BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **16**, No 1
Sofia • 2016 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.1515/cait-2016-0015

Addendum to "Mining Similar Traces of Entities on Web"
(published in Vol. 15, No 6)

Xinyan Huang[1,3], *Xinjun Wang*[1,2], *Hui Li*[1,2]

[1]*Shandong University, Num 1500, SunHua Road in High Tech Industrial Development Zone, Ji'nan, China*
[2]*Dareway Software Co., Ltd, Num 1500, SunHua Road in High Tech Industrial Development Zone, Ji'nan, China*
[3]*Shandong University of Finance and Economics, Num 7366, East Erhuan Road in Lixia Zone, Ji'nan, China*
Emails: 20063462@sdufe.edu.cn wxj@sdu.edu.cn lih@sdu.edu.cn

Publiher's note: The Editorial Board asked the authors to write the present Addendum in order to discuss with explicit explanations their view on the similarities and differences of the approaches, presented in: *Xinyan Huang, Xinjun Wang, Hui Li – Mining Similar Traces of Entities on Web "Cybernetics and Information Technologies", Vol. 15, No 6, pp. 219-229, 2015* (paper [1] in References) *and Andreas Weiler, Michael Grossniklaus, and Marc H. Scholl – Event Identification and Tracking in Social Media Streaming Data. Proceedings of MSDM 2014, EDBT Workshop on Multimodal Social Data Management, Athens, Greece, March 2014, pp. 282-287* (paper [2] in References).

The work presented in Section 5 of [1] differs from the work in Section 2 of [2] due to the following reasons.

1. **Different research objects and different meaning of a sliding time window.** Our work presented in [1] is based on massive events data sets in a long history which have already been extracted, while the work presented in [2] is based on the context in real-time from the live public data stream of Twitter. So the scale of data to process in the case of [2] isn't in the same order of magnitude as ours, where the same approach has a different meaning for our work. Take a sliding time window model, for example, taking into consideration such a big scale of events. The main aim of applying time windows in our work is to facilitate parallel processing and achieve high computational efficiency. A sliding window model is applied in [2] to extract events.

2. **Different background and different meaning of id*f*(*e*).** In our work, if the Target Entity is TE, the pre-processing work before significant events identification is done through a separate task to aggregate all the events of TE, then the value of

id$f$(e) of each event e is evaluated and finally the event relationship graph of TE is constructed where all the edges are based on co-occurrence of events. Here, id$f$(e) in our work is the total occurrence number of the event e in historical events data sets and isn't just one in a time window (as in [2]), the name of which comes from a statistical method TF-IDF used to evaluate the significance of a word to a document set or a document of a corpus. In our work, the significance of each event is similarly assessed through analyzing id$f$(e).

3. **Different focus and different approaches.** Identifying the most significant events is to identity the events with locally the highest id$f$(e). It is well-known that the issue of discovering the local maximum of a curve is generally solved through slopes of Tangent Lines. So naturally, the most obvious approach to identify these events with locally the highest id$f$(e) is to analyze sid$f$(e) which is the change ratio of id$f$(e) from an event to next one. For an event e_i, the change ratio of e_i is expressed as sid$f$(e_i) and is calculated through the following definition:

$$\mathrm{sid}f(e_i)=(\mathrm{id}f(e_{i+1}) - \mathrm{id}f(e_i))/(\mathrm{id}f(e_i) - \mathrm{id}f(e_{i-1})).$$

For each most significant event e with locally highest id$f$(e), its characteristics are that the numerator of sid$f$(e) is negative, the denominator of sid$f$(e) is positive and sid$f$(e) is negative. We need to discover all the events owning these characteristics in all time windows, while they use shifts in the Inverse Document Frequency (IDF) to capture trending terms in their work. We focus on different aims and the whole algorithms in our work are quite different.

In order to further distinguish between the two works mentioned above, Section 5 of [1] is extended with more details below.

Our work presented in [1] is based on events data sets in a long history which have been extracted already. If the target entity is TE, the pre-processing work before significant events identification is done through a separate task to aggregate all the events of TE, then count the value of id$f$(e) of each event e and finally construct the event relationship graph of TE in which all the edges are based on the co-occurrence of events. In addition, id$f$(e) is the total occurrence number of the event e in historical events data sets and isn't just one in a time window, the name of which comes from a statistical method TF-IDF which is used to evaluate the significance of a word to a document set or a document of a corpus. In our approach, the significance of each event is similarly assessed through analyzing the value change of id$f$(e).

In [1] most significant event is an event e which has higher id$f$(e) than both the events before it and the events after it in a time span, such as event $e1$, $e2$ and $e3$ in Fig. 3. In practice, each significant event and its two second significant events, such as $e1'$ and $e1''$ in Fig. 3, are captured together and collectively referred to as significant events, because, three events intuitively make it easier and faster to be extended to a topic. Taking into account a vast number of events generated each second and no explicit knowledge about current or future events, identifying significant events becomes a crucial task, which makes it possible to handle so massive events [3]. Significant events are identified first and then are extended according to topics which are subgraphs of the events relationship graph of the target entity and naturally consist of co-occurrence events of the significant events.

