A Multi-Relational Term Scheme for First Story Detection[☆]

Yanghui Rao[a], Qing Li[b], Qingyuan Wu[d], Haoran Xie[c,*],
Fu Lee Wang[c], Tao Wang[e]

[a]*School of Mobile Information Engineering, Sun Yat-Sen University, China*
[b]*Department of Computer Science, & Multimedia software Engineering Research Centre, City University of Hong Kong, Hong Kong*
[c]*Caritas Institute of Higher Education, New Territories, Hong Kong*
[d]*School of Management, Beijing Normal University, Zhuhai, China*
[e]*Department of Economics, University of Southampton, UK*

Abstract

First Story Detection (FSD) aims to identify the first story for an emerging event that had not been reported before, which is essential to practical applications in news analysis, intelligence gathering, and national security. Compared to information retrieval, text clustering, text classification and other subject-based tasks, FSD is event-based and thus faces the challenging issues of multiple events on the same subject and the evolution of events. To tackle these challenges, several schemes of exploiting temporal information, named entity, and topic modeling have been proposed for FSD. In this paper, we present a new term weighting scheme called *LGT* scheme which models the local element, global element and topical association of each story jointly. An unsupervised algorithm based on *LGT* scheme is then devised and applied to FSD. We evaluate four feature reduction strategies and also online model on *LGT* scheme. Experiments show that our approach yields better results than existing baseline schemes on retrospective and online FSD.

Keywords: First story detection, latent Dirichlet allocation, feature reduction, synonymous, polysemous

1. Introduction

Facing the vast amount of streaming data from newswire services, it is essential to organize news stories effectively and detect breaking news events efficiently.

[☆]A preliminary version of this paper has been published in [21].

[*]Corresponding author. Tel.: +852 68502340
Email address: hrxie2@gmail.com (Haoran Xie)

First story detection (FSD), also referred as new event detection, aims to identify the first stories that discuss emerging events, and has practical applications in domains like news analysis, intelligence gathering, and national security [15, 18, 6]. FSD arises out of, and has been recognized as, the most difficult task in topic detection and tracking (TDT) [31]. The purpose of TDT is to organize broadcast news stories by real world events that they discuss. In particular, FSD can be divided into retrospective FSD and online FSD. For retrospective FSD, all the stories are known in advance, and the task is to group the stories in the complete corpus into clusters, where each cluster represents an event, and stories in the cluster discuss that event. For online FSD, stories are generated continuously and ordered chronologically, and the task is to flag the onset of a previously unseen news event by marking stories as "new" or "old" as the stories arrive on the stream.

In the research area of FSD, a story is defined as a newswire article or a segment of news broadcast with a coherent news focus. The notion of a "topic" is sharpened to be an "event", i.e., a particular thing that happens at a specific time and place, along with all necessary preconditions and unavoidable consequences. For example, "The winter Olympics in Sochi" is considered as an event, whereas "The winter Olympics" is considered as a class of events (i.e., a subject). According to the definition, FSD is event-based, rather than subject-based as in information retrieval, text clustering and classification. Consequently, when applying a traditional term weighting scheme such as *TFIDF* to FSD, some problems will arise due to the facts that multiple events may belong to the same subject, and that an event can evolve. On one hand, multiple events on the same subject can lead to overestimation of the similarities between the seed story of an emerging event and old stories. This problem is akin to the phenomenon of polysemy, in which different semantic events are referred to by the same words. On the other hand, the evolution of events may cause the similarities between follow-up stories and the seed story on the same event to be underestimated. This problem is akin to the synonymous phenomenon, in which the same semantic event is referred to by different words. This paper focuses on addressing these polysemous and synonymous issues which are crucial to FSD. The main contributions of our work are as follows.

- A new term weighting scheme (viz, *LGT*) is proposed for FSD. The scheme can capture the uniqueness of each story; meanwhile, it can smooth the disturbing effect of synonymous and polysemous phenomenons.

- Four feature reduction strategies are implemented and evaluated on the proposed *LGT* scheme. Experimental results show that two parametric strategies are beneficial to FSD on diverse events, but they are sensitive to the thresholds and perform negatively on multi-events under the same subject.

For the nonparametric strategies, one of them deteriorates the performance, yet the other one not only may reduce redundant features, but also can improve the performance.

The remainder of this paper is organized as follows. The next section presents related works on FSD. In Section 3 we study existing schemes exploiting topic modeling and introduce some improved ones. In Section 4 we show the evaluation and comparison of our scheme against the classical *TFIDF* and two existing schemes exploiting topic modeling on retrospective and online FSD. Conclusions and discussion of further research follow in Section 5.

2. Related Work

Most algorithms for FSD are derived from information retrieval, text clustering and classification. As a result, one of the most popular term weighting schemes for FSD is *TFIDF*, including the group average clustering (GAC)-based hierarchical clustering algorithm, incremental clustering algorithm (also called story-story algorithm or single pass clustering), and story-cluster algorithm [26, 20, 4].

The GAC-based hierarchical clustering algorithm is designed for retrospective FSD, which detects events by maximizing the average similarity between story pairs in the resulting cluster. The story-story and story-cluster algorithms are employed for both retrospective and online FSD. The general idea of these algorithms is to first calculate the similarity between story pairs or between a story and cluster, then use the highest similarity to determine whether the current story is about an emerging event. As FSD focuses on events, which are described by news stories, rather than their broader subjects, several research efforts have been made on improving *TFIDF* by exploiting other technologies and information for the task of FSD, as discussed below.

2.1. Named entity-based schemes

The first line of research related to the improvement of *TFIDF* scheme is based on named entity recognition. Named entity such as person, organization, location, and date, is a kind of particular features. The underlying assumption of such named entity-based schemes is that named entities possess much more discrimination than other features in a news story. However, previous experiments found that the effectiveness of using named entities is dependent on the topics [16]. Thus, classification algorithms were then used to estimate the weights of named entities within different topics [30, 27].

These studies reveal several challenges of a term weighting scheme exploiting named entities. Firstly, classification algorithms need a set of tagged corpus

with topic labels, in order to generate a classifier for named entities, and there is no guarantee that a classifier trained from one topic can perform well on diverse events, which means it is sensitive to the domain of the training data. Secondly, the weighting functions of named entities, linear or nonlinear, have a certain effect on the performance of FSD. Some of them could improve the quality on several data sets, but others deteriorate the quality [29].

2.2. Schemes exploiting temporal information

Another line of research has improved *TFIDF* by considering the temporal information for FSD. Term weighting schemes exploiting temporal information attempt to adjust dynamically the similarity between stories, or the threshold of algorithms by using the publish time. The underlying assumption of such improved schemes is that stories within an event are likely to be adjacent in timeline [26, 20].

