Information & Management 57 (2020) 103348

Contents lists available at ScienceDirect

Information & Management

journal homepage: www.elsevier.com/locate/im

Discovering event episodes from sequences of online news articles: A time-adjoining frequent itemset-based clustering method

Yen-Hsien Lee[a,*], Paul Jen-Hwa Hu[b], Hongquan Zhu[c], Hsin-Wei Chen[d]

[a] Department of Management Information Systems, National Chiayi University, Taiwan
[b] Department of Operations and Information Systems, David Eccles School of Business, University of Utah, USA
[c] Department of Finance, School of Economics and Management, Southwest Jiaotong University, China
[d] AdvancedTEK International Corp., Taiwan

ARTICLE INFO

Keywords:
Event episode discovery
Retrospective event detection
Event evolution
Temporal frequent itemset-based clustering

ABSTRACT

Firms perform environmental surveillance to identify important events and their developments. To alleviate the stringent information processing and analysis requirements, automated methods are needed to discover from online news articles distinct episodes (stages) of an important event. We propose a time-adjoining frequent itemset-based method that incorporates essential temporal characteristics of news articles for event episode discovery. With a corpus of 1468 news articles that pertain to 248 episodes of 53 different events, we empirically evaluate the proposed method and include several prevalent techniques as benchmarks. The results show that our method outperforms the benchmark techniques consistently and significantly, attaining the cluster recall, cluster precision, and F-measure values at 0.706, 0.593, and 0.584, respectively.

1. Introduction

Firms' ability to identify and monitor essential changes in the environment and incorporate such information in strategy formulations, decision making, and business actions is crucial to their performance and competitiveness [1–4]. Many firms continuously surveil the business environment to identify important events concerning customers, competitors, industry, technology, broad economic conditions, and government policies and regulations [5,6]. Because of the increasing prevalence of the Internet, online news articles represent a common and critical source for firms' environmental surveillance [7–10]. As Haase and Franco [11] note, these voluminous news articles can be rapidly created, globally disseminated, conveniently accessed, and easily processed with increasing effectiveness and efficiency. Yet, the fast-growing quantity of online news articles that pertain to various events now poses a fundamental challenge to firms that strive to stay abreast of the emerging events as they develop [12,13].

Event evolution patterns (EEPs) are central to firms' environmental surveillance and can support their market predictions, business decisions, scenario analyses, and dialog system designs [12,14,15]. Firms seek to unfold EEPs that depict the general (overall) evolvement of a particular event type over time (e.g., merge and acquisition), so that they can be aware of and get prepared for monitoring such events in the near future [15,16]. In essence, EEPs produce high-level depictions of the evolution of different event types (categories), such as Initial Public Offerings warranting firms' attention and surveillance [12,13,17], by discovering from sequences of news articles (or documents) a common evolution pattern for distinct events of the same type, together with their temporal relationships [12,14,18–20].

Typically, news stories reveal different episodes. Previous research has adopted different event structures or taxonomies. For example, [17,18] employ Story→Event→Topic; [21] adopt Story→Simple Event→Complex Event; [22] utilize Story→Component Event→Event; and [12] use Story→Episode→Event. These studies also use different concepts and terminologies. Specifically, "Topic" in [17,18] is identical to "Event" in [12,22] or "Complex Event" in [21], while "Event" refers to a distinct development stage of an event and is highly similar to "Simple Event" in [21], "Component Event" in [22], or "Episode" in [12].

In this study, we follow the Story→Episode→Event event structure [12]. As we illustrate in Fig. 1, one or more news stories describe a particular earthquake event, with each article denoting a distinct development stage or subevent of the focal event [12]. As shown, a common evolution pattern for earthquake events involves multiple episodes (development stages), such as epicenter and magnitude, casualties and damages, rescue actions, and rebuilding and restoration

* Corresponding author.
E-mail addresses: yhlee@mail.ncyu.edu.tw (Y.-H. Lee), paul.hu@eccles.utah.edu (P.J.-H. Hu), hqzhu@swjtu.edu.cn (H. Zhu), samtry_chen@advtek.com.tw (H.-W. Chen).

https://doi.org/10.1016/j.im.2020.103348
Received 3 May 2019; Received in revised form 21 July 2020; Accepted 22 July 2020
Available online 05 August 2020

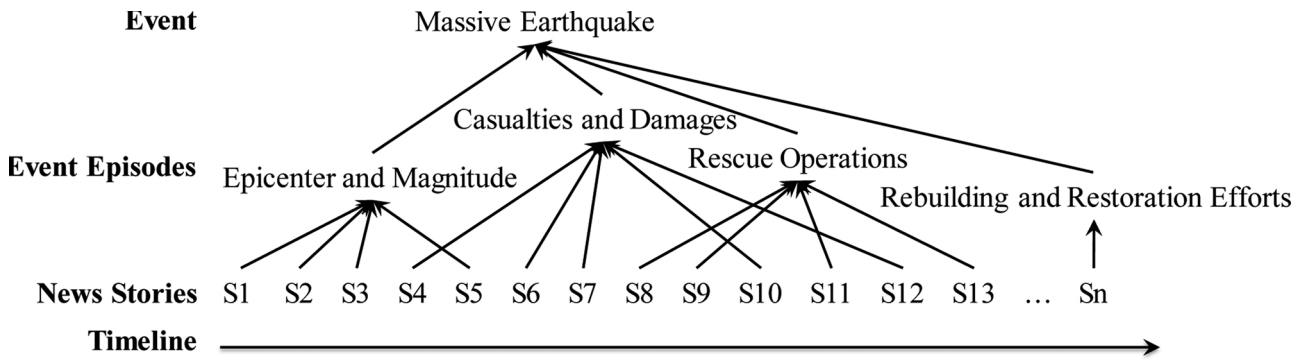

Fig. 1. Example of Relationships among Events, Event Episodes, and News Stories.

efforts. Supported by effective EEP discovery, firms can identify distinct stages of an emerging event, which they otherwise might overlook, and thereby can adapt better to the changing environment with agility and appropriate responses [23–26]. All else being equal, the common evolution pattern of a specific event type, if effectively identified and tracked, enables firms to anticipate and respond better to subsequent developments of different events of that type [27].

A critical precursor to EEP discovery is identifying distinct episodes of an event from a sequence of news articles (documents) that pertain to that event, by grouping the articles that describe each episode [12,17–19,28,29]. The enormous quantity of online news articles available on the Internet makes the traditional, manual approach to event episode discovery ineffective, if feasible at all [30]. Previous EEP discovery research assumes the availability (existence) of different episodes of an event to be analyzed [21,22], then takes a conventional document clustering approach to discover episodes from a sequence of news articles pertinent to an event, without considering their temporal relationships [14,19,31–33], or extends existing document clustering techniques by considering characteristics specific to event-based documents [17,18]. For example, Nallapati, Feng, Peng, and Allan [17] incorporate temporal locality by applying a time-decaying function to adjust the similarity of two news articles. In general, news articles that describe a particular event often are temporally proximate; the greater the temporal difference between two articles, the lower their similarity.

Although empirical evidence indicates temporal difference-based clustering more effective for detecting events than traditional feature-based clustering [30], the use of a document-based time-decaying function may not be appropriate for event episode discovery, mainly because distinct episodes can have different temporal patterns. Typically, a time-decaying function tends to lower the similarity between news articles that are temporally distant and may not recognize persistent episodes of an event, which could vary in their time interval, such that the separation of different episodes becomes difficult. Also, distinct episodes of an event may be temporally adjacent or even overlap to some extent [30]. Even if the episodes are distinct, they remain interrelated and cannot be separated effectively by a linear time penalty function or an exponential time-decaying function. In addition, the features (terms) appearing in news articles describing the same event often share a considerable similarity, which often makes conventional feature-based document clustering less effective for grouping these articles according to feature similarity [30]. In either case then, the effectiveness of a document-based time-decaying function may be diminished.

This study addresses the limitations of conventional document-based event episode discovery techniques by incorporating essential temporal characteristics of news articles about a focal event in feature-level analyses. The rationale is that temporally proximate features (terms) should be more representative of the underlying episodes than temporally proximate news articles (documents). These temporally proximate features also should be more important than features that are farther apart temporally. We identify and select temporally proximate features and then use them to represent distinct episodes to be discovered, rather than relying on feature similarity for discovery. Some important temporal characteristics of features in distinct but related episodes of an event are incorporated in the proposed time-adjoining frequent itemset-based event-episode discovery (TAFIED) method.[1] Overall, our method extends frequent itemset-based hierarchical clustering (FIHC) by incorporating temporal locality, according to the fit between a cluster and a document. Unlike typical FIHC that uses frequent items to group documents (features appearing frequently in news articles), our measure instead assesses the fit of documents within a cluster and estimates the temporal adjacency of different features intuitively. We target event episode discovery and seek to identify different subevents of a focal event to support EEP discovery.

The organization of the remaining paper is as follows. Section 2 provides an overview of retrospective event detection and event episode discovery, reviews representative studies of FIHC, and highlights the gaps that motivate our study. In Section 3, we elaborate the proposed method and its overall processing. We describe our data and evaluation design in Section 4, followed by data analyses and important results in Section 5. We conclude with a summary of the study and its contributions, together with several future research directions, in Section 6.

2. Literature review and motivation

Several streams of research closely relate to our study, including retrospective event detection and event episode discovery as well as FIHC. Herein, we review these streams of extant literature to indicate the gaps that we seek to address.

2.1. Retrospective event detection and event episode discovery

Previous research has examined retrospective event detection [34–38]. In general, retrospective event detection partitions or clusters a corpus of news articles into distinct topics or events; it shares several characteristics with event episode discovery but differs in both the unit and the granularity of analysis. Typical retrospective event detection techniques discover events from a stream of news articles; they target an event-based classification by clustering a sequence of chronologically ordered news articles, available in different sources or languages, to identify a set of coherent topics (events) inherent to the news articles [39,40].

Event episode discovery explicitly aims at discovering an event as it evolves through different development stages and identifying news articles that pertain to each stage. That is, retrospective event detection may reveal distinct events (e.g., Indian Ocean tsunami, the trade war

[1] In the proposed method, an item refers to a term or feature that appears in a news article.

between the United States and China, and Tesla going private) and find news articles associated with each event; event episode discovery instead seeks to identify the different development stages of these events over time and find news articles pertinent to each stage. Overall, event episode discovery identifies distinct episodes of an event from a sequence of news articles pertinent to that event, whereas retrospective event detection identifies events from a stream of articles by segmenting the different events described by these articles. As a result, event episode discovery tends to perform analyses at a finer-grained level than retrospective event detection.