In the following, we describe the process of significant events identification:

As shown in Fig. 3, the blue dots represent different events, and each event *e* has its own id*f*(*e*). The first step is to put all the events data *E*(*e*1, *e*2, *e*3, ...) of the entity TE into chronological order. In order to facilitate parallel processing in practice, all the events data *E*(*e*1, *e*2, *e*3, ...) are subdivided according to fixed sized windows (such as *w*1, *w*2, *w*3 in Fig. 3).

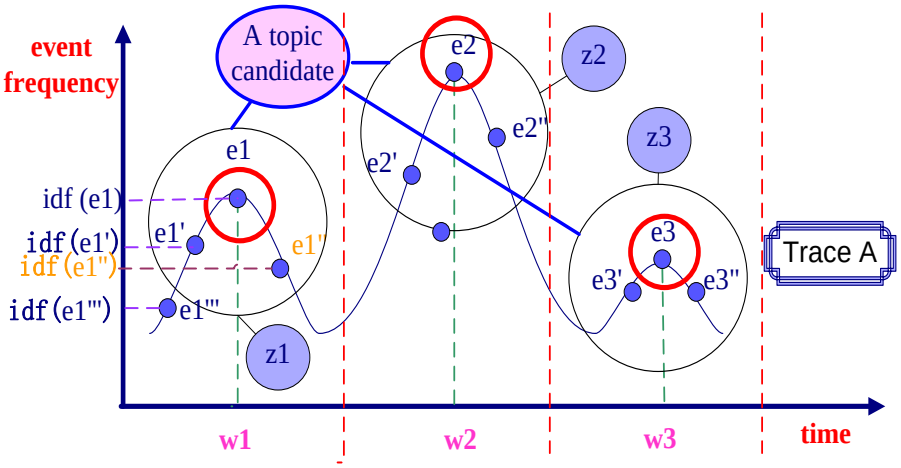

Fig. 3. Finding significant events and topic sequences

By our definition, identifying the most significant events is to identity the events with locally the highest id*f*(*e*), such as *e*1, *e*2 and *e*3 in the Fig. 3. It is well-known that the issue of discovering the local maximum of a curve is generally solved through slopes of Tangent Lines. So naturally, the most obvious approach to identify these events with locally highest id*f*(*e*) is to analyze sid*f*(*e*) which is the change ratio of id*f*(*e*) from an event to next one. For an event e_i, the change ratio of e_i is expressed as sid*f*(e_i) and is calculated through the following definition (1):

$$\text{sid}f(e_i)=(\text{id}f(e_{i+1}) - \text{id}f(e_i))/(\text{id}f(e_i) - \text{id}f(e_{i-1})). \quad (1)$$

Take event *e*1′ in Fig. 3 for example, the change ratio of *e*1′, as follows: sid*f*(*e*1′)=(id*f*(*e*1) – id*f*(*e*1′))/(id*f*(*e*1′) – id*f*(*ei*′′′)). For each most significant event e with locally highest id*f*(*e*), such as *e*1, *e*2 and *e*3 in the Fig. 3, its characteristics are that the numerator of sid*f*(*e*) is negative, the denominator of sid*f*(*e*) is positive and sid*f*(*e*) is negative. We need to discover all the events owning these characteristics in all time windows. At the same time, the event just before the most significant event and the event just after the most significant event could be captured together, such as *e*1′ and *e*1″ in Fig. 3.

Algorithm 1 describes the process of significant events identification in a time window w_j. Because all the processes of significant events identification in each window are independent, all the time windows can work in parallel.

Algorithm 1. Significant event identification

Input: Time window w_j, all the events data $E(e_1, e_2, e_3, \ldots, e_n)$ in w_j and their own id$f(e_i)$

Output: all the significant events T

Step 1. Initialize S←Φ, T←Φ

Step 2. For i=1 to n

Step 3. s=(idf(e_{i+1}) – idf(e_i))/(idf(e_i) – idf(e_{i-1}))

Step 4. if (s<0 and (idf(e_{i+1}) – idf(e_i))<0 and (idf(e_i) – idf(e_{i-1}))>0)

Step 5. S=ei∪ei–1∪ei+1

Step 6. endif

Step 7. T=T∪S

Step 8. end

Step 9. Return T

Finally, each the most significant event and its second the most significant events are organized as a whole, which are regarded as a candidate topic. For example, all the significant events in Fig. 3 can be organized as ($e1'$, $e1$, $e1''$), ($e2'$, $e2$, $e2''$), ($e3'$, $e3$, $e3''$).

Then, for all topic candidates, a clustering method is employed. These topic candidates with at least a similar event are clustered into a group and get an identical topic label, such as *A'*, *B'*, etc. In addition, add the edges to each candidate topic if the events in it are related, according to the event relationship graph of the target entity TE.

These labelled candidate topic sequences, such as *B'D'F'A'C'*, are passed on to the phase of mining of similar topic sequences, which is described in Section 6 [1].

References

1. Huang, X., X. Wang, H. Li. Mining Similar Traces of Entities on Web. – Cybernetics and Information Technologies, Vol. **15**, 2015, No 6, pp. 219-229.
2. Weiler, A., M. Grossniklaus, M. H. Scholl. Event Identification and Tracking in Social Media Streaming Data. – In: Proc. of MSDM 2014, EDBT Workshop on Multimodal Social Data Management, Athens, Greece, March 2014, pp. 282-287.
3. Zhang, J., J. Tang, J. Li. Expert Finding in a Social Network. – In: DASFAA'07, 2007, pp. 1066-1069.