The temporal relation is usually modeled in a time window with a decay function. The size of a time window specifies the number of previous stories or events to be considered when doing clustering. The decay function weighs the influence of a story s in the window based on the gap between s and current story. The influence of s is larger if the time gap between s and the current story is narrower. However, Brants et al. [4] tested two different time decay models, a linear and an exponential, and found that all time-based results were worse than the baseline not using time information. The reason is that methods like time window and decay function are not suitable for both long-lasting and short-running events Chen et al. [7] which, nevertheless, are quite normal in real news corpus. Motivated by this, Chen et al. incorporated the aging theory to capture the life cycle of different events [7]. But it needs a set of manually tagged corpus to train the decay rate of events, in order to realize their self-adaptive event life cycle mechanism.

2.3. Schemes by topic modeling

Recently, several improved schemes exploiting topic modeling have been proposed, in which probabilistic latent semantic indexing (PLSI) and latent Dirichlet allocation (LDA) were used as context-dependent models for term smoothing.

As a model to deal with polysemy, PLSI as an enhanced model of latent semantic indexing (LSI) has a solid statistical foundation [14]. LSI is based on singular value decomposition (SVD) of the feature-by-document matrix of a corpus. Both PLSI and LSI project the original matrix to a lower rank space, so as to cope with synonymous and related features. Chou et al. [8] and Zhang et al. [31] proposed an incremental PLSI to capture the latent semantics within a corpus, and applied it to FSD by modifying the *TFIDF*. The PLSI parameters are first estimated by EM algorithm, and then the similarities between new and existing stories are calculated by smoothed feature vectors.

Latent Dirichlet allocation (LDA) is another popular topic model, where LDA parameters are estimated by the approximate inference algorithms, such as variants of EM and Gibbs sampling [12, 2, 24]. LDA estimates latent topics by hypothesizing that a document is constructed by words which are generated based on several topics. Unlike PLSI, LDA uses a detailed nonparametric Bayesian model of the prior probability over all the topics, in this way, the parameters will not grow with the increase of documents in the training set, and the model does not suffer from the overfitting problem [5].

Smet and Moens [10] utilized topics trained by LDA as one aspect of documents, and named entities as another aspect; the dissimilarities between each aspect of the documents are then combined to detect event clusters. As different types of name entities have different effects on different kinds of events, the output of the event clustering was further used to adjust the weight of factors. However, the output of this (preliminary) event clustering step might import noises, and the system has no guarantee of getting boosted from it. Furthermore, the weighting functions of different kinds of aspects and the method of combining aspects are hard to decide. Consequently, the results of their experiments were unstable, and some of them were even far less precise than the classical algorithm by *TFIDF*.

In the next section, through exploiting LDA we present an improved scheme of FSD which outperforms *TFIDF* and two other existing schemes. The result of our scheme is stable, and it can also be extended to online environments easily.

3. Term Weighting Schemes for First Story Detection

In this section, we first give a quick introduction to the existing schemes exploiting topic modeling, and then describe our approach in detail.

3.1. Existing schemes

Latent Dirichlet allocation (LDA) is a generative probabilistic model of a corpus [3]. The power of LDA lies in the natural modeling of synonymous and polysemous words [10]. As a result, it is natural for FSD algorithms to use LDA in order to alleviate the synonymous and polysemous problems. In the LDA model, latent topics are sampled from a topic distribution, and such a distribution is denoted as $p(z|d, \theta)$. The parameter is chosen from a Dirichlet prior . Each word w is sampled from each topics word distribution, and such a distribution is denoted as $p(w|z, \beta)$. LDA model learns both kinds of distributions in an unsupervised way, and typical values for the number of topics K to be useful lie in the range of 100 to 300 for the English data sets [10]. Besides, by learning additional latent variables which are independent of the training corpus, the topic distributions of new or previously unseen documents can be inferred.

Generally, there are two different schemes of exploiting LDA (or PLSI) for term weighting in the research areas of IR and FSD: 1) as a context-dependent unigram model [31, 8] to smoothen the empirical word distributions in documents; 2) as a latent space model [10] which provides a low-dimensional document representation.

In the first scheme [31, 8], a document d is represented by a smoothed version of the feature vector:

$$\vec{d} = \left(p(w_1|d) \times IDF(w_1), p(w_2|d) \times IDF(w_2), ..., p(w_v|d) \times IDF(w_v)\right), \tag{1}$$

where

$$p(w|d) = \sum_{z \in Z} p(z|d)p(w|z), \tag{2}$$

$$IDF(w) = \log(M/df_w). \tag{3}$$

In the above, v is the number of distinct features in story d, Z is the set of all topics, M is the total number of documents, and df_w is the number of documents that contain feature w.

Then, the similarity between any two documents can be calculated by cosine function as follows:

$$sim(\vec{d_i}, \vec{d_j}) = \frac{\vec{d_i} \cdot \vec{d_j}}{|\vec{d_i}| \times |\vec{d_j}|}. \tag{4}$$

In the second scheme [10], a document d is represented by the distribution of K topics associated with it, i.e.,

$$\vec{d} = \left(p(z_1|d), p(z_2|d), ..., p(z_K|d)\right). \tag{5}$$

Then, the distances are calculated by the symmetric Kullback-Leibler divergence of the K-dimensional probability distributions d_i and d_j:

$$KL(d_i, d_j) = \frac{1}{2}\left(\sum_{z \in Z} d_i^{(z)} \log \frac{d_i^{(z)}}{d_j^{(z)}} + \sum_{z \in Z} d_j^{(z)} \log \frac{d_j^{(z)}}{d_i^{(z)}}\right) \tag{6}$$

Although named entities can be combined into these schemes, the result is quite unstable. According to Banerjee and Basu, existing schemes exploiting LDA may perform significantly worse than the classical *TFIDF* [1].

3.2. Proposed approach

In this section, we first present the basic scheme of our approach, and then describe four feature reduction strategies, as well as a model for online environments.

3.2.1. The LGT scheme

In the basic scheme, we divide the features into the local element, global element and topical association. Fig. 1 illustrates the framework of our proposed *LGT* scheme. The local element is extracted from documents, which is used to capture the uniqueness of each story. The statistics of the whole corpus is modeled by the global element. Furthermore, we exploit the latent topics to construct the topical association.

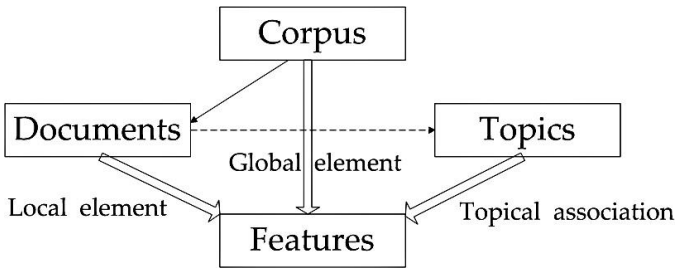


Figure 1: The framework of LGT scheme.

Local element (Document-level): To model the statistics and capture the uniqueness of individual stories, the local element can be represented by the Bernoulli model, relative term frequency, or the smoothed term frequency. We use the last one to represent the local element:

$$le(d, w) = \log(tf_d^{(w)} + 1), \tag{7}$$

where $tf_d^{(w)}$ means how many times word w occurs in news story d.

