EET: Efficient Event Tracking over Emergency-Oriented Web Data. – In: Proc. of International Joint Conference on Neural Networks (IJCNN), 12-17 July 2015, pp. 3666-3673.
36. R o d r i g o, A., A. X a b i e r, B. Z u h a i t z, R. G e r m a n, S. A i t o r. Big Data for Natural Language Processing: A Streaming Approach. – Knowledge-Based Systems, Vol. **79**, 2015, pp. 36-42.
37. A m i n u l, I., I. D i a n a, K. I l u j u. Applications of Corpus-Based Semantic Similarity and Word Segmentation to Database Schema Matching. – The VLDB Journal, Vol. **17**, 2008, Issue 5, pp. 1293-1320.