Nallapati, Feng, Peng, and Allan [17] and Wei and Chang [12] attempt event episode discovery by applying a hierarchical agglomerative clustering (HAC) algorithm [41] to identify distinct episodes of an event from sequences of news articles. Nallapati, Feng, Peng, and Allan [17] describe an event structure as interconnected, threading subjects, and apply a Story→Event→Topic event taxonomy. To discern different episodes of an event, they leverage the temporal localization of news stories, using a time-decaying function to estimate the similarity between two news stories and applying a penalty to news story pairs that are temporally distant. Empirical results confirm that the use of a time-decaying function improves the effectiveness of an existing event episode discovery technique [17,30]. Wei and Chang [12] instead propose a Story→Episode→Event taxonomy, which emphasizes essential intra- and intersequence episode relationships and aims to capture the event evolution by unfolding the temporal patterns of the respective episodes of an event. However, this taxonomy does not consider temporal localization of news stories, because the goal is to generalize episodes across different events and discover frequent event episodes in the underlying temporal relationships, that is, event evolution patterns.

2.2. Frequent itemset-based hierarchical clustering

FIHC leverages association rule mining by considering the documents as transactions and the features of a document as items [42]. In general, FIHC selects features (items) with a document frequency greater than the prespecified minimum (g_f) threshold and uses the identified frequent features (items) as cluster centroids to group documents. That is, the identified frequent features serve as cluster labels; a document d_j is initially assigned a set of candidate clusters, on the basis of its own frequent items, so that the most appropriate cluster for the document can be determined according to the goodness of fit that indicates the score of retaining document d_j in the cluster c_x. The score can be calculated as

$$Score(c_x \longleftarrow d_j) = [\sum_i (n(t_i) \times Cluster\_Support(t_i))] - [\sum_i (n(t_i') \times Global\_Support(t_i'))] \quad (1)$$

where t_i represents a global frequent item in document d_j and is also a frequent item in cluster c_x, t_i' denotes a global frequent item in d_j but not a frequent item in c_x, $n(t_i)$ indicates the weight of t_i in d_j, $n(t_i')$ is the weight of t_i' in d_j, $Cluster\_Support(t_i)$ reveals the percentage of the documents in c_x that contain t_i, and $Global\_Support(t_i')$ depicts the percentage of the entire documents that contain t_i'.

Next, FIHC retains each document in the cluster with the highest goodness-of-fit score, such that each document belongs to one and only one cluster, and empty clusters get removed. To avoid the possibility that documents describe the same topic (event) but get assigned to multiple clusters, FIHC uses an intercluster similarity measure and collapses any cluster whose similarity exceeds the prespecified threshold. Furthermore, FIHC can generate a natural topic hierarchy, with increasing ease of browsing and cluster accuracy. For example, it may measure intercluster similarity by assessing the viability of merging cluster c_y with c_x by aggregating all the documents in c_y into a document, then consolidating the goodness of fit of the merging cluster c_x with c_y by aggregating all the documents in c_x into a document, calculated by Eq. (2). The score for merging cluster c_y with c_x then would be defined as in Eq. (3).

$$Inter\_Sim(c_x \leftrightarrow c_y) = \sqrt{Sim(c_x \longleftarrow c_y) \times Sim(c_y \longleftarrow c_x)} \quad (2)$$

and

$$Sim(c_x \longleftarrow c_y) = \frac{Score(c_x \longleftarrow doc(c_y))}{\sum n(t_i) + \sum n(t_i')} + 1 \quad (3)$$

where c_x and c_y represent two clusters, $doc(c_y)$ is the aggregation of all the documents in cluster c_y, t_i denotes a global frequent item in $doc(c_y)$ and a frequent item in cluster c_x, t_i' indicates a global frequent item in $doc(c_y)$ but not a frequent item in c_x, $n(t_i)$ is the weight of t_i in $doc(c_y)$, and $n(t_i')$ is the weight of t_i' in $doc(c_y)$.

2.3. Gap analysis and motivation

Retrospective event detection identifies different events that exist in a sequence of news articles and then groups them accordingly. Different news events might not be temporally adjacent and could vary substantially in content. In contrast, event episode discovery unfolds distinct (development) stages of a focal event from the related news articles. News articles that pertain to the focal event might have similar content, themes (main storyline), and wording. Retrospective event detection applies to the entire corpus of documents, whereas event episode discovery groups only those news articles specific to an episode of the focal event. As a result, existing retrospective event detection techniques may not be effective for event episode discovery.

Most existing event episode discovery techniques adopt a conventional document clustering approach and address the temporal locality of news articles by using a time-decaying function to adjust the probability that different news stories belong to the same episode. News articles pertinent to an event episode in principle should be temporally adjacent; yet, existing techniques overlook the likelihood that distinct episodes may develop concurrently, such as when multiple distinct episodes occur within a time period, with some degree of overlap among them.

To be effective, event episode discovery should properly consider two issues: news articles describing different episodes of a particular event have similar content, and different episodes could emerge concurrently within a time window. Existing techniques for event episode discovery or retrospective event detection may not identify event episodes from sequences of news articles effectively, even with the inclusion of a document-based time-decaying function. Event episode discovery instead should proceed at the feature level, rather than the document level, to analyze whether different news articles pertain to the same episode. Although articles about an event may have similar descriptions and wordings, they tend to vary in focus (main storyline) and have specific features (terms) that differ from those frequently appearing in the related news articles. For example, regarding a particular earthquake event, an initial episode may focus on the date and time, location, and magnitude of the quake; subsequent episodes could relate to casualties and damages, rescue operations, logistic support and survivor placement, and rebuilding and restoration efforts. Although these news articles would have similar content, their focus (theme) differs noticeably. We thus propose a novel time-adjoining frequent itemset-based method for event episode discovery, as we elaborate about this proposal in the next section.

3. Proposed method

The proposed TAFIED method extends FIHC, in an attempt to increase its ability to cope with the burst characteristic of features for event episode discovery. Our method can address the limitations of document-based time-decaying functions that often constrain existing document clustering techniques by considering several essential

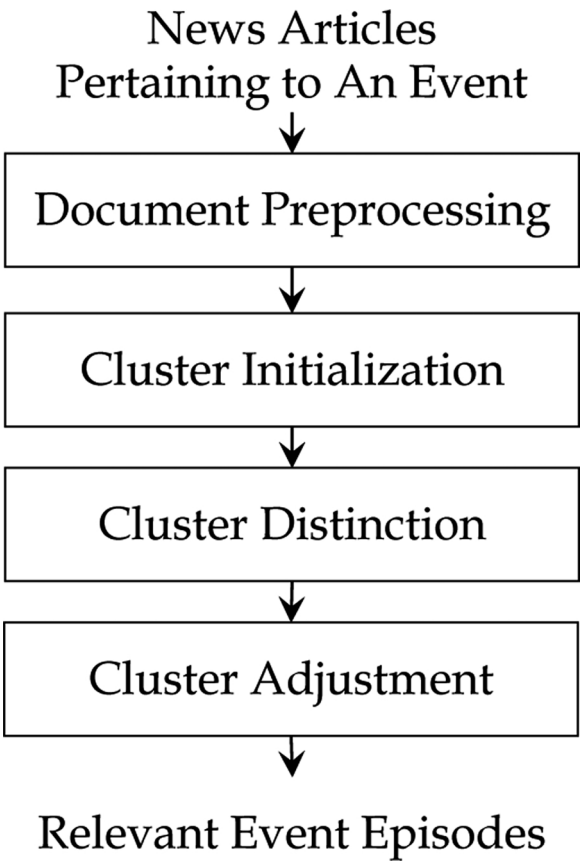

Fig. 2. Proposed TAFIED Method.

characteristics of online news articles: burst, new terms and developments, and the temporal adjacency of features. TAFIED groups news articles (documents) by using frequent items as cluster centroids; it addresses the temporal adjacency of documents in a cluster to ensure goodness of fit between a cluster and a document by properly weighing the adjacency of the respective time stamps of different news articles that belong to the same cluster (event episode). Thus, TAFIED can create clusters in which documents are temporally adjacent and share features that frequently appear in a stream of news articles. As we depict in Fig. 2, the input for the proposed method is a sequence of news articles about an event, which are used to discover a set of distinct episodes. The overall processing of our method consists of *document preprocessing*, *cluster initialization*, *cluster distinction*, and *cluster adjustment*.

In *document preprocessing*, TAFIED extracts meaningful terms (e.g., nouns, noun phrases, and verbs) from each news article, applies a rule-based part-of-speech tagger to tag each word in the article [43], then uses a parser to select nouns, noun phrases, and verbs from the article. Stop words such as nonsemantic-bearing words get removed, and the remaining words are stemmed into their respective original forms.

For *cluster initialization*, TAFIED constructs a set of initial clusters and assigns each news article to candidate clusters according to its own frequent items. Frequent items (terms) are first identified from the entire corpus of news articles under analysis. Term t_i is a frequent item if the ratio between its document frequency (number of news articles with term t_i) and the total number of news articles exceeds the pre-specified minimum global support g_t. By viewing each frequent item as a class label, our method can create a set of initial clusters; each news article gets assigned to candidate clusters on the basis of its own frequent items (class labels). We may have as many clusters as the number of frequent items identified, and a news article can be labeled as a member of multiple clusters. For an illustration, assume that we have ten news articles, identified frequent items, with $g_t = 0.4$, and their respective term frequencies in each document, as shown in Table 1.

Table 1
Example of Term Frequency of Frequent Items in Each Document.

	t_1	t_2	t_3	t_4	t_5
d_1	3	-	2	-	-
d_2	5	-	2	2	-
d_3	7	-	-	3	4
d_4	1	-	1	4	-
d_5	2	-	-	5	-
d_6	-	2	-	-	2
d_7	-	3	-	-	-
d_8	-	1	16	-	2
d_9	-	1	-	-	1
d_{10}	2	-	12	-	-

The cluster initialization creates five initial clusters, with five identified frequent items (t_1 to t_5) as cluster labels. Then a news document gets assigned repeatedly to the corresponding clusters, according to its own frequent items. For example, document d_1 is assigned to clusters c_{t1} and c_{t3}; document d_2 is assigned to clusters c_{t1}, c_{t3}, *and* c_{t4}. As a result, we have a set of initial clusters with their respective member documents, as shown in Table 2.

After cluster initialization, each news document has been assigned to at least one candidate cluster. We assume that each news article pertains to one and only one event episode, so in *cluster distinction*, TAFIED assesses the fit between a document and each candidate cluster, selects the most appropriate cluster, and generates a final set of clusters. We develop a fitness function to measure the likelihood that a document d_j belongs to a cluster c_x. Because we consider the temporal characteristics of a sequence of news articles, any features (terms) of news articles that describe the same event episode in principle should exhibit essential temporal characteristics (burst, new terms and developments, and temporal proximity) and share more important features than others. That is, a news article describing a specific event episode should share features of high appearance frequency and temporal adjacency with articles about that same episode, compared with articles pertaining to another episode. Formally, the fitness function between a document d_j and a cluster c_x is as follows:

$$Fitness(c_x \longleftarrow d_j) = \sum_{i=1}^{|T|} (\alpha \times CS(t_i,\ c_x) \times TFIDF(t_i,\ d_j) \times TP(c_x)) \quad (4)$$

where T is a set of frequent items, t_i denotes a frequent item, $CS(t_i, c_x)$ is the cluster support calculated as the percentage of documents in c_x that contain t_i, α is a parameter to control the impact direction of t_i, $TFIDF(t_i, d_j)$ represents the within-document term frequency × inverted document frequency for t_i appearing in d_j, and $TP(c_x)$ is a temporal proximity (TP) function for measuring the temporal adjacency of documents when d_j is assigned to c_x.

For TAFIED, we employ the TF × IDF measure to evaluate the novelty of feature t_i in the entire corpus of news articles; lower document frequency should produce a higher TF × IDF value. Furthermore, the frequent item t_i is essential to cluster c_x if it appears frequently in many documents of c_x, because a document that shares more important features with other documents in the same cluster should have a higher

Table 2
Example Initial Clusters and Their Respective Member Documents.

Initial Set of Clusters	Member Documents
c_{t1}	$d_1, d_2, d_3, d_4, d_5, d_{10}$
c_{t2}	d_6, d_7, d_8, d_9
c_{t3}	$d_1, d_2, d_4, d_8, d_{10}$
c_{t4}	d_2, d_3, d_4, d_5
c_{t5}	d_3, d_6, d_8, d_9

likelihood of belonging to that cluster. Inversely, the probability may decrease if a document shares fewer important features with the documents in c_x.[2] We thus use a parameter (α) to control the effect direction of frequent item t_i; specifically, α is set to -1 if the cluster support of t_i, CS(t_i, c_x), is lower than a prespecified significance threshold s_t (i.e., frequent item in d_j not satisfying the minimum cluster support of c_x), or to 1 otherwise.

Previous event episode discovery research considers temporal characteristics of news articles (or documents) by incorporating a linear time-penalty function or a nonlinear time-decaying function in the content-similarity measure to adjust the similarity between pairs of articles and then employing the adjusted similarity to group articles with a traditional document clustering algorithm. The underlying assumption is that the probability of two news articles pertaining to the same event episode would become lower if they are temporally distant; i.e., a large time interval between their published time. In this study, we instead approach event episode discovery from a different perspective by considering important characteristics of news articles that belong to an event episode. Specifically, news documents pertaining to an event episode contain similar content and often are published in close temporal adjacency (proximity). Unlike previous research that directly employs temporal distance to adjust interdocument similarity, we develop a TP function to measure the temporal adjacency of news articles in a cluster to determine the soundness of grouping them together; i.e., belonging to a specific event episode. The TP function, a criterion for our fitness measure, essentially considers that the articles in a cluster should be published in close temporal adjacency.

When assigning document d_j to a cluster, the fitness function considers the temporal adjacency of the documents in cluster c_x. The fitness score decreases if assigning document d_j to a cluster c_x is likely to increase the temporal difference among documents in that cluster. We develop a TP function $TP(c_x)$, defined as $\frac{e^{\lambda-\theta}}{1+e^{\lambda-\theta}}$ if $|c_x| > 1$ and 0.5 otherwise, where $\lambda = (|c_x|\text{-}1)\times w^2$ is the theoretically maximal temporal difference allowed between two time-ordered documents in cluster c_x, w is a parameter denoting the tolerant temporal difference of two time-ordered documents in c_x, and $\theta = \sum_{k=1}^{|c_x|-1} |t(d_k) - t(d_{k+1})|^2$ represents the sum of the squared actual temporal difference between two time-ordered documents in c_x, with $t(d_k)$ being the time stamp of document d_k in cluster c_x. The value of $TP(c_x)$ ranges between 0 and 1. A smaller temporal difference between the documents in c_x implies that documents are in temporal adjacency (i.e., proximity), thus resulting in a higher value of $TP(c_x)$. Furthermore, the value of $TP(c_x)$ gradually decreases as θ increases from 0 and decreases sharply as θ exceeds a threshold value. The proposed TP function can accommodate news articles pertinent to an event episode that are published in a later time period, so that the relevance between these news articles would not be unduly discounted by their temporal distance. Specifically, our TP function incorporates a parameter (w) to account for the tolerance in temporal difference between two time-ordered documents within a cluster. Thus, two articles are considered as relevant (pertaining to an event episode) if they are published within the tolerant temporal difference. As we illustrate in Fig. 3, with a cluster of five documents and w set to 2, so that $\lambda = 16$, the variance of $TP(c_x)$ increases with θ; the $TP(c_x)$ value decreases gradually as θ increases from 0 to 12, then decreases sharply as θ increases further. The value of $TP(c_x)$ reaches 0.5 when $\theta = 16$, such that $\theta = \lambda$.

[2] In TAFIED, we replace global support in FIHC with cluster support, because global support depicts the ratio between document frequency of a global frequent item and the entire document corpus, whereas cluster support indicates the ratio between document frequency of a global frequent item and the number of documents in a particular cluster. The objective of our fitness function is to determine which cluster is relatively appropriate to a document, which makes the use of cluster support intuitive for assessing the effect of global frequent items in a document with respect to a particular cluster.

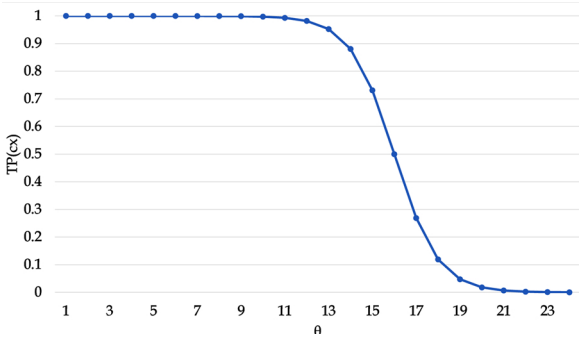

Fig. 3. Variance of Temporal Proximity Function.

To illustrate how we calculate the fitness score, we continue with the example of an initial set of clusters in Table 2. We set the significance threshold s_t to 0.3 and the tolerant time gap w to 2, then calculate the fitness score for document d_3 to each of its candidate clusters as follows:

$$Fitness(c_{t1} \longleftarrow d_3) = ((1 \times \tfrac{6}{6} \times 7 \times \log_2 \tfrac{10}{6}) + (1 \times \tfrac{4}{6} \times 3 \times \log_2 \tfrac{10}{4}) + (-1 \times \tfrac{1}{6} \times 4 \times \log_2 \tfrac{10}{4})) \times \frac{e^{20-29}}{1+e^{20-29}} = 0.001,$$

where $\lambda_{t1} = |6-1| \times 2^2 = 20$ and $\theta_{t1} = |2-1|^2 + |3-2|^2 + |4-3|^2 + |5-4|^2 + |10-5|^2 = 29$.

$$Fitness(c_{t4} \longleftarrow d_3) = ((1 \times \frac{4}{4} \times 7 \times \log_2 \frac{10}{6}) + (1 \times \frac{4}{4} \times 3 \times \log_2 \frac{10}{4}) + (-1 \times \frac{1}{4} \times 4 \times \log_2 \frac{10}{4})) \times \frac{e^{12-3}}{1+e^{12-3}} = 7.802$$

where $\lambda_{t4} = |4-1| \times 2^2 = 12$ and $\theta_{t4} = |3-2|^2 + |4-3|^2 + |5-4|^2 = 3$.

$$Fitness(c_{t5} \longleftarrow d_3) = ((-1 \times \frac{1}{4} \times 7 \times \log_2 \frac{10}{6}) + (-1 \times \frac{1}{4} \times 3 \times \log_2 \frac{10}{4}) + (1 \times \frac{4}{4} \times 4 \times \log_2 \frac{10}{4})) \times \frac{e^{12-14}}{1+e^{12-14}} = 0.358,$$

where $\lambda_{t5} = |4-1| \times 2^2 = 12$ and $\theta_{t5} = |6-3|^2 + |8-6|^2 + |9-8|^2 = 14$.

The fitness scores of d_3 with respect to candidate clusters c_{t1}, c_{t4}, and c_{t5} thus are 0.001, 7.802, and 0.358, respectively. Thus, document d_3 remains in cluster c_{t4}.

The use of frequent items as the base to cluster articles could lead to news articles that describe the same event episode scattered across multiple clusters after cluster distinction. For example, an episode, like cluster c_{t3} in Table 3 mainly related to frequent items{*t1*, *t3*} and documents that primarily contain frequent items{*t1*, *t3*, *t4*} form another cluster c_{t4}. Intuitively, the documents in cluster c_{t4} are a subset of those in cluster c_{t3}. However, in this example, they are separated to form another cluster after cluster distinction. As a remedy, we assess the need to merge two clusters in the cluster adjustment step. To perform *cluster adjustment*, TAFIED merges the clusters that contain highly similar or relevant documents. A combined cohesion measure evaluates the appropriateness of merging two clusters, and the combined cohesion of two clusters c_x and c_y is calculated as follows:

$$\text{Combined-Cohesion}(c_x \longleftarrow c_y) = \sqrt{Cohesion(c_x \longleftarrow c_y) \times Cohesion(c_y \longleftarrow c_x)} \times TP(c_x \leftrightarrow c_y) \quad (5)$$

Therefore, TAFIED might merge two clusters if their combined cohesion score exceeds a specified merging threshold η. To measure the cohesion of two clusters, the cluster to be merged (e.g., c_y) serves as the document, for which the fitness with respect to the other cluster (e.g., c_x) can be calculated. The cohesion function extends the fitness function to assess the fit of a document and a cluster (in the cluster distinction phase), and it normalizes the output value to between 0 and 2, to avoid any negative values. Formally, the cohesion function is defined as:

Table 3
Distinct Clusters and Respective Member Documents.