Global element (Corpus-level): To model the statistics of the whole corpus and reflect the salience of each story, the global element can be captured by the smoothed inverse document frequency (IDF) [30], as follows:

$$ge(d, w) = \log\left(\frac{M+1}{df_w + 0.5}\right), \tag{8}$$

where M is the total number of stories in the corpus, and df_w denotes the number of stories that contain word w.

Topical association (Topic-level): To model the intrinsic common relationships among disjoint components (i.e., the synonymous and polysemous words), topical association can be represented by topic modeling. As the LDA model captures the low-rank aspect and latent topics well, we use $p(w|d; \alpha, \beta)$ to represent the topical association, so as to smooth the disturbing effect of synonymous and

polysemous in the stories. The formula of the topical association is given below:

$$ta(d, w) = \sum_{z \in Z} p(z|d, \theta)p(w|z, \beta), \tag{9}$$

where $p(z|d, \theta)$ and $p(w|z, \beta)$ denote, respectively, topic zs distribution of a given document d, and topic zs word distribution; $\theta \sim Dir(\alpha)$, α and β are hyper parameters specified by users. Because it is intractable to perform an exact inference, an approximate inference method based on Gibbs sampling can be used to estimate the parameters. For the i-th word w_i, the conditional posterior distribution $p(z_i = z|z_{-i}, w; \alpha, \beta)$ can be derived as follows:

$$p(z_i = z|z_{-i}, w; \alpha, \beta) \propto (n^{(z)}_{d_i,-i} + \alpha) \times \frac{n^{w_i}_{z,-i} + \beta}{\sum_{w'}(n^{(w')}_{z,-i} + \beta)}, \tag{10}$$

where z_i is the candidate topic that w_i is assigned to, z_{Ci} refers to the topic assignments of all other words, d_i indicates the document from which word w_i is sampled, $n^{(z)}_d$ is the number of words in document d assigned to topic z, $n^{(w)}_z$ is the number of instances of word w assigned to topic z. The superscript Ci means the number that does not include the current assignment of word w_i.

After the above sampling, it is convenient to estimate the probability of topic z conditioned on document d, and the probability of word w conditioned on topic z by

$$p(z|d, \theta) = \frac{n^{(z)}_d + \alpha}{\sum_{z'}(n^{z'}_d + \alpha)}, \tag{11}$$

$$p(w|z, \beta) = \frac{n^{(w)}_z + \beta}{\sum_{w'}(n^{w'}_z + \beta)}. \tag{12}$$

Then, each story d is represented by a vector below:

$$\vec{d} = (weight(d, w_1), weight(d, w_2), ..., weight(d, w_v)), \tag{13}$$

where $weight(d, w)$ denotes the combined weight of features in d, which is normalized by the sum of the combined weight of all features in the story, i.e.,

$$weight(d, w) = \frac{le(d, w) \times ge(d, w) \times ta(d, w)}{\sum_{w'} le(d, w') \times ge(d, w') \times ta(d, w')}. \tag{14}$$

The combined weight of word w in story d given above, i.e., $weight(d, w)$ or $p(w|d)$, could be deduced by the following processes.

The probability of word w conditioned on story d can be estimated by introducing a topic level, as follows:

$$p(w|d) = \sum_{z \in Z} p(z|d)p(w|d,z), \tag{15}$$

where $p(w|d,z)$ denotes the probability of word w conditioned on both the story d and the latent topic z. In the previous studies [31, 8, 14], a conditional independence assumption is usually made to approximately estimate $p(w|d,z)$. The conditional independence assumption means that, conditioned on the latent topic z, all features w are generated independently of the specific document d. Based on this assumption, $p(w|d,z)$ equals to $p(w|z)$. The same convention is followed in FSD (ref. Eq. 2). In our scheme, we try to partially relax the approximate estimation of $p(w|d,z)$.

We treat each word w in story d by an independent dual occurrence, in which we represents ws explicit occurrence that only depends on story d, and w_i denotes ws implicit occurrence that only depends on latent topic z. As a result,

$$p(w|d) = \sum_{z \in Z} p(z|d)p(w_e, w_i|d,z) = p(w_e|d) \sum_{z \in Z} p(z|d)p(w_i|z), \tag{16}$$

where the probability of we conditioned on d, i.e., $p(w_e|d)$ on the right-hand side of the above equation, can be represented by the smoothed term frequency-inverse document frequency, thus $p(w_e|d) = le(d, w_e) \times ge(d, w_e)$. The rest part on the right-hand side is $ta(d, w_i)$ according to Eq. 9. Since each word w is represented by the dual occurrence, both w_e and w_i have the same typeface (while different semantemes) with w, and each story d is represented as in Eq. 13.

Finally, the similarity between any two stories d_1 and d_2 is calculated by the Hellinger distance [30]:

$$sim(\vec{d_1}, \vec{d_2}) = \sum_{w \in d_1, d_2} \sqrt{weight(d_1, w) \times weight(d_2), w)}. \tag{17}$$

In our experiment, we use an incremental clustering algorithm to test the effectiveness of all term weighting schemes. The algorithm is one of the basic yet most popular algorithms for FSD, and is suitable for both retrospective and online FSD [20, 4]. In this algorithm, the current story is compared to each previous story in memory, and the highest similarity is used to determine whether the current story is on an emerging event.

Note that our scheme is different from the two schemes of using LDA (or PLSI) described in Section 3.1. For the first scheme there, the story is represented by the combination of $p(z|d)$, $p(w|z)$ and $IDF(w)$ (ref. Eq. 1). While it captures

the intrinsic common relationships among disjoint components, there is no aspect designed to model the statistics of individual stories. For the second scheme, the story is represented by the documents topical distributions $p(z|d)$ as given in Eq. 5. As our experimental study is going to reveal, using it as story representation is not recommendable for FSD either.

3.2.2. Feature reduction strategies

Feature reduction is a process that removes a subset from the original feature set according to some criteria [28]. It is useful to conduct feature reduction for FSD, since rare words with extremely low document frequency may not be influential in performance [4, 25], and topical common words may cause events in the same subject to be mutually confusing [27]. For evaluation purpose, we have implemented four feature reduction strategies, three of which are from the literature, and the fourth one is devised by us. Note that each strategy uses a different criterion to eliminate a desired degree of features.

The first strategy removes those features whose document frequency is less than some predetermined threshold. The basic assumption is that extremely rare features are either non-informative, or not influential in performance. In either case, the removal of rare features reduces the feature dimension and improves the efficiency of FSD [4, 25].

The second strategy takes the opposite process, which eliminates features whose document frequency is larger than a threshold. The assumption is that topical common features are a potential cause for FSD to miss the first story of an emerging event, since those features can cause events on the same topic to be mutually confusing [27]. However, such mechanism requires manual parameters setting, and is dependent on topics.

The third strategy is a compromised form of the above two [28]. Given a set of word document frequency $DF = df_1, df_2, ..., df_v$, it first calculates the means (μ) and deviation (σ) of DF, and then keeps features whose document frequency lies in $\mu C\sigma < df_w < \mu + \sigma$. All features with low and high document frequency are removed from the original feature set. The advantage of this method is that no threshold is needed. However, it is unreasonable to treat the distribution of word document frequency as bell-shaped.

The last strategy reduces the feature set based on the logarithmic normal distribution (long-tail or positive asymmetry distribution) of word document frequency. The assumption is that most features document frequency lies in the median, and few features occur in all or only one of the documents. The mechanism involves two steps. First, the original set of word document frequency is transformed to the logarithmic form $LDF = \log(df_1), \log(df_2), ..., \log(df_v)$. It is common to take logarithms of positively skewed data to make the distribution more symmetric

(so that it approximates a normal distribution more closely) [13]. Second, features are reduced by the boxplot of LDF. The boxplot bases use the minimum value ($MinV$), the maximum value ($MaxV$), the first and third quartile (Q_1 and Q_3) and the interquartile range (IQR). Then, the lower and upper limits (L and U) can be defined according to [9]. Specifically,

$$L = max(MinV, Q_1 - 1.5 \times IQR), \tag{18}$$

$$U = min(MaxV, Q_3 + 1.5 \times IQR). \tag{19}$$

where IQR is the difference between the third quartile and the first quartile, i.e., $IQR = Q_3 C Q_1$. The boxplot indicates the presence of anomalous observations or outliers. The strategy removes those features whose logarithmic document frequency lies outside the lower and upper bounds. Since the document frequency of each feature has been calculated in the initialization step, it is convenient to embed these strategies into the LGT scheme.

In Section 4.4.2, we will examine the performance of all the above four feature reduction methods with respect to the task of FSD.

3.2.3. Online model

Our scheme can be extended for online FSD as follows. For the local element, we compute the smoothed term frequency of the current story according to Eq. 7. For the global element, we use the incremental IDF model from [26, 30], namely:

$$ge(d, w, T) = \log\left(\frac{M_T + 1}{df_T(w) + 0.5}\right), \tag{20}$$

where T is the current time point, M_T is the number of accumulated documents up to the current point, and $df_T(w)$ is the number of documents which contain feature w up to the current point.

In the incremental IDF model, the document vocabulary is incrementally updated and IDF recomputed each time a new document is processed. An empirical analysis shows that incremental IDF can be effective in document retrieval after a sufficient number of"past" documents have been processed. Obviously, the time overhead will be high if we update the IDF with each incoming document. Zhang et al. [30] proposed to update document frequencies dynamically in each time window, within which 50 news stories are included. We also start with a retrospective corpus containing initially 50 news stories, and update it whenever there are 50 new stories come in. The formula is given below:

$$df_T(w) = df_{T-1}(w) + df_{D_t}(w), \tag{21}$$

where D_T represents the news story set received in time T, $+df_{D_t}(w)$ means the number of documents in which feature w occurs, and $df_{T-1}(w)$ denotes the total number of documents in which feature w occurs before time T.

For the topical association $p(w|d, \alpha, \beta)$ trained by LDA, it can also be extended to online environments reasonably well. In fact, one of the advantages of LDA is the possibility of inferring the topic distributions of new documents [10], so it is applicable to deal with a stream of news stories in which new events are added continuously. While there are many online training models for both LDA [2] and PLSI [31, 8, 24], we use the batched LDA training method for our study, the effect of which is evaluated in Section 4.4.3.

4. Experiment and Evaluation

In this section, we conduct experimental studies to evaluate the effect of our scheme on both detecting emerging events under the same subject, and detecting emerging events under diverse subjects. The classical *TFIDF* scheme and two existing schemes of exploiting topic modeling are implemented, and compared to our proposed *LGT* scheme on retrospective and online FSD.

4.1. Data sets

TDT5 data set, provided by the Linguistic Data Consortium (LDC), is the latest corpus from English, Chinese and Arabic news sources[1]. It has labeled 250 events, and 10,002 stories belonging to at least one of these events. All the events are classified according to 13 "Rules of Interpretation" (ROI) which state the general category or "subject" of them.

For our evaluation, we use two subsets from TDT5 with each reflecting different aspects of news events. The first sub-set contains events from different subjects with diversity, which is suitable for us to evaluate the effect of schemes on detecting emerging events under diverse subjects. The second sub-set contains events with the same ROI, which can facilitate us to evaluate the effect of schemes on distinguishing similar events under the same subject.

TDT5-diversity: This first subset is a diverse data set which contains English only stories under 11 subjects. Each subject has one or multiple events (from 1 to 15), and each event contains multiple stories (from 5 to 270). This data set totally has 1,403 stories which are divided into 63 events. After removing stop words and stemming by Porter Stemmer algorithm, it contains 12,175 distinct features.

[1] `http://projects.ldc.upenn.edu/TDT5/.`

Table 1: Statistics of the Data Sets

Data Set	Subject	♯events	♯stories
TDT5-diversity	Accidents	4	56
	Acts of Violence or War	7	206
	Celebrity and Human Interest News	11	185
	Elections	1	80
	Financial News	1	5
	Legal/Criminal Cases	15	270
	Miscellaneous News	11	160
	Natural Disasters	1	17
	New Laws	4	93
	Scandals/Hearings	3	169
	Sports News	5	162
TDT5-centralization	Acts of Violence or War	36	4,076

TDT5-centralization: Events under the subject of "Acts of Violence or War" are chosen as the second data set, which covers 40.8% of the whole labeled 10,002 stories under the single subject. In particular, it contains 4,076 stories which are divided into 36 events, such as "Murder-suicide in San Diego", "Bomb explosion in Pakistan", "Taliban Attack in Afghanistan". By removing stop words and stemming via Porter stemmer algorithm, 13,267 distinct features are retained.

Table 1 summarizes the two data sets. In terms of the number of stories and distinct features, we can see the diversity of the first data set: on one hand, the number of stories of TDT5-diversity is only 34.4% of that in TDT5-centralization; on the other hand, the number of distinct features of TDT5-diversity is 91.8% of that in TDT5-centralization.

4.2. Experimental setup

To test the effectiveness of the various approaches on solving the two general problems in FSD (i.e., multiple events on the same subject, and the evolution of events), we run all the term weighting schemes on TDT5-centralization and TDT5-diversity. Our proposed *LGT* and the existing schemes are denoted as follows:

LGT: This is our scheme which divides stories into local element, global element and topical association. For retrospective FSD, the weight of features is represented according to Eq. 13. Hellinger distance is used to measure story simi-

larities. For online FSD, both batched LDA model and incremental IDF model are used to generate feature weights.

PIDF: This is the first existing scheme of exploiting topic modeling. For retrospective FSD, each story is represented as a smoothed feature vector according to Eq. 1, and cosine function is used to measure story similarities. The model was estimated by incremental PLSI in [31, 8]. Here we use LDA to avoid the local optima, slow converging and overfitting of PLSI [12]. For online FSD, both batched LDA model and incremental IDF model are used to generate feature weights.

PZ: This is the second existing scheme of exploiting topic modeling. For retrospective FSD, each story is represented as a low-dimensional probability distribution according to Eq. 5, and symmetric Kullback-Leibler divergence is used to measure story dissimilarities. Named entities were used as the other aspect of stories in [10]. However, as the weighting function of different aspects and the method of aspect combination are hard to decide, the result is unstable according to [10]. So we adopt the latent space model as the representation of stories in this study. For online FSD, batched LDA model is used to generate document weights.

TFIDF: This scheme is also used as the baseline in [30, 19]. For retrospective FSD, each story is represented as the *TFIDF* vector, and Hellinger distance is used to measure story similarities. For online FSD, incremental IDF model is used to generate feature weights.

Note that the term weighting of each story in the *PIDF* scheme is unnormalized, so we adopt the cosine function to measure story similarities (as in [31, 8].), since Hellinger distance for *TFIDF* is only suitable for normalized term weighting [30, 19]. Also note that the term representation of the *PZ* scheme is at topic level, rather than word level as in other schemes. The former is essentially the probability distribution, so we use symmetric Kullback-Leibler divergence to measure the story distance (as in [10]). That is to say, these schemes include both their specific term representations and the corresponding measures of story similarities or distances. Besides, for all runs of LDA model, we set the initial super parameters $\alpha = 50/T$ and $\beta = 0.01$. These parameters were found to work well for LDA with many different text collections [23]. The number of iterations is set to 2,000 to ensure the convergence of the model.

We also implement and evaluate the four feature reduction strategies (cf. Section 3.3.2) on TDT5-centralization and TDT5-diversity, as follows:

textbfDFr: This is the first method of feature reduction by eliminating rare features whose document frequency is less than some predetermined threshold.

DFc: This is the second method of feature reduction by removing common features whose document frequency is larger than a manual parameter.

DFmd: This is the third method based on the means and deviation of word document frequency. The mechanism treats the distribution of word document

Table 2: Cluster-Event Contingency Table

	In event	Not in event
In cluster	*a*	*b*
Not in cluster	*c*	*d*

frequency as standard normal, and it does not need to predetermine any threshold.

DFbp: This is the last method based on the boxplot of logarithmic word document frequency. The mechanism uses the quartiles of the transformed word document frequency to reduce the feature set, without setting any parameters or thresholds.

4.3. Evaluation metrics

Two evaluation metrics used are the Detection Error Tradeoff (DET) curve provided by the Defense Advanced Research Projects Agency (DARPA), and a cost function from the National Institute of Standards and Technology (NIST), respectively.

DET Curve: In the DET curve, systems error rates are plotted on two axes, the abscissa axis shows the false alarm probability while the ordinate axis shows the miss probability. False alarm probability (FAP) means the proportion of stories which are actually old but get assigned as new. Miss probability (MP) means the proportion of stories which are actually new but get annotated as old [Yang et al. 1998]. Table 2 illustrates a contingency table for a cluster-event pair, where a, b, c and d are the numbers of stories in the corresponding cases. FAP and MP are defined as follows:

- FAP = $b/(b+d)$ if $b+d>0$; otherwise, it is undefined.
- MP = $c/(a+c)$ if $a+c>0$; otherwise, it is undefined.

Each plot in the DET curve is determined by both FAP and MP under a certain threshold, with the curve closer to the origin suggesting better performance.

Cost Function: A cost functionC_{Det} from [4] is also adopted which combines the probabilities of missing a new story and a false alarm in the following way:

$$C_{Det} = C_{Miss} \cdot P_{Target} + F_{FA} \cdot FAP \cdot P_{Nontarget}, \tag{22}$$

where C_{Miss} is the cost of missing a new story, P_{Target} is the probability of seeing a new story in the data. Furthermore, F_{FA} is the cost of a false alarm, $P_{Nontarget}$ is the probability of seeing an old story and $P_{Nontarget} = q = P_{Target}$. While

false alarm (old stories which superficially look new to the system) can be noticed and inspected by the user, miss errors (new stories which the system failed to find) will normally go unnoticed because the user cannot look through the entire story collection. So miss errors are considered as a more severe problem than false alarm. In our experiment, we set $C_{Miss} = 1.0$ and $F_{FA} = 0.1$, as in [18, 6, 27, 8]. In TDT benchmark evaluations, P_{Target} is set to 0.02 for all events, and we follow the same convention here.

As in [4], the cost C_{Det} is usually normalized such that a perfect system would score 0 and a trivial system score 1, i.e.,

$$Norm(C_{Det}) = \frac{C_{Det}}{min(C_{Miss} \cdot P_{Target}, F_{FA} \cdot P_{Nontarget})}. \tag{23}$$

Naturally, the smaller the normalized detection cost, the better the quality of the FSD system.

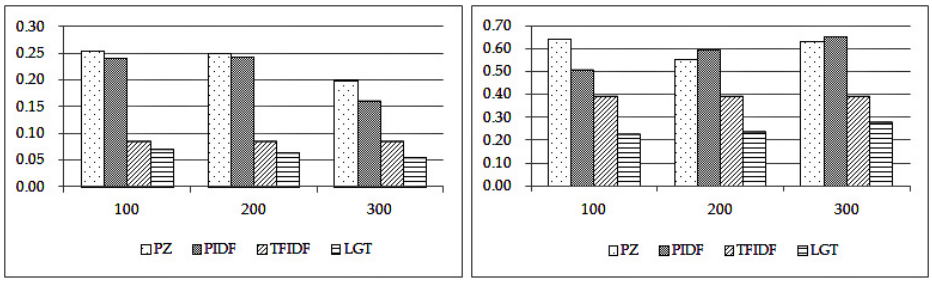
4.4. Results and analysis

We first present the results with different number of topics, and then analyze the effects of feature reduction. Finally, we give the results of online FSD on events under the same subject and on diverse events.

4.4.1. Influence of the number of topics

The number of topics indicates how many latent clusters of documents can be derived, which is a parameter of schemes exploiting topic modeling. For the English data sets, typical values of the number of topics lie in the range of 100 to 300 for topic models [10]. The minimum normalized detection cost of all schemes with different topic numbers are presented in Fig. 2. The results show that the number of topics has a different impact on diverse events and events under the same subject.

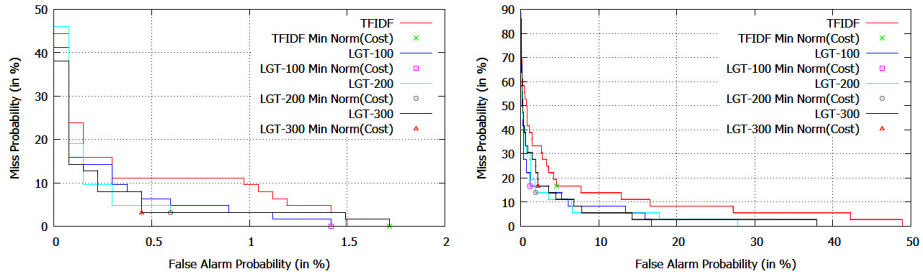
For **TDT5-diversity** (diverse events), our scheme achieves the best result for 300 topics, at a false alarm probability of 0.45% and a miss probability of 3.17%. As the number of topics increases from 100 to 300, the normalized detection cost of our scheme decreases from 0.0695 to 0.0537. Compared to *TFIDF*, the performance of our scheme improves 17.39%, 27.47% and 36.17% for 100, 200 and 300 topics, respectively. The two existing schemes exploiting LDA perform worse than *TFIDF*, which are consistent with their results on unsupervised learning under abundant experiments [1, 11]. However, their detection costs also decrease with the increase of the topic numbers. The reason is that for diverse events, the latent topics of documents are decentralized, and they are better modeled by a larger number of topics.



(a) (b)

Figure 2: The minimum normalized detection cost of FSD with different topic numbers, where (a) is on TDT5-diversity, (b) is on TDT5-centralization.

For **TDT5-centralization** (events under the same subject), our scheme achieves the best result for 100 topics, at a false alarm probability of 1.14% and a miss probability of 16.67%. Compared to *TFIDF*, the performance of our scheme improves 43.11%, 40.91% and 29.78% for 100, 200 and 300 topics, respectively. The two existing schemes exploiting LDA also perform worse than *TFIDF* and our scheme. Generally, their detection costs are lower as the decrease of topic numbers, since for events under the same subject, the latent topics of documents are centralized, and they are modeled better by less number of topics. However, the *PZ* scheme performs worst when the topic number is 100. That is because for such a topic-level scheme, the vector dimension equals to the number of topics. A too small topic number may induce the problem of under-fitting and cannot distinguish an event from another, so it will hurt the performance of the scheme.



(a) (b)

Figure 3: The DET curve of FSD with different topic numbers, where (a) is on TDT5-diversity, (b) is on TDT5-centralization.

Fig. 3 plots the DET curve of our *LGT* scheme under 100, 200 and 300 topics. As can be seen, our scheme achieves regions closer to the origin than *TFIDF*, which

Table 3: The Proportions of Features Reduced by DFr and DFc

	Data Set	Scale	$p1$	$p2$	$p3$	$p4$	$p5$
LGT+ DFr	TDT5 diversity	10^{-3}	29.40%	42.16%	55.91%	60.28%	66.10%
	TDT5 centralization	10^{-3}	50.18%	61.77%	67.93%	71.95%	74.82%
LGT+ DFc	TDT5 diversity	10^{-1}	2.42%	0.53%	0.19%	0.06%	0.02%
	TDT5 centralization	10^{-1}	2.13%	0.64%	0.26%	0.12%	0.04%

means lower error rates (i.e., FAP and MP). A possible reason of the proposed *LGT* scheme achieving the best performance in FSD is that the scheme not only models the characteristics of unique and salient features, but also explores the intrinsic relationships among latent topics to deal with the polysemous and synonymous problems.

4.4.2. Effects of feature reduction strategies

In this section, we first study the influence of *DFr* and *DFc* on the performance of FSD, since these two feature reduction strategies require to set parameters manually. Then, we evaluate the effects of *DFmd* and *DFbp*, which do not need to predetermine any parameters or thresholds. To evaluate the effects of feature reduction on FSD, we use the proposed *LGT* scheme by setting the number of topics to 100.

For parametric methods of feature reduction (i.e., *DFr* and *DFc*), we use the proportion p of all stories to determine the document frequency threshold. Specifically, we select $p = 0.1\%, 0.2\%, 0.3\%, 0.4\%, 0.5\%$ as the parameters of DFr, and $p = 10\%, 20\%, 30\%, 40\%, 50\%$ as the parameters of DFc. Thus, the threshold scale is 0.001 for *DFr* and 0.1 for *DFc*, respectively. It follows that *DFr* keeps only the features that occur in more than p of all stories, and *DFc* deletes the features that occur in more than p of all stories.

Given those parameters, the proportions of rare words reduced by *DFr* are 29.40%, 42.16%, 55.91%, 60.28%, 66.10% for TDT5-diversity, and 50.18%, 61.77%, 67.93%, 71.95%, 74.82% for TDT5-centralization (see Table 3). The proportions of words deleted approximately obey linear distribution for both data sets. The values on TDT5-diversity are less than those on TDT5-centralization, be-

cause words occur in stories of TDT5-diversity are more decentralized than those of TDT5-centralization. In other words, TDT5-diversity is more scattered than TDT5-centralization in terms of the word document frequency distribution, thus less features are reduced by the same scale of threshold. On the other hand, the proportions of common words reduced by *DFc* are 2.42%, 0.53%, 0.19%, 0.06%, 0.02% for TDT5-diversity, and 2.13%, 0.64%, 0.26%, 0.12%, 0.04% for TDT5-centralization. Fig. 4 presents the minimum normalized detection cost of FSD by *DFr* and *DFc*. The results show that both *DFr* and *DFc* are beneficial to FSD on TDT5-diversity, and *DFc* performs better than *DFr*. However, they perform negatively on TDT5-centralization, especially for *DFc*. Besides, they are sensitive to the thresholds for both data sets, meaning that the performance is unstable and it is hard to optimize the parameters.

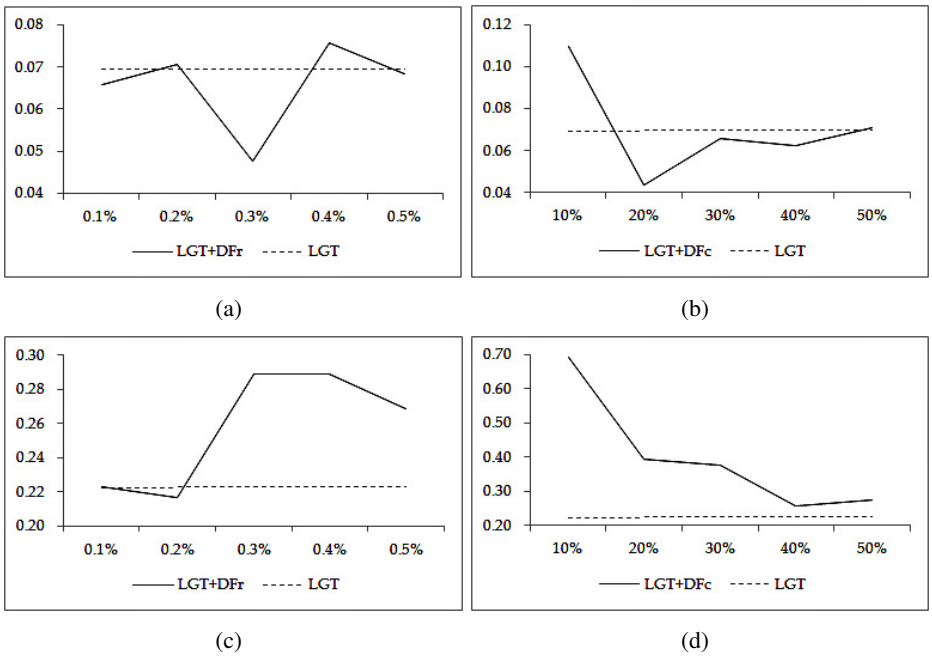


Figure 4: The minimum normalized detection cost of FSD by LGT+DFr and LGT+DFc, where (a) and (b) are on TDT5-diversity, (c) and (d) are on TDT5-centralization.

