TopicPanorama: A Full Picture of Relevant Topics

Xiting Wang, Shixia Liu, Junlin Liu, Jianfei Chen, Jun Zhu, and Baining Guo

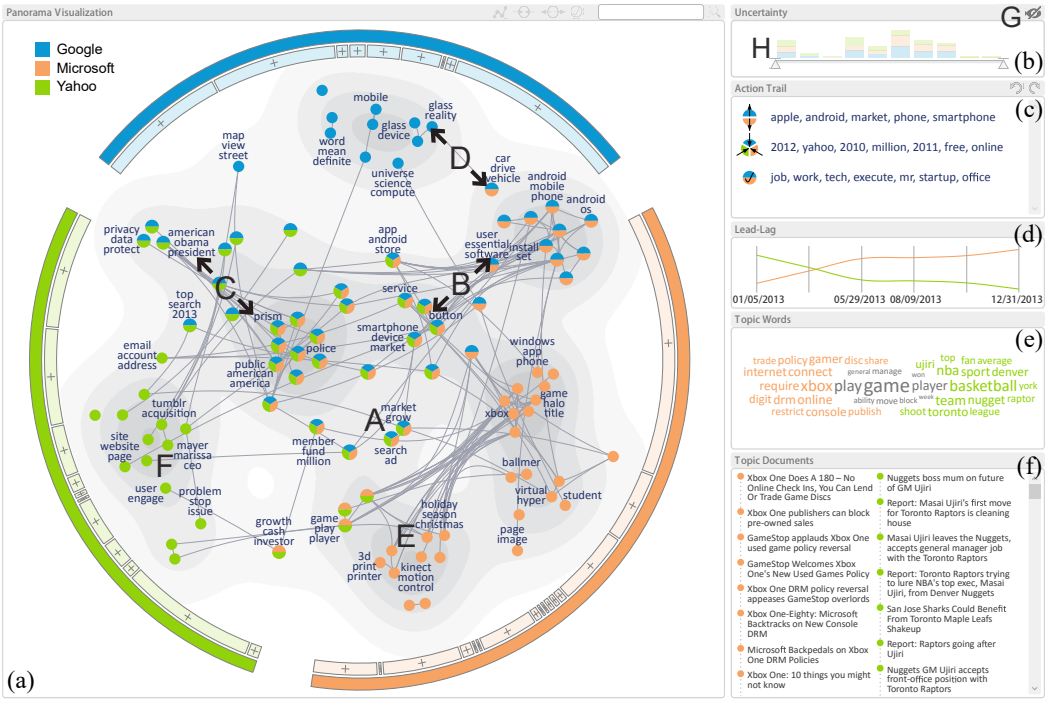


Figure 1: A full picture of topics related to Google, Microsoft, and Yahoo: (a) LOD visualization to examine the matched graph; (b) uncertainty filter; (c) match modification trail; (d) lead-lag analysis; (e) topic word; (f) document.

Abstract—This paper presents a visual analytics approach to analyzing a full picture of relevant topics discussed in multiple sources, such as news, blogs, or micro-blogs. The full picture consists of a number of common topics covered by multiple sources, as well as distinctive topics from each source. Our approach models each textual corpus as a topic graph. These graphs are then matched using a consistent graph matching method. Next, we develop a level-of-detail (LOD) visualization that balances both readability and stability. Accordingly, the resulting visualization enhances the ability of users to understand and analyze the matched graph from multiple perspectives. By incorporating metric learning and feature selection into the graph matching algorithm, we allow users to interactively modify the graph matching result based on their information needs. We have applied our approach to various types of data, including news articles, tweets, and blog data. Quantitative evaluation and real-world case studies demonstrate the promise of our approach, especially in support of examining a topic-graph-based full picture at different levels of detail.

Index Terms—Topic graph, graph matching, graph visualization, user interactions, level-of-detail.

1 Introduction

In real-world communications, relevant topics concerning events (e.g., the Ebola outbreak) or organizations (e.g., IT companies) are often heavily discussed in multiple sources, such as news, blogs, or micro-blogs. These sources share a number of common topics while also having their own distinctive topics, which together form a full picture of the relevant topics. Here, a full picture is a comprehensive visual summary that presents different aspects of relevant topics discussed in multiple sources. Several recent studies have suggested that a better understanding of the full picture provides new insights for decision-making [1].

However, users often take great pains to develop a comprehensive understanding of the whole story. They have to repeatedly switch back and forth from one source to another in order to gradually form a clear picture of given

- *X. Wang, S. Liu, J. Liu, J. Chen, and J. Zhu are with Tsinghua University. E-mail: {wang-xt11,liujl12,chenjian14}@mails.tsinghua.edu.cn; {shixia,dcszj}@tsinghua.edu.cn.*
- *B. Guo is with Microsoft Research. E-mail: bainguo@microsoft.com.*

topics. To facilitate such an analysis, it is important to be able to gather separate pieces of information about these topics, which are scattered among different sources, and reconstruct the full picture.

Topic graphs, in which a node represents a topic and an edge represents a type of correlation between topics, are very important for providing an efficient but comprehensive understanding of topics of interest through correlation [2], [3]. As a result, a straightforward way of developing a full picture is to merge all the data collected from different sources and then utilize the correlated topic model (CTM) [2] to build a topic graph on the merged data. However, there are two drawbacks to this approach. First, different text corpora contain text strings of different lengths and language usages. For example, news articles are long and well formed, while tweets are short and noisy. This makes it difficult to use a unified topic graph generation method to build a single topic graph that fits each corpus well. Second, even when document lengths and language usages are similar, different corpora may have their own unique topics. Direct use of the topic graph construction method (with the same parameters) on all data may fail to model the diversity across different corpora because the model uses a common set of topics to model all the data [1]. We reported on the deficiency of using a unified topic graph construction method in Sec. 8.1 of our previous work [4].

To solve these issues, we have developed an interactive visual analytics tool called TopicPanorama. The main objective of TopicPanorama is to help users analyze common topics within the context of each individual topic graph. Fig. 1 shows an application of TopicPanorama in analyzing the full picture of topics related to three IT companies, Google, Microsoft, and Yahoo. Several common topics and the distinctive topics of each corpus are identified in Fig. 1(a). For example, some government-related topics were referred to by the three companies and some of them were shared between Google and Yahoo (**C**). Kinect-related topics were most often mentioned in the Microsoft corpus (**E**). Starting from this overview, users can zoom in to find the topics of interest.

Technically, TopicPanorama aims to consistently integrate multiple topic graphs to support iterative, progressive topic graph synthesis and analysis.

First, we have developed a multiple graph matching algorithm to derive consistent matching results among multiple topic graphs. Our algorithm is based on graph edit distance [5], one of the most widely used pairwise graph matching metrics. The major feature of our proposed graph matching algorithm is that it jointly optimizes related pairwise matches instead of performing a sequence of pairwise matches with a reference graph, which may introduce inconsistency.

Second, we have developed a match modification algorithm to allow users to modify the matching results. We combine metric learning with feature selection to reduce user effort when modifying the matching results. In contrast to our previous rule-based method [4], our algorithm automatically learns the costs and then updates the relevant matching results. By combining metric learning and feature selection with the incremental Hungarian algorithm [6], we allow users to interactively modify the matching results. Experiments on three real-world datasets show that our algorithm is at least 20% more effective than the rule-based method and can be done in less than 0.4 seconds.

Third, we have developed an LOD-based visualization for understanding the matched graph, which combines a radial icicle plot with a density-based graph visualization. With this combination, TopicPanorama enables users to examine both the overarching concepts and finer details in each corpus. We employ a constrained spring model in the layout algorithm to balance both readability and stability. Moreover, a set of rich interactions is provided to allow users to understand the full picture from multiple perspectives.

This paper is an extension of our previous work [4], where a full picture was generated by integrating a graph-edit-distance-based matching algorithm with a density-based graph visualization. In this paper, we instead focus on the analytical lifecycle of TopicPanorama, including

- **A more effective interactive match modification method** to reduce user effort when modifying the graph matching results.
- **A constrained spring model** that manages the trade-off between readability and stability.
- **A set of rich interactions** (e.g., lead-lag analysis) to help users analyze and understand the full picture.

2 Related Work

2.1 Graph Matching

In this section, we only review error-tolerant graph matching methods since they can flexibly accommodate the differences between graphs by relaxing matching constraints. Such a relaxation is useful for topic graph matching, which often matches related graphs rather than exactly the same ones. The widely used error-tolerant graph matching method is based on the edit distance of graphs [5]. The basic idea is to measure the structural difference of graphs by the number of edit operations needed to transform one graph into another.

Directly using these pairwise matching methods to match multiple graphs may introduce inconsistency [7]. Simply removing the inconsistent results may lead to suboptimal results [4].

To tackle this issue, there has been some effort to match multiple graphs. Williams et al. [8] presented a proof-of-concept for multiple graph matching. They adopted a Bayesian framework to construct an inference matrix and used it to measure the mutual consistency of multiple graph matching. The framework looks promising, but no solver is provided. To compute a representative of a set of graphs, a common labeling algorithm [9] has been developed. The algorithm learns common labels through a consistent multiple isomorphism. It can find consistent, common labeling among multiple graphs. However, it assumes that each graph has the same number of nodes.

Yan et al. [7] provided a multiple graph matching algorithm based on the pairwise matching solver and constrained integer quadratic programming (IQP). However,

IQP is known to be computationally expensive, which means this algorithm is not applicable for real-time interactions. Furthermore, it may fail to infer matching relationships among non-common parts of graphs [4]. Compared with [7], our method addresses the bottleneck of computation and missing matches. We formulate multiple graph matching as a unified optimization approach based on graph edit distance and the incremental Hungarian algorithm [6]. Accordingly, an effective match modification algorithm is developed based on metric learning and feature selection. We have also designed an LOD-based visualization to better understand and analyze the matched graph from multiple perspectives.

2.2 Visual Graph Comparison

Visual graph comparison aims to analyze the similarities and differences between graphs [10]. A number of graph comparison methods have been proposed. Among them, the most closely related is that of Hascoët et al. [11], who developed an interactive graph matching tool that combines node-link diagrams with graph matching techniques. A heuristic rule based on the layout positions of nodes was used to approximately match nodes from different graphs. Although the matching method is simple and easy to implement, it may introduce more errors/uncertainties since the node position is not a reliable metric for matching nodes. The adopted layout method does not distinguish between common and distinctive topics perceptually. Compared with this method, TopicPanorama consistently integrates multiple topic graphs together to form a full picture of relevant topics, based on their content and relationships with each other. With this, TopicPanorama enables users to easily see the matching result, including the matched graph as well as individual ones.

2.3 Topic Visualization

Topic visualization can be classified into two categories: dynamic topic visualization and static topic visualization [12], [13]. Most existing dynamic topic visualizations focus on analyzing evolving topics based on a river metaphor. For example, Havre et al. [14] made an initial effort to employ a river metaphor to convey evolving topics over time. TIARA [15] tightly integrates the stacked graph visualization with the LDA model to illustrate topic evolution patterns over time. Visual Backchannel [16] was developed to visualize keyword-based topics that are extracted from tweets. ParallelTopics [17] employs ThemeRiver to illustrate topic evolution over time and parallel coordinate plots to convey the probabilistic distribution of a document on different topics. TextFlow [18] and RoseRiver [19] leverage Sankey diagrams to visually convey topic merging and splitting relationships over time. Inspired by the visual representation of TextFlow, ThemeDelta [20] was developed for discovering how trend keywords converge into topics and diverge into different topics, as well as identifying temporal trends, clustering, and significant shifts in topics.

Several approaches have also been developed to help analyze evolving topics on social media [21], [22], [23], [24]. For example, a visual analytics system was designed by Xu et al. [23] to allow users to understand the dynamic competition relationships among topics on social media. Zhao et al. [24] developed FluxFlow to reveal and analyze anomalous information as it spreads on social media. The aforementioned approaches focus on the visual exploration of evolving topics from a single source. In contrast to these approaches, our method aims to provide a full picture of relevant topics from multiple sources.

Static topic visualizations leverage word lists or word clouds to visualize topic models. For example, Chaney and Blei [25] employed word lists to illustrate the hidden structure discovered by a topic model. HierarchicalTopics [26] hierarchically organizes the learned topics and thus can represent a large number of topics without being cluttered. Serendip [27] supports multi-level serendipitous discovery in text corpora, including the corpus, passage, and word levels. Most static topic visualizations aim to provide an overview of the topics extracted from one text corpus. Our method provides a full picture of relevant topics from multiple corpora and allows users to examine common topics among corpora as well as distinctive topics of each corpus.

Other methods related to ours are FacetAtlas [28] and SolarMap [29]. They also adopt the density-based graph visualization to represent the multifaceted relationships of documents within or across document clusters. However, they might fail to easily distinguish the common and distinctive topics across multiple corpora if we directly employed them in TopicPanorama.

Recently, several methods have been introduced that analyze and compare content in multiple corpora [30], [31], [32], [33]. Oelke et al. [32] have developed a closely related method that aims to extract and reveal distinctive and common topics in order to compare multiple text corpora. In addition to supporting this comparison function, TopicPanorama also allows users to examine the correlations between topics and supports navigation of a large number of topics.

3 TopicPanorama

3.1 Task Analysis

We designed TopicPanorama through a participatory design session with several experts, including two public relations managers (R1, R2), two journalists (J1, J2), and two professors who major in media and communications (P1, P2). The experts usually form an overall picture by manually analyzing all the available documents, which is very time-consuming and requires a great deal of expertise. They need a toolkit that allows them to effectively conduct analysis on a much larger dataset and can greatly advance their understanding of a full picture of the relevant topics of interest.

In the design session, we focused on probing the experts' analysis needs by asking the following questions:

- How do you create a full picture of the relevant topics?
- How do you explore and understand the full picture?
- How do you use the full picture in your work?

Accordingly, we identified the following high-level tasks to guide the design of TopicPanorama:

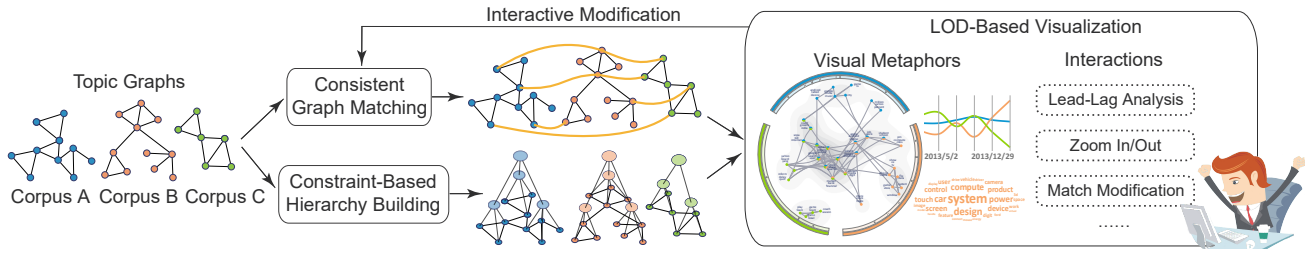


Figure 2: TopicPanorama overview.

T1 - obtaining an overview of relevant topics. All experts expressed the need to smoothly navigate a full picture when analyzing relevant topics that are discussed in multiple sources, from the high-level topics to the detailed documents. They stated that they would benefit from a toolkit that can effectively integrate two or three sources. This is consistent with the conclusion of previous experiments, namely that about four objects can be tracked in visual comparison [34].

T2 - examining the common topics and distinctive topics of each source. When analyzing a full picture, the experts often compare topics across sources, including the common and distinctive ones. As a result, the experts require the ability to examine common topics across multiple corpora as well as the distinctive topics of each corpus in the same view.

T3 - examining the correlations between topics. All the experts wanted to understand the correlations between topics, especially the correlations between the common topics and distinctive topics of each source, because such correlations help them find information of interest more quickly. For example, professor P1 commented, "When analyzing the media framing of events, I need to understand how two discursive spaces (i.e., mass media and grass roots) interact with each other."

T4 - exploring the full picture at different levels of granularity. In many applications, a source may contain hundreds or even thousands of topics. Quickly getting an overview of these topics and then zooming into the detailed content gradually is a very important step for the experts to perform various analysis tasks. For example, R1 said, "In my daily work, I often process multiple sources that contain thousands of topics. A toolkit that efficiently organizes a large number of topics could help me a lot."

T5 - analyzing the temporal patterns of the matched topics. When approaching a specific topic of interest, experts often require examining the idea propagation among multiple sources. In our case, three of the experts expressed the need to analyze the temporal lead-lag changes of the matched topics. For example, journalist J1 said, "When comparing event propagation among different media sources, identifying the leading source is very useful for me to find breaking news to report."

T6 - customizing the full picture based on user needs. In real-world applications, given several sources, the experts often need different full pictures for different tasks. As a result, they requested the ability to tailor the full picture based on their own information needs. For example, when analyzing the Ebola outbreak, R1 cared more about game-related topics, so she hoped to unmatch the incorrect matches.

3.2 System Overview

To help users better perform the aforementioned tasks, TopicPanorama contains the following features:

- The ability to leverage a topic graph to represent each source and hierarchically organize the graph (**T3**, **T4**);
- Able to match multiple topic graphs to form a full picture (**T1**);
- An LOD visualization that places the common parts near the area of each related source and the distinctive parts in the corresponding area of each source (**T2**);
- Rich interactions such as lead-lag analysis and interactive match modification (**T5**, **T6**).

Accordingly, TopicPanorama consists of four major modules: graph matching, hierarchy building, a visualization module, and an interaction module (Fig. 2). Given several topic graphs, the graph matching module generates consistent matching results among them. To handle large topic graphs effectively, the hierarchy-building module generates a topic hierarchy based on the constraint-based Bayesian Rose Tree (BRT) model [35]. The graph matching results and the topic hierarchies are then fed to the visualization module. The visualization combines a radial icicle plot with a density-based graph visualization to illustrate the graph matching results. Users can interact with the generated visualizations for further analysis. For example, the user can modify one of the matching results, then TopicPanorama will incrementally update the matching results.

4 Consistent Graph Matching

Graph edit distance is a widely used metric in graph matching algorithms to match two graphs [5]. It measures the structural differences of graphs by the number of edit operations (e.g., *insertions*, *deletions*, and *substitutions* of nodes/edges) needed to transform one graph into another. Given two graphs $G_1 = (\mathcal{V}_1, \mathcal{E}_1)$ and $G_2 = (\mathcal{V}_2, \mathcal{E}_2)$, where $\mathcal{V}_1$, $\mathcal{V}_2$ are the node sets and $\mathcal{E}_1$, $\mathcal{E}_2$ are the edge sets, we denote the matching between them as $f_{G_1G_2}$. The graph edit distance between G_1 and G_2 is defined as the minimal cost of all sequences of edit operations between them:

$$dis(G_1, G_2) = \min\ c(f_{G_1G_2}), \qquad c(f_{G_1G_2}) = \sum_{o_i} c(o_i), \tag{1}$$

where $c(f_{G_1G_2})$ is the edit cost that matches G_1 to G_2 and $c(o_i)$ denotes the cost function of the edit operation o_i.

Given N graphs, a natural extension of the pairwise matching method for multi-graph matching is to summarize the graph edit distance of each pairwise matching:

$$dis(G_1, G_2, ..., G_N) = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} dis(G_i, G_j). \tag{2}$$

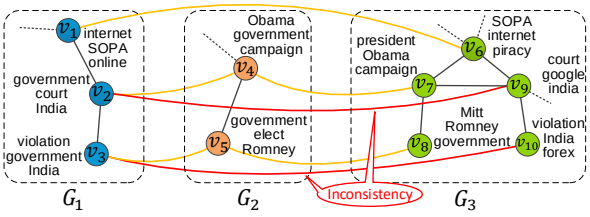


Figure 3: Inconsistency caused by directly applying pairwise matching. The node matches derived from the matching results of $f_{G_1G_2}$ and $f_{G_2G_3}$, $v_2 \mapsto v_4$ and $v_4 \mapsto v_7$, are inconsistent with the direct matching of $f_{G_1G_3}$, $v_2 \mapsto v_9$.

However, this formulation may introduce inconsistency into the matching results. As shown in Fig. 3, $f_{G_1G_2}$ matches v_2 to v_4 ($v_2 \mapsto v_4$). Here $v_i \mapsto v_j$ indicates that v_i is matched to v_j. $f_{G_2G_3}$ matches v_4 to v_7. From these two matching results, we note that node v_2 matches node v_7, which conflicts with the direct matching result of $f_{G_1G_3}$ ($v_2 \mapsto v_9$). Similar inconsistency is observed in the matches among the nodes v_3, v_5, v_8, and v_{10}.

To solve this issue, in our previous work, we develop a consistent graph matching method that minimizes the cost of all pairwise graph matchings, with the constraint that all node matching relationships are transitive [4]. By ensuring such transitive relationships (consistency constraint), the proposed method derives globally consistent matching results across multiple graphs. Mathematically, the proposed graph matching method is formulated as

$$\begin{aligned} &dis(G_1,...,G_N) = \min\ c(f_{G_1...G_N}),\ c(f_{G_1...G_N}) = \sum_{i=1}^{N-1}\sum_{j=i+1}^{N} c(f_{G_iG_j}) \\ &s.t.\ v_l \mapsto v_m, v_m \mapsto v_n \Rightarrow v_l \mapsto v_n \\ &\forall G_i, G_j, G_k \in \{G_1, G_2, ..., G_N\},\ \forall v_l \in \mathcal{V}_i, \forall v_m \in \mathcal{V}_j, \forall v_n \in \mathcal{V}_k, \end{aligned} \tag{3}$$

The graph matching problem is NP-hard in general [7]. It becomes even harder when the consistency constraints are considered. To solve this problem, we introduce the concept of *meta-graph*. The meta-graph is constructed by merging the matched nodes (or edges) as a meta-node (or meta-edge). The meta-graph is comprised of the consistently matched results of N graphs that contain both the common topics and distinctive topics of each topic graph. Fig. 4 shows an example of a meta-graph $M(G_1G_2)$ for the matching $f_{G_1G_2}$. When matching a meta-graph and a normal graph, we define the cost of each edit operation of a meta-node as the sum of the cost that matches each node in the meta-node to the normal node. Accordingly, the cost function of matching a meta-graph and a normal graph is

$$c(f_{M(G_1...G_{N-1})G_N}) = \sum_{i=1}^{N-1} c(f_{G_iG_N}). \tag{4}$$

By leveraging this cost function, Eq. (3) is simplified as:

$$\begin{aligned} c(f_{G_1G_2...G_N}) = &\ \textstyle\sum_{i=1}^{N-2}\sum_{j=i+1}^{N-1} c(f_{G_iG_j}) + \sum_{i=1}^{N-1} c(f_{G_iG_N}) \\ = &\ c(f_{G_1G_2...G_{N-1}}) + c(f_{M(G_1G_2...G_{N-1})G_N}). \end{aligned} \tag{5}$$

5 Interactive Match Modification Model

Although the proposed graph matching method can successfully generate optimal matching among multiple graphs, it may still be imperfect. Furthermore, different users may have different information needs. Thus one graph-mining model cannot meet all the possible requirements. To compensate for this, TopicPanorama allows users to interactively modify the graph matching result.

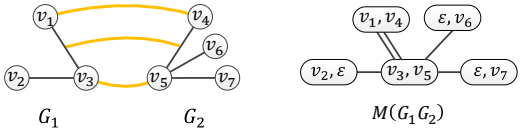


Figure 4: Meta-graph example. ε represents a null node.

5.1 Problem Formulation

In TopicPanorama, a user can provide feedback on whether two topics should be matched or not (user-provided constraints) based on his/her knowledge or information needs. These operations can be regarded as a node substitution in graph matching. Thus, to support these operations, we formulate such user feedback as the node substitution cost and modify it in the interactive match modification algorithm. In particular, our interactive match modification algorithm consists of two steps:

- Modify the substitution cost based on user-provided constraints.
- Incrementally update the matching results based on the incremental Hungarian algorithm [6].

The second step was introduced in our previous work [4]. Here we focus on the first step.

Previously, we leveraged a rule-based method to change the node substitution costs [4]. In particular, the rule-based method sets the substitution cost of the matched (or unmatched) topics to 0 (or ∞) while keeping other costs unchanged. The method works well when two topics share a set of the same words, which plays an important role in matching them. However, this method assumes that different words in the two topics are not relevant and all the words are equally important for matching two topics, which in many cases is not true. For example, "xbox" and "playstation" are different words, but they are relevant since they both refer to video game consoles.

To solve this problem, we employ **metric learning** to automatically learn node substitution costs. We also leverage several **feature selection** techniques to speed up the algorithm without sacrificing its effectiveness (see Table 2).

5.2 Metric Learning

Distance metric learning learns a distance function between samples with the intention of assigning small distances between similar samples [36]. It aims to satisfy the maximum number of constraints (e.g., similar sample pairs) by considering certain features more important while also incorporating relevance between features. The weight is derived from feature co-occurrence statistics in the given pairs. In our case, the distance is the node substitution cost. The constraints are a set of similar or dissimilar topic pairs. The node substitution cost is defined as a squared Mahalanobis distance metric:

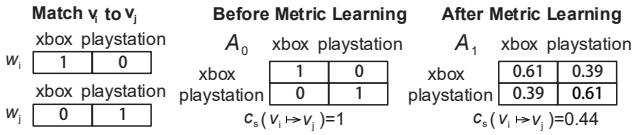


Figure 5: An example of applying metric learning.

$$c(v_i \mapsto v_j) = d_A(\mathbf{w}_i, \mathbf{w}_j) = (\mathbf{w}_i - \mathbf{w}_j)^T A (\mathbf{w}_i - \mathbf{w}_j). \quad (6)$$

Here A is a positive definite matrix to be learned, which represents the relevance between different words (off-diagonal elements) and the importance of the words (diagonal elements). $\mathbf{w}_i = (w_i^1, ..., w_i^n) \in \mathcal{R}^n$ is the word distribution vector of topic v_i, where n is the vocabulary size and w_i^k is the probability that the kth word occurs in topic v_i.

In principle, learning the distance is equivalent to learning matrix A based on the user-provided constraints. Basically, there are two challenges in deriving A. First, the number of constraints is usually small (< 50), which may lead to poor learning performance due to insufficient training samples. Second, the constraint arrives one by one, and thus we need to update A incrementally. To address these problems, we employ an online information-theoretic metric learning algorithm [36]. The major goal of this algorithm is to minimize the loss, which is the difference between the target distance (d_t) at time t and $d_A(\mathbf{w}_i, \mathbf{w}_j)$:

$$A_{t+1} = \underset{A \succ 0}{arg\,min} \left[D(A, A_t) + \eta_t \left(d_t - d_A(\mathbf{w}_i, \mathbf{w}_j) \right)^2 \right]. \quad (7)$$

Here $A \succ 0$ indicates A is a positive definite matrix. The first term ensures that Mahalanobis matrix A is close to the current model A_t (regularization term). The second term indicates that the derived A satisfies the target distance specified at current time t (loss term). $\eta_t > 0$ is a parameter to balance regularization and loss, and $D(A, A_t)$ measures the difference between A and A_t.

To reduce the user's workload, we do not require the user to specify the exact target distance. Instead, we automatically compute it based on user feedback (e.g., match or unmatch). For topics matched (unmatched) by the user, the target distance should be smaller (larger) than $d_{A_t}(\mathbf{w}_i, \mathbf{w}_j)$. Accordingly, we set $d_t = \alpha d_{A_t}(\mathbf{w}_i, \mathbf{w}_j)$, where $0 < \alpha < 1$ for matched topics and $\alpha > 1$ for unmatched topics. In the experiment, we did a grid search on α to find the best parameter that corresponds to the smallest constraint number.

According to the derivation in [36], the resulting solution to the minimization of Eq. (7) is

$$A_{t+1} = A_t - \left(\eta_t (\bar{d}_t - d_t) A_t \Delta\mathbf{w}_t \Delta\mathbf{w}_t^T A_t \right) / \left(1 + \eta_t (\bar{d}_t - d_t) \hat{d}_t \right), \quad (8)$$

where $\Delta\mathbf{w}_t = \mathbf{w}_i - \mathbf{w}_j$, $\hat{d}_t = \Delta\mathbf{w}_t^T A_t \Delta\mathbf{w}_t$, and

$$\bar{d}_t = \left(\eta_t d_t \hat{d}_t - 1 + \sqrt{(\eta_t d_t \hat{d}_t - 1)^2 + 4\eta_t \hat{d}_t^2} \right) / \left(2\eta_t \hat{d}_t \right). \quad (9)$$

An example of applying metric learning is shown in Fig. 5. After a user matches the "xbox" topic with the "playstation" topic, our algorithm learns the relevancy between "xbox" and "playstation," which increases from 0 to 0.39. Accordingly, the corresponding node substitution cost is decreased.

5.3 Feature Selection

If all the words (features) in a text corpus are used in metric learning, the algorithm will be too slow for real-time interactions. To reduce the number of features used without sacrificing performance, we use feature selection to find the most salient features for metric learning.

Existing feature selection methods can be classified into three main categories: filter, wrapper, and embedded methods [37]. The filter method is usually the fastest of the three. Thus we employ this method to speed up metric learning in TopicPanorama. Filter methods measure the relevancy between the candidate features and a class label and then select the features with the highest relevancy as the salient features. In TopicPanorama, we regard the user-provided constraints as the class label. Specifically, topic pair (v_i, v_j) is labelled as 1 (or 0) if v_i and v_j are (or not) matched according to the user. If there are not enough user-provided constraints, the most certain topic matches (with the lowest cost values) calculated by our graph matching algorithm are added into the feature selection process. To calculate the relevancy between a feature and the class label, we compare three commonly used relevancy measures: Pearson correlation, mutual information, and Relief [38]. Experimental results are discussed in Sec. 7.1. Based on the experimental results, we adopt the mutual information method in TopicPanorama.

6 Panorama Visualization

In this section, we describe the visual design, layout algorithm, and interactions of TopicPanorama.

6.1 Visual Design

We have designed the visualization based on the analysis needs of domain experts (Sec. 3.1). In particular, the visualization aims to make it easy to reveal the topic hierarchy, matching results, and uncertainty. The design of each visualization component is introduced as follows.

Topic hierarchy as radial icicle plot. To handle a large corpus with a large number of topics, we build hierarchies for topic graphs based on the constraint-based BRT model [35] with each non-leaf node representing a topic cluster. The BRT model greedily estimates the tree structure with higher marginal likelihood. It can produce trees with an arbitrary branching structure at each node. We utilize the constraint-based algorithm to ensure that the hierarchies built for different graphs have similar structures. More specifically, we generate a hierarchy for each graph and iteratively refine each hierarchy by regarding the hierarchies of the other graphs as constraints. We employ a radial icicle plot (Fig. 7(b)) to display topic hierarchies (**T4**). They are placed on the circumference of the radial layout, with the sector angle encoding the topic number of the corpus.

Matching as density-based visualization. Previous research has shown that a familiar visual representation lowers the cognitive load imposed on a user and benefits the learning process by employing the user's knowledge and experience [39]. Thus, the basic principle of our design is

to employ a familiar visual metaphor when appropriate. We also employ a superposition comparison because this design is more efficient for comparing multiple graphs [40].

Inspired by these two principles, we developed a density-based graph visualization that combines a node-link diagram with a density map to display the nodes at the selected level of the topic hierarchies (Fig. 7(e)). We extract representative nodes for each of the cluster nodes of the selected tree level and assign other non-representative nodes to their closest representative nodes. Accordingly, the node-link diagram is utilized to explain the relationships between representative nodes and the density is employed to illustrate global context consisting of non-representative nodes (**T1**, **T3**). In the node-link diagram, the topic nodes of different corpora are encoded by different colors and the ones in common are represented by a pie chart (Fig. 7) with each of the slices corresponding to the matched corpus (**T2**). The parts in common are placed near the area of each related corpus and the distinctive parts are placed in the area corresponding to the corpus (**T2**). For example, the shared parts of all corpora are placed in the center of the layout area. The common parts of corpora **A** and **B** are placed in between the two related corpus areas (Fig. 7). In each part, the topic nodes under the same parent are placed near each other. This design was positively received by our target users. They all liked the hybrid visualization design, in which both the focus and context are well conveyed.

Uncertainty as glyph. After engaging with the first prototype, users identified some incorrect matches. They expressed the need to be prompted with an explicit request to examine such uncertain matches. This is consistent with the conclusion of previous research that effectively conveying uncertainty in the matching results is very important in the data analysis process [41], [42]. As shown in Fig. 6, we designed a glyph to represent the uncertainty matches with larger cost values. The glyph design was inspired by the iconic symbol called *filled bar and slider*, which is one of the intuitiveness winners for representing attribute uncertainty [43]. In this metaphor, we use the angle between the two sliders to encode the degree of uncertainty. A larger angle indicates a higher degree of uncertainty.

Figure 6: Uncertainty glyph.

6.2 Layout

In this section, we introduce the topic hierarchy layout and density-based graph layout.

Topic hierarchy. Given N corpora, the layout of the radial icicle plot is quite straightforward. We put the unique nodes and common nodes of the N corpora in the middle of the corresponding arc. Other common nodes that are matched to fewer than N corpora are placed on a part of the arc that is close to the related tree nodes in other corpora.

Density-based graph layout. We formulate the layout problem as a constrained spring model that manages the trade-off between readability and stability.

Readability. We use two measures to evaluate the readability of the matched graph layout. The first is a widely used measure suggested by Kamada et al. [44], E_k, which ensures that the distance between two nodes is close to their graph theoretic distance: $E_k = \sum_{v_i \neq v_j} \left[(|p_i - p_j| - l_{ij})^2 / l_{ij}^2 \right]$. Here p_i represents the position of v_i in the matched graph and l_{ij} is the graph theoretic distance between v_i and v_j. The second measure is based on the law of proximity in Gestalt theory, which ensures the nodes are placed close to their corresponding positions on the radial icicle plot: $E_h = \sum_{v_i} |p_i - \hat{p}_i|^2$. Here $\hat{p}_i$ represents the position of v_i on the radial icicle plot. Furthermore, to satisfy the visualization design described above, we define two hard constraints:

- Corpus constraint C_p: the nodes that are shared by the same corpora are placed in the same area (corpus area).
- Cluster constraint C_l: in each corpus area, the nodes under the same parents are placed in the same area.

Stability. To preserve a user's mental map when zooming in/out to explore the full picture, a stability measure is added to assess how stable two adjacent layouts are during any interactions (E_s). According to Misue et al. [45], preserving the orthogonal ordering (i.e., up-down and left-right relationship) of nodes is important for maintaining the mental map when a layout is changed. Inspired by their idea, we examine every pair of nodes and check whether their up-down (left-right) relationship is changed compared with the previous layout. If the relationship changes, the corresponding cost is measured by the distance between v_i and v_j along the y-axis (x-axis).

$$E_s = \sum_{(x_i - x_j)(x_i' - x_j') < 0} (x_i - x_j)^2 + \sum_{(y_i - y_j)(y_i' - y_j') < 0} (y_i - y_j)^2, \qquad (10)$$

where x_i/y_i is the x/y-coordinate of v_i and x_i'/y_i' is the x/y-coordinate of v_i in the previous layout.

Constrained spring model. By combining the above measures and constraints, we formulate the layout problem as a constrained spring model:

$$\min \; E = E_k + \lambda_h E_h + \lambda_s E_s, \; s.t., \; C_p, \; C_l, \qquad (11)$$

where $\lambda_h > 0$, $\lambda_s > 0$ are parameters to balance E_k, E_h, and E_s. In our implementation, λ_h is set to 1 and λ_s is set to 0.5.

The energy function E in Eq. (11) can be locally minimized by using three spring forces. The first is spring force F_k, which is between two nodes as introduced in [44]. This force is used to minimize E_k. The second is spring force F_h, which is between each node and the corresponding node on the radial icicle plot and utilized to minimize E_h. The third spring force F_s is added between two nodes whose up-down or left-right relationship is changed. This force is parallel to the y-axis (x-axis) and used to minimize E_s. To satisfy constraints C_p and C_l, we use Voronoi tessellation to find the corpus area and the area for each cluster. We choose Voronoi tessellation for two reasons [46]: 1) it generates layout areas with an overall aspect ratio converging to 1; and 2) it can generate larger layout areas for a corpus or cluster with more nodes. To solve the constrained spring model, we combine Voronoi tessellation with a force-directed graph layout. In particular, the layout algorithm is divided into three steps.

The first step involves deriving the layout centers of the common and distinctive parts in each corpus, respectively.

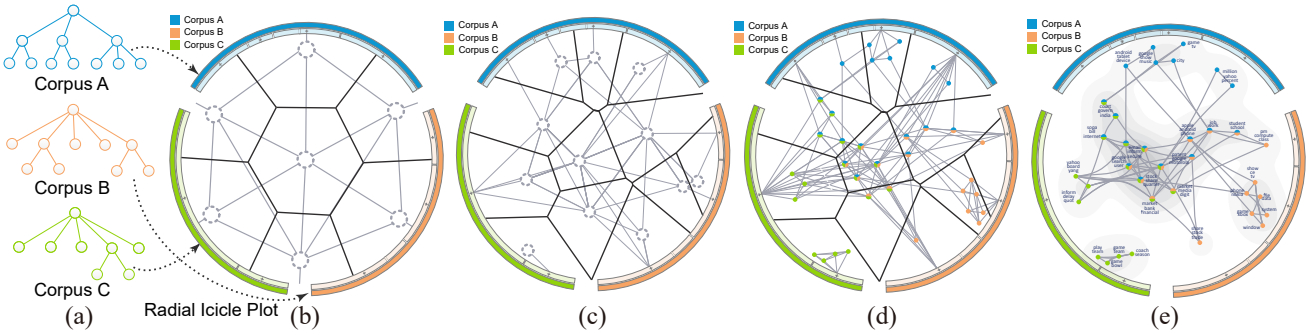


Figure 7: The basic idea of the layout algorithm: (a) topic hierarchies; (b) layout of the common and distinctive parts and computing the corresponding Voronoi tessellation; (c) layout of the cluster nodes of the selected tree level within each generated tessellation cell and computing a new Voronoi tessellation based on the new layout; (d) layout of representative nodes; (e) the final layout.

The basic idea is to employ the force-directed graph layout method to compute the center position of each part. To this end, we build a graph according to the relationships between individual parts as well as the relationships between the radial icicle plots and each part. Next, the graph is laid out using the force-directed model, which provides the center position of each layout area (Fig. 7(b)). Based on the center positions, a Voronoi tessellation is computed to allocate the corpus area. Within each corpus area, we then place the cluster nodes of the selected tree level. Then, based on the calculated node position, we compute another Voronoi tessellation to derive the layout area for each cluster (Fig. 7(c)). For each cluster node of the selected tree level, we extract several representative leaf topics to represent the content of this node. We follow the topic ranking techniques, namely, coverage and variance, as well as distinctiveness proposed in TIARA [15], to select the representative leaf topics. This method assigns higher ranks to topics that cover a significant portion of the cluster content (coverage), do not appear in all the clusters (variance), and are distinctly different from one another (distinctiveness). Naturally, the leaf topics that belong to the same cluster are placed in the corresponding tessellation cell by a force-directed layout, which maintains the cluster structure among topics (Fig. 7(d)). In the third step, we assign each hidden leaf topic to the closest representative leaf topic and utilize kernel density estimation [47] to visually illustrate the global cluster context (Fig. 7(e)).

6.3 Interaction

The following interactions are provided to help users understand the full picture.

Smooth exploration of different tree levels (T4). A user can zoom into a cluster of interest by clicking the corresponding tree node. The interaction works in the following way. First, the user can hover over a topic cluster in the hierarchy, and then a word cloud is displayed to convey the content of the topic cluster (Fig. 8(a)). The topics belonging to this cluster are highlighted in the node-link diagram. Based on these cues, the user can decide whether s/he needs to zoom in or not. If the user is interested, s/he can click the tree node to zoom into the corresponding cluster. As shown in Fig. 8(b), more topics from this cluster

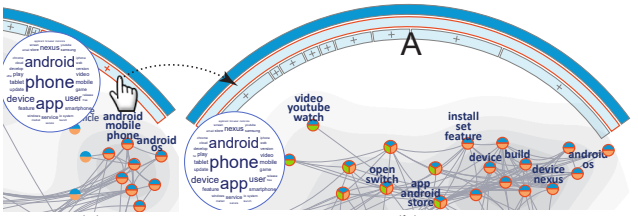


Figure 8: Exploration of different tree levels.

are then displayed. In addition, the children of the clicked tree node are shown on the radial icicle plot for further exploration (**A**). We use staged animations [48] to smooth the zoom in/out process and preserve a user's mental map.

Examining the topics of interest and their correlations (T3). To help users understand the topics of interest, we extract keywords and representative documents of each topic. Users can click a topic node to examine its keywords (Fig. 1(e)) and documents (Fig. 1(f)), or select a group of topics using a lasso to examine their keyword distribution. After a user finds an interesting topic, s/he can highlight the correlated topics to find more relevant topics. S/he can also hover over the topic to see its hierarchy in the tree. We also allow users to find interesting topics with the search function.

Analyzing lead-lag relationships of matched topics (T5). To help users analyze the temporal patterns of the matched topics, TopicPanorama visually illustrates the lead-lag relationships across corpora of a selected topic. The lead-lag relationships show which text corpus (lead) is followed by the others (lags) in regards to a specific common topic. We employ the algorithm developed by Liu et al. [31] to learn the lead-lag changes over time. Given two corpora, this algorithm considers a corpus to be the lead at a time point if its content at this time point is more similar to the future content of the other corpus rather than past content in the context of a given topic. The algorithm in [31] can only visualize the temporal lead-lag changes of two corpora. We have designed a twisted-line-like visualization to convey the lead-lag evolution patterns across multiple corpora (Fig. 9). In this visualization, each line represents a topic from a corpus with the x-axis encoding time. The lead/lag of a topic is encoded according to the spatial positions of the lines. The one with the higher position is the lead,

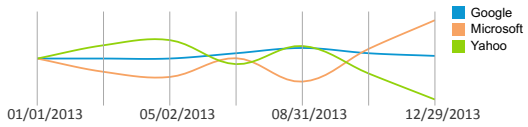


Figure 9: Temporal lead-lag visualization.

while the one with the lower position is the lag. A twisted point indicates the lead and lag have been switched.

Interactive match modification (**T6**). Inspired by semantic interactions designed by Endert et al. [49], we allow users to modify the matching results with three types of actions: *match*, *unmatch*, and *confirm*. *Match* and *unmatch* are used to modify incorrect matches, while *confirm* is used to lower the uncertainty of correct matches. In the match modification algorithm, *confirm* is treated in the same way as *match*, that is, the target distance is multiplied by α, with $0 < \alpha < 1$. After the modification, the full picture will be updated accordingly (Fig. 11). As users modify the matching results, a trail of modification actions is presented in the match modification trail panel (Fig. 1(c)). The action trail panel enables a user to undo and redo previous modifications as well as examine what matches are changed due to the corresponding modification. Potential incorrect matches will be highlighted if the user clicks the eye icon (**G**). A scented widget [50] (**H**) is then employed to filter the unwanted uncertainty glyphs.

7 Evaluation

In this section, we conduct a quantitative evaluation and two case studies to demonstrate the usefulness and effectiveness of TopicPanorama. The following datasets are used to evaluate the performance of TopicPanorama.

- **Dataset A** was collected from Boardreader [51] (from Jul. 2008 to Apr. 2009). It contains a news corpus, a blog corpus, and a BBS corpus.
- **Dataset B** and **Dataset C** include news articles related to Google, Microsoft, and Yahoo (from Jan. 2013 to Dec. 2013). Dataset B was sampled from Dataset C to reduce labeling efforts.
- **Dataset D** and **Dataset E** were collected from news and Twitter by using the keyword "Ebola" (from Jul. 27, 2014 to Feb. 21, 2015). Dataset D was sampled from Dataset E to reduce labeling efforts.

Table 1 shows a summary of these datasets.

7.1 Quantitative Evaluation

In our previous work [4], we compared the precision, recall, and conflicts of our graph matching algorithm with the baselines. Here we focus on evaluating the effectiveness and efficiency of the interactive match modification algorithm.

Experimental Settings. Three human labeled datasets, datasets A, B, and D, were used in the experiments. Two PhD students who are majoring in text mining labeled the matching results and the inter-annotator agreement was 83.8%. The experiments were conducted on a workstation with an Intel Xeon E5620 CPU (2.4 GHz) and 12GB Memory. We did a grid search (100, 110, ..., 200) to determine the number of features used in the feature-selection-based methods.

Criteria. We used average response time to measure the efficiency. To measure the effectiveness, the constraint number ($|\mathcal{U}|$) that is needed to correct all the matching errors was utilized. Specifically, we first compared the current matching result with the human labeled ground-truth to find incorrect matches. Then we randomly selected an incorrect match and treated the corresponding correct match as a constraint. Next, the constraint was input into the interactive match modification algorithm to generate a new matching result. The above process was repeated until all the incorrect matches were eliminated. Generally, the

(a) Dataset A

	$\|\mathcal{D}\|$	$\|\mathcal{V}\|$	$\|\mathcal{E}\|$
News	26,538	60	68
Blog	13,424	50	51
BBS	15,272	59	86

(b) Dataset B

	$\|\mathcal{D}\|$	$\|\mathcal{V}\|$	$\|\mathcal{E}\|$
Google	54,338	93	152
Microsoft	37,001	115	230
Yahoo	1,701	112	176

(c) Dataset C

	$\|\mathcal{D}\|$	$\|\mathcal{V}\|$	$\|\mathcal{E}\|$
Google	147,887	260	713
Microsoft	100,134	314	1285
Yahoo	6,280	246	872

(d) Dataset D

	$\|\mathcal{D}\|$	$\|\mathcal{V}\|$	$\|\mathcal{E}\|$
News	100,450	108	161
Twitter	6,381,868	130	222

(e) Dataset E

	$\|\mathcal{D}\|$	$\|\mathcal{V}\|$	$\|\mathcal{E}\|$
News	207,406	324	817
Twitter	15,565,532	390	833

Table 1: Summary statistics of five datasets. $|\mathcal{D}|$: document number; $|\mathcal{V}|$ and $|\mathcal{E}|$: node and edge numbers in a topic graph.

	Dataset A			Dataset B			Dataset D		
	$\|\mathcal{U}\|_{avg}$	$[\|\mathcal{U}\|_{min}, \|\mathcal{U}\|_{max}]$	Time	$\|\mathcal{U}\|_{avg}$	$[\|\mathcal{U}\|_{min}, \|\mathcal{U}\|_{max}]$	Time	$\|\mathcal{U}\|_{avg}$	$[\|\mathcal{U}\|_{min}, \|\mathcal{U}\|_{max}]$	Time
NoML	17.00	[17, 17]	0.0046	35.00	[34, 36]	0.0085	28.40	[28, 29]	0.0018
ML-NoFS	15.75	[15, 17]	2.9602	23.60	[21, 26]	0.8382	23.15	[20, 28]	342.00
ML-PC	15.35	[15, 16]	0.0755	23.20	[21, 25]	0.0366	22.80	[21, 24]	0.1145
ML-MI	13.35	[11, 17]	0.0850	23.10	[21, 26]	0.1062	19.65	[19, 20]	0.2113
ML-Relief	15.05	[13, 17]	0.3201	24.45	[23, 26]	0.0981	19.75	[18, 24]	0.3154

Table 2: Comparison of five match editing algorithms in terms of the average, minimum, and maximum constraint numbers needed to correct all the incorrect matches ($|\mathcal{U}|_{avg}$, $|\mathcal{U}|_{min}$, and $|\mathcal{U}|_{max}$), and average response time in seconds (Time). Here NoML and ML denotes without and with metric learning. NoFS represents without feature selection and PC, MI, and Relief denote Pearson Correlation, mutual information, and Relief relevancy measures employed in feature selection.

smaller the constraint number, the more effective the match modification algorithm. To reduce bias caused by incorrect match selection, we repeated the experiment 20 times in different selection orders and reported the average, minimum, and maximum numbers of $|\mathcal{U}|$ needed in all experiments.

Results. Table 2 compares the results of five match modification algorithms in terms of effectiveness and efficiency. The following observations can be made from the results.

With metric learning vs. without metric learning. Table 2 shows that the metric-learning-based algorithms were more effective than the algorithm without using metric learning (NoML). This demonstrates that the metric learning algorithm is more effective at learning the node substitution cost by using user-provided constraints. However, metric-learning-based algorithms are slower than NoML. In particular, ML-NoFS fails to respond in real-time in Dataset D. This indicates that a high computational cost is a key issue for metric learning.

With feature selection vs. without feature selection. The performance of the metric-learning-based algorithms with feature selection was comparable to that of an algorithm that does not have feature selection (ML-NoFS). In most cases, the metric-learning-based algorithms with feature selection were even more effective than ML-NoFS. This demonstrates that the feature selection methods can successfully find most of the salient words used in metric learning as well as avoid noisy words in most cases. Moreover, the feature selection based algorithms were at least 8 times faster than ML-NoFS. The lowest response time of the feature-selection-based algorithms was 0.3201 seconds (Dataset A, ML-Relief), which is fast enough for real-time interactions. This shows that feature selection techniques successfully speed up the metric learning algorithm without sacrificing its effectiveness.

Comparison of feature-selection-based methods. Among the three feature-selection-based algorithms, ML-MI performed a little better than the others. The reason ML-MI is more effective than ML-PC is that mutual information is a more comprehensive measure. Typically, Pearson Correlation works best for measuring the linear dependence between real-valued variables, while mutual information also takes higher order dependencies into account and takes both continuous and discrete variables. In our case, the class label is a discrete variable (matched or unmatched), and mutual information is thus a better choice. The reason why ML-MI is more effective than ML-Relief is because of the insufficient training samples for ML-Relief. The difference between the two methods is that ML-Relief considers the complicated interactions between features while ML-MI treats each feature independently. As a result, ML-Relief needs more training samples. In our case, the number of class labels (matched or unmatched) is not very large, making it not enough to derive the complicate interactions between features.

7.2 Case Study

We have worked closely with domain experts to develop scenarios and conduct the following case studies.

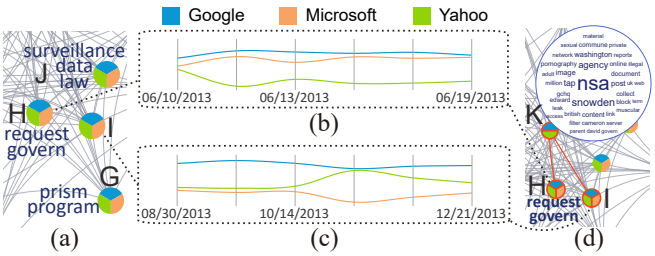


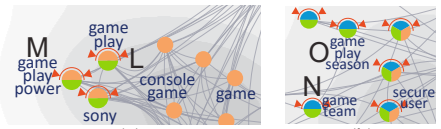
Figure 10: NSA Prism spying scandal shared by Google, Microsoft, and Yahoo: (a) relevant topics; (b) lead-lag relationship of topic **H**; (c) lead-lag relationship of topic **I**; (d) a correlated topic (**K**) of topics **H** and **I**.

7.2.1 IT Companies

This case study aims to illustrate how TopicPanorama helps analysts meet their analytical needs and points out what functions are useful for performing related tasks. Table 1(c) summarizes the statistics of Dataset C utilized here. One expert, who has been a public relations manager for over 10 years, participated in the case study. She used TopicPanorama to find a set of patterns within 2 hours and with some minor guidance from us.

Overview (tasks **T1** and **T2**). We first provided the expert with a full picture of the three companies (Fig. 1(a)). From the overview, she identified several common topics and the distinctive topics of each corpus. For instance, search and market related topics were shared by three corpora (**A**). Most phone-related topics were shared between Google and Microsoft and a few of them were shared by all three companies (**B**). Some government-related topics were referred to by the three companies and some of them were shared between Google and Yahoo (**C**). Car-related topics were mainly discussed in the Google corpus (**D**). Kinect-related topics were most often mentioned in the Microsoft corpus (**E**). The Yahoo corpus had some distinctive topics related to its CEO, Marissa Mayer (**F**).

Exploring the government-related cluster at different granularities and analyzing the temporal patterns (tasks **T4** and **T5**). The expert wanted to understand why so many government-related topics were shared by these companies. She zoomed into the fourth level of the topic tree by selecting the largest common tree node each time. Most of the topics talked about the Prism scandal. She further explored the NSA Prism spying scandal information shared by the three companies. As shown in Fig. 10(a), the four topics were classified into two groups. The first group concerned the disclosure of the scandal (**G**). The second category talked about actions taken by the three companies (**H**, **I**, and **J**), specifically, how they responded to this scandal in a similar manner. First, they denied they cooperated with the government in disclosing user data (**H**). Fig. 10(b) shows the lead-lag relationships of this topic. Google and Microsoft published transparency reports to disclose information about secret government requests for data. Later, Yahoo also disclosed the data requests from the US government. Second, the three companies encrypted information flowing between its various data centers (**I**). As shown in Fig. 10(c), Google took the lead, with Yahoo



(a) (b)

Figure 11: Interactive modification of the matching result.

responding similarly, and Microsoft later joining Google and Yahoo in beefing up encryption. The expert originally believed that only Google and Yahoo encrypted their data centers. After exploring the related topics with our tool, she found that Microsoft also stepped up encryption to thwart the NSA. She commented, "This is a surprise to me. I really appreciate this tool because it corrects my incorrect understandings." Finally, the three companies and other major tech companies asked the US government to reform its surveillance laws (**J**).

Examining correlations between topics (task **T3**). In the above exploration, the expert found one interesting pattern. When publishing the reports, Yahoo followed Google and Microsoft (Fig. 10(b)). However, Yahoo was more active in making plans to encrypt information (Fig. 10(c)). The expert was curious about such a change, so she continued to explore the topics correlated to both topics **H** and **I**. After some exploration, she found a relevant topic concerning "NSA statement on Washington Post report on infiltration of Google, Yahoo data center links" (**K**), which was connected to each of these topics, respectively.

Customizing the full picture (task **T6**). The expert was interested in game-related topics, so she entered "game" into the search box. A part of the search results is shown in Fig. 11(a). She enabled the tool to show the uncertainty glyph of the matched topics. After examining the results, she found two incorrect matches, **L** and **M**, which match Microsoft Xbox games to Yahoo-sports-related games. By unmatching **M**, she found **L** was changed to **N** and **M** was changed to **O** (Fig. 11(b)), which correctly matches Google sport games to Yahoo sport games. By leveraging the metric learning algorithm, the two incorrect matches can be fixed in one operation, instead of two operations in our previous method [4].

7.2.2 The Ebola Epidemic

The case study was conducted with professor P2. She is interested in how news media impacts the public in a health crisis like the Ebola outbreak. Table 1(e) shows the statistics of Dataset E used here.

Overview (tasks **T1** and **T2**). We first provided the professor with an overview of the Ebola epidemic (Fig. 12). By examining the keywords, she found the topics could be classified into four categories: Ebola outbreak in West Africa ($\mathbf{A_n}$, $\mathbf{A_c}$, and $\mathbf{A_t}$), U.S. Ebola suspects and patients ($\mathbf{B_n}$, $\mathbf{B_c}$, and $\mathbf{B_t}$), general knowledge of Ebola ($\mathbf{C_n}$, $\mathbf{C_c}$, and $\mathbf{C_t}$), and joining the fight against Ebola (**D**). Here subscripts **n**, **t**, and **c** represent the distinctive news topics, the distinctive Twitter topics, and the common topics, respectively.

Zooming in cluster "general knowledge of Ebola" (task **T4**). The professor wanted to know how the knowledge

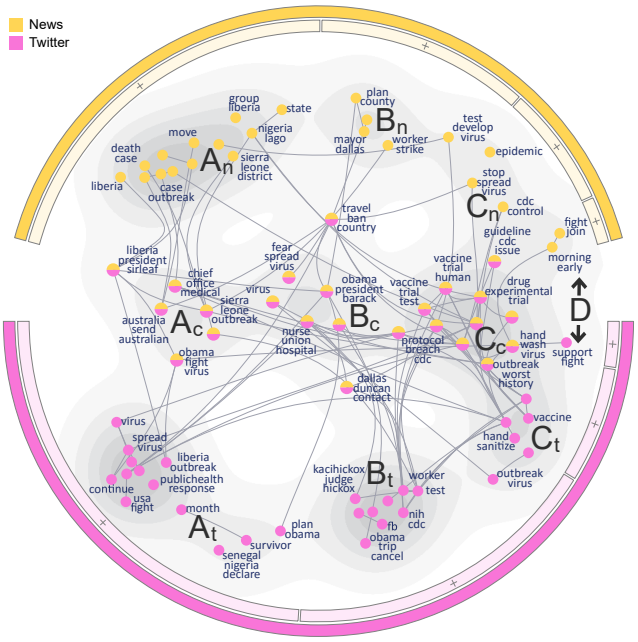


Figure 12: Overview of Ebola epidemic. Subscripts **n**, **t**, and **c** represent the distinctive news topics, the distinctive Twitter topics, and the common topics, respectively.

from the news media impacted the attitudes and practices of the public. Thus, she zoomed into the third category: general knowledge of Ebola. By selecting the largest tree node each time, she zoomed into a cluster related to general knowledge of the Ebola virus (Fig. 13(a)).

To understand how the news media impacted the public, she further investigated the topics in common. Most of the common topics talked about how the Ebola virus was transmitted. For example, topic **E** is about "How is Ebola spread? CDC expert explains 'direct contact' with bodily fluid." After some exploration, the professor found a common topic that reflected the public's attitudes and practices from Twitter (**F**). This topic was about hand washing, with keywords "hand," "wash," and "virus." News articles emphasized the significance of hand washing ("Preventing cholera, Ebola: Hand washing should be our top priority"), and related tweets demonstrated the attitudes and practices of the public: "I wash my hands like every hour I prevent from touching stuff and people like I don't want ebola." By analyzing the documents and checking the lead-lag relationships (Fig. 13(d)), the professor learned that the knowledge provided by the news media did impact the attitudes and practices of the public.

Two distinctive Twitter topics **G** and **H** that are correlated to **F** attracted the professor's attention. After examining them, the professor found that the hand washing topic created anxiety among the public. The public exhibited irrational behavior. For example, "I be pouring like a gallon of hand sanitizer on my hands after I shake up with Africans now BC of that Ebola shit." The professor then commented, "In crisis communications, the government and mainstream news media first need to warn the public to *watch out*, then tell the public to *calm down*. The news

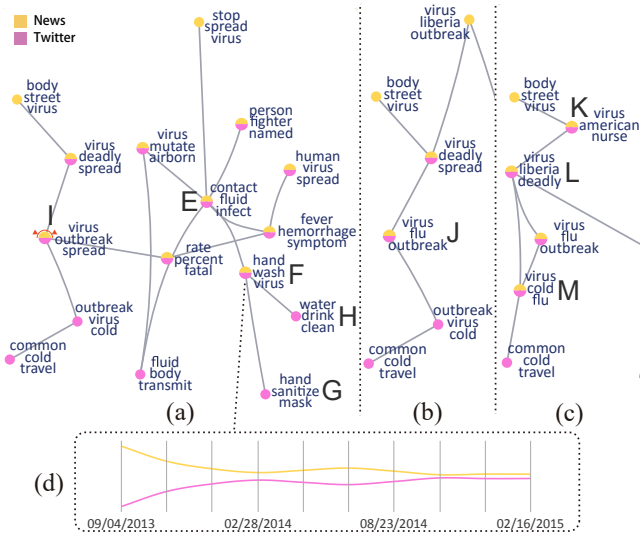


Figure 13: General knowledge of the Ebola virus. (a) The matched topic graph. (b) A topic that compares Ebola with flu (**J**) occurs after unmatching topic **I**. (c) Specific topics related to the Ebola virus in America (**K**), the Ebola virus in Liberia (**L**), and the common cold (**M**) occur after confirming topic **J**. (d) Lead-lag relationship of topic **F**.

media was already successful at delivering the *watch out* message, but needed do more to convey the *calm down* message." Professor P2 commented that there are two typical ways to calm down the public.

- Give specific and practical guidance for hand washing.
- Deliver three positive messages per negative message.

Customizing the full picture (task **T6**). The expert then continued to check whether the government and the mainstream news media made any efforts to calm the public down. When exploring the "general knowledge about the Ebola virus" cluster, P2 found most common topics were quite specific and illustrative, except for **I**. The news topic and Twitter topic in **I** were matched because of general keywords like "virus" and "outbreak." Professor P2 decided to unmatch the topic. After unmatching it, a new common topic **J** occurred. In this topic, the news media encouraged people to be more concerned about the flu than the Ebola virus ("Doctors More Concerned With Flu Season Than Ebola Virus"). The professor was glad to see this topic because it shows that the news media guided the public towards a more rational understanding of Ebola. The professor then confirmed this topic to lower its uncertainty. After confirming the topic, three specific common topics emerged (**K**, **L**, and **M**). They were the Ebola virus in America (**K**), the Ebola virus in Liberia (**L**), and the common cold (**M**). These topics reveal the news media's efforts at guiding the public towards a rational understanding of Ebola. For example, one of the news article in topic **M** has the title "Gov. Perry: Ebola is much harder to get than the cold."

7.3 Expert Interview

We interviewed the six domain experts working with us. Each of the evaluations took 90 minutes, including 10 minutes for system introduction, 50 minutes for the case study and free exploration, and 30 minutes for the post interview. Overall, TopicPanorama has been well received by the experts. We have summarized the feedback into four themes.

Graph matching. All the experts agreed that the graph matching component is very useful for developing a full picture. They especially liked the incremental graph matching algorithm. One expert commented, "I love the interactive editing function that allows me to modify the graph matching results according to my needs."

Interactive visualization. The experts were impressed by the power of the visualization components. They all liked the hybrid visualization that allows them to understand the full picture at different granularity levels. Furthermore, the uncertainty glyph provides an easy way to examine the mapping results with lower scores. The experts can then freely modify the error matching results based on their knowledge.

Insight discovery. All the experts were able to easily use TopicPanorama to form a full picture of relevant topics across multiple sources. They were able to find topics of interest and then drill down to examine their relationships with other topics. With TopicPanorama, experts were even able to gain insights they did not have before. For example, a senior public relations manager of a large IT company believed that in the NSA Prism spying scandal, only Google and Yahoo encrypted data while Microsoft did not. Based on what she saw by exploring the related topics in our tool, she learned that Microsoft also beefed up encryption following the actions of the other two companies.

Usage patterns. The frequency of match editing is determined by the user task and the application. According to our domain experts, if they simply want to obtain a general overview of the document collections, three to five editing operations are acceptable per task. If they need to perform a specific task (e.g., guide public opinion in a health crisis), more editing operations are acceptable. For example, one expert said that it is acceptable if the editing number is around dozens of times for one important task.

8 Conclusions and Future Work

We have worked with a group of experts, including public relations managers, journalists, and professors, to derive several high-level visual analytic tasks for understanding multi-source textual data. Based on these tasks, we developed TopicPanorama to help users develop a full picture of relevant topics that are discussed in multiple sources.

The system provides three advantages. First, it efficiently derives consistent graph matching results among multiple graphs. Second, it provides an LOD-based visualization that allows users to examine the matching results globally and locally and to switch between the global overview and local details smoothly. Third, