Bring you to the past: Automatic Generation of Topically Relevant Event Chronicles

Tao Ge[1,2]**, Wenzhe Pei**[1]**, Heng Ji**[3]**, Sujian Li**[1,2]**, Baobao Chang**[1,2]**, Zhifang Sui**[1,2]

[1]Key Laboratory of Computational Linguistics, Ministry of Education,
School of EECS, Peking University, Beijing, 100871, China
[2]Collaborative Innovation Center for Language Ability, Xuzhou, Jiangsu, 221009, China
[3]Computer Science Department, Rensselaer Polytechnic Institute, Troy, NY 12180, USA
getao@pku.edu.cn, wenzhepei@pku.edu.cn, jih@rpi.edu,
lisujian@pku.edu.cn, chbb@pku.edu.cn, szf@pku.edu.cn

Abstract

An event chronicle provides people with an easy and fast access to learn the past. In this paper, we propose the first novel approach to automatically generate a topically relevant event chronicle during a certain period given a reference chronicle during another period. Our approach consists of two core components – a time-aware hierarchical Bayesian model for event detection, and a learning-to-rank model to select the salient events to construct the final chronicle. Experimental results demonstrate our approach is promising to tackle this new problem.

1 Introduction

Human civilization has developed for thousands of years. During the long period, history witnessed the changes of societies and dynasties, the revolution of science and technology, as well as the emergency of celebrities, which are great wealth for later generations. Even nowadays, people usually look back through history either for their work or interests. Among various ways to learn history, many people prefer reading an event chronicle summarizing important events in the past, which saves much time and efforts.

The left part of Figure 1 shows a disaster event chronicle from Infoplease[1], by which people can easily learn important disaster events in 2009. Unfortunately, almost all the available event chronicles are created and edited manually, which requires editors to learn everything that happened in the past. Even if an editor tries her best to generate an event chronicle, she still cannot guarantee that all the important events are included. Moreover, when new events happen in the future, she

needs to update the chronicle in time, which is laborious. For example, the event chronicle of 2010 in Wikipedia[2] has been edited 8,488 times by 3,211 distinct editors since this page was created. In addition, event chronicles can vary according to topic preferences. Some event chronicles are mainly about disasters while others may focus more on sports. For people interested in sports, the event chronicle in Figure 1 is undesirable. Due to the diversity of event chronicles, it is common that an event chronicle regarding a specific topic for some certain period is unavailable. If editing an event chronicle can be done by computers, people can have an overview of any period according to their interests and do not have to wait for human editing, which will largely speed up knowledge acquisition and popularization.

Based on this motivation, we propose a new task of automatic event chronicle generation, whose goal is to generate a topically relevant event chronicle for some period based on a reference chronicle of another period. For example, if an disaster event chronicle during 2009 is available, we can use it to generate a disaster chronicle during 2010 from a news collection, as shown in Figure 1.

To achieve this goal, we need to know what events happened during the target period, whether these events are topically relevant to the chronicle, and whether they are important enough to be included, since an event chronicle has only a limited number of entries. To tackle these challenges, we propose an approach consisting of two core components – an event detection component based on a novel time-aware hierarchical Bayesian model and a learning-to-rank component to select the salient events to construct the final chronicle. Our event detection model can not only learn topic preferences of the reference chronicle and measure topical relevance of an event to the chronicle

---

[1]http://www.infoplease.com/world/disasters/2009.html

[2]http://en.wikipedia.org/wiki/2010

20090109: At least 20 are killed and thousands more are left homeless after a 6.2 magnitude earthquake strikes the verdant mountains of northern Costa Rica, setting off landslides.

20090112: More than 200 people are missing and feared dead when a 250-passenger ferry sinks off the coast of the Indonesian island of Sulawesi during a storm.

20090124: At least 15 people die and more than one million homes are left without power when winds of more than 100 mph swept across France and Spain during the most severe storm to hit the region since 1999.

20090126: An avalanche slams into a group of 17 Turkish hikers on Mount Zigana, dragging them more than 1,640 ft and killing 10 of them.
...

News during 2009-2010

Output: an event chronicle about disasters during 2010

20100112: The beleaguered country of Haiti is dealt a catastrophic blow when a magnitude 7.0 earthquake strikes 10 miles southwest of Port-au-Prince, the country's capital.

20100227: An 8.8 magnitude earthquake rocks Chile. Fatalities are relatively low, with some 750 people killed in the devastation.

20100404: A 7.2 earthquake, centered in Mexico but felt for miles, shakes California and kills two.

20100405: An explosion in a West Virginia coal mine kills at least 25 people and leaves 4 unaccounted for.

20100410: The President of Poland, Lech Kaczynski, and his wife are among the 96 people on board a flight from Poland to Smolensk, Russia that crashed while attempting to land in thick fog.
...

Figure 1: Example for automatic generation of a topically relevant event chronicle.

but also can effectively distinguish similar events by taking into account time information and event-specific details. Experimental results show our approach significantly outperforms baseline methods and that is promising to tackle this new problem.

The major novel contributions of this paper are:

- We propose a new task automatic generation of a topically relevant event chronicle, which is meaningful and has never been studied to the best of our knowledge.

- We design a general approach to tackle this new problem, which is language-independent, domain-independent and scalable to any arbitrary topics.

- We design a novel event detection model. It outperforms the state-of-the-art event detection model for generating topically relevant event chronicles.
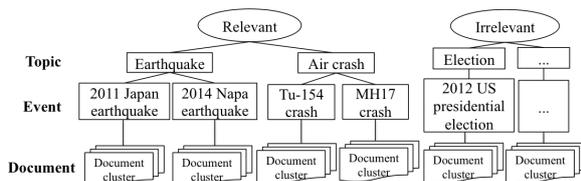
2 Terminology and Task Overview

Figure 2: An example of relevance-topic-event hierarchical structure for a disaster event chronicle.

As shown in Figure 1, *an event (entry)* in an event chronicle corresponds to a specific occurrence in the real world, whose granularity depends on the chronicle. For a sports chronicle, an event entry may be a match in 2010 World Cup, while for a comprehensive chronicle, the World Cup is regarded as one event. In general, an event can be represented by a cluster of documents related to

it. *The topic* of an event can be considered as the event class. For example, we can call the topic of MH17 crash as air crash (fine-grained) or disaster (coarse-grained). The relation between topic and event is shown through the example in Figure 2.

An event chronicle is a set of important events occurring in the past. Event chronicles vary according to topic preferences. For the disaster chronicle shown in Figure 1, earthquakes and air crashes are relevant topics while election is not. Hence, we can use a hierarchical structure to organize documents in a corpus, as Figure 2 shows.

Formally, we define an event chronicle $\mathbb{E} = \{e_1, e_2, ..., e_n\}$ where e_i is an event entry in $\mathbb{E}$ and it can be represented by a tuple $e_i = \langle D_{e_i}, t_{e_i}, z_{e_i} \rangle$. D_{e_i} denotes the set of documents about e_i, t_{e_i} is e_i's time and z_{e_i} is e_i's topic. Specially, we use Λ to denote the time period (interval) covered by $\mathbb{E}$, and θ to denote the topic distribution of $\mathbb{E}$, which reflects $\mathbb{E}$'s topic preferences.

As shown in Figure 1, the goal of our task is to generate an (target) event chronicle $\mathbb{E}_T$ during Λ_T based on a reference chronicle $\mathbb{E}_R$ during Λ_R. The topic distributions of $\mathbb{E}_T$ and $\mathbb{E}_R$ (i.e., θ_T and θ_R) should be consistent.

3 Event Detection

3.1 Challenges of Event Detection

(d_1) A magnitude-6.1 earthquake in southern China's Yunnan province destroyed thousands of home on Sunday, killing at least 367 people. (Aug 2014)

(d_2) A massive earthquake devastated Sichuan province in southwest China, leaving five million people homeless and taking some 87,000 lives. (May 2008)

(d_3) A 3.4-magnitude earthquake occurred in and around the city of Napa, California on Sunday, killing one person and injuring about 200. (Aug 2014)

Figure 3: Documents that are lexically similar but refer to different events. The underlined words are event-specific words.

For our task, the first step is to detect topically relevant events from a corpus. A good event detection model should be able to

(1) measure the topical relevance of a detected event to the reference chronicle.

(2) consider document time information.

(3) look into a document's event-specific details.

The first requirement is to identify topically relevant events since we want to generate a topically relevant chronicle. The second and third requirements are for effectively distinguishing events, especially similar events like the example in Figure 3. To distinguish the similar events, we must consider document time information (for distinguishing events in d_1 and d_2) and look into the document's event-specific details (the underlined words in Figure 3) (for distinguishing events in d_1 and d_3).

3.2 TaHBM: A Time-aware Hierarchical Bayesian Model

To tackle all the above challenges mentioned in Section 3.1, which cannot be tackled by conventional detection methods (e.g., agglomerative clustering), we propose a **T**ime-**a**ware **H**ierarchical **B**ayesian **M**odel (TaHBM) for detecting events.
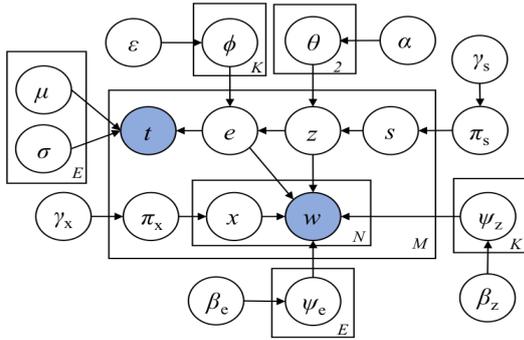
Model Overview

Figure 4: The plate diagram of TaHBM. The shaded nodes are observable nodes.

The plate diagram and generative story of TaHBM are depicted in Figure 4 and Figure 5 respectively. For a corpus with M documents, TaHBM assumes each document has three labels – s, z, and e. s is a binary variable indicating a document's topical relevance to the reference event chronicle, whose distribution is a Bernoulli distribution π_s drawn from a Beta distribution with

---

Draw $\pi_s \sim Beta(\gamma_s)$
For each $s \in \{0, 1\}$: draw $\theta^{(s)} \sim Dir(\alpha)$
For each $z = 1, 2, 3, ..., K$: draw $\phi^{(z)} \sim Dir(\varepsilon), \psi_z^{(z)} \sim Dir(\beta_z)$
For each $e = 1, 2, 3, ..., E$: draw $\psi_e^{(e)} \sim Dir(\beta_e)$
For each document $m = 1, 2, 3, ..., M$:
    Draw $s \sim Bernoulli(\pi_s)$
    Draw $z \sim Multi(\theta^{(s)})$
    Draw $e \sim Multi(\phi^{(z)})$
    Draw $t' \sim Gaussian(\mu_e, \sigma_e), t \leftarrow \lfloor t' \rfloor$
    Draw $\pi_x \sim Beta(\gamma_x)$
    For each word w in document m:
        Draw $x \sim Bernoulli(\pi_x)$
        If $x = 0$: draw $w \sim \psi_z^{(z)}$
        Else: draw $w \sim \psi_e^{(e)}$

Figure 5: The generative story of TaHBM

symmetric hyperparameter γ_s. s=1 indicates the document is topically relevant to the chronicle while s=0 means not. z is a document's topic label drawn from a K-dimensional multinomial distribution θ, and e is a document's event label drawn from an E-dimensional multinomial distribution ϕ. θ and ϕ are drawn from Dirichlet distributions with symmetric hyperparameter α and ε respectively. For an event e', it can be represented by a set of documents whose event label is e'.

In TaHBM, the relations among s, z and e are similar to the hierarchical structure in Figure 2. Based on the dependencies among s, z and e, we can compute the topical relevance of an event to the reference chronicle by Eq (1) where $P(e|z)$, $P(e)$, $P(s)$ and $P(z|s)$ can be estimated using Bayesian inference (some details of estimation of $P(s)$ and $P(s|z)$ will be discussed in Section 3.3) and thus we solve the first challenge in Section 3.1 (i.e., topical relevance measure problem).

$$P(s|e) = \frac{P(s) \times P(z|s) \times P(e|z)}{P(e)} \quad (1)$$

Now, we introduce how to tackle the second challenge – how to take into account a document's time information for distinguishing events. In TaHBM, we introduce t, document timestamps. We assume $t = \lfloor t' \rfloor$ where t' is drawn from a Gaussian distribution with mean μ and variance σ^2. Each event e corresponds to a specific Gaussian distribution which serves as a temporal con-

straint for e. A Gaussian distribution has only one peak around where the probability is concentrated. Its value trends to zero if a point lies far away from the mean. For this reason, a Gaussian distribution is suitable to describe an event's temporal distribution whose probability usually concentrates around the event's burst time and it will be close to zero if time lies far from the burst time. Figure 6 shows the temporal distribution of *the July 2009 Urumqi riots*[3]. The probability of this event concentrates around the 7th day. If we use a Gaussian distribution (the dashed curve in Figure 6) to constrain this event's time scope, the documents whose timestamps are beyond this scope are unlikely to be grouped into this event's cluster.

Now that the problems of topical relevance measure and temporal constraints have been solved, we discuss how to identify event-specific details of a document for distinguishing events. By analyzing the documents shown in Figure 3, we find that general words (e.g., *earthquake, kill, injury, devastate*) indicate the document's topic while words about event-specific details (e.g., *Napa, California, 3.4-magnitude*) are helpful to determine what events the document talks about. Assuming a person is asked to analyze what event a document discusses, it would be a natural way to first determine topic of the document based its general words, and then determine what event it talks about given its topic, timestamp and event-specific details, which is exactly the way our TaHBM works.

For simplicity, we call the general words as **topic words** and call the words describing event-specific information as **event words**. Inspired by the idea of Chemudugunta et al. (2007), given the different roles these two kinds of words play, we assume words in a document are generated by two distributions: topic words are generated by a topic word distribution ψ_z while event words are generated by an event word distribution ψ_e. ψ_z and ψ_e are $|V|$-dimensional multinomial distributions drawn from Dirichlet distributions with symmetric hyperparameter β_z and β_e respectively, where $|V|$ denotes the size of vocabulary V. A binary indicator x, which is generated by a Bernoulli distribution π_x drawn from a Beta distribution with symmetric hyperparameter γ_x, determines whether a word is generated by ψ_z or ψ_e. Specifically, if $x = 0$, a word is drawn from ψ_z; otherwise the

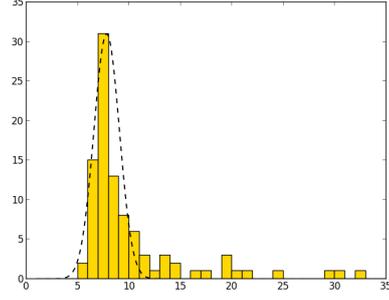[3]http://en.wikipedia.org/wiki/July_2009_Urumqi_riots

Figure 6: The temporal distribution of documents about the Urumqi riots, which can be described by a Gaussian distribution (the dashed curve). The horizontal axis is time (day) and the vertical axis is the number of documents about this event.

word is drawn from ψ_e. Since ψ_z is shared by all events of one topic, it can be seen as a background word distribution which captures general aspects. In contrast, ψ_e tends to describe the event-specific aspects. In this way, we can model a document's general and specific aspects and use the information to better distinguish similar events[4].

Model Inference

Like most Bayesian models, we use collapsed Gibbs sampling for model inference in TaHBM. For a document m, we present the conditional probability of its latent variables s, z and x for sampling:

$$P(s_m|\vec{s}_{\neg m}, \vec{z}, \gamma_s, \alpha) = \frac{c_s + \gamma_s}{\sum_s(c_s + \gamma_s)} \times \frac{c_{s,z_m} + \alpha}{\sum_z(c_{s,z} + \alpha)} \quad (2)$$

$$P(z_m|\vec{z}_{\neg m}, \vec{e}, \vec{s}, \vec{w}_m, \vec{x}_m, \alpha, \varepsilon, \beta_z)$$
$$= \frac{c_{s_m,z} + \alpha}{\sum_z(c_{s_m,z} + \alpha)} \times \frac{c_{z,e_m} + \varepsilon}{\sum_e(c_{z,e} + \varepsilon)}$$
$$\times \prod_{n=1}^{N_m} \left(\frac{c_{z,w_{m,n}} + \sum_{i=1}^{n-1} \mathbf{1}(w_{m,i} = w_{m,n}) + \beta_z}{\sum_{w \in V}(c_{z,w} + \beta_z) + n - 1} \right)^{(1-x_{m,n})}$$
$$(3)$$

$$P(x_{m,n}|\vec{w}_m, \vec{x}_{\neg m,n}, z_m, e_m, \gamma_x)$$
$$= \frac{c_{m,x} + \gamma_x}{N_m + 2\gamma_x} \times \left(\frac{c_{z_m,w_{m,n}} + \beta_z}{\sum_{w \in V}(c_{z_m,w} + \beta_z)} \right)^{(1-x)}$$
$$\times \left(\frac{c_{e_m,w_{m,n}} + \beta_e}{\sum_{w \in V}(c_{e_m,w} + \beta_e)} \right)^{x} \quad (4)$$

where V denotes the vocabulary, $w_{m,n}$ is the nth word in a document m, c_s is the count of documents with topic relevance label s, $c_{s,z}$ is the count

[4]TaHBM is language-independent, which can identify event words without name tagging. But if name tagging results are available, we can also exploit them (e.g., we can fix x of a named entity specific to an event to 1 during inference.).

of documents with topic relevance label s and topic label z, $c_{z,w}$ is the count of word w whose document's topic label is z, $c_{m,x}$ is the count of words with binary indicator label x in m and $\mathbf{1}(\cdot)$ is an indicator function.

Specially, for variable e which is dependent on the Gaussian distribution, its conditional probability for sampling is computed as Eq (5):

$$
\begin{aligned}
&P(e_m | \vec{e}_{\neg m}, \vec{z}, \vec{w}_m, \vec{x}_m, t_m, \varepsilon, \beta_e, \mu_e, \sigma_e) \\
&= \frac{c_{z_m,e} + \varepsilon}{\sum_e (c_{z_m,e} + \varepsilon)} \times \int_{t_m}^{t_m+1} p_G(t_m; \mu_e, \sigma_e') \\
&\times \prod_{n=1}^{N_m} \left(\frac{c_{e,w_{m,n}} + \sum_{i=1}^{n-1} \mathbf{1}(w_{m,i} = w_{m,n}) + \beta_e}{\sum_{w \in V}(c_{e,w} + \beta_e) + n - 1} \right)^{x_{m,n}}
\end{aligned}
\tag{5}
$$

where $p_G(x; \mu, \sigma)$ is a Gaussian probability mass function with parameter μ and σ.

The function $p_G(\cdot)$ can be seen as the temporal distribution of an event, as discussed before. In this sense, the temporal distribution of the whole corpus can be considered as a mixture of Gaussian distributions of events. As a natural way to estimate parameters of mixture of Gaussians, we use EM algorithm (Bilmes, 1998). In fact, Eq (5) can be seen as the E-step. The M-step of EM updates μ and σ as follows:

$$
\mu_e = \frac{\sum_{d \in D_e} t_d}{|D_e|}, \sigma_e = \sqrt{\frac{\sum_{d \in D_e}(t_d - \mu_e)^2}{|D_e|}}
\tag{6}
$$

where t_d is document d's timestamp and D_e is the set of documents with event label e.

Specially, for sampling e we use σ_e' defined as $\sigma_e' = \sigma_e + \tau$ (τ is a small number for smoothing[5]) because when σ is very small (e.g., $\sigma = 0$), an event's temporal scope will be strictly constrained. Using σ_e' can help the model overcome this "trap" for better parameter estimation.

Above all, the model inference and parameter estimation procedure can be summarized by algorithm 1.

3.3 Learn Topic Preferences of the Event Chronicle

A prerequisite to use Eq (1) to compute an event's topical relevance to an event chronicle is that we know $P(s)$ and $P(z|s)$ which reflects topic preferences of the event chronicle. Nonetheless, $P(s)$ and $P(z|s)$ vary according to different event chronicles. Hence, we cannot directly estimate

---

Algorithm 1 Model inference for TaHBM
1: Initialize parameters in TaHBM;
2: **for** each $iteration$ **do**
3:      **for** each document d in the corpus **do**
4:          sample s according to Eq (2)
5:          sample z according to Eq (3)
6:          sample e according to Eq (5)
7:          **for** each word w in d **do**
8:              sample x according to Eq (4)
9:          **end for**
10:      **end for**
11:      **for** each event e **do**
12:          update μ_e, σ_e according to Eq (6)
13:      **end for**
14: **end for**

---

them in an unsupervised manner; instead, we provide TaHBM some "supervision". As we mentioned in section 3.2, the variable s indicates a document's topical relevance to the event chronicle. For some documents, s label can be easily derived with high accuracy so that we can exploit the information to learn the topic preferences.

To obtain the labeled data, we use the description of each event entry in the reference chronicle $\mathbb{E}_R$ during period Λ_R as a query to retrieve relevant documents in the corpus using Lucene (Jakarta, 2004) which is an information retrieval software library. We define $\mathscr{R}$ as the set of documents in hits of any event entry in the reference chronicle returned by Lucene:

$$
\mathscr{R} = \cup_{e \in \mathbb{E}_R} Hit(e)
$$

where $Hit(e)$ is the complete hit list of event e returned by Lucene. For document d with timestamp t_d, if $d \notin \mathscr{R}$ and $t_d \in \Lambda_R$, then d is considered irrelevant to the event chronicle and thus it would be labeled as a negative example.

To generate positive examples, we use a strict criterion since we cannot guarantee that all the documents in $\mathscr{R}$ are actually relevant. To precisely generate positive examples, a document d is labeled as positive only if it satisfies the **positive condition** which is defined as follows:

$$
\exists_{e \in \mathbb{E}_R} 0 \le t_d - t_e \le 10 \wedge sim(d, e) \ge 0.4
$$

where t_e is time[6] of event e, provided by the reference chronicle. $sim(d, e)$ is Lucene's score of d given query e. According to the positive condition, a positive document example must be lexi-

---

cally similar to some event in the reference chronicle and its timestamp is close to the event's time.

As a result, we can use the labeled data to learn topic preferences of the event chronicle. For the labeled documents, s is fixed during model inference. In contrast, for documents that are not labeled, s is sampled by Eq (2). In this manner, TaHBM can learn topic preferences (i.e., $P(z|s)$) without any manually labeled data and thus can measure the topical relevance between an event and the reference chronicle.

4 Event Ranking

Generating an event chronicle is beyond event detection because we cannot use all detected events to generate the chronicle with a limited number of entries. We propose to use learning-to-rank techniques to select the most salient events to generate the final chronicle since we believe the reference event chronicle can teach us the principles of selecting salient events. Specifically, we use SVM-Rank (Joachims, 2006).

4.1 Training and Test Set Generation

The event detection component returns many document clusters, each of which represents an event. As Section 3.2 shows, each event has a Gaussian distribution whose mean indicates its burst time in TaHBM. We use the events whose burst time is during the reference chronicle's period as training examples and treat those during the target chronicle's period as test examples. Formally, the training set and test set are defined as follows:

$$Train = \{e|\mu_e \in \Lambda_R\}, Test = \{e|\mu_e \in \Lambda_T\}$$

In the training set, events containing at least one positive document (i.e. relevant to the event chronicle) in Section 3.3 are labeled as high rank priority while those without positive documents are labeled as low priority.

4.2 Features

We use the following features to train the ranking model, all of which can be provided by TaHBM.

- $P(s = 1|e)$: the probability that an event e is topically relevant to the reference chronicle.

- $P(e|z)$: the probability reflects an event's impact given its topic.

- σ_e: the parameter of an event e's Gaussian distribution. It determines the 'bandwidth'

of the Gaussian distribution and thus can be considered as the time span of e.

- $|D_e|$: the number of documents related to event e, reflecting the impact of e.

- $\frac{|D_e|}{\sigma_e}$: For an event with a long time span (e.g., Premier League), the number of relevant documents is large but its impact may not be profound. Hence, we use $\frac{|D_e|}{\sigma_e}$ to normalize $|D_e|$, which may better reflect the impact of e.

5 Experiments

5.1 Experiment Setting

Data: We use various event chronicles during 2009 as references to generate their counterparts during 2010. Specifically, we collected disaster, sports, war, politics and comprehensive chronicles during 2009 from mapreport[7], infoplease and Wikipedia[8]. To generate chronicles during 2010, we use 2009-2010 APW and Xinhua news in English Gigaword (Graff et al., 2003) and remove documents whose titles and first paragraphs do not include any burst words. We detect burst words using Kleinberg algorithm (Kleinberg, 2003), which is a 2-state finite automaton model and widely used to detect bursts. In total, there are 140,557 documents in the corpus.

Preprocessing: We remove stopwords and use Stanford CoreNLP (Manning et al., 2014) to do lemmatization.

Parameter setting: For TaHBM, we empirically set $\alpha = 0.05$, $\beta_z = 0.005$, $\beta_e = 0.0001$, $\gamma_s = 0.05$, $\gamma_x = 0.5$, $\varepsilon = 0.01$, the number of topics $K = 50$, and the number of events $E = 5000$. We run Gibbs sampler for 2000 iterations with burn-in period of 500 for inference. For event ranking, we set regularization parameter of SVMRank $c = 0.1$.

Chronicle display: We use a heuristic way to generate the description of each event. Since the first paragraph of a news article is usually a good summary of the article and the earliest document in a cluster usually explicitly describes the event, for an event represented by a document cluster, we choose the first paragraph of the earliest document written in 2010 in the cluster to generate the event's description. The earliest document's timestamp is considered as the event's time.

---

[7]http://www.mapreport.com
[8]http://en.wikipedia.org/wiki/2009

5.2 Evaluation Methods and Baselines

Since there is no existing evaluation metric for the new task, we design a method for evaluation.

Although there are manually edited event chronicles on the web, which may serve as references for evaluation, they are often incomplete. For example, the 2010 politics event chronicle on Wikipedia has only two event entries. Hence, we first pool all event entries of existing chronicles on the web and chronicles generated by approaches evaluated in this paper and then have 3 human assessors judge each event entry for generating a ground truth based on its topical relevance, impact and description according to the standard of the reference chronicles. An event entry will be included in the ground-truth only if it is selected as a candidate by at least two human judges. On average, the existing event chronicles on the web cover 50.3% of event entries in the ground-truth.

Given the ground truth, we can use *Precision@k* to evaluate an event chronicle's quality.

$$Precision@k = |\mathbb{E}_G \cap \mathbb{E}_{topk}|/k$$

where $\mathbb{E}_G$ and $\mathbb{E}_{topk}$ are ground-truth chronicle and the chronicle with top k entries generated by an approach respectively. If there are multiple event entries corresponding to one event in the ground-truth, only one is counted.

For comparison, we choose several baseline approaches. Note that event detection models except TaHBM do not provide features used in learning-to-rank model. For these detection models, we use a criterion that considers both relevance and importance to rank events:

$$rankscore_{basic}(e) = \sum_{d \in D_e} max_{e' \in \mathbb{E}_R} sim(d, e')$$

where $\mathbb{E}_R$ is the reference chronicle and $sim(d, e')$ is Lucene's score of document d given query e'. We call this ranking criterion as **basic criterion**.

- Random: We randomly select k documents to generate the chronicle.

- NB+basic: Since TaHBM is essentially an extension of NB, we use Naive Bayes (NB) to detect events and basic ranking criterion to rank events.

- B-HAC+basic: We use hierarchical agglomerative clustering (HAC) based on BurstVSM

schema (Zhao et al., 2012) to detect events, which is the state-of-the-art event detection method for general domains.

- TaHBM+basic: we use this baseline to verify the effectiveness of learning-to-rank.

As TaHBM, the number of clusters in NB is set to 5000 for comparison. For B-HAC, we adopt the same setting with (Zhao et al., 2012).

5.3 Experiment Results

Using the evaluation method introduced above, we can conduct a quantitative evaluation for event chronicle generation approaches[9].

Table 1 shows the overall performance. Our approach outperforms the baselines for all chronicles. TaHBM beats other detection models for chronicle generation owing to its ability of incorporating the temporal information and identification of event-specific details of a document. Moreover, learning-to-ranking is proven more effective to rank events than the basic ranking criterion.

Among these 5 chronicles, almost all approaches perform best on disaster event chronicle while worst on sports event chronicle. We analyzed the results and found that many event entries in the sports event chronicle are about the opening match, or the first-round match of a tournament due to the display method described in Section 5.1. According to the reference sport event chronicle, however, only matches after quarterfinals in a tournament are qualified to be event entries. In other words, a sports chronicle should provide information about the results of semi-final and final, and the champion of the tournament instead of the first-round match's result, which accounts for the poor performance. In contrast, the earliest document about a disaster event always directly describes the disaster event while the following reports usually concern responses to the event such as humanitarian aids and condolence from the world leaders. The patterns of reporting war events are similar to those of disasters, thus the quality of war chronicle is also good. Politics is somewhat complex because some political events (e.g., election) are arranged in advance while others (e.g., government shutdown) are unexpected. It is notable that for generating comprehensive event chronicles, learning-to-rank does

---

[9]Due to the space limitation, we display chronicles generated by our approach in the supplementary notes.

	sports		politics		disaster		war		comprehensive	
	P@50	P@100	P@50	P@100	P@50	P@100	P@50	P@100	P@50	P@100
Random	0.02	0.08	0	0	0.02	0.04	0	0	0.02	0.03
NB+basic	0.08	0.12	0.18	0.19	0.42	0.36	0.18	0.17	0.38	0.31
B-HAC+basic	0.10	0.13	0.30	0.26	0.50	0.47	0.30	0.22	0.36	0.32
TaHBM+basic	0.18	**0.15**	0.30	0.29	0.50	0.43	0.46	0.36	0.38	**0.33**
Our approach	**0.20**	**0.15**	**0.38**	**0.36**	**0.64**	**0.53**	**0.54**	**0.41**	**0.40**	**0.33**

Table 1: Performance of event chronicle generation.

	Topically Irrelevant	Trivial Events	Indirect Description	Redundant Entries
disaster	31.91%	17.02%	44.68%	6.38%
sports	38.82%	55.29%	3.52%	2.35%
comp	-	67.16%	31.34%	1.49%

Table 2: Proportion of errors in disaster, sports and comprehensive event chronicles.

not show significant improvement. A possible reason is that a comprehensive event chronicle does not care the topical relevance of a event. In other words, its ranking problem is simpler so that the learning-to-rank does not improve the basic ranking criterion much.

Moreover, we analyze the incorrect entries in the chronicles generated by our approaches. In general, there are four types of errors.
Topically irrelevant: the topic of an event entry is irrelevant to the event chronicle.
Minor events: the event is not important enough to be included. For example, *"20100828: Lebanon beat Canada 81-71 in the opening round of the basketball world championships"* is a minor event in the sports chronicle because it is about an opening-round match and not important enough.
Indirect description: the entry does not describe a major event directly. For instance, *"20100114: Turkey expressed sorrow over the Haiti earthquake"* is an incorrect entry in the disaster chronicle though it mentions the Haiti earthquake.
Redundant entries: multiple event entries describe the same event.

We analyze the errors of the disaster, sports and comprehensive event chronicle since they are representative, as shown in Table 2.

Topical irrelevance is a major error source for both disaster and sports event chronicles. This problem mainly arises from incorrect identification of topically relevant events during detection. Moreover, disaster and sports chronicles have their own more serious problems. Disaster event chronicles suffer from the indirect description problem since there are many responses (e.g., humanitarian aids) to a disaster. These responses are topically relevant and contain many documents, and

thus appear in the top list. One possible solution might be to increase the event granularity by adjusting parameters of the detection model so that the documents describing a major event and those discussing in response to this event can be grouped into one cluster (i.e., one event). In contrast, the sports event chronicle's biggest problem is on minor events, as mentioned before. Like the sports chronicle, the comprehensive event chronicle also has many minor event entries but its main problem results from its strict criterion. Since comprehensive chronicles can include events of any topic, only extremely important events can be included. For example, *"Netherlands beat Uruguay to reach final in the World Cup 2010"* may be a correct event entry in sports chronicles but it is not a good entry in comprehensive chronicles. Compared with comprehensive event chronicles, events in other chronicles tend to describe more details. For example, a sports chronicle may regard each match in the World Cup as an event while comprehensive chronicles consider the World Cup as one event, which requires us to adapt event granularity for different chronicles.

Also, we evaluate the time of event entries in these five event chronicles because event's happening time is not always equal to the timestamp of the document creation time (UzZaman et al., 2012; Ge et al., 2013). We collect existing manually edited 2010 chronicles on the web and use their event time as gold standard. We define a metric to evaluate if the event entry's time in our chronicle is accurate:

$$diff = \sum_{e \in \mathbb{E} \cap \mathbb{E}^*} |(t_e - t_e^*)| / |\mathbb{E} \cap \mathbb{E}^*|$$

where $\mathbb{E}$ and $\mathbb{E}^*$ are our chronicle and the manually edited event chronicle respectively. t_e is e's

time labeled by our method and t_e^* is e's correct time. Note that for multiple entries referring the same event in event chronicles, the earliest entry's time is used as the event's time to compute *diff*.

sports	politics	disaster	war	comprehensive
0.800	3.363	1.042	1.610	2.467

Table 3: Difference between an event's actual time and the time in our chronicles. Time unit is a day.

Table 3 shows the performance of our approach in labeling event time. For disaster, sports and war, the accuracy is desirable since important events about these topics are usually reported in time. In contrast, the accuracy of political event time is the lowest. The reason is that some political events may be confidential and thus they are not reported as soon as they happen; on the other hand, some political events (e.g., a summit) are reported several days before the events happen. The comprehensive event chronicle includes many political events, which results in a lower accuracy.

6 Related Work

To the best of our knowledge, there was no previous end-to-end topically relevant event chronicle generation work but there are some related tasks.

Event detection, sometimes called topic detection (Allan, 2002), is an important part of our approach. Yang et al. (1998) used clustering techniques for event detection on news. He et al. (2007) and Zhao et al. (2012) designed burst feature representations for detecting bursty events. Compared with our TaHBM, these methods lack the ability of distinguishing similar events.

Similar to event detection, event extraction focuses on finding events from documents. Most work regarding event extraction (Grishman et al., 2005; Ahn, 2006; Ji and Grishman, 2008; Chen and Ji, 2009; Liao and Grishman, 2010; Hong et al., 2011; Li et al., 2012; Chen and Ng, 2012; Li et al., 2013) was developed under Automatic Content Extraction (ACE) program. The task only defines 33 event types and events are in much finer grain than those in our task. Moreover, there was work (Verhagen et al., 2005; Chambers and Jurafsky, 2008; Bethard, 2013; Chambers, 2013; Chambers et al., 2014) about temporal event extraction and tracking. Like ACE, the granularity of events in this task is too fine to be suitable for our task.

Also, timeline generation is related to our work. Most previous work focused on generating a time-line for a document (Do et al., 2012), a centroid entity (Ji et al., 2009) or one major event (Hu et al., 2011; Yan et al., 2011; Lin et al., 2012; Li and Li, 2013). In addition, Li and Cardie (2014) generated timelines for users in microblogs. The most related work to ours is Swan and Allan (2000). They used a timeline to show bursty events along the time, which can be seen as an early form of event chronicles. Different from their work, we generate a topically relevant event chronicle based on a reference event chronicle.

7 Conclusions and Future Work

In this paper, we propose a novel task – automatic generation of topically relevant event chronicles. It can serve as a new framework to combine the merits of Information Retrieval, Information Extraction and Summarization techniques, to rapidly extract and rank salient events. This framework is also able to rapidly and accurately capture a user's interest and needs based on the reference chronicle (instead of keywords as in Information Retrieval or event templates as in Guided Summarization) which can reflect diverse levels of granularity.

As a preliminary study of this new challenge, this paper focuses on event detection and ranking. There are still many challenges for generating high-quality event chronicles. In the future, we plan to investigate automatically adapting an event's granularity and learn the principle of summarizing the event according to the reference event chronicle. Moreover, we plan to study the generation of entity-driven event chronicles, leveraging more fine-grained entity and event extraction approaches.

References

David Ahn. 2006. The stages of event extraction. In *Workshop on Annotating and Reasoning about Time and Events*.

James Allan. 2002. *Topic detection and tracking: event-based information organization*, volume 12. Springer Science & Business Media.

Steven Bethard. 2013. Cleartk-timeml: A minimalist approach to tempeval 2013. In *Second Joint Conference on Lexical and Computational Semantics*.

Jeff A Bilmes. 1998. A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. *International Computer Science Institute*, 4(510):126.

Nathanael Chambers and Dan Jurafsky. 2008. Jointly combining implicit constraints improves temporal ordering. In *EMNLP*.

Nathanael Chambers, Taylor Cassidy, Bill McDowell, and Steven Bethard. 2014. Dense event ordering with a multi-pass architecture. *TACL*, 2:273–284.

Nathanael Chambers. 2013. Navytime: Event and time ordering from raw text. Technical report, DTIC Document.

Chaitanya Chemudugunta and Padhraic Smyth Mark Steyvers. 2007. Modeling general and specific aspects of documents with a probabilistic topic model. In *NIPS*.

Zheng Chen and Heng Ji. 2009. Language specific issue and feature exploration in chinese event extraction. In *NAACL*.

Chen Chen and Vincent Ng. 2012. Joint modeling for chinese event extraction with rich linguistic features. In *COLING*.

Quang Xuan Do, Wei Lu, and Dan Roth. 2012. Joint inference for event timeline construction. In *EMNLP*.

Tao Ge, Baobao Chang, Sujian Li, and Zhifang Sui. 2013. Event-based time label propagation for automatic dating of news articles. In *EMNLP*.

David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. 2003. English gigaword. *Linguistic Data Consortium, Philadelphia*.

Ralph Grishman, David Westbrook, and Adam Meyers. 2005. Nyu's english ace 2005 system description. In *ACE 2005 Evaluation Workshop*.

Qi He, Kuiyu Chang, and Ee-Peng Lim. 2007. Using burstiness to improve clustering of topics in news streams. In *ICDM*.

Yu Hong, Jianfeng Zhang, Bin Ma, Jian-Min Yao, Guodong Zhou, and Qiaoming Zhu. 2011. Using cross-entity inference to improve event extraction. In *ACL*.

Po Hu, Minlie Huang, Peng Xu, Weichang Li, Adam K Usadi, and Xiaoyan Zhu. 2011. Generating breakpoint-based timeline overview for news topic retrospection. In *ICDM*.

Apache Jakarta. 2004. Apache lucene-a high-performance, full-featured text search engine library.

Heng Ji and Ralph Grishman. 2008. Refining event extraction through cross-document inference. In *ACL*.

Heng Ji, Ralph Grishman, Zheng Chen, and Prashant Gupta. 2009. Cross-document event extraction and tracking: Task, evaluation, techniques and challenges. In *RANLP*.

Thorsten Joachims. 2006. Training linear svms in linear time. In *SIGKDD*.

Jon Kleinberg. 2003. Bursty and hierarchical structure in streams. *Data Mining and Knowledge Discovery*, 7(4):373–397.

Jiwei Li and Claire Cardie. 2014. Timeline generation: Tracking individuals on twitter. In *WWW*.

Jiwei Li and Sujian Li. 2013. Evolutionary hierarchical dirichlet process for timeline summarization. In *ACL*.

Peifeng Li, Guodong Zhou, Qiaoming Zhu, and Libin Hou. 2012. Employing compositional semantics and discourse consistency in chinese event extraction. In *EMNLP*.

Qi Li, Heng Ji, and Liang Huang. 2013. Joint event extraction via structured prediction with global features. In *ACL*.

Shasha Liao and Ralph Grishman. 2010. Using document level cross-event inference to improve event extraction. In *ACL*.

Chen Lin, Chun Lin, Jingxuan Li, Dingding Wang, Yang Chen, and Tao Li. 2012. Generating event storylines from microblogs. In *CIKM*.

Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *ACL System Demonstrations*.

Russell Swan and James Allan. 2000. Automatic generation of overview timelines. In *SIGIR*.

Naushad UzZaman, Hector Llorens, James Allen, Leon Derczynski, Marc Verhagen, and James Pustejovsky. 2012. Tempeval-3: Evaluating events, time expressions, and temporal relations. *arXiv preprint arXiv:1206.5333*.

Marc Verhagen, Inderjeet Mani, Roser Sauri, Robert Knippen, Seok Bae Jang, Jessica Littman, Anna Rumshisky, John Phillips, and James Pustejovsky. 2005. Automating temporal annotation with tarsqi. In *ACL demo*.

Rui Yan, Xiaojun Wan, Jahna Otterbacher, Liang Kong, Xiaoming Li, and Yan Zhang. 2011. Evolutionary timeline summarization: a balanced optimization framework via iterative substitution. In *SIGIR*.

Yiming Yang, Tom Pierce, and Jaime Carbonell. 1998. A study of retrospective and on-line event detection. In *SIGIR*.

Wayne Xin Zhao, Rishan Chen, Kai Fan, Hongfei Yan, and Xiaoming Li. 2012. A novel burst-based text representation model for scalable event detection. In *ACL*.