EventRiver: Visually Exploring Text Collections With Temporal References

Dongning Luo, Jing Yang, Milos Krstajic, William Ribarsky, and Daniel Keim

Abstract—Many text collections with temporal references, such as news corpora and weblogs, are generated to report and discuss real life events. Thus, *event-related tasks*, such as detecting real life events that drive the generation of the text documents, tracking event evolutions, and investigating reports and commentaries about events of interest, are important when exploring such text collections. To incorporate and leverage human efforts in conducting such tasks, we propose a novel visual analytics approach named *EventRiver*. EventRiver integrates event-based automated text analysis and visualization to reveal the events motivating the text generation and the long term stories they construct. On the visualization, users can interactively conduct tasks such as event browsing, tracking, association, and investigation. A working prototype of EventRiver has been implemented for exploring news corpora. A set of case studies, experiments, and a preliminary user test have been conducted to evaluate its effectiveness and efficiency.

Index Terms—Visual Analytics, Information Visualization, Topic, Event, Text, Clustering.

1 Introduction

Text collections with temporal references, such as news corpora, weblogs, and email archives, consist of text documents with time stamps that are critical to the understanding and analysis of the text collection. They are important information sources in a wide variety of applications, including social and cultural studies, government intelligence, and business decision making. Text collections with temporal references are often generated to report and discuss real life events, which happen at specific times and draw continuous attention [3]. To understand them, it is necessary *to detect the real life events motivating the text generation, to learn their semantics and temporal context, and to track their evolution over time* [16]. It is also important for users to *find documents related to events of interest and investigate them in full detail* [16]. We refer to all the above tasks as **event-related tasks**, or **tasks** for short. Since it is effort-intensive to manually conduct these tasks on large text collections, there is a need for tools that aid human beings in conducting these tasks effectively and efficiently.

Towards this goal, many efforts have been made to automatically detect and track events in text collections under the name of Topic Detection and Tracking [4], [24]. Unfortunately, these approaches usually focus on system-provided answers [15] while many event-related tasks require incorporation of human efforts. There also exist a range of visualization systems in which users can interactively explore text collections, but most of them do not directly support event-related tasks. The reason is that they do not tailor the text collection in the form that suits event-related tasks. Thus there is a *world view gap*, namely the gap between what is being shown and what actually needs to be shown to draw a straightforward representational conclusion for decision making [5], when users perform these tasks. Therefore, to incorporate and leverage human efforts in conducting event-related tasks, visual-based techniques that fuel themselves with event-based automated analysis are needed.

In this paper, we propose such a visual analytics approach. It aims to visually support the following event-related tasks on text collections driven by real life events:

- **Event Browsing:** To allow users to detect the major events motivating the text collections and the long term stories consisting of these events without prior knowledge, and to learn their semantics, temporal context, and the attention received without reading the documents.
- **Event Search, Tracking, and Association:** To allow users to search events by keywords or example text, to track the evolution of an event of interest, and to examine the possible relationships among multiple events within the temporal context.
- **Event Investigation:** To allow users to examine the documents about events of interest in full detail and conduct investigative analysis.

On facilitating the above tasks, the proposed approach integrates automated text analysis with visualization to reveal the events motivating a text collection and the long term stories they construct. To be specific, a novel event-based text analysis technique is proposed and employed to extract document clusters that can be mapped to real life events. The semantics of the events and their temporal influences, namely the continuous attention they drew, are detected according to the characteristics of the clusters. The whole text collection is then

- *D. Luo, J. Yang, and W. Ribarsky are with the University of North Carolina at Charlotte, USA;*
- *M. Krstajic and D. Keim are with the University of Konstanz, Germany.*

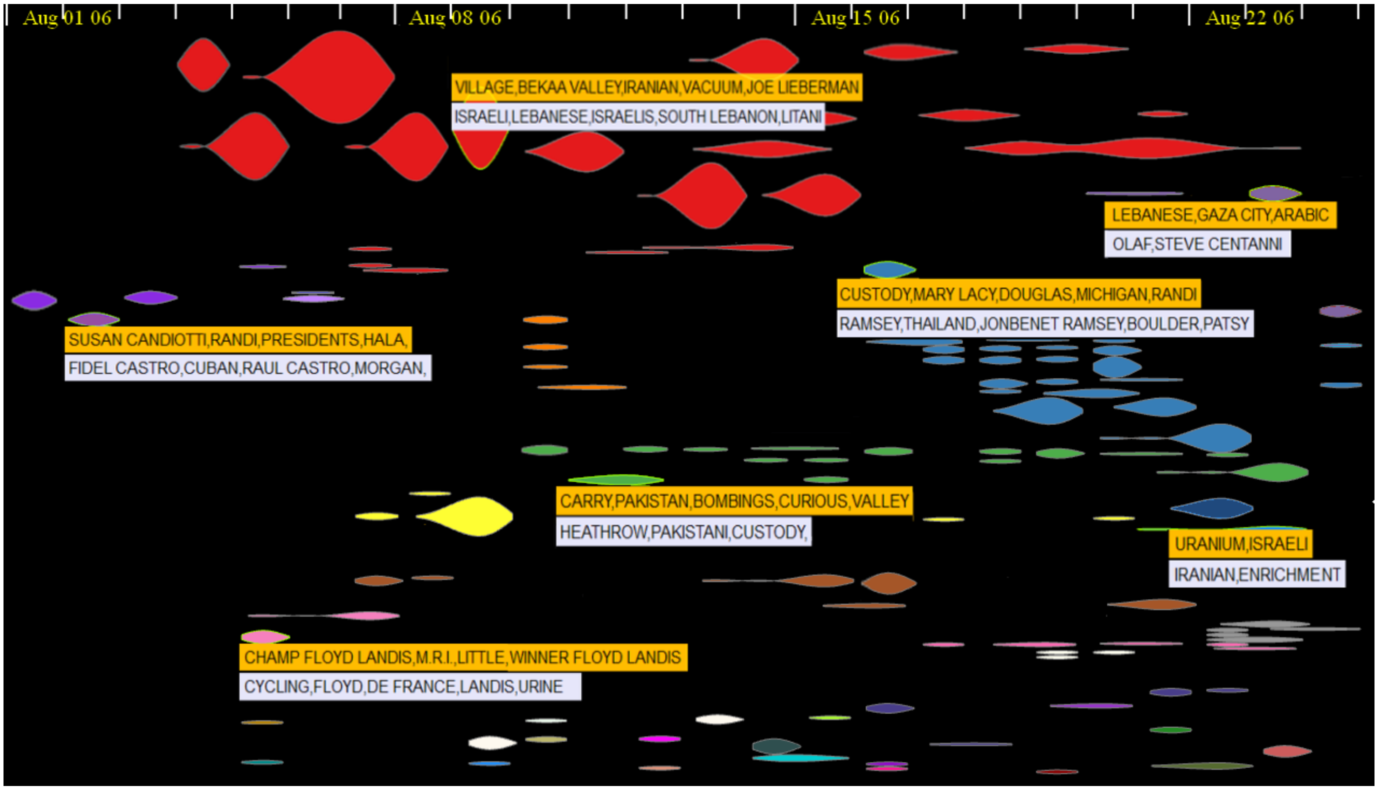


Fig. 1. *CNN* news from Aug. 1 to 24, 2006 ($29,211$ closed-caption documents) in EventRiver. The horizontal axis of the display is a time axis where time flows from left to right. Each bubble represents an event mapped to a cluster of documents in the collection. Its changing vertical dimension represents how it drew continuous attention after it had happened in terms of the number of documents reporting or discussing it. A pointy left end of a bubble indicates the earliest reports about the event and a big body conveys the information that the event drawn significant attention. Events with the same color and adjacent to each other in their vertical positions are closely related and construct a long term story.

visually presented in a display where the semantics and temporal influences of the events are visually depicted in a temporal context to reveal the narrative arcs of the long term stories they construct (see Fig. 1 for an example). Therefore, users can interactively conduct event-related tasks on the display. Since the display looks like a river of events flowing over time, our approach is named **EventRiver**. It narrows the world view gap for event-related tasks by organizing and depicting text collections in terms of events.

A fully working prototype of EventRiver has been implemented and used to explore large collections of close-captioned broadcast news videos. A set of case studies has been conducted to show how to use EventRiver to undertake the three event-related tasks listed above. A set of experiments and a preliminary user test have been conducted to evaluate the effectiveness and efficiency of EventRiver in event extraction and interactive exploration.

The rest of the paper is organized as follows. Section 2 discusses related work. Section 3 discloses the automatic analysis techniques used in EventRiver. Section 4 introduces the visualizations and interactions for event browsing, retrieval, tracking, association, and detail investigation. Sections 5, 6 and 7 report the case studies, the experiments, and the preliminary user test. Section 8 concludes the paper.

2 Related Work

Many efforts have been made on automated topic detection and tracking [4], [24]. For example, Mei and Zhai [19] extract events using a language model-based approach and analyze the temporal and evolutionary structure of the events to discover the evolutionary theme patterns. Fung et al. [11] propose a *Time Driven Documents Partition* algorithm to construct an event hierarchy in a text corpus based on a given query. Allan [3] presents a survey of recent work on topic detection and tracking techniques. They usually focus on system-provided answers [15].

Most traditional visualization approaches for exploring text collections with temporal references fall into either of the following two categories, namely *keyword tracing techniques* and *time slicing techniques*. Keyword tracing techniques visually depict the strength changes of individual keywords in a text collection over time. A representative approach in this category is *ThemeRiver* [14], which depicts the strength changes of individual keywords as currents within a river flowing along a time axis. *LensRiver* [12] is a variation of *ThemeRiver*. It reveals the global relationships among the keywords by constructing a keyword hierarchy for the whole text collection and bundling the currents according to the hierarchy. *Narratives* [10] uses time plots to visualize how concepts (keywords) change over time in weblog archives and introduces several methods to explore how these concepts relate to each other.

Keyword tracing techniques do not visually present event information to users and directly support event-related tasks. The users have to speculate about events and their semantics, temporal influences, and evolution by observing the changing strengths of individual keywords. Since the same keyword can contribute to multiple events even at the same time period, and the semantics of an event are usually conveyed by a set of co-occurring keywords in a short time period, this speculation is not only time consuming, but also inaccurate. In addition, to access documents related to an event, the users may need to conduct a set of searches. Time slicing techniques divide a text collection into multiple time slices, generate a static view for each slice, and dynamically display the static views in a time sequence to reveal the topic changes. For instance, Hetzler et al. [15] visualize text collections in a 2D projection space with fresh and stale documents visually distinguished. An evolving window of time is used to control the animation. Erten et al. [9] present a temporal graph layout algorithm to visualize the topic evolution of a computing literature collection. Textpool [2] buffers live text streams into a pool, extracts the most frequently occurring salient terms from the buffered streams, and displays them in a dynamic text collage. TagLine [8] characterizes the most interesting tags associated with a sliding interval of time and uses an animation to reveal how the interesting tags evolve over time. Although time slicing techniques can be good at revealing topics in each time slice, users often suffer from change blindness [20] when investigating event evolutions.

There exist several visualization approaches that organize text into topics and display the topics along a time axis, including a couple of recent visual analytics approaches. *Conversation landscape* [7] groups postings in the same conversation together and displays them as horizontal lines along y axis representing time. CAST [21] applies a hierarchical agglomerative clustering algorithm on keywords extracted from a news corpus to generate themes (clusters of keywords). The themes are visualized as an array of the theme words over time with width-changing lines connecting words in the same theme to indicate the flow of stories. News Cycle [17] clusters textual variants of short, distinctive, and quoted phrases and uses a stacked plot to reveal the daily rhythms and their temporal patterns in news media and blogs. The analysis algorithms underlying the above approaches require reprocessing the whole text collections when new data arrive. The consistency among the results is not guaranteed after each reprocessing. Differently, EventRiver uses an incremental algorithm without requiring reprocessing the whole dataset when new data arrive, and thus has the potential to visualize fast evolving text collections.

3 Automatic Analysis

3.1 Problem Definition

An **event** refers to an occurrence that happens at a specific time and draws continuous attention [3]. We assume the following **document generation model** for text collections driven by real life events: once an event happens, documents recording or discussing it will be generated when it draws continuous attention. Since the documents are about the same event, they will have *closely related contents*. Since the event draws continuous attention, the documents will *coincide or be adjacent in time*. Thus, if we discover a cluster of documents that have closely related contents and coincide or are adjacent in time, named a **temporal-locality cluster** or a **cluster** for short, we can establish a direct mutual mapping between this cluster *and* the real life event motivating the documents in it.

The mapping relation between a cluster and a real life event allows us to learn the event by analyzing the cluster. For example, we can get strong clues for what happened in the event by summarizing the semantics of the cluster. We can also learn when the event occurred, how long it attracted continuous attention, and how significant the attention was by analyzing the temporal features of the cluster. Characterizing events in this way and presenting event characteristics to users will greatly narrow the world view gap when they conduct event-related tasks. Therefore, two major tasks of the automatic analysis component are to *discover temporal-locality clusters* and to *characterize their mapped events using the cluster semantics and temporal features*. We call the first task **temporal-locality clustering** and the second one **event characterization**.

An event may have its triggering events and follow-up events. They form the narrative arc of a **long term story**. For example, a murder case may be the triggering event for the event of the arrest of a suspect after a while. The latter may have a follow-up event, for example, the conviction of the suspect. Documents about these events may share common contents, but the temporal-locality clustering will separate them into different clusters if the time spans when they draw continuous attention do not overlap. This clustering approach significantly distinguishes EventRiver from existing clustering-based text collection visualization approaches that only consider content coherency. The latter often mix the documents about an event and its triggering and follow-up events into the same cluster if the differences in their contents are subtle. Consequently, the narrative arc is lost.

To reveal the narrative arcs of long running stories and allow users to track event evolution, EventRiver constructs **cluster groups** consisting of temporal-locality clusters with related contents regardless of their temporal spans. The clusters in a group can be sequential (i.e., they are mapped to triggering and follow-up events) and reveal how a story develops. They can also be contemporary and reflect different aspects of a complex story.

3.2 Dynamic Data Processing

Currently the visualization of EventRiver does not support data streams. However, the automatic analysis of EventRiver is built upon a *streaming* data model and an *incremental* data processing mechanism. By "streaming" we mean that all text documents are divided into a sequence of batches based on their time stamps and the intake and processing of the documents are in a batch-by-batch manner. By "incremental" we mean that processing the current batch of data does not involve re-processing of data in previous batches, and its processing results are seamlessly merged into the final outputs. The incremental feature brings great computational efficiency to the approach. The whole dynamic data processing mechanism endows EventRiver the potential to enlarge its territory to real-time text stream applications.

3.3 Temporal-Locality Clustering

We propose an incremental streaming text analysis algorithm for temporal-locality clustering. Following the two-phase data stream clustering paradigm proposed by Aggarwal et al. [1], the algorithm consists of an online component, named the **Parallel-Processing Component**, which periodically stores detailed summary statistics of newly arrived raw data, and an offline component, named the **Sequential-Processing Component**, which uses the summary statistics generated by the Parallel-Processing Component to conduct higher level data aggregation without processing the raw data.

Definition 1: (**Particle Time Zone**) The time horizon of a text collection with temporal references is divided into a sequence of non-overlapping and equal-length short parts, each of which is called a **particle time zone** ζ. They are sorted in chronological order ζ_i $(i = 0, 1, \dots)$, and have time length $l(\zeta_i) = l^{\zeta} = CONST$.

Definition 2: (**Document Batch**) Let $doc_j@\zeta_i$ mean that the time stamp of document doc_j $(j = 0, 1, \dots)$ falls into particle time zone ζ_i. We define a **document batch** $B_i = \{doc_j | doc_j@\zeta_i\}$.
Note: we assume that l^{ζ} is so small that the time differences among documents in B_i $(i = 0, 1, \dots)$ are trivial to the application. Therefore, the value of l^{ζ} is application-related. For example, we set l^{ζ} to be 24 hours when we examine news articles within several months.

A text collection is divided into a sequence of document batches, which are the input of the temporal-locality clustering algorithm. The Parallel-Processing Component processes B_i $(i = 0, 1, \dots)$ one by one in the chronological order of ζ_i. Whenever it finishes processing a document batch, the Sequential-Processing Component will generate and maintain temporal-locality clusters using the output of the Parallel-Processing Component. The details are described as follows.

Fig. 2. SPC Pseudo Code

```
Begin $SPC(\Gamma, \Gamma_i^c, \Gamma_i^a, ts(\zeta_i), te(\zeta_i), \delta^t)$
for all $\gamma_j \in \Gamma_i^a$ do
    for all $\nu_k \in \Gamma_i^c$ do
        Set $cov_{j,k}$ :=the Jaccard Coefficient of $\gamma_j$ and $\nu_k$;
    end for
    Set $kmax(j) := k_m$, where $cov_{j,k_m} == \max_{\forall \nu_k \in \Gamma_i^c} cov_{j,k}$;
end for
for all $kmax(j)$ do
    if $cov_{j,kmax(j)} == \max_{\forall \gamma_{j'} \in \Gamma_i^a, j' \neq j} cov_{j',kmax(j)}$ then
        for all document $doc_x \in \nu_{kmax(j)}$ do
            Add $doc_x$ into $\gamma_j$
        end for
        $tl(\gamma_j) = te(\zeta_i)$;
        $Remove(\nu_{kmax(j)}, \Gamma_i^c)$;
    end if
end for
for all $\gamma_j \in \Gamma_i^a$ do
    if $te(\zeta_i) - tl(\gamma_j) \geq \delta^t$ then
        $Remove(\gamma_j, \Gamma_i^a)$;
    end if
end for
for all $\nu_k \in \Gamma_i^c$ do
    $\gamma_m := MakeNew(\nu_k), m \in N$;
    $Insert(\gamma_m, \Gamma_i^a)$;
    $Insert(\gamma_m, \Gamma)$;
end for
End $SPC(\Gamma, \Gamma_i^c, \Gamma_i^a, ts(\zeta_i), te(\zeta_i), \delta^t)$
```

3.3.1 Parallel-Processing Component (PPC)

The Parallel-Processing Component (PPC) processes one batch of documents in each run, ignoring their time differences and forgetting other documents in the text collection. The input documents have been converted into keyword vectors representations using the algorithm presented by Luo et al. [18] before they are sent to PPC.

PPC clusters the documents into a set of **sub-clusters** by content similarity and outputs these sub-clusters. Each sub-cluster records the IDs and the keyword vectors of the member documents. The clustering algorithm used by PPC is Rock [13], a robust hierarchical clustering algorithm for data with Boolean and categorical attributes. It uses Jaccard Coefficients to define the neighborhood among the data items. The similarity between two data items is measured by the number of their common neighbors. We chose Rock since it generates high quality clusters and scales to large datasets (its worst-case time complexity is $O(n^2 + nm_m m_a + n^2 logn)$, where n is the number of input data items and m_m and m_a are the maximum and average numbers of neighbors for all data items) [13].

3.3.2 Sequential-Processing Component (SPC)

The Sequential-Processing Component (SPC) creates and maintains temporal-locality clusters. Once PPC finishes processing a document batch, its output (i.e., sub-clusters denoted by ν), are sent to SPC. According to the content coherence and temporal-locality criteria, SPC either merges a sub-cluster into an existing temporal-locality cluster or makes it a new temporal-locality cluster. To be specific, suppose that $\zeta_i(i = 0, 1, \dots)$ is the particle time zone of the upcoming document batch B_i that SPC is about to process. Let $\gamma_j(j = 0, 1, \dots)$ denote a temporal-locality cluster, and $\gamma_j @ \zeta_i$ denote that "temporal-locality cluster γ_j covers zone ζ_i". Let $ts(\zeta_i)$ and $te(\zeta_i)$ be the starting time and ending time of ζ_i (i.e. $l^\zeta = te(\zeta_i) - ts(\zeta_i)$). We define Γ_i^c to be the set of sub-clusters ν_k $(k = 0, 1, \dots)$ detected from B_i. We also define Γ to be the set of ALL temporal-locality clusters that have been detected from the text collection before current batch B_i is processed. Γ is empty before ζ_0 is processed. Let $tl(\gamma_j)$ be the latest time when cluster γ_j was updated. We define the **Active Cluster Set** of batch B_i as $\Gamma_i^a = \{\gamma_j | \gamma_j \in \Gamma$ and $(ts(\zeta_i) - tl(\gamma_j)) < \delta^t\}$, where δ^t is a threshold to ensure *temporal-locality* (heuristically, we set $\delta^t = C \cdot l^\zeta$, $C \geq 1$ is a constant). Sub-clusters can only be merged into clusters belonging to Γ_i^a so as to ensure temporal-locality. The pseudo code of SPC is shown in Fig. 2.

The time complexity of SPC is $O(mn^2)$, where n and m are the sizes of Γ_i^c and Γ_i^a respectively. To speed up the calculation of the Jaccard Coefficient between a sub-cluster and a temporal-locality cluster, sampling the documents to be calculated can be an option.

3.4 Event Characterization

According the document generation model in Section 3.1, each temporal-locality cluster is mapped to a real life event, and is used to characterize the temporal and semantic aspects of the event. Let ϵ_γ denote an event which is mapped to cluster γ. The temporal influence, namely to what extend event ϵ_γ draws continuous attention, and the semantic summary of event ϵ_γ are modeled by analyzing cluster γ.

3.4.1 Temporal Influence

***Definition** 3:* (**Temporal Influence**) The **temporal influence** of event ϵ_γ, within particle time zone ζ, is defined as the number of γ's documents that fall into the particle time zone ζ, and is denoted by $f(\epsilon_\gamma, \zeta)$.

We assume that the more significant an event is; the more documents will be published to report and discuss it. In other words, significant events will have big temporal influences. Thus temporal influence can help user identify significant events.

3.4.2 Semantic Summary

The semantic content of a temporal-locality cluster provides clues to the answer of "what happened in the mapped event". EventRiver summarizes the semantics of the cluster and uses them to annotate the event. The document vectors in a cluster usually involve a large number of keywords, each of which has its specific weight in representing the semantics of the cluster. Therefore, it is inappropriate to indifferently use all the keywords to describe the event. We propose a dual labeling approach to depict the semantics of the event. To begin with, we define two concepts.

***Definition** 4:* The **Intra-Link Co-Strength** $ILC(k_i, k_j)$ between keyword k_i and k_j in a temporal-locality cluster is the number of its documents in which k_i and k_j co-occur.

***Definition** 5:* The **Intra-Link Strength**, or **strength** for short, of keyword k_i in a temporal-locality cluster is defined as $ILS(k_i) = \sum_{k_j \in U^k, k_j \neq k_i} ILC(k_i, k_j)$, where U^k is the union of all keywords used to characterize documents in the cluster.

The strength of each keyword in U^k is calculated. The keywords with high strengths and low strengths are used to represent the semantics of the event. The keywords with high strengths are the most shared keywords in the cluster and describe the semantic context of the event. They are named **Context-Keywords** of the event. The keywords with low strengths convey unique contents in the event and are named **Core-Keywords** of the event. Assuming that the distribution of keywords is normal, we set the thresholds for context-keywords and core-keywords as $\mu + \sigma$ and $\mu - \sigma$ respectively, where μ and σ are the mean and standard deviation of the keyword strengths. The semantic representation of an event is the combination of the **context-keywords** and the **core-keywords**, and thus called **dual labels**.

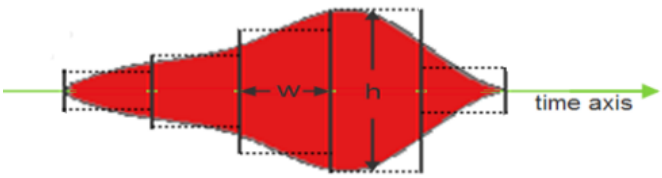


Fig. 3. The bubble representation of an event.

3.5 Cluster Group Construction

Clusters are organized into **cluster groups** $\gamma_i^g(i = 0, 1, \dots)$ using an algorithm similar to the SPC algorithm, where clusters replace sub-clusters and cluster groups replace clusters. We also define the **group time horizon** ζ_i^g as the union of the particle time zones covered by any clusters in γ_i^g . Cluster groups are updated after each run of SPC. The similarity between two clusters is calculated based on their context-keywords using the Jaccard Coefficient.

A cluster group is considered an **Outlier** if it meets one of the following criteria: (1) it contains only one cluster (i.e., the event has no related events); (2) there are more than one clusters in the group, but all the clusters only occupy the same single particle time zone (i.e. the events do not receive continuous attention). In the rest of this paper, cluster groups refer to groups that are not outliers.

4 Visualization and Interactions

Semantics and time are two key aspects to the understanding of a real life event [3]. To narrow the

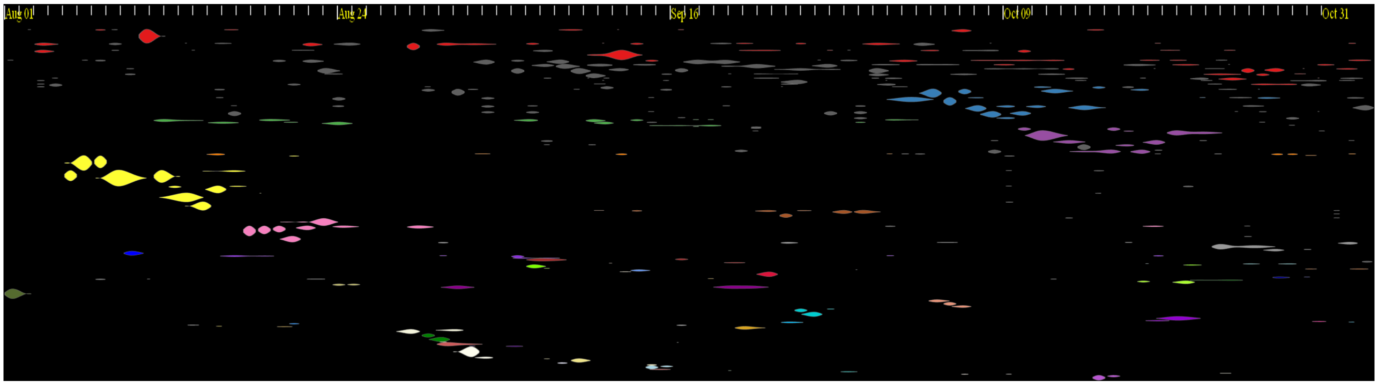


Fig. 4. The bubble layout with 1901 clusters. The corpus in display is CNN news from Aug. 1 to Oct 31, 2006 (72,435 closed-caption documents).

world view gap for event-related tasks, it is desired to intuitively present the events, their semantics, temporal information, and influences in a temporal context provided by the other events to users so that they can browse events and retrieve events of interest. It is also important to reveal long-term stories consisting of related events and to visually present their narrative arcs to the users, so that they can discover high level stories and trace event evolution. Furthermore, the users should be allowed to retrieve and investigate stories and events of interest effectively and efficiently even when the text collection explored is large. The visualizations and interactions in EventRiver are designed targeting at the above goals.

4.1 Visualization Design

EventRiver visually presents events and long term stories to users. First, the events are displayed as bubbles whose sizes pre-attentively represent the temporal influences of the events. Second, the colors and layouts of the bubbles highlight long term stories consisting of related events in a temporal context. Third, the semantics of the events and stories are displayed in their semantic representations.

4.1.1 Visual Representation of Events

As shown in Fig. 3, EventRiver displays an event as a bubble in a river of time. The horizontal dimension of a bubble represents the time span when it draws continuous attention, namely the life span of the mapping cluster. Its changing vertical dimension represents its temporal influence in terms of the number of documents reporting or discussing it (refer to Section 3.4). By comparing the sizes of different bubbles, users can find events that received most significant attention since their bubbles are bigger and longer than the other bubbles. Thus this design benefits event-related tasks such as event browsing and retrieval.

The curved outline of a bubble is a smoothed approximation of a set of rectangles as illustrated in Fig. 3. The rectangles depict temporal influence $f(\epsilon_\gamma, \zeta)$ of event ϵ_γ. Each rectangle corresponds to the particle time zone ζ covered by γ. Its width (w) represents l^ζ and its height (h) represents $f(\epsilon_\gamma, \zeta)$. The curve boundaries of the bubbles are the *Cubic Spline Interpolation* of these rectangles.

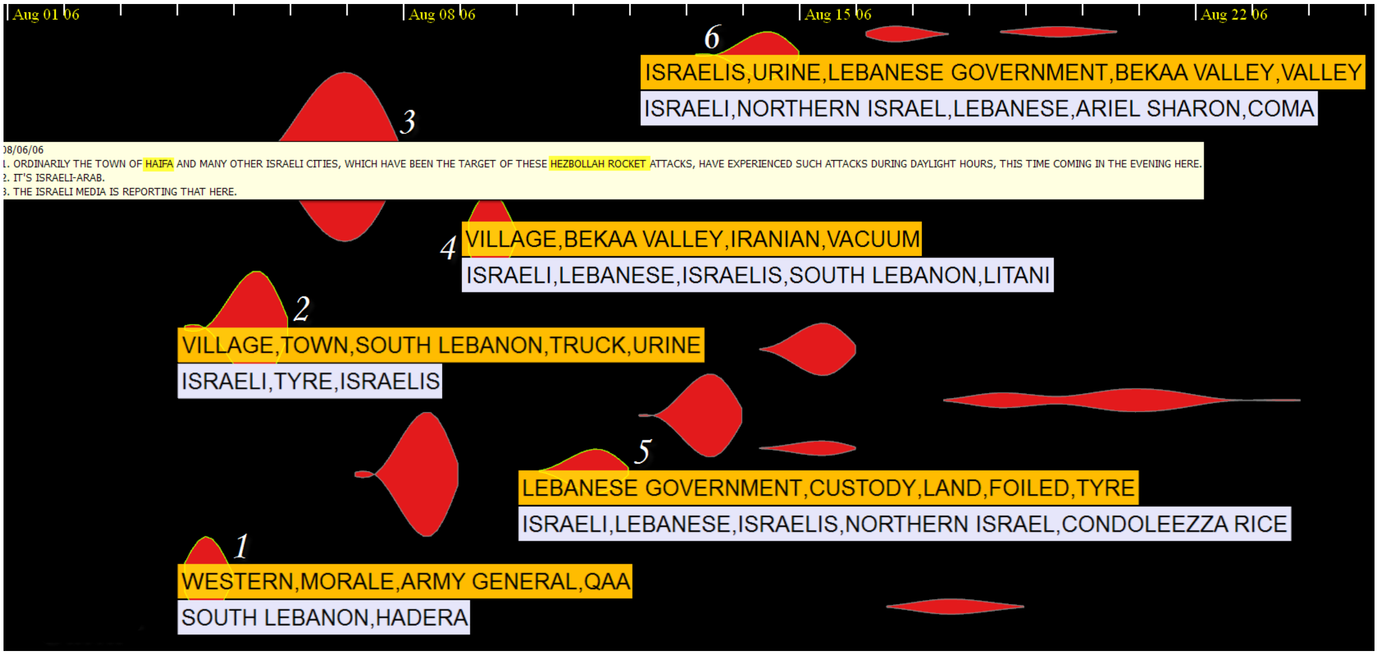
4.1.2 Visual Representation of Long Term Stories

A **long term story**, **story** for short, consists of events mapped to clusters in the same cluster group. We thus assign the same color to their bubbles to emphasize their relationships. Different colors are assigned to different stories to distinguish them from each other. All outliers (refer to 3.5) are colored in dark grey to reduce color clutter on the screen.

In addition, the bubbles are laid out in a river like display (see Fig. 1) to show the narrative arcs of the stories in a temporal context. The horizontal axis of the display is a time axis flowing from left to right. The bubbles are positioned along the time axis according to the time spans their clusters cover. Users can thus observe the temporal influence of an event in a context consisting of other events in the same story and other stories. The vertical positions of the bubbles distinguish different stories. Four rules are followed in the vertical position assignment: **(1)** Events belong to the same story should be adjacent in their vertical positions to reveal their relationships. **(2)** The more important a story is, the higher its vertical position. Thus the reading manner of human beings, which is usually from top to bottom, can be supported. The **importance of a story** can be measured using different criteria, such as the number of related documents, the length, starting time, or ending time of its group time zone, or the maximum number of related documents in any particle time zone. Users can interactively set the criterion. **(3)** Within a story, the events are positioned one by one. The positioning priority from high to low is starting time, peak strength, and duration. **(4)** No overlaps are allowed among the bubbles.

The positioning algorithm is described as follows.

Step 1: Put all cluster groups into a sorted queue where the one with large *Importance* (see rule 2 above) comes first into the queue.



08/06/06
1. ORDINARILY THE TOWN OF HAIFA AND MANY OTHER ISRAELI CITIES, WHICH HAVE BEEN THE TARGET OF THESE HEZBOLLAH ROCKET ATTACKS, HAVE EXPERIENCED SUCH ATTACKS DURING DAYLIGHT HOURS, THIS TIME COMING IN THE EVENING HERE.
2. IT'S ISRAELI-ARAB.
3. THE ISRAELI MEDIA IS REPORTING THAT HERE.

Aug 01 06
Aug 08 06
Aug 15 06
Aug 22 06
6
ISRAELIS,URINE,LEBANESE GOVERNMENT,BEKAA VALLEY,VALLEY
ISRAELI,NORTHERN ISRAEL,LEBANESE,ARIEL SHARON,COMA
3
4
VILLAGE,BEKAA VALLEY,IRANIAN,VACUUM
ISRAELI,LEBANESE,ISRAELIS,SOUTH LEBANON,LITANI
2
VILLAGE,TOWN,SOUTH LEBANON,TRUCK,URINE
ISRAELI,TYRE,ISRAELIS
5
LEBANESE GOVERNMENT,CUSTODY,LAND,FOILED,TYRE
ISRAELI,LEBANESE,ISRAELIS,NORTHERN ISRAEL,CONDOLEEZZA RICE
1
WESTERN,MORALE,ARMY GENERAL,QAA
SOUTH LEBANON,HADERA

Fig. 5. Zooming into the long term story about the *2006 Lebanon War*. The on-the-fly snippet of event 3 is displayed.

Step 2: Take the group at the front of the queue (referred as γ^g_{top}). Given the mapping relation between the clusters in the group and a couple of real life events, each of these events is placed into the group's "playground". The **playground** of γ^g_{top} is a space whose horizontal dimension is confined by the group time zone ζ^g_{top}; and its vertical dimension is "half-open": its upper boundary is the topmost position of the display area tailored by ζ^g_{top} where there are no other events placed underneath; and there is no constrain to the lowest vertical position. The events are placed one by one into the playground in the positioning priority defined in rule 3. Each event is placed at the highest possible vertical position within the playground without overlapping the settled events.

Step 3: Repeat *Step 2* until the queue of cluster groups is empty.

Step 4: Insert the outlier events to the topmost unoccupied space one by one according to the same priority list used in settling events in the same story.

Step 5: Scale the positions and the vertical sizes of all the events to fit them into the screen space. The relative vertical sizes of the events are kept so that users can compare the temporal influence of different events.

Fig. 4 gives an example of the bubble layout. This figure proves that even with large number of bubbles, the narrative arcs of significant long term stories (seen as the groups in yellow, in pink, in blue, and in red) are still well preserved in the visualization.

4.1.3 Semantic Representation

The semantics of events are displayed as labels and snippets (see Fig. 5 for an example) in the visualization. An event is labeled by **dual-labels**, where core-keywords and context-keywords (refer to Section 3.4) are displayed in parallel and in distinct background colors.

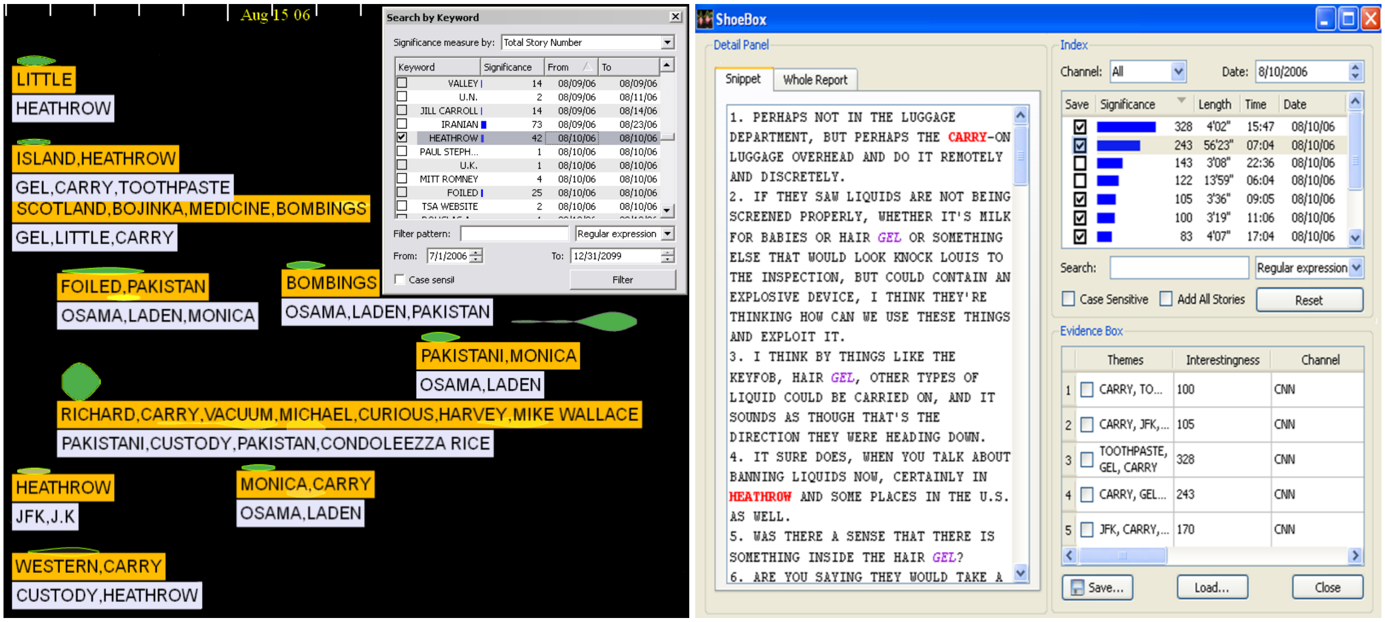
Dual labels help users learn the common theme in a long term story and detect the unique content of the individual events. For example, Fig. 5 shows events in the long term story about the *2006 Lebanon War* with their labels displayed. The labels in white background provide context information while the labels in yellow background display the unique content of each event. They allow users to build a rough mental map of the long term story without reading the documents. Note that there are a few irrelevant keywords like "IRANIAN" in the yellow labels. They are introduced by errors in the segmentation of the closed caption documents.

To reduce clutter, EventRiver provides a few automatic labeling strategies to selectively label the events. **Representative event labeling** automatically labels a representative event, such as the event mapped to the biggest cluster, for each story. **Outlier labeling** labels outlier events only. In addition, when users zoom into a long term story to examine it in detail, all of its events will be automatically labeled. Users can click on an event to manually turn on/off its labels.

When the labels do not fulfill the information needs of a user, the user can trigger the **on-the-fly snippet** by hovering the mouse over an event. The snippet shows a few sentences from the documents in the cluster that contain one or more core-keywords and context-keywords. Fig. 5 gives an example of the snippet.

4.2 Interactions for Event-Related Tasks

EventRiver provides a set of interactions to assist event-related tasks, as described below.



Significance measure by: Total Story Number

	Keyword	Significance	From	To
☐	VALLEY	14	08/09/06	08/09/06
☐	U.N.	2	08/09/06	08/11/06
☐	JILL CARROLL	14	08/09/06	08/14/06
☐	IRANIAN	73	08/09/06	08/23/06
☑	HEATHROW	42	08/10/06	08/10/06
☐	PAUL STEPH...	1	08/10/06	08/10/06
☐	U.K.	1	08/10/06	08/10/06
☐	MITT ROMNEY	4	08/10/06	08/10/06
☐	FOILED	25	08/10/06	08/10/06
☐	TSA WEBSITE	2	08/10/06	08/10/06

Filter pattern: Regular expression

From: 7/1/2006 To: 12/31/2099

☐ Case sensil

1. PERHAPS NOT IN THE LUGGAGE DEPARTMENT, BUT PERHAPS THE CARRY-ON LUGGAGE OVERHEAD AND DO IT REMOTELY AND DISCRETELY.
2. IF THEY SAW LIQUIDS ARE NOT BEING SCREENED PROPERLY, WHETHER IT'S MILK FOR BABIES OR HAIR GEL OR SOMETHING ELSE THAT WOULD LOOK KNOCK LOUIS TO THE INSPECTION, BUT COULD CONTAIN AN EXPLOSIVE DEVICE, I THINK THEY'RE THINKING HOW CAN WE USE THESE THINGS AND EXPLOIT IT.
3. I THINK BY THINGS LIKE THE KEYFOB, HAIR GEL, OTHER TYPES OF LIQUID COULD BE CARRIED ON, AND IT SOUNDS AS THOUGH THAT'S THE DIRECTION THEY WERE HEADING DOWN.
4. IT SURE DOES, WHEN YOU TALK ABOUT BANNING LIQUIDS NOW, CERTAINLY IN HEATHROW AND SOME PLACES IN THE U.S. AS WELL.
5. WAS THERE A SENSE THAT THERE IS SOMETHING INSIDE THE HAIR GEL?
6. ARE YOU SAYING THEY WOULD TAKE A

Channel: All Date: 8/10/2006

Save	Significance		Length	Time	Date
☑	█████	328	4'02"	15:47	08/10/06
☑	████	243	56'23"	07:04	08/10/06
☐	██	143	3'08"	22:36	08/10/06
☐	██	122	13'59"	06:04	08/10/06
☑	█	105	3'36"	09:05	08/10/06
☑	█	100	3'19"	11:06	08/10/06
☑	█	83	4'07"	17:04	08/10/06

Search: Regular expression

☐ Case Sensitive ☐ Add All Stories

	Themes	Interestingness	Channel
1	☐ CARRY, TO...	100	CNN
2	☐ CARRY, JFK,...	105	CNN
3	☐ TOOTHPASTE, GEL, CARRY	328	CNN
4	☐ CARRY, GEL...	243	CNN
5	☐ JFK, CARRY,...	170	CNN

(a) Zooming into selected events

(b) A shoebox for the event labeled “GEL, LITTLE, CARRY”

Fig. 6. Investigating the events about the *2006 Transatlantic Aircraft Plot*. The events in (a) were selected using the keyword *Heathrow*.

Interactions for Event Browsing:

- **Filtering-by-Influence** - Users can set an influence range to hide events whose peak influences are out of the range. Unhidden events are repositioned and proportionally deformed to fill the room made by hidden events. In this way, users can examine the major events driving the text without the distraction from the small ones, as shown in Fig. 1.
- **Semantic Zooming** - Users can use semantic zooming to remove unselected events from the screen and rescale the selected events to fulfill the screen. Meanwhile, the labels of the selected events will be automatically turned on.
- **Temporal Zooming** - Users can brush the time axis to select a specific time period as a focus time slot. This time slot will be rescaled to fulfill the horizontal screen space.
- **Group Sorting** - Users can sort the stories by a different importance criterion (refer to Section 4.1.2) and regenerate the vertical layout so that the stories considered more important will be displayed higher in the display.
- **Manual Relocation** - Users can manually change the vertical positions of individual events to reduce overlaps among the labels.

Interactions for Event Search, Tracking, and Association:

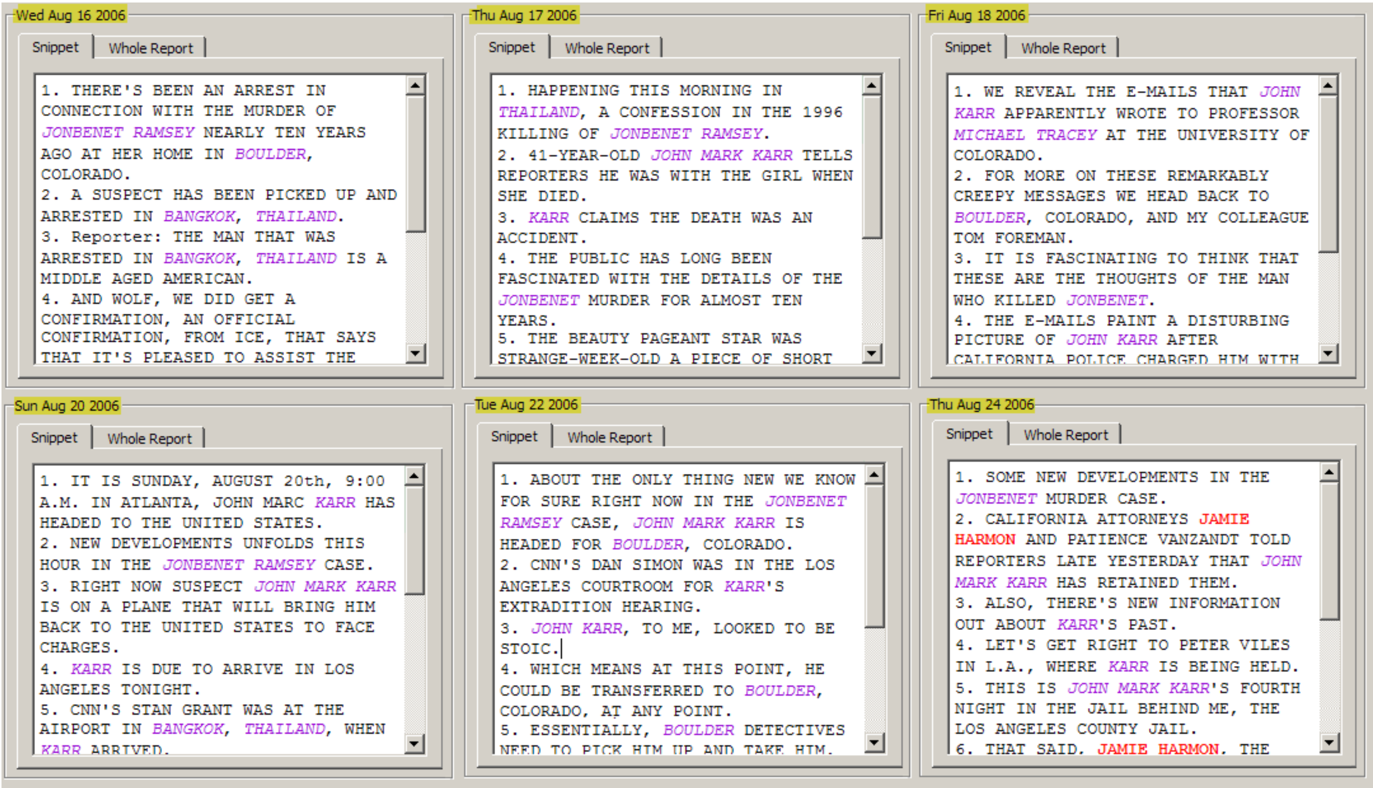
- **Search-by-Keywords** - Users can search events whose mapping clusters include/exclude any documents containing a few input keywords. Users can type the keywords or select them from a preset keyword list (see Fig. 6(a) for an example). The list contains all the keywords used to characterize documents in the collection and can be sorted by different criteria, such as the total occurrences, the peak counts within the particle time zones, and the first/last appearance time.
- **Search-by-Example** - Users can interactively select an event and search all the events that share any core-keywords or context keywords with it. In this way they can find associated events and track event evolution. They can also search events by a piece of example text.

Interactions for Event Investigation:

Shoebox - Users can investigate an event by opening its shoebox with a mouse-click. The shoebox allows the users to examine full details of the related documents and conduct investigative analysis (see Fig. 6(b) for an example). Its interface consists of three components:

- **Index Panel** lists all documents within the cluster, which can be sorted by their lengths, releasing time, or other criteria. Typing a keyword in the search box will remove all documents with no occurrence of the keyword from the list. Clicking a document in the list will load the document to the detail panel.
- **Detail Panel** allows users to read a document in snippets or full text. In the snippet mode, only the sentences containing core-keywords or context-keywords are displayed. In both modes, the keywords are highlighted by colors.
- **Evidence Box** allows users to save documents of interest into evidence files for external uses, such as evidence exchange or hypothesis evaluations.

Storyboard - users can use a storyboard to examine all selected events at the same time to construct a metal map of the story consisting of these events. It displays the



Wed Aug 16 2006

1. THERE'S BEEN AN ARREST IN CONNECTION WITH THE MURDER OF JONBENET RAMSEY NEARLY TEN YEARS AGO AT HER HOME IN BOULDER, COLORADO.
2. A SUSPECT HAS BEEN PICKED UP AND ARRESTED IN BANGKOK, THAILAND.
3. Reporter: THE MAN THAT WAS ARRESTED IN BANGKOK, THAILAND IS A MIDDLE AGED AMERICAN.
4. AND WOLF, WE DID GET A CONFIRMATION, AN OFFICIAL CONFIRMATION, FROM ICE, THAT SAYS THAT IT'S PLEASED TO ASSIST THE

Thu Aug 17 2006

1. HAPPENING THIS MORNING IN THAILAND, A CONFESSION IN THE 1996 KILLING OF JONBENET RAMSEY.
2. 41-YEAR-OLD JOHN MARK KARR TELLS REPORTERS HE WAS WITH THE GIRL WHEN SHE DIED.
3. KARR CLAIMS THE DEATH WAS AN ACCIDENT.
4. THE PUBLIC HAS LONG BEEN FASCINATED WITH THE DETAILS OF THE JONBENET MURDER FOR ALMOST TEN YEARS.
5. THE BEAUTY PAGEANT STAR WAS STRANGE-WEEK-OLD A PIECE OF SHORT

Fri Aug 18 2006

1. WE REVEAL THE E-MAILS THAT JOHN KARR APPARENTLY WROTE TO PROFESSOR MICHAEL TRACEY AT THE UNIVERSITY OF COLORADO.
2. FOR MORE ON THESE REMARKABLY CREEPY MESSAGES WE HEAD BACK TO BOULDER, COLORADO, AND MY COLLEAGUE TOM FOREMAN.
3. IT IS FASCINATING TO THINK THAT THESE ARE THE THOUGHTS OF THE MAN WHO KILLED JONBENET.
4. THE E-MAILS PAINT A DISTURBING PICTURE OF JOHN KARR AFTER CALIFORNIA POLICE CHARGED HIM WITH

Sun Aug 20 2006

1. IT IS SUNDAY, AUGUST 20th, 9:00 A.M. IN ATLANTA, JOHN MARC KARR HAS HEADED TO THE UNITED STATES.
2. NEW DEVELOPMENTS UNFOLDS THIS HOUR IN THE JONBENET RAMSEY CASE.
3. RIGHT NOW SUSPECT JOHN MARK KARR IS ON A PLANE THAT WILL BRING HIM BACK TO THE UNITED STATES TO FACE CHARGES.
4. KARR IS DUE TO ARRIVE IN LOS ANGELES TONIGHT.
5. CNN'S STAN GRANT WAS AT THE AIRPORT IN BANGKOK, THAILAND, WHEN KARR ARRIVED.

Tue Aug 22 2006

1. ABOUT THE ONLY THING NEW WE KNOW FOR SURE RIGHT NOW IN THE JONBENET RAMSEY CASE, JOHN MARK KARR IS HEADED FOR BOULDER, COLORADO.
2. CNN'S DAN SIMON WAS IN THE LOS ANGELES COURTROOM FOR KARR'S EXTRADITION HEARING.
3. JOHN KARR, TO ME, LOOKED TO BE STOIC.
4. WHICH MEANS AT THIS POINT, HE COULD BE TRANSFERRED TO BOULDER, COLORADO, AT ANY POINT.
5. ESSENTIALLY, BOULDER DETECTIVES NEED TO PICK HIM UP AND TAKE HIM.

Thu Aug 24 2006

1. SOME NEW DEVELOPMENTS IN THE JONBENET MURDER CASE.
2. CALIFORNIA ATTORNEYS JAMIE HARMON AND PATIENCE VANZANDT TOLD REPORTERS LATE YESTERDAY THAT JOHN MARK KARR HAS RETAINED THEM.
3. ALSO, THERE'S NEW INFORMATION OUT ABOUT KARR'S PAST.
4. LET'S GET RIGHT TO PETER VILES IN L.A., WHERE KARR IS BEING HELD.
5. THIS IS JOHN MARK KARR'S FOURTH NIGHT IN THE JAIL BEHIND ME, THE LOS ANGELES COUNTY JAIL.
6. THAT SAID, JAMIE HARMON, THE

Fig. 7. A *storyboard* for the story about the *Ramsey Murder Case*.

events as an array of shoeboxes in chronological order. See Fig. 7 for an example.

5 Case Study

In this section, we present a few case studies. They show how EventRiver help users quickly browse a large text collection for major driving events, track their evolution, and dive in an event for investigative analyses. The data explored is CNN news from Aug. 1 to Aug. 24, 2006, containing $29,211$ closed-caption documents.

Case Study 1: Event Browsing - In this case, we, without any prior knowledge, quickly browsed the major events and long term stories that drive the news development. When the dataset was opened in EventRiver, *all* events detected from it were displayed. We interactively *filtered* out events with relatively low influences and *labeled* the representative events of the significant stories. The resulting display is shown in Fig. 1. Several long term stories stand out. The story in *red* contains context-keywords "Israeli" and "Lebanese". From the snippets we learn that it is about the **2006 Lebanon War**. The story in *blue* has the context-keywords "Jonbenet Ramsey" and "Thailand". From the snippets we learn that it reports the new leads in the **Jonbenet Ramsey Murder Case**. Similarly, several other stories, such as the **2006 Transatlantic Aircraft Plot** in *green* and **Floyd Landis Drug Scandal of La Tour de France** in *pink*, are detected.

Case Study 2: Event Search, Tracking, and Association - In this case, we took a closer look at the **Jonbenet Ramsey Murder Case**. We selected all events in this story using the *Search-by-Example* interaction and opened a *storyboard* for them (Fig. 7). In Fig. 7, the snippets in shoeboxes exhibit the narratives of this story, starting with an motivating event on *Aug.16* - "There's been an arrest in connection with the murder of Jonbenet Ramsey nearly ten years ago at her home in Boulder, Colorado". Then there were the follow-ups of this event: *Aug.17*, Karr admitted being in the company with Jonbenet when she was killed; the next day, *Aug.18*, an e-mail written by Karr was revealed as a piece of importance evidence in this murder case; on *Aug.20*, suspect John Karr was deported to the United States from Thailand for trial; Karr waived extradition in Los Angeles County Superior Court, clearing the way for his transfer to Boulder (CO) on *Aug.22*; and *Aug.24*, Karr retained his California attorneys for the upcoming trial in Boulder (CO).

Case Study 3: Event Investigation - When we zoomed into the story of **2006 Transatlantic Aircraft Plot** (see Fig. 6(a)), we noticed that the keyword "gel" appears in the labels of a few events. We wondered how *gel* was related to this terrorism and clicked on an event with "gel" in its label to open a shoebox (see Fig. 6(b)) to investigate it. In the shoebox, we searched documents containing "gel" and learned that "gel" had been listed as a suspicious and banned item in air flights to prevent the carrying of disguised liquid explosives.

6 Experiment

The effectiveness of EventRiver in supporting event-related tasks heavily depends on the quality of the temporal-locality clusters. Its scalability to large text collections is related to the time efficiency of the clustering algorithm. We have conducted a set of experiments to evaluate the efficiency and effectiveness of this algorithm.

6.1 Settings

A real dataset consisting of $29,211$ closed-caption documents of CNN news from Aug. 1 to 24, 2006 was used in the experiment. On average 9092 keywords per day were used to characterize the documents.

EventRiver processed the dataset on a PC with Intel Core 2 Duo processor and $2GB$ memory. The parameters of the algorithms were set as follows: the length of particle time zone $l^{\zeta} = 24h$; A sampling rate of 10% was used in SPC Jaccard Coefficient calculation for sub-clusters and clusters which sizes are larger than 10. Otherwise, the sampling rate is 100%.

6.2 Time Efficiency

The time of cluster extraction and cluster group construction for the whole text collection was $5m13s$, which means that it takes about $0.01s$ to process the keywords of each document on average. In particular, given that the average size of a document batch is 1221, the average processing time for one particle time zone, namely the average incremental processing time, was $13.064s$. The results showed that the cluster extraction and group construction were time efficient.

6.3 Quality

There were 363 events and 17 long term stories discovered from the text collection (see Fig. 1). A multi-resolution approach was used to examine the results. First, to make sure that no major stories in the dataset were missing in EventRiver, we visualized the same dataset using IN-SPIRE [23] (see Fig. 8) and compared the stories discovered by EventRiver with the clusters revealed in IN-SPIRE. IN-SPIRE clustered the whole text collection without considering the time stamps of the documents and displayed the clusters as mountains. As shown in Fig. 8, IN-SPIRE revealed 18 significant clusters excluding the similar ones. Comparing them with the 17 stories discovered by EventRiver, we found that the two systems detected 15 topics in common. The two exclusive topics from EventRiver were "Floyd Landis drug scandal of La Tour de France" and "the kidnap of the two journalists Olaf Wiig and Steve Centanni in Gaza". By scanning the original dataset, we were sure that both of them were significant topics. The three exclusive topics from IN-SPIRE were labeled as "larry, ve, didn't", "hot, companies, women", and "larry, ve, ricky, yeah". After reading documents in these topics, we learned that "larry" refers to the famous *CNN* anchor *Larry King* and the topics consist of several small events in Larry King's show. The topic "hot, companies, women" consists of a set of trivial daily news with these three words appearing prominently. The results show that EventRiver captured the major stories from the text collection as effectively as or even better than IN-SPIRE. EventRiver also exhibited significant advantages in visually depicting the narrative arcs of the stories in one view. IN-SPIRE depended on time slicing techniques to reveal the narratives, which was not efficient and suffered from change blindness [20].

We next examined several long term stories detected by EventRiver in detail by comparing their events with an external reference source, namely **Wikipedia.org** [22]. Eight stories with the largest numbers of events were examined, each of which was listed as one of the weekly top 5 most popular topics at CNN.com at least once for the time period during which it occurred. In Wikipedia, the major events of these stories were manually summarized and listed as timelines. We believe that such human-generated summaries are usually more accurate than any automatic results, which make them a benchmark for evaluation.

For each of the eight stories, we first manually compared their events discovered by EventRiver with the events listed in the Wikipedia timeline to identify matching events. Then, we measured the quality of EventRiver results against Wikipedia using *precision* [6], namely the ratio between the number of matching events and the total number of events of this story in EventRiver, and *recall* [6], namely the ratio between the number of matching events and the total number of events in this story listed in the Wikipedia timeline. This process is illustrated in the following example.

The 2006 Lebanon War - A Close Look at the Event Extraction Results

Wikipedia has a timeline [22] for the military operations of the 2006 Lebanon War where 17 events are listed in the time period of the experiment dataset. For EventRiver, there were 15 events in the story of the 2006 Lebanon War. 14 events matched, which led to a precision of $93.3\%(14/15)$ and a recall of $82.4\%(14/17)$.

Here we list a few example matching events. For each event, we cite the event description in Wikipedia and highlight the keywords appearing in the labels of the matching event in EventRiver in bold (see Fig. 5 for the events, labels, and snippets in EventRiver): (1) On *Aug.4* , "Israel targeted the **southern** outskirts of Beirut, and later in the day, Hezbollah launched rockets at the **Hadera** region. 33 civilian farm workers are killed and 20 wounded after an Israeli airstrike in a farm near **Qaa** in **Lebanon**." (2) On *Aug.5*, "**Israeli** commando soldiers landed in **Tyre**, where fighting erupted with Hezbollah forces." (3) On *Aug.6*, "12 army reservists resting near the Lebanon border were killed in the deadliest barrage of **Hezbollah rocket** attacks so far. Three **Israeli** civilians

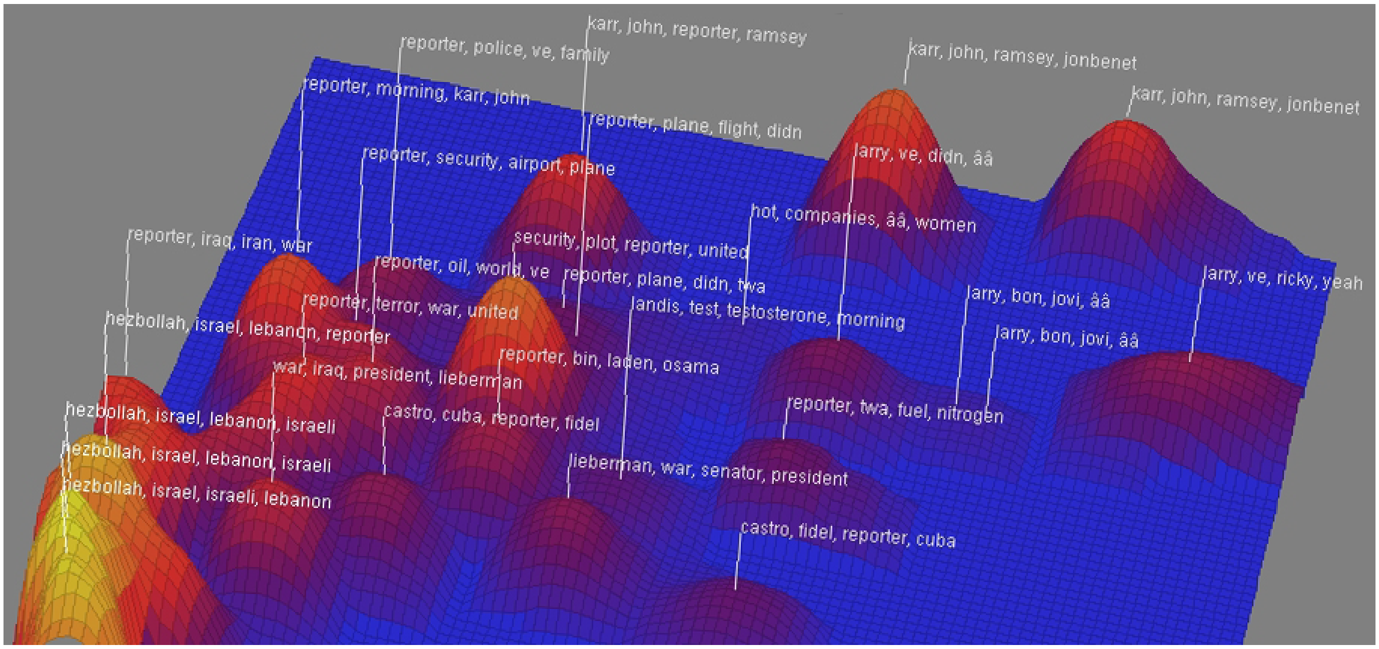


Fig. 8. *CNN* news from Aug. 1 to 24, 2006 $(29,211$ closed-caption documents) in IN-SPIRE.

	Topics (Event Groups)	Precision (%)	Recall (%)
1	Jonbenet Ramsey Murder Case	40.0 (8/20)	72.7 (8/11)
2	2006 Lebanon War	93.3 (14/15)	82.4 (14/17)
3	2006 Transatlantic Aircraft Plot	70.0 (7/10)	77.7 (7/9)
4	Cuban transfer of presidential duties	80.0 (4/5)	80.0 (4/5)
5	Landis' drug use in the 2006 Le tour de France	66.7 (2/3)	100.0 (2/2)
6	Joe Lieberman vs. Ned Lamond in Democratic Party Senate election	75.0 (6/8)	85.7 (6/7)
7	The kidnap of Fox journalists Olaf Wiig and Steve Centanni	66.7 (2/3)	66.7 (2/3)
8	U.N. Security council demands Iran suspend Uranium enrichment	66.7 (2/3)	100.0 (2/2)
	Average	69.8	83.2

TABLE 1
The scores of *precision* and *recall* from the experiment.

were also killed in a dusk attack in the port of **Haifa**." A snippet of this event is shown in Fig. 5. (4) On *Aug.9*, "in the eastern **Bekaa Valley** five people were reported killed and two feared dead after an **Israeli** airraid." (5) *On Aug.10*, "**Condoleezza Rice** formally explains the resolution plan of the U.N. for reconciliation between Lebanon and **Israel**."; (6) On *Aug.14*, "The Former **Israeli** prime minister **Ariel Sharon**'s health condition gets worse because of a new finding of pneumonia, and he is still in a **coma**."

The experiment results demonstrated that EventRiver provides an effective and efficient event-based text analysis approach that successfully maps a text corpus to its driving events and helps users discover narrative arcs of the long term stories. The meaningful real life events discovered in the experiment also indirectly proved the effectiveness of the document generation model on which the event-based text analysis establishes itself. The analysis offered by EventRiver can scale up to much larger text collections with temporal references and can give overviews even when human analyses (demonstrated via Wikipedia here) have not been performed. Further, the temporal analysis approach provided in this paper is unique and gives EventRiver capabilities other tools do not have.

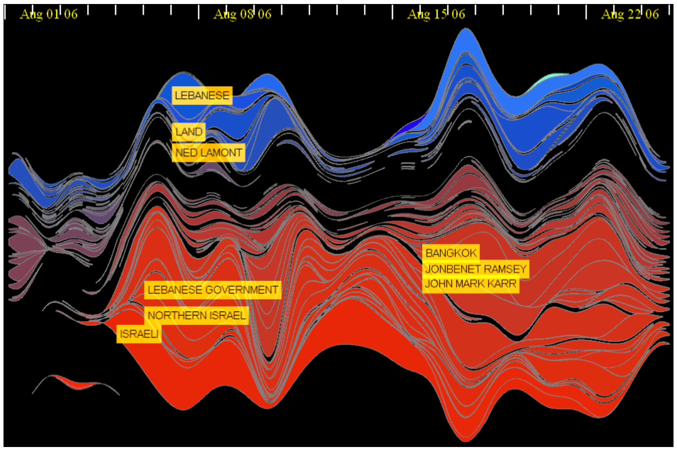


Fig. 9. *CNN* news from Aug. 1 to 24, 2006 $(29,211$ closed-caption documents) in LensRiver. Each current represents a keyword. The x-axis is the time line. The width of a current along the y-axis indicates the strength of the keyword.

7 Preliminary User Feedbacks

We report user feedbacks from a preliminary user test of EventRiver. This test was conducted to evaluate the effectiveness and efficiency of EventRiver in supporting the event-related tasks as a human-centered visual analytics solution. We performed this test by comparing EventRiver with LensRiver [12] (see Fig. 9), which is

Story 1.	**Fidel Castro and Cuba Situation**
HE 1	Fidel Castro meets General Abizaid in Cuba.
HE 2	Juanita Castro, sister of Fidel Castro took an interview about Fidel's health condition.
HE 3	Raul Castro takes over the power from Fidel Castro.
Story 2.	**2006 Transatlantic Aircraft Plot**
HE 1	The Homeland Security secretary Michael Chertoff talked about the London airport plot and explained the new carry-on rules of flights.
HE 2	Some terrorists tried to carry liquid explosives disguised as gel, toothpaste, etc. onto the flights from London Heathrow Airport to US and Canada.
HE 3	Some Pakistani arrested for the London Heathrow airport terrorist plot were originally connected to Osama Bin Laden.

TABLE 2
User Study Task 1 - Hypothetical Event Evaluation

Story 1.	**The 2006 Lebanon War (Lebanon vs. Israel)**
E 1	The Former Israeli prime minister Ariel Sharon's health condition got worse because of a new finding of pneumonia, and he was still in coma.
E 2	Israel expanded the military operations in southern Lebanon trying to pushing their troops into Latini River.
E 3	Condoleezza Rice formally explains the resolution plan of U.N. for reconciliation between Lebanon and Israel.
E 4	Israel's military action in the Bekaa Valley was a violation of the cease-fire agreement with Lebanon.
Story 2.	**Jonbenet Ramsey Murder Case (the suspect John Karr)**
E 1	John Karr took the DNA test in Boulder, CO.
E 2	John Karr's arrest in Bangkok, Thailand.
E 3	John Karr was sent to Boulder, CO.

TABLE 3
User Study Task 2 - Event Ordering

		EventRiver	LensRiver
Hypotheses Evaluation	Time (min)	4.35	5.78
	Correctness (%)	62.50	56.25
	Confidence (1-5)	3.13	2.48
Event Ordering	Time (min)	5.8	7.1
	Correctness (%)	75.00	49.75
	Confidence (1-5)	3.75	2.75
Preference	Usefulness (1-5)	4.2	2.4
	Ease of Use (1-5)	3.9	2.8
	Awareness of Context (1-5)	4.3	1.9

TABLE 4
Measures on average of the user study.

a Keyword-Tracing Technique (refer to Section 2). Our assumption was that the EventRiver was more effective in helping users discover major events and learn their semantic contents and evolution than Keyword Tracing Techniques such as LensRiver.

To form a comparable study, labels and snippets were provided in both EventRiver and LensRiver. The shoebox was disabled to make sure that users learned the semantics without reading the documents. CNN news from Aug. 1 to 24 in 2006 was used in the formal test while CNN news from Sept.1 to 15 in 2006 was used in the training. Each document is characterized by the same keyword vector in both systems. Fig. 1 and Fig. 9 show the test dataset in EventRiver and LensRiver respectively.

The test was a within-subjects, balanced user study. Twelve graduate students majoring in computer science participated in this test. All subjects claimed that they were unfamiliar with or forgot about the details about the news during Aug. 2006. The subjects took the test one by one on the same desktop PC.

The procedure of this test was indicated as follows. Each subject worked through two sections. In each section, a subject was asked to complete the same set of tasks with either EventRiver or LensRiver. Half of the subjects worked with EventRiver in the first section and LensRiver in the second section. Another half worked in reversed order. We asked the subjects to use both systems to learn their preference. In our task performance statistics, only the results from the first section were calculated to avoid the case that what a subject learned from the first section helped him/her in the second section. Each section started with a 10 minutes training period in which the instructor introduced the interface and the interactions to the subject and the subject freely explored the tool using the training dataset. The instructor answered questions from the subject during the training.

The formal tasks followed the training. It contained two tasks. The first task tested how the visualization helped users discover significant events in the major stories that drive the text collection and learn their semantics. As shown in Table 2, two groups of hypothetical events were given and the subject was asked to judge whether each hypothetical event was true or false. The subject was asked to rate his/her confidence about the judgment on a 1 (low confidence) to 5 (high confidence) scale. The second task tested how the visualization helped users learn event evolution. As shown in Table 3, two groups of associated real life events were given and the subject was asked to number the chronological order of the events within each group. The subject was again asked to rate his/her confidence about the answers on the same scale. Note that all the events used in the tasks were based on news events that were listed in the weekly top

The total time the subject used to finish each task was manually recorded by the instructor. After the test, the subjects were asked to complete a post-test questionnaire on their preference to EventRiver and LensRiver with regard to their usefulness, ease of use, and awareness of context. Scales of 1 (low preference) to 5 (high preference) were used for the measure. Free-style comments were also collected.

The results of this test are reported in Table 4. The results showed that EventRiver had advantages over LensRiver for the given tasks with regard to time efficiency, correctness, user confidence, and user preference on usefulness, ease of use, and awareness of context. Users commented that it was easy to find keywords associations and track event evolution using EventRiver. By contrast, users found that it was hard to find keyword associations from LensRiver, especially when the keywords under inspection had a relatively long life span.

8 Conclusion and Future Work

In conclusion, EventRiver advances analysis on text collections with temporal references in that:

- EventRiver exemplifies the integration of novel analytical components with an expressive visual representation and interaction methods to visualize text collections in support of event-related analytical tasks. The whole approach is established upon an event-based perspective so that the *world view gap* can be narrowed to a great extend. Its effectiveness and efficiency have been evaluated by the case studies, the experiments, and the preliminary user test.
- EventRiver employs a novel event-based text analysis approach upon a streaming data model. Besides computational efficiency, this dynamic data process mechanism brings EventRiver the potential to be applied to real-time text stream applications in the future.
- EventRiver employs a visualization that reveals the narrative arcs of a text collection in terms of events and offers intuitive visual clues with ready accessibility to full text. It also provides a multi-resolution visual exploration pipeline from long term stories, events, to full text of a document, which allows users to explore the text collection in a scalable and manageable way.

In the future, we are going to extend EventRiver in the following directions:

- to allow the individual user to influence the determination of importance of the events by feeding personal preferences into the system, particularly into the analytic part, and thus to further narrow the world view gap;
- to scale to larger text collections by enabling multi-resolution temporal granularity and allowing user to interactively change the granularity (for example, the particle time zone can be dynamically adjusted by users during their tasks), and design corresponding visualizations;
- to design a dynamic visualization mechanism, which will seamlessly integrate with the current dynamic data procession mechanism, so that EventRiver can be applied to real-time text streams.

In addition, EventRiver will be strengthened to support visual sense making better by integrating more analytical components such as opinion analysis. We also plan to support comparative analysis of multiple text collections using EventRiver.

Acknowledgments

This material is based upon work supported by the U.S. Department of Homeland security under Grant Award Number 2008-ST-108-000002. We also thank Dr. Hangzai Luo and Dr. Jianping Fan for their help on data collection and preprocessing.

Disclaimer: the views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Department of Homeland Security.

References

[1] C. Aggarwal, J. Han, J. Wang, and P. Yu, "A framework for clustering evolving data streams," *VLDB*, pp. 81–92, 2003.

[2] C. Albrecht-Buehler, B. Watson, and D. A. Shamma, "Visualizing live text streams using motion and temporal pooling," *IEEE Computer Graphics and Application*, vol. 25, no. 3, pp. 52–59, 2005.

[3] J. Allan, Ed., *Topic Detection and Tracking, Event-based Information Organization*. Kluwer Academic Publishers, 2002.

[4] J. Allan, J. Carbonell, G. Doddington, J. Yamron, and Y. Yang, "Topic detection and tracking pilot study: Final report," in *Proc. of Broadcast News Transcription and Understanding Workshop*, 1998, pp. 194–218.

[5] R. Amar. and J. Stasko, "Knowledge task-based framework for design and evaluation of information visualizations," in *Proc. IEEE Symposium on Information Visualization*, 2004, pp. 143–149.

[6] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*. New York: Addison Wesley, 1999.

[7] J. Donath, K. Karahalios, and F. B. Viégas, "Visualizing conversation," in *HICSS '99: Proc. of the Thirty-Second Annual Hawaii International Conference on System Sciences*, vol. 2, 1999, p. 2023.

[8] M. Dubinko, R. Kumar, J. Magnani, J. Novak, P. Raghavan, and A. Tomkins, "Visualizing tags over time," in *WWW '06: Proc. of the 15th international conference on World Wide Web*, 2006, pp. 193–202.

[9] C. Erten, P. Harding, S. Kobourov, K. Wampler, and G. Yee, "Exploring the computing literature using temporal graph visualization," in *Conference on Visualization and Data Analysis*, 2004, pp. 45–56.

[10] D. Fisher, A. Hoff, G. Robertson, and M. Hurst, "Narratives: A visualization to track narrative events as they develop," in *Proc. of IEEE Symposium on Visual Analytics Science and Technology*, 2008, pp. 115–122.

[11] G. Fung, J. Yu, H. Liu, and P. Yu, "Time-dependent event hierarchy construction," in *Proc. of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2007, pp. 300–309.

[12] M. Ghoniem, D. Luo, J. Yang, and W. Ribarsky, "Newslab: Exploratory broadcast news video analysis," in *Proc. of IEEE Symposium on Visual Analytics Science and Technology*, 2007, pp. 123–130.

[13] S. Guha, R. Rastogi, and K. Shim, "Rock: A robust clustering algorithm for categorical attributes," in *ICDE*, 1999, pp. 512–521.

[14] S. Havre, E. Hetzler, P. Whitney, and L. Nowell, "Themeriver: Visualizing thematic changes in large document collections," *IEEE TVCG*, vol. 8, no. 1, pp. 9–20, 2002.

[15] E. Hetzler, V. Crow, D. Payne, and A. Turner, "Turning the bucket of text into a pipe," in *Proc. IEEE Symposium on Information Visualization*, 2005, pp. 12–18.

[16] J. Kleinberg, *Temporal Dynamics of On-Line Information Streams*. Springer, 2006.

[17] J. Leskovec, L. Backstrom, and J. Kleinberg, "Meme-tracking and the dynamics of the news cycle," in *Proc. of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2009, pp. 497–506.

[18] H. Luo, J. Fan, J. Yang, W. Ribarsky, and S. Satoh, "Mining large-scale news video databases for supporting knowledge visualization and intuitive retrieval," in *Proc. of IEEE Symposium on Visual Analytics Science and Technology*, 2007, pp. 107–114.

[19] Q. Mei, "Discovering evolutionary theme patterns from text: an exploration of temporal text mining," in *Proc. of the 11th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2005, pp. 198–207.

[20] L. Nowell, E. Hetzler, and T. Tanasse, "Change blindness in information visualization: A case study," in *Proc. IEEE Symposium on Information Visualization*, 2001, pp. 15–22.

[21] S. Rose, S. Butner, W. Cowley, M. Gregory, and H. Walker, "Describing story evolution from dynamic information streams," in *Proc. of IEEE Symposium on Visual Analytics Science and Technology*, 2009, pp. 99–106.

[22] WikiPedia, http://en.wikipedia.org/wiki/Main_Page, http://en.wikipedia.org/wiki/Timeline_of_Military_Operations_in_the_2006_Lebanon_War, http://en.wikipedia.org/wiki/2006_Lebanon_War, http://en.wikipedia.org/wiki/John_Mark_Karr, http://en.wikipedia.org/wiki/Timeline_of_the_2006_transatlantic_aircraft_plot, http://en.wikipedia.org/wiki/Floyd_Landis, http://en.wikipedia.org/wiki/Democratic_Party_primary,_Connecticut_United_States_Senate_election,_2006, http://en.wikipedia.org/wiki/2006_Cuban_transfer_of_presidential_duties, http://en.wikipedia.org/wiki/2006_Fox_journalists_kidnapping, http://en.wikipedia.org/wiki/Nuclear_program_of_Iran.

[23] J. Wise, J. Thomas, K. Pennock, D. Lantrip, M. Pottier, A. Schur, and V. Crow, "Visualizing the non-visual: spatial analysis and interaction with information from text documents," in *Proc. IEEE Symposium on Information Visualization*, 1995, pp. 51–58.

[24] Y. Yang, T. Pierce, and J. Carbonell, "A study on retrospective and on-line event detection," in *Proc. of the 21st annual international ACM SIGIR conference on research and development in information retrieval*, 1998, pp. 28–36.