Distinct Set of Clusters	Member Documents
c_{t2}	d_6, d_7, d_9
c_{t3}	d_1, d_8, d_{10}
c_{t4}	d_2, d_3, d_4, d_5

$$\text{Cohesion}(c_x \longleftarrow c_y) = \frac{\sum_{i=1}^{|F|}(\alpha \times CS(f_i, c_x)) \times \sum_{dj \in c_y} TFIDF(f_i, d_j)}{\sum_{i=1}^{|F|}\sum_{dj \in c_y} TFIDF(f_i, d_j)} + 1 \quad (6)$$

where c_x and c_y are clusters to be considered for merging, $\sum_{dj \in c_y} TFIDF(f_i, d_j)$ is the sum of the TF × IDF values of f_i in each document d_j in cluster c_y, and $\sum_{i=1}^{|F|}\sum_{dj \in c_y} TFIDF(f_i, d_j)$ is the sum of the TF × IDF values of all frequent items in each document in cluster c_y. We next calculate the combined cohesion score of merging clusters c_{t3} and c_{t4} for the example in Table 3:

$$\text{Cohesion}(c_{t3} \longleftarrow c_{t4}) = 1.321 = \frac{(\frac{2}{3} \times (5+7+1+2) \times \log\frac{10}{6} \times 1) + (\frac{3}{3} \times (2+1) \times \log\frac{10}{5} \times 1) + (\frac{0}{3} \times (2+3+4+5) \times \log\frac{10}{4} \times (-1)) + (\frac{1}{3} \times (4) \times \log\frac{10}{4} \times 1)}{(5+7+1+2) \times \log\frac{10}{6} + (2+1) \times \log\frac{10}{5} + (2+3+4+5) \times \log\frac{10}{4} + (4) \times \log\frac{10}{4}} + 1$$

$$\text{Cohesion}(c_{t4} \longleftarrow c_{t3}) = 1.478 = \frac{(\frac{4}{4} \times (3+2) \times \log\frac{10}{6} \times 1) + (\frac{0}{4} \times (1) \times \log\frac{10}{4} \times (-1)) + (\frac{2}{4} \times (2+16+12) \times \log\frac{10}{5} \times 1) + (\frac{1}{4} \times (2) \times \log\frac{10}{4} \times (-1))}{(3+2) \times \log\frac{10}{6} + (1) \times \log\frac{10}{4} + (2+16+12) \times \log\frac{10}{5} + (2) \times \log\frac{10}{4}} + 1$$

$$\text{Combined-Cohesion}(c_{t3} \leftrightarrow c_{t4}) = \sqrt{1.321 \times 1.478} \times \frac{e^{24-17}}{1+e^{24-17}} = 1.396$$

4. Data and evaluation design

We empirically evaluated the proposed method, including several prevalent techniques as benchmarks. In this section, we describe the corpus of articles used in the evaluation and detail the benchmark techniques, performance measures, and parameter tuning analyses.

4.1. Document collection

We used the event corpus from Nallapati, Feng, Peng, and Allan [17], which contains 248 event episodes associated with 53 distinct events described by 1468 related news articles (stories), selected from TDT2 and TDT3 corpora. In this sample, the news articles are relatively balanced across the 53 events, and each event consists of a relatively modest number of news articles. The length of news documents averages 64.2 words, and the average number of features identified for each event, after feature extraction, is 520. In Table 4, we summarize the event corpus.

4.2. Benchmark techniques

Three prevalent techniques served as performance benchmarks: FIHC, HAC, and HAC augmented with a time-decaying function (HAC + TD). FIHC partitions (clusters) documents on the basis of the frequent itemset and its inclusion provides a base for the proposed TAFIED method. With this benchmark technique, we can evaluate the use of conventional frequent itemset-based clustering for event episode discovery. In line with previous research that applies feature-based clustering to discover event episodes, we also included HAC as a benchmark; this feature-based document clustering technique can support event episode discovery [12] and retrospective event detection [34], with promising effectiveness. Finally, we augmented HAC with the time-decaying function [17], such that HAC + TD can adjust the similarity of different news articles according to their temporal distance. The time-decaying similarity function is defined as $sim_{time\text{-}decaying}(d_i, d_j) = sim(d_i, d_j) \times \exp(-\frac{t(dj) - t(di)}{T})$, where $sim(d_i, d_j)$ is the cosine similarity between d_i and d_j, $t(d_i)$, which indicates the timestamp of the ith document, and T is the time interval between the first and last document in the time-ordered sequence of documents pertinent to the focal event, $t(d_{|S|})$ - $t(d_1)$, in which $|S|$ is the total number of documents describing that event. Overall, these benchmarks represent prevalent techniques for event episode discovery.

Table 4
Summary of Event Corpus.

	Average	Minimum	Maximum
Number of stories per event	27.7	16	30
Number of stories per episode	5.92	1	25
Number of episodes per event	4.68	2	8
Duration of episode (Days)	8.32	1	103
Duration of event (Days)	31.55	2	138

4.3. Performance measures

We used cluster recall and cluster precision to measure the performance of each technique for event episode discovery. Both measures reflect the association of documents in the same cluster (episode) and represent crucial performance measures for document clustering [44]. Given each event in the corpus of news articles, we considered known event episodes as the true (correct) episodes of the target event. The cluster recall (CR) and cluster precision (CP) of the target event are $CR = \frac{|CA|}{|TA|}$ and $CP = \frac{|CA|}{|GA|}$, where TA refers to the set of associations of documents in the true event episodes, GA denotes the set of associations of documents in the event episodes generated by a technique under evaluation, and CA is the set of associations of documents that exists in both the true and generated event episodes.

As an illustration, consider a sequence of documents $S = < d_1, d_2, d_3, d_4, d_5, d_6, d_7 >$ pertaining to an event, such that S can be classified into three true event episodes, EP_1, EP_2, and EP_3, where $EP_1 = \{d_1, d_2\}$, $EP_2 = \{d_3, d_4, d_5, d_6\}$, and $EP_3 = \{d_7\}$. Thus, seven associations of documents exist: $\{(d_1, d_2), (d_3, d_4), (d_3, d_5), (d_3, d_6), (d_4, d_5), (d_4, d_6), (d_5, d_6)\}$. Alternatively, we could let the event episodes identified by an investigated technique be G_1 and G_2, where $G_1 = \{d_1, d_2, d_3\}$ and $G_2 = \{d_4, d_5, d_6, d_7\}$. Then nine associations of documents, including $\{(d_1, d_2), (d_1, d_3), (d_2, d_3), (d_4, d_5), (d_4, d_6), (d_4, d_7), (d_5, d_6), (d_5, d_7), (d_6, d_7)\}$, would exist in the generated event episodes. In this case, four associations of documents, $\{(d_1, d_2), (d_4, d_5), (d_4, d_6), (d_5, d_6)\}$, are consistent in both the true and generated event episodes, so $CR = 4/7$ and $CP = 4/9$.

Using the cluster recall and cluster precision attained for each event in the corpus of news articles, we apply a weighted average method to measure overall effectiveness across all events. To assess the trade-off between cluster precision and recall, we used the precision/recall trade-off (PRT) curve [45] to reveal the effectiveness of a technique with respect to different merging threshold values, such as the intercluster similarity threshold for HAC and the cluster merging threshold for FIHC and TAFIED. For HAC, we examined merging thresholds between 0 and 1, in increments of 0.02. For both FIHC and TAFIED, the value of combined cohesion ranged from 0 to 2, and two clusters would be merged if the value exceeded 1. In general, PRT curves close to the

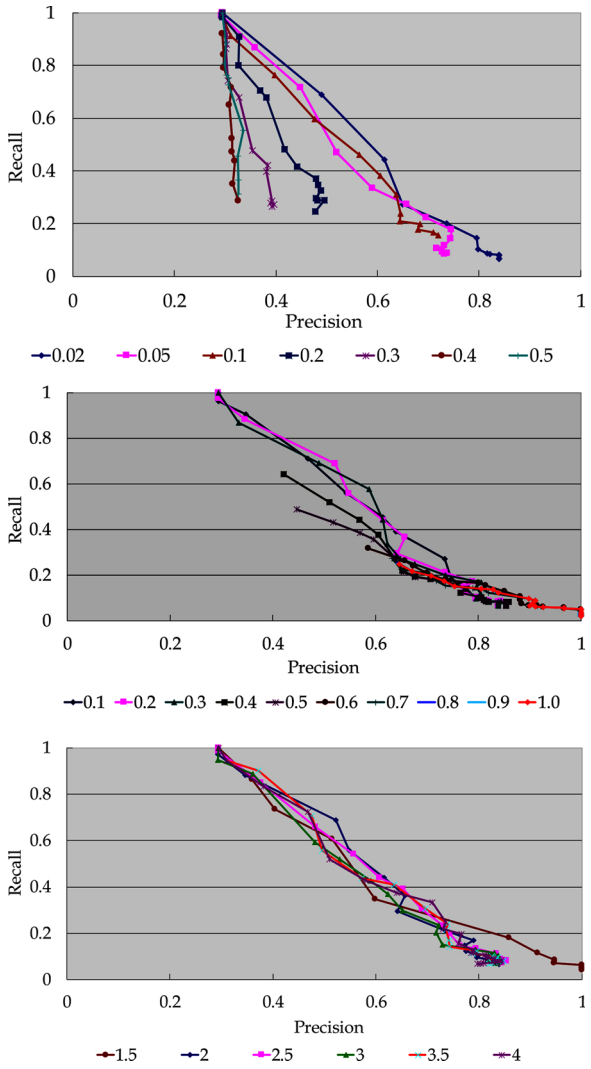

Fig. 4. (a) Minimum Global Support (g_t) of TAFIED Parameter Tuning. (b) Significance Threshold for Cluster Support (s_t) of TAFIED Parameter Tuning. (c) Tolerant Time Gap (w) of TAFIED Parameter Tuning.

upper right corner are more desirable than those near the point of origin.

4.4. Parameter tuning analyses

We collected a sample of news articles pertaining to 10 events and performed a series of parameter tuning experiments to determine appropriate values of the parameters essential to each investigated technique. In particular, several parameter values are important to TAFIED: minimum global support (g_t), the significance threshold for cluster support (s_t), and the tolerant time gap (w). We first evaluated g_t over the range of 0.02, 0.05, 0.1, and 0.5 (in increments of 0.1 between 0.1 and 0.5) with the default setting of $s_t = 0.2$ and $w = 2.0$. As we show in Fig. 4a, TAFIED appears effective with g_t set to 0.02.

We then assessed the effect of s_t (between 0.1 and 1.0, in increments of 0.1) on clustering effectiveness, with $g_t = 0.02$ and $w = 2.0$. As Fig. 4b indicates, the performance of TAFIED has a tradeoff between recall and precision when $s_t = 0.1$, 0.2, and 0.3. According to the overall results, we set $s_t = 0.2$ as the average performance.

Next, we examined the effectiveness of TAFIED with w ranging from 1.5 to 4, in increments of 0.5. As Fig. 4c shows, we also observe a trade off between $w = 2.0$ and 2.5. Overall, TAFIED achieves better results when $w = 2.0$, and appears most effective when $g_t = 0.02$, $s_t = 0.2$, and $w = 2.0$. We therefore adopted these parameter values to perform the subsequent evaluation.

For FIHC, we need to determine the minimum global support (g_f) and the significance threshold for cluster support (s_f). We followed the same procedure to tune essential parameters by fixing the value of s_f and then assessing values of g_f over the range of 0.02, 0.05, 0.1, to 0.5 (in increments of 0.1 between 0.1 and 0.5). As we show in Fig. 5a, the smaller the value minimum global support, the greater effectiveness FIHC achieves. We set g_f to 0.02 according to the performance tuning results.

We then set g_f to 0.02 to examine the effect of s_f (between 0.1 and 1.0, in increments of 0.1) on clustering effectiveness. As we indicate in Fig. 5b, FIHC achieved the best performance when $g_f = 0.02$ and $s_f = 0.8$.

Because HAC uses TF × IDF as the feature selection metric, we need to determine two important parameters: the number of features (k_h) and the document representation scheme (r_h). We examined the effects of k_h over the range between 50 and 250 (in increments of 50). As shown in Fig. 6a, setting k_h to 150 appeared most appropriate.

We also examined the effects of document representation scheme (r_h) on the effectiveness of HAC, using the binary and TF × IDF schemes. According to the results shown in Fig. 6b, we chose a binary scheme for r_h.

Finally, the parameters that need to be turned for HAC + TD are identical to those of HAC; we therefore followed the same procedure for HAC + TD and set the number of features to 150 and adopted TF × IDF as the document representation scheme.

5. Results and discussion

5.1. Comparative evaluation results

According to our comparative assessment of the effectiveness of TAFIED and three benchmark techniques (FIHC, HAC, and HAC + TD), as we show in Fig. 7, TAFIED is consistently more effective for event episode discovery than any other benchmark techniques across the different merging thresholds we consider. The traditional feature-based HAC, which does not consider essential temporal characteristics of news articles, appears least effective; FIHC, which uses frequent items to cluster news articles and considers the burst of features (terms) in the same event episodes to some degree, groups (clusters) articles that share more features and thus results in better performance than the feature-based HAC. Finally, HAC + TD, which considers temporal localization by using a time-decaying function to adjust the similarity between news articles, outperforms both HAC and FIHC, neither of which considers this temporal characteristic. This result is in line with previous research that suggests the use of a time-decaying function to improve the effectiveness of HAC for discovering event episodes [17]. Overall, the comparative results show our proposed method, which considers the frequent term set (burst and new terms) and temporal characteristics of news articles, capable of identifying different event episodes more effectively than the benchmarks.

Because of the trade-off between cluster recall and cluster precision, we performed a best-versus-best comparison by examining the best F-measure values achieved by the respective techniques for each of the 53 discovered events.[3] As Table 5 summarizes, TAFIED attains a F-measure value higher than that of any benchmark technique, in congruence with its lower cluster recall and the higher cluster precision values.

In addition, we assessed the statistical significance of effectiveness differentials of the respective techniques across the 53 events. Specifically, we conducted the Wilcoxon signed-rank test to examine the statistical significance of the effectiveness differences between pairs

[3] F-measure the harmonic average of precision and recall, defined as $\frac{2 \times Precision \times Recall}{Precision + Recall}$.

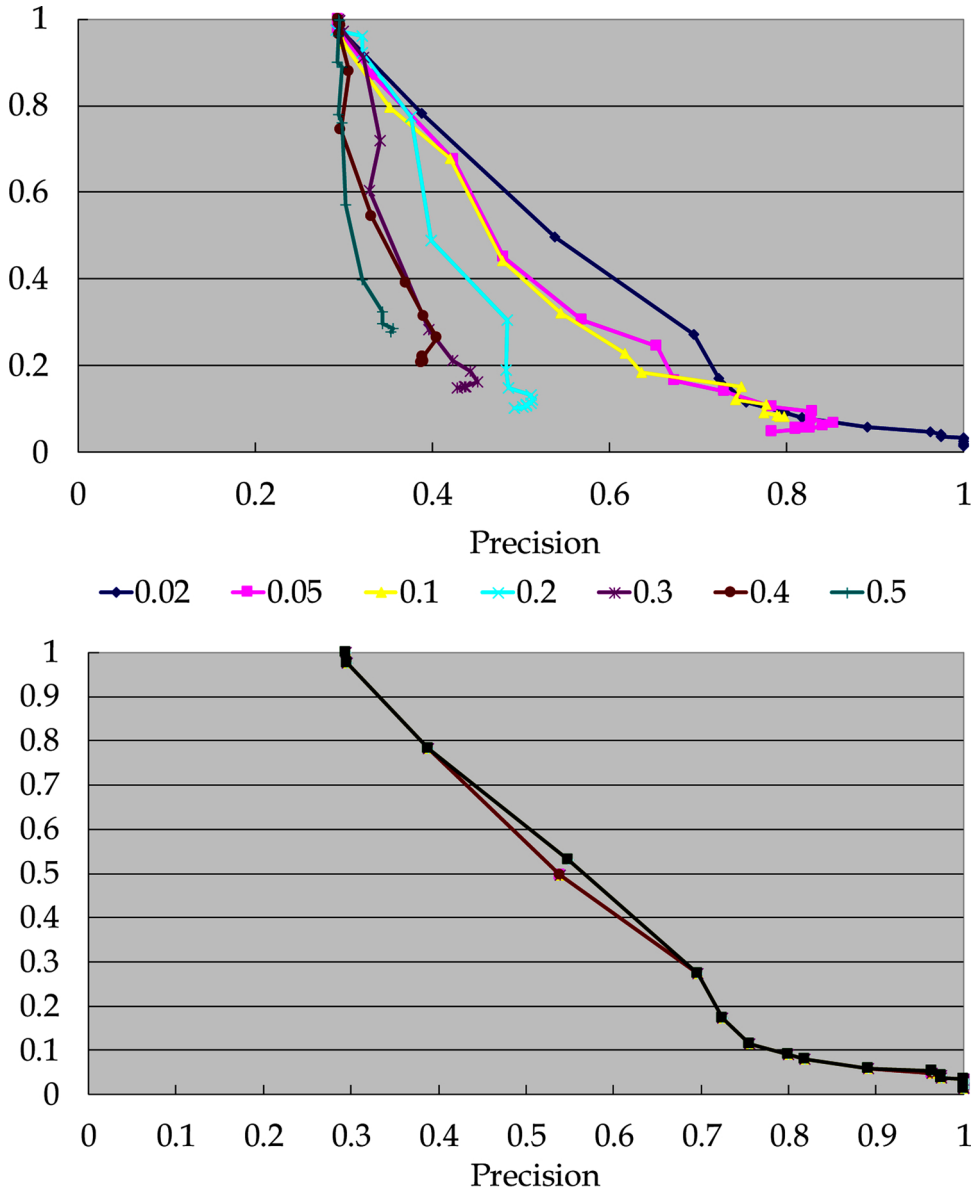

Fig. 5. (a) Minimum Global Support (g_f) of FIHC Parameter Tuning. (b) Significance Threshold for Cluster Support (s_f) of FIHC Parameter Tuning.

of F-measure values. The Wilcoxon signed-rank test, a nonparametric statistical hypothesis test, represents a legitimate alternative to the paired *t*-test when the distribution of the difference between two samples' means cannot be assumed to be normally distributed.[4] As we show in Table 6, the F-measure value attained by TAFIED is higher than those attained by HAC, FIHC, and HAC + TD significant at the 0.01, 0.05, and 0.1 level, respectively, while the F-measure values of the benchmark techniques were not significantly different. The results suggest TAFIED significantly more effective for discovering event episodes than the benchmark techniques.

5.2. Effect of temporal proximity on TAFIED performance

According to the evaluation results, TAFIED is more effective for discovering event episodes than either HAC or FIHC. To analyze the performance enhancement achieved by including the temporal characteristics of news articles, we examined its effect by removing the TP function from the proposed method and then re-evaluating its effectiveness. As we show in Fig. 8, the effectiveness of TAFIED without the TP function decreased substantially. This result explicitly indicates the value of TP function and reinforces the significance of considering temporal characteristics of news articles when clustering them for event episode discovery.

5.3. Importance of temporal characteristic for event episode discovery

Because the clustering process of FIHC is similar to that of TAFIED, we incorporated the TP function with FIHC to create FIHC + TP, which considers the temporal differences of the news documents within a cluster to measure the goodness-of-fit score for document assignments to clusters and cluster merge evaluation in the cluster distinction and cluster aggregation steps. We followed the same procedure to tune important parameters for FIHC + TP, including the parameter values for the tolerant time gap (w), the minimum global support (g_f), and the significance threshold for cluster support (s_f). The performance-tuning analyses suggested setting values of 0.1, 0.3, and 2 for the minimum

[4] McDonald, J.H. 2014. Handbook of Biological Statistics (3rd ed.). Sparky House Publishing, Baltimore, Maryland, pp. 180 – 185 (http://www.biostathandbook.com/pairedttest.html).

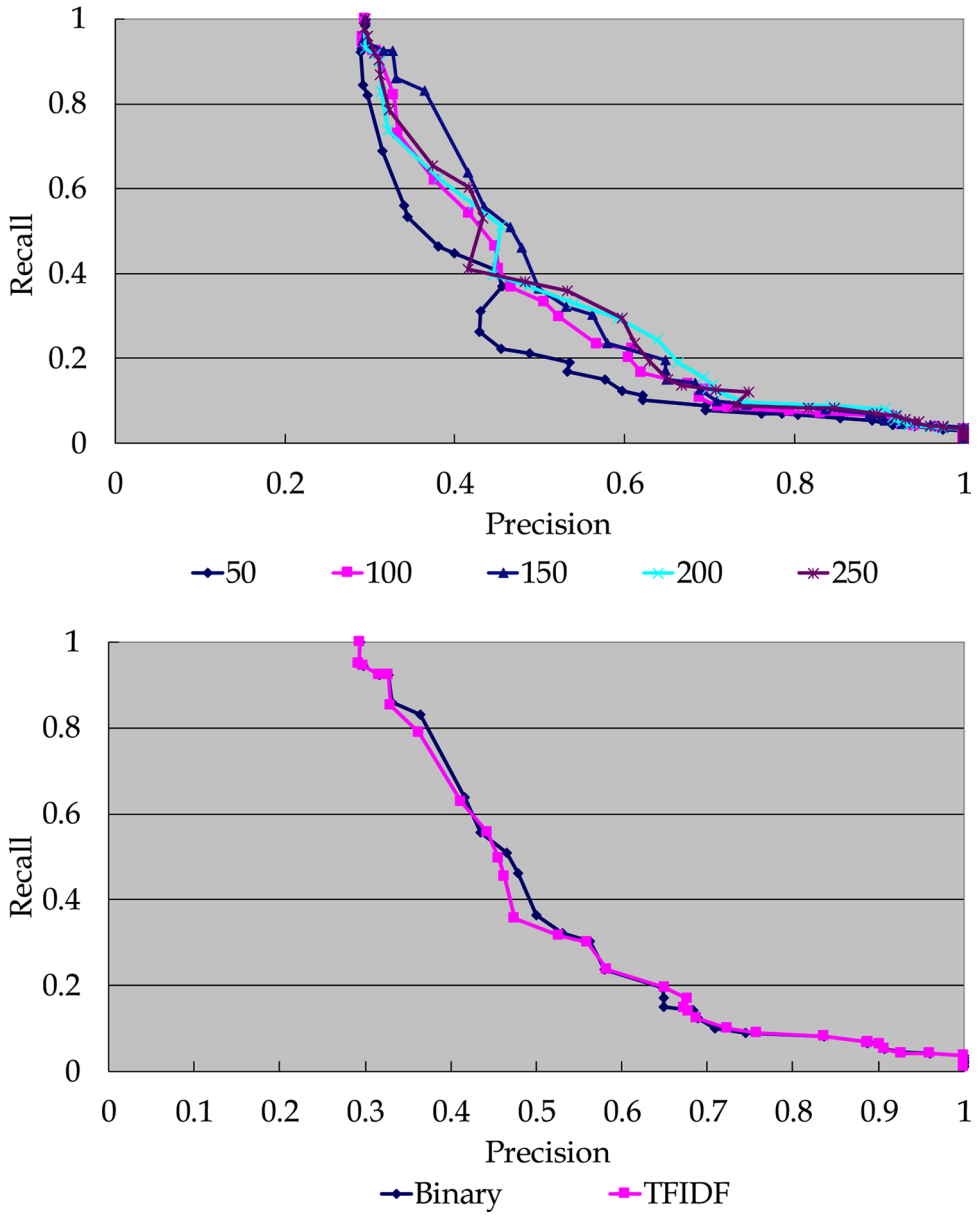

Fig. 6. (a) Number of Features (k_h) of HAC Parameter Tuning. (b) Document Representation Scheme (r_h) of HAC Parameter Tuning.

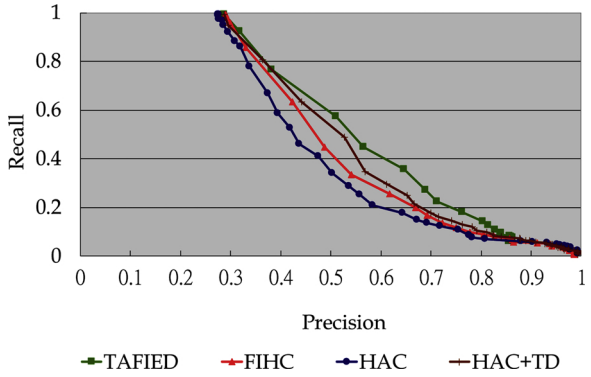

Fig. 7. Comparative Evaluation Results.

global support, significance threshold for cluster support, and tolerant time gap for FIHC + TP, respectively.

As we illustrate in Fig. 9, TAFIED remains more effective than FIHC + TP or HAC + TD, partially because it considers the essential frequent term set and temporal characteristics of news articles to discover event episodes. As noted, HAC + TD considers temporal localization by using a time-decaying function to adjust the similarity between news articles, and outperforms HAC and FIHC, neither of which takes this temporal characteristic into account. These comparative results reinforce the value of a time-decaying function to improve the effectiveness of HAC for event episode discovery. The performance of FIHC + TP is lower than that of HAC + TD but notably not better than that of FIHC. Together, these results suggest a relatively limited value of

Table 5
Best F-Measure Values with Associated Cluster Recall and Cluster Precision Values.

	Cluster Recall	Cluster Precision	F-measure
TAFIED	0.706	0.593	0.584
FIHC	0.724	0.498	0.543
HAC	0.741	0.472	0.533
HAC + TD	0.740	0.520	0.567

Table 6
Wilcoxon Signed-Rank Test on F-Measure Values of Investigated Methods.

	FIHC	HAC	HAC + TD
TAFIED	0.017**	0.002***	0.072*
FIHC	—	0.642	0.108
HAC	—	—	0.101

* *p-value* < 0.1.
** *p-value* < 0.05 and.
*** *p-value* < 0.01.

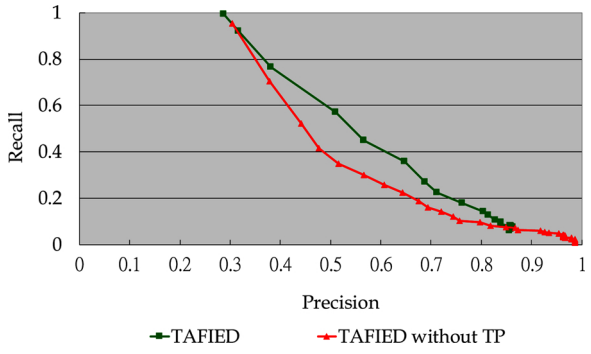

Fig. 8. Effects of Temporal Proximity on TAFIED Technique.

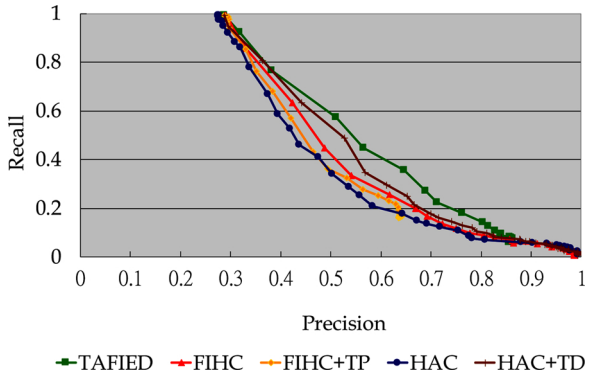

Fig. 9. Effects of Temporal Characteristics on Event Episode Discovery Techniques.

TP function to FIHC, possibly because FIHC depends on frequent terms for document clustering, yet the number of frequent terms in news articles pertinent to an event episode may not be small, and the articles describing different episodes of an event likely share similar frequent terms. These frequent terms then become less representative of the different event episodes and FIHC groups articles pertinent to distinct, temporally adjacent episodes in a cluster, thereby hindering its performance.

We also consider the computational processing requirements of the respective techniques. Overall, HAC is least computationally efficient; it performs the similarity analysis at the document level and exhaustively combines articles or article clusters that share the greatest similarity. In contrast, both FIHC and TAFIED determine a set of frequent terms, use these terms to cluster (group) news articles, and then make cluster assignments, distinctions, and merging decisions. The computational efficiency and scalability of HAC rely on the quality of news articles; those of FIHC or TAFIED depend on the number of initial clusters. When the number of initial clusters is smaller than the number of news articles, FIHC and TAFIED are more efficient than HAC. The incorporation of essential temporal characteristics, such as using a TP or time-decaying function, is not likely to affect computational efficiency significantly, because these functions are based on the temporal difference between documents.

6. Discussion and implications

The evaluation results reveal the criticality of frequent terms and temporal characteristics of news articles for event episode discovery. Existing techniques that cluster documents on the basis of feature similarity of pairwise documents cannot effectively identify distinct episodes from a news corpus, because the important features (terms) appearing in the documents about a specific event often highly overlap. As revealed in our analyses, methods that follow the frequent term-based approach (TAFIED and FIHC) outperform techniques using the feature-based similarity approach (HAC). As noted, the TAFIED groups documents on the basis of the set of frequent terms to gradually differentiate the documents pertaining to distinct episodes. The findings imply that the use of a set of frequent terms is both viable and advantageous to discover event episodes from a news corpus. By developing a method that is built on the use of frequent terms, we extend the event episode discovery research by highlighting another promising clustering strategy that groups news documents that pertain to a focal event into episodes, according to their frequent terms rather than the similarity of their feature sets.

Furthermore, temporal characteristics of news articles also are crucial to event episode discovery, particularly when considering various sources capable of publishing (releasing) a vast quantity of online news articles within a short time frame. As the evaluation results show, methods that consider temporal characteristics can support event episode discovery more effectively than prevalent techniques that do not consider such characteristics. Unlike the common use of a linear time-penalty function or a nonlinear time-decaying function to measure the temporal difference between pairs of documents and adjust their similarity, the proposed TAFIED method considers temporal characteristics by incorporating a TP function to measure the temporal adjacency of news articles and determine the soundness of grouping them in a cluster (the same episode), and is shown more effective for event episode discovery. This finding implies that a TP function that considers temporal tolerance to group news documents is more effective and flexible for identifying event episodes, because these episodes tend to vary in the length of their temporal intervals, which in turn indicates the need to properly measure temporal adjacency and further analyze its effects on event episode discovery.

After observing the proposed method's greater effectiveness, we also explored the existence of conditions that would favor our method or a benchmark technique. Because of the infeasibility of exhaustively considering all the different conditions across the 53 news events, we instead concentrated on the article distribution of the event that our method attains the highest F-measure value (0.908) while HAC + TD (best-performing benchmark) has a value of 0.731 as well as the article distribution of the event that HAC + TD achieves the highest F-measure value (0.905) while the proposed TAFIED has a value of 0.771. Fig. 10 illustrates the distribution of articles for two events, with x-axis being the publishing time, and the size of the triangle or circle indicating the number of articles published in the same day. As shown, both events

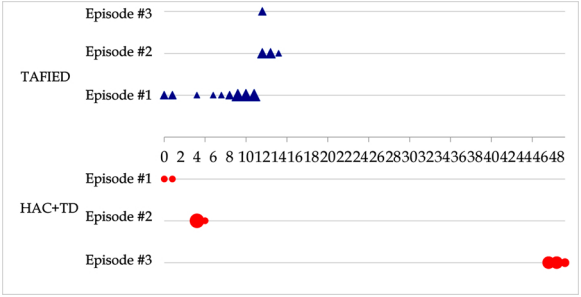

Fig. 10. Article Distribution of Event TAFIED and HAC + TD Shows Highest F-measure Value.

have three episodes and their first article is published at x = 0. In the event that TAFIED performs the best, Super Bowl XXXII between Denver and Green Bay (shown in the upper panel), we note a noticeable overlap in publishing time of the articles pertinent to Episode #2 and #3, with some articles of Episode#1 published in a small time difference and within the tolerance interval between articles (w). For the event that HAC + TD performs the best, a murder trial of two New Jersey teenagers accused of killing their newborn son (shown in the lower panel), articles that belong to distinct episodes are published in different time periods and those pertaining to the same episode are published in close temporal adjacency. As described, the proposed method considers two issues: news articles that describe different episodes of an event and have similar content, and different episodes that could emerge concurrently within a time window. According to the analysis results, our method achieves better performance when articles about different episodes of an event are published in an overlapped manner (temporal concurrency), which is common to the development and reporting of many events. On the other hand, conventional clustering techniques that rely on the temporal distance of the publishing time between articles for adjusting the inter-article similarity seem to perform well when different episodes of the event are separated by obvious time intervals.

Finally, the proposed TAFIED is more effective than benchmark techniques for discovering different episodes of an event from sequences of news articles. Our method can advance firms' practices by supporting automatic processing and analyses of online news articles (documents) for event evolution pattern discovery, so that firms can unfold distinct episodes of important events as they develop over time. By effectively clustering news articles into appropriate episodes, the proposed method enables firms to better predict important trends and changes in the environment for increased competitiveness. Additionally, the proposed method groups documents on the basis of their frequent items, which could be considered for labeling event episodes; i.e., suggesting appropriate labels for different event episodes. We use the news articles regarding "Hurricane Mitch" in our evaluation data set as an illustration. As shown in Fig. 11, this event comprises five episodes: "emergence of tropical storm Mitch," "hurricane Mitch upgrade," "path and impact forecasts," "damage estimations and precautionary measures," and "update and warning before landing," respectively. For episode labeling, we presented the frequent items identified by our method to a researcher knowledgeable about the events in the data set, who then selected those appropriate to label each episode. For example, frequent items such as "tropical storm Mitch," "Atlantic ocean," and "Jamaica" are selected as labels for "Emergence of tropical storm Mitch" episode. Example labels for "hurricane Mitch upgrade" episode include "watch," "strengthen," and "Hurricane warning."

7. Conclusion and future research directions

When performing environmental surveillance, companies may encounter multiple events and topics related to customers, market, competitors, industry, technology, or government regulations. Online news articles represent a common but crucial source for monitoring and tracking such events. To mitigate information overload and tedious processing required to monitor and track important events, firms need automated event episode discovery methods to classify and organize news articles about the same event that pertain to different episodes (subevents). We propose TAFIED and empirically examine its effectiveness, in comparison with feature-based HAC, HAC augmented with a time-decaying function, and FIHC. The evaluation results confirm that TAFIED outperforms the benchmark techniques as manifested by significantly better cluster recall and cluster precision values. In addition, incorporating a TP function can increase event episode discovery effectiveness.

This study contributes to extant literature in several ways. First, our novel TAFIED method can discover event episodes embedded in sequences of news articles that pertain to a specific news event. The evaluation results affirm that TAFIED is capable of automatically identifying distinct episodes of an event from news articles to enable subsequent event evolution pattern discovery. Second, most existing document clustering techniques, which rely on the similarity of important features in news articles to determine whether those articles belong to the same event episode, are not appropriate for event episode discovery, because articles pertinent to the same event likely have similar content (feature), except for the portion that is specific to its different episodes. We therefore stress the frequent itemset and consider features (terms) associated with different episodes to cluster news articles, which can group articles specific to an episode. In this effort, our method extends FIHC by considering essential temporal characteristics of news articles and thereby specifying distinct episodes from among sequences of news articles. The proposed TAFIED method emerges as more effective for discovering different episodes of an event than any benchmark techniques. Third, though previous research has cited temporal differences to assess the similarity of news articles and used a linear time-penalty function or nonlinear time-decaying function to adjust similarity accordingly, this approach is not effective for persistent episodes, because it offers limited temporal tolerance. For example, the similarity of two news articles, published one day apart, could be unduly discounted, which is ineffective for discovering longer episodes (e.g., reducing the similarity of news articles describing an episode with a 10-day duration by 10% or more). Our proposed method addresses the temporal tolerance issue and provides a TP function to evaluate the similarity of different news articles according to their temporal relationships. The evaluation results affirm that the incorporation of the TP function enhances the performance of the proposed TAFIED method for event episode discovery.

This proposed method can also support other applications. For example, distinct events could be categorized; events of the same category (type) might follow a defined progression pattern that consists of different episodes that occur in a temporal or causal sequence [12]. By discovering episodes of separate events of the same category (type), we can generalize the underlying event evolution patterns, through the identified episodes and their associations with respect to the focal event category, and thus better support event tracking by firms [12,14,18,19]. Furthermore, we consider temporal adjacency in the context of event episode discovery, which offers an important temporal characteristic of essential features that exist in other application domains. For example, discussion threads in an online forum have similar

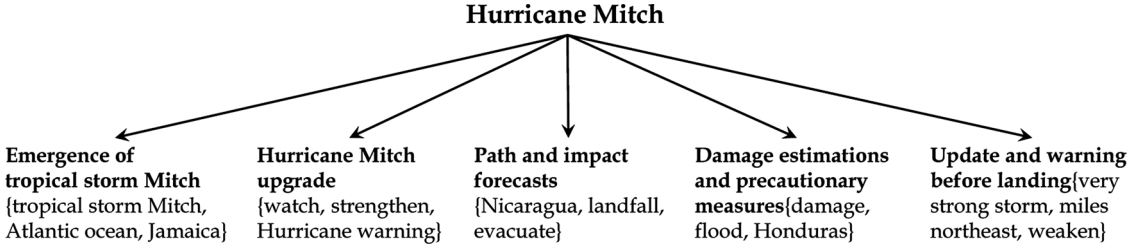

Fig. 11. Subset of Frequent Items in Each Episode of Event "Hurricane Mitch".

features, and discussions pertinent to a topic can be clustered according to their different subtopics, which should aid in managing the content in the forum [46].

This study can be extended in several promising directions. First, the news articles collected from TDT2 and TDT3 that are used in the evaluation include 53 events, each comprised of a relatively small number of episodes that range from 2 to 8 episodes. To produce more generalizable and robust results, additional sequences of news articles that describe more distinct, complex events should be included to evaluate the proposed method. Second, we assume that a news article pertains to one and only one episode; additional studies could relax this assumption by extending the proposed method to multi-episode analyses. Conceivably, a news article pertains to multiple episodes of an event, such as one that describes several event developments. Third, event episode discovery is crucial to multi-document summaries, so a promising extension would detail episode-based, multi-document summarization, as is essential for many application domains. Fourth, this study focuses on developing an effective method to discover episodes from news articles associated with an event. Although our results suggest its effectiveness, the proposed method's value and utility need further evaluations, including the use of additional sequences of real-world news articles in extrinsic evaluations. Last but not least, label choices for different episodes (from their respective frequent items) by targeted practitioners also deserve future research attention, so that the chosen labels are meaningful and appropriate for their practices. Although we observe that a researcher's selecting labels from the identified frequent items appears straight-forward (without any difficulty), we nevertheless acknowledge the subjectivity of label choices and recognize the need to assess the selected (suggested) labels' appropriateness by targeted practitioners using qualitative interview and case study methods.

CRediT authorship contribution statement

Yen-Hsien Lee: Conceptualization, Methodology, Writing - original draft, Writing - review & editing. **Paul Jen-Hwa Hu:** Validation, Formal analysis, Writing - original draft, Writing - review & editing. **Hongquan Zhu:** Validation, Formal analysis. **Hsin-Wei Chen:** Software, Investigation, Visualization.

Acknowledgment

This work was partially supported by the Ministry of Science Technology of the Republic of China (Taiwan) under Grants 99-2410-H-415-019-MY2 and 107-2410-H-415-011-MY3.

References

[1] C.W. Choo, Environmental scanning as information seeking and organizational learning, Inf. Res. 7 (2001) 1–14.
[2] C.V. Robinson, J.E.L. Simmons, Organising environmental scanning: exploring information source, mode and the impact of firm size, Long Range Plann. (2017).
[3] R.Y.K. Lau, S.S.Y. Liao, K.F. Wong, D.K.W. Chiu, Web 2.0 environmental scanning and adaptive decision support for business mergers and acquisitions, Mis Q. 36 (2012) 1239–1268.
[4] X.M. Xu, G.R. Kaye, Y. Duan, UK executives' vision on business environment for information scanning: a cross industry study, Inf. Manag. 40 (2003) 381–389.
[5] R.L. Daft, J. Sormunen, D. Parks, Chief executive scanning, environmental characteristics and company performance: an empirical study, Strateg. Manage. J. 9 (1988) 123–140.
[6] K.J. Sund, Scanning, perceived uncertainty, and the interpretation of trends: a study of hotel director's interpretation of demographic change, Int. J. Hosp. Manag. 33 (2013) 294–303.
[7] S.C. Jain, Environmental scanning in U.S. corporations, Long Range Plann. 17 (1984) 117–128.
[8] G. Jogaratnam, R. Law, Environmental scanning and information source utilization: exploring the behavior of Hong Kong Hotel and tourism executives, J. Hosp. Tour. Res. 30 (2006) 170–190.
[9] A.S.A. du Toit, Using environmental scanning to collect strategic information: a South African survey, Int. J. Inf. Manage. 36 (2016) 16–24.
[10] S. Tan, H.H. Teo, B. Tan, K. Wei, Environmental scanning on the internet, International Conference on Information Systems, Helsinki, 1998, pp. 76–87.
[11] H. Haase, M. Franco, Information sources for environmental scanning: do industry and firm size matter? Manage. Decis. 49 (2011) 1642–1657.
[12] C.P. Wei, Y.S. Chang, Discovering event evolution patterns from document sequences, IEEE Trans. Syst. Man Cybern. A. Syst. Hum. 37 (2007) 273–283.
[13] D.C. Luckham, Event Processing for Business: Organizing the Real-Time Enterprise, Wiley, 2011.
[14] Z. Li, S. Zhao, X. Ding, T. Liu, EEG: Knowledge Base for Event Evolutionary Principles and Patterns, Springer Singapore, Singapore, 2017, pp. 40–52.
[15] D.-R. Liu, M.-J. Shih, C.-J. Liau, C.-H. Lai, Mining the change of event trends for decision support in environmental scanning, Expert Syst. Appl. 36 (2009) 972–984.
[16] J. Yang, J. McAuley, J. Leskovec, P. LePendu, N. Shah, Finding progression stages in time-evolving event sequences, The 23th International World Wide Web Conference, ACM, Seoul, Korea, 2014.
[17] R.M. Nallapati, A. Feng, F. Peng, J. Allan, Event threading within news topics, Thirteenth ACM Conference on Information and Knowledge Management, Washington, D.C., 2004, pp. 425–432.
[18] C.C. Yang, X. Shi, C.P. Wei, Discovering event evolution graphs from news corpora, IEEE Trans. Syst. Man Cybern. A. Syst. Hum. 39 (2009) 850–863.
[19] S. Guo, K. Xu, R. Zhao, D. Gotz, H. Zha, N. Cao, EventThread: visual summarization and stage analysis of event sequence data, IEEE Trans. Vis. Comput. Graph. 24 (2018) 56–65.
[20] M. Ubaidullah Bokhari, K. Adhami, Event Evolution Modeling for Efficient News Search, (2015).
[21] X. Li, Y. Zheng, Y. Dong, Discovering evolution of complex event based on correlations between events, 11th Web Information System and Application Conference, Tianjin, China, 2014, pp. 47–50.
[22] Y. Cai, Q. Li, H. Xie, T. Wang, H. Min, Event relationship analysis for temporal event search, in: W. Meng, L. Feng, S. Bressan, W. Winiwarter, W. Song (Eds.), Database Systems for Advanced Applications, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 179–193.
[23] R.L. Liu, Collaborative multiagent adaptation for business environmental scanning through the Internet, Appl. Intell. 20 (2004) 119–133.
[24] C.P. Wei, Y.H. Lee, Event detection from online news documents for supporting environmental scanning, Decis. Support Syst. 36 (2004) 385–401.
[25] J. Granat, Event mining based on observations of the system, J. Telecommun. Inf. Technol. 3 (2005) 87–90.
[26] L. Fahed, A. Brun, A. Boyer, DEER: Distant and essential episode rules for early prediction, Expert Syst. Appl. 93 (2018) 283–298.
[27] Y. Ning, S. Muthiah, H. Rangwala, N. Ramakrishnan, Modeling precursors for event forecasting via nested multi-instance learning, Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, San Francisco, California, USA, 2016, pp. 1095–1104.
[28] R. Ahirrao, S. Patel, An overview on event evolution technique, Int. J. Comput. Appl. 77 (2013) 7–11.
[29] L. Kong, R. Yan, H. Jiang, Y. Zhang, Y. Gao, L. Fu, Mining event temporal boundaries from news corpora through evolution phase discovery, in: H. Wang, S. Li, S. Oyama, X. Hu, T. Qian (Eds.), International Conference on Web-Age Information Management, Springer, Berlin, Heidelberg, Wuhan, China, 2011, pp. 554–565.
[30] C.P. Wei, Y.H. Lee, Y. Chiang, C. Chen, C.C. Yang, Exploiting temporal characteristics of features for effectively discovering event episodes from news corpora, J. Assoc. Inf. Sci. Technol. 65 (2014) 621–634.
[31] D. Huang, S. Hu, Y. Cai, H. Min, Discovering event evolution graphs based on News articles relationships, 2014 IEEE 11th International Conference on e-Business engineering (2014) 246–251.
[32] A. Wen, W. Lin, Y. Ma, H. Xie, G. Zhang, News event evolution model based on the reading willingness and modified TF-IDF formula, J. High Speed Networks 23 (2017) 33–47.
[33] P. Zhou, B. Wu, Z. Cao, EMMBTT: a novel event evolution model based on TFxIEF and TDC in tracking News streams, 2017 IEEE Second International Conference on Data Science in Cyberspace (DSC) (2017) 102–107.
[34] Y. Yang, T. Pierce, J.G. Carbonell, A study on retrospective and on-line event detection, 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Melbourne, Australia, ACM Press, 1998, pp. 28–36.
[35] G. Kumaran, J. Allan, Text classification and named entities for new event detection, 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Sheffield, United Kingdom, ACM Press, 2004.
[36] G. Kumaran, J. Allan, Using names and topics for new event detection, Conference on Human Language Technology and Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Vancouver, British Columbia, Canada, 2005, pp. 121–128.
[37] K. Zhang, J.Z. Li, G. Wu, New event detection based on indexing-tree and named entity, 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Amsterdam, Netherlands, 2007, pp. 215–222.
[38] Q.H. Ramadan, M. Mohd, A review of retrospective news event detection, International Conference on Semantic Technology and Information Retrieval, Putrajaya, Malaysia, 2011, pp. 209–214.
[39] J. Allan, V. Lavrenko, R. Swan, Explorations within topic tracking and detection, in: J. Allan (Ed.), Topic Detection and Tracking: Event-Based Information Organization, Kluwer Academic Publishers, 2002, pp. 197–224.
[40] J. Allan, S. Harding, D. Fisher, A. Bolivar, S. Guzman-Lara, P. Amstutz, Taking topic detection from evaluation to practice, 32th Annual Hawaii International Conference on System Sciences, Big Island, Hawaii, 2005.

[41] E.M. Voorhees, Implementing agglomerative hierarchical clustering algorithms for use in document retrieval, Inf. Process. Manag. 22 (1986) 465–476.
[42] B. Fung, K. Wang, M. Ester, Hierarchical document clustering using frequent itemsets, SIAM International Conference on Data Mining (2003) 59–70.
[43] E. Brill, Some advances in rule-based part of speech tagging, 12th National Conference on Artificial Intelligence (AAAI-94), AAAI Press, Seattle, WA, 1994, pp. 722–727.
[44] D.G. Roussinov, H. Chen, Document clustering for electronic meetings: an experimental comparison of two techniques, Decis. Support Syst. 27 (1999) 67–79.
[45] M. Gordon, M. Kochen, Recall-precision trade-off: a derivation, J. Am. Soc. Inf. Sci. 40 (1989) 145–151.
[46] P.K. Srijith, M. Hepple, K. Bontcheva, D. Preotiuc-Pietro, Sub-story detection in Twitter with hierarchical Dirichlet processes, Inf. Process. Manag. 53 (2017) 989–1003.

Yen-Hsien Lee received his Ph.D. in Information Management from National Sun Yat-Sen University in Taiwan. He is currently an associate professor of the Department of Management Information Systems at the National Chiayi University in Taiwan. He was a visiting scholar at University of Utah in Fall 2002 and at University of Florida in Fall 2016. His papers have appeared in *Journal of Management Information Systems*, *Journal of Organizational Computing and Electronic Commerce*, *ACM Transactions on Management Information Systems*, *Artificial Intelligence in Medicine, Journal of the American Society for Information Science and Technology, IEEE Transactions on Systems*, *Man and Cybernetics*, and *Decision Support Systems*. His current research interests include knowledge discovery and data mining, knowledge management, information retrieval, text mining, and web mining.

Paul Jen-Hwa Hu is David Eccles Chair Professor at the David Eccles School of Business, the University of Utah. He has a Ph.D. in Management Information Systems from the University of Arizona. His current research interests include information technology in health care, technology implementation management, business analytics, e-commerce and digital government, technology-enabled innovation, human-computer interactions, and knowledge management. Hu has published papers in *Management Information Systems Quarterly*; *Information Systems Research*; *Journal of Management Information Systems*; *Decision Sciences*; *Decision Support Systems*; *Journal of Association for Information Systems*; *Journal of Information Systems* (*AAA*); *Journal of Service Research*; *Journal of Business Research*; and various ACM and IEEE journals and transactions.

Hongquan Zhuis a professor at the School of Economics and Management, the Southwest Jiaotong University. He had a Ph.D. in Management Science and Engineering from the Academy of Mathematics and Systems Science, Chinese Academy of Science in 2001. His current research interests include financial markets and institutions and empirical asset pricing. He was a visiting scholar at the University of Utah in Spring 2013 and Fall 2015. Zhu has published papers in *Journal of Banking and Finance*; *Journal of International Accounting Research*; *and International Review of Economics and Finance.*

Hsin-Wei Chen received an MBA in Management Information Systems from National Chiayi University, Taiwan. He is currently a senior consultant in the AdvancedTEK International Corporation, Taiwan. His research interests include data mining, text mining, and knowledge management.