For nonparametric methods of feature reduction, the method *DFmd* is based on the assumption that the distribution of word document frequency is standard normal. The proportions of words reduced by *DFmd* are 6.89% and 5.48% on TDT5-diversity and TDT5-centralization, respectively. However, compared to the pure *LGT* scheme, the performance of *LGT+DFmd* decreases by 221.29% and 258.28%, respectively. On the other hand, the performance of *LGT+DFbp* im-

Table 4: The Proportions of Features Reduced by DFmd and DFbp and Performance of Schemes

Scheme	Data Set	Features Reduced	The minimum $Norm(C_{Det})$
LGT+DFmd	TDT5-diversity	6.89%	0.2232
	TDT5-centralization	5.48%	0.797
LGT+DFbp	TDT5-diversity	29.49%	0.0586
	TDT5-centralization	26.65%	0.2212
LGT	TDT5-diversity	NA	0.0695
	TDT5-centralization	NA	0.2225

proves by 15.67% and 0.55%, and the proportions of words reduced by *DFbp* are 29.49% and 26.65%, respectively (see Table 4). From these results we can observe that, *LGT* coupled with *DFbp* reduce almost 30% of features without setting any parameters, while achieving improved (+15.67%) or relatively stable (+0.55%) performance with respect to the pure *LGT* scheme.

The reasons for negative or non-substantially improved performance of all feature reduction methods on TDT5-centralization can be explained as follows: First, low document frequency features (i.e., rare words) are informative for FSD on events under the same subject, since these features capture the uniqueness of each story and are beneficial to discriminate similar events. Second, although the elimination of high document frequency features (i.e., topical common features) is useful to a general data set, it may be unsuitable for news events under the same subject. For TDT5-centralization, words describing these events are highly overlapping, yet they may have different frequencies and meanings in their context. As a result, the common features are also helpful to FSD on events under the same subject.

4.4.3. Results of online FSD

To evaluate the performance of online FSD, we set factors such as time window, topic numbers and feature reduction *uniformly* for all schemes. First, time window indicates the number of stories fold-in each time. We set the interval to 50 by following [30]. Second, the number of stories to be used for training batched LDA is quite small, especially in the initial step (only 50 stories are used), so we set the number of topics to 100 for all schemes exploiting LDA. Finally, since the feature reduction only changes the dimensions of word-level schemes, while the dimension of topic-level scheme (i.e., *PZ*) remains the same, we exclude feature

reduction operation so as to keep the results of all schemes comparable to online FSD.

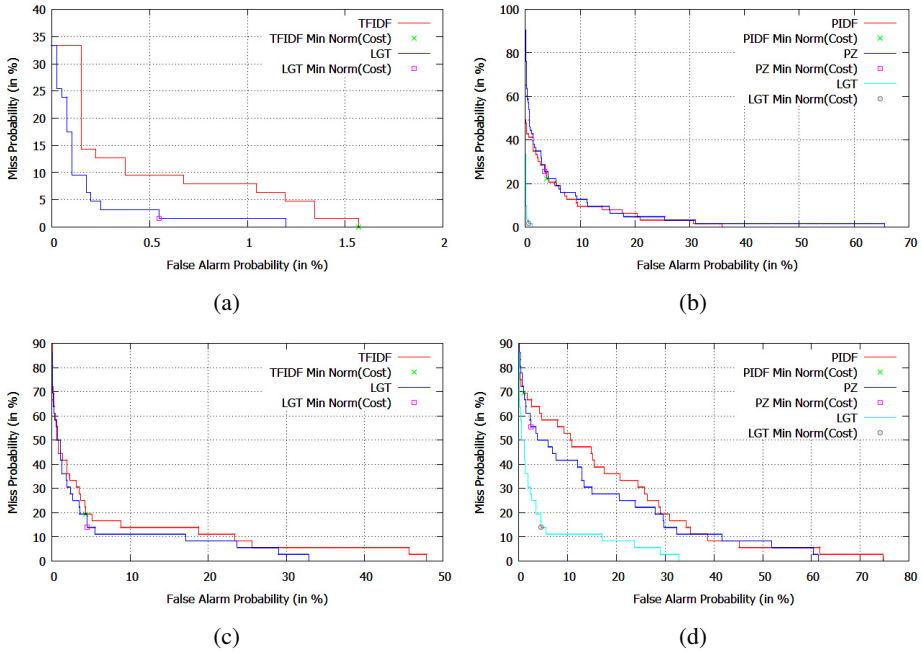


Figure 5: Comparison with other schemes of online FSD, where (a) and (b) are on TDT5-diversity, (c) and (d) are on TDT5-centralization.

Fig. 5 plots the DET curve of online FSD for all schemes. Our proposed scheme (*LGT*) achieves regions of lower error rates than *TFIDF*, as well as the two existing schemes of exploiting topic modeling (i.e., *PIDF* and *PZ*).

For TDT5-diversity (diverse events), our scheme achieves the best result at a false alarm probability of 0.55%, a miss probability of 1.59%, and the corresponding normalized detection cost is 0.0427. Compared to *TFIDF*, the performance of our scheme improves 44.35%. Compared to *PIDF* and *PZ*, the performance of our scheme improves 89.64% and 89.96%, respectively. These results confirm that it is reasonable and effective to use the local element to capture the uniqueness of each story, and the topical association to deal with synonymous and polysemous in online FSD.

For TDT5-centralization (events under the same subject), our scheme achieves the best result at a false alarm probability of 4.50%, a miss probability of 13.89%, and the corresponding normalized detection cost is 0.3596. Compared to *TFIDF*, the performance of our scheme improves 10.77%. Compared to the two exist-

ing schemes of exploiting LDA, the performance of our scheme improves 51.44% and 46.58%, respectively. However, the improvement of our scheme on TDT5-centralization is smaller than that on TDT5-diversity. This is because we uniformly set the number of topics to 100 for each time window/interval that only contains 50 new stories. But for events under the same subject, the latent topics of stories are centralized, and they may be modeled better by less number of topics for such small intervals.

5. Conclusion

First Story Detection (FSD) as one of the most fundamental tasks in Topic Detection and Tracking (TDT) is important to information, security or stock analysts. Compared to information retrieval, text clustering and classification, FSD is event-based rather than subject-based. As a result, when applied to FSD, classical term weighting schemes falls short of addressing the problems of multiple events on the same subject and evolution of events. While there have been some existing schemes for FSD which exploit named entity, temporal information, and topic modeling, they all suffer from limited accuracy. In this paper, we have proposed a new scheme called *LGT* which advocates to combine the local element, global element and topical association of features. The rationales underlying *LGT* lie in that:

1. the local element, which represents the uniqueness of each story, has a significant impact on the performance both for events under the same subject and for diverse events in FSD;
2. the topical association exploiting LDA model is powerful in modeling multiple events on the same subject (polysemous), and evolution of events (synonymous).

Overall our scheme works well in FSD and incremental clustering, as shown by experiments on the two subsets of TDT5. We also note that a nonparametric feature reduction strategy, *LGT* coupled with *DFbp*, can reduce many features while achieving good performance on diverse events, and is relatively stable with respect to multiple events under the same subject. In our subsequent study, we plan to conduct further research on other text mining tasks, in which polysemous and synonymous words are common and can affect the performance. In addition, with the increase of online social media streams (blogs, tweets, etc.), learning evolving and emerging events in social media [17, 22] becomes increasingly important and deserves our further research too.

Acknowledgement

The work described in this paper was fully supported by the National Natural Science Foundation of China (No. 61502545), the Fundamental Research Funds for the Central Universities under Project 46000-31610009, a grant from the Soft Science Research Project of Guangdong Province (No. 2014A030304013) and a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (UGC/FDS11/E06/14).

References

[1] A. Banerjee, S. Basu, Topic models over text streams: A study of batch and online unsupervised learning., in: SDM, volume 7, SIAM, pp. 437–442.

[2] D.M. Blei, J.D. Lafferty, Dynamic topic models, in: Proceedings of the 23rd international conference on Machine learning, ACM, pp. 113–120.

[3] D.M. Blei, A.Y. Ng, M.I. Jordan, Latent dirichlet allocation, the Journal of machine Learning research 3 (2003) 993–1022.

[4] T. Brants, F. Chen, A. Farahat, A system for new event detection, in: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval, ACM, pp. 330–337.

[5] D. Cai, Q. Mei, J. Han, C. Zhai, Modeling hidden topics on document manifold, in: Proceedings of the 17th ACM conference on Information and knowledge management, ACM, pp. 911–920.

[6] F. Can, S. Kocberber, O. Baglioglu, S. Kardas, H.C. Ocalan, E. Uyar, New event detection and topic tracking in turkish, Journal of the American Society for Information Science and Technology 61 (2010) 802–819.

[7] C.C. Chen, Y.T. Chen, Y. Sun, M.C. Chen, Life cycle modeling of news events using aging theory, in: Machine Learning: ECML 2003, Springer, 2003, pp. 47–59.

[8] T.C. Chou, M.C. Chen, Using incremental plsi for threshold-resilient online event analysis, IEEE Transactions on Knowledge and Data Engineering 20 (2008) 289–299.

[9] R.J. Cleary, Applied data mining: Statistical methods for business and industry, Journal of the American Statistical Association 101 (2006) 1317–1318.

[10] W. De Smet, M.F. Moens, An aspect based document representation for event clustering, in: Proceedings of the 19th meeting of computational linguistics in the Netherlands.

[11] W. De Smet, M.F. Moens, Representations for multi-document event clustering, Data Mining and Knowledge Discovery 26 (2013) 533–558.

[12] T.L. Griffiths, M. Steyvers, Finding scientific topics, Proceedings of the National Academy of Sciences 101 (2004) 5228–5235.

[13] D.J. Hand, H. Mannila, P. Smyth, Principles of data mining, MIT press, 2001.

[14] T. Hofmann, Probabilistic latent semantic indexing, in: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, ACM, pp. 50–57.

[15] S.P. Kasiviswanathan, P. Melville, A. Banerjee, V. Sindhwani, Emerging topic detection using dictionary learning, in: Proceedings of the 20th ACM international conference on Information and knowledge management, ACM, pp. 745–754.

[16] G. Kumaran, J. Allan, Text classification and named entities for new event detection, in: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval, ACM, pp. 297–304.

[17] Y. Lee, H.y. Jung, W. Song, J.H. Lee, Mining the blogosphere for top news stories identification, in: Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval, ACM, pp. 395–402.

[18] G. Luo, C. Tang, P.S. Yu, Resource-adaptive real-time new event detection, in: Proceedings of the 2007 ACM SIGMOD international conference on Management of data, ACM, pp. 497–508.

[19] G. Luo, R. Yan, P.S. Yu, Real-time new event detection for video streams, in: Proceedings of the 17th ACM conference on Information and knowledge management, ACM, pp. 379–388.

[20] R. Papka, J. Allan, On-Line New Event Detection Using Single Pass Clustering TITLE2:, Technical Report, Amherst, MA, USA, 1998.

[21] Y. Rao, Q. Li, Term weighting schemes for emerging event detection, in: Web Intelligence and Intelligent Agent Technology (WI-IAT), 2012

IEEE/WIC/ACM International Conferences on, volume 1, IEEE, pp. 105–112.

[22] A. Saha, V. Sindhwani, Learning evolving and emerging topics in social media: a dynamic nmf approach with temporal regularization, in: Proceedings of the fifth ACM international conference on Web search and data mining, ACM, pp. 693–702.

[23] M. Steyvers, T. Griffiths, Probabilistic topic models, Handbook of latent semantic analysis 427 (2007) 424–440.

[24] H. Wu, Y. Wang, X. Cheng, Incremental probabilistic latent semantic analysis for automatic question recommendation, in: Proceedings of the 2008 ACM conference on Recommender systems, ACM, pp. 99–106.

[25] Y. Yang, J.O. Pedersen, A comparative study on feature selection in text categorization, in: ICML, volume 97, pp. 412–420.

[26] Y. Yang, T. Pierce, J. Carbonell, A study of retrospective and on-line event detection, in: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, ACM, pp. 28–36.

[27] Y. Yang, J. Zhang, J. Carbonell, C. Jin, Topic-conditioned novelty detection, in: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, pp. 688–693.

[28] J. Zeng, J. Duan, W. Cao, C. Wu, Topics modeling based on selective zipf distribution, Expert Systems with Applications 39 (2012) 6541–6546.

[29] K. Zhang, J. Li, G. Wu, K. Wang, New event detection and topic tracking in turkish, A new event detection model based on term reweighting 19 (2008) 817–828.

[30] K. Zhang, J. Zi, L.G. Wu, New event detection based on indexing-tree and named entity, in: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval, ACM, pp. 215–222.

[31] X. Zhang, Z. Li, Online new event detection based on iplsa, in: Advanced Data Mining and Applications, Springer, 2009, pp. 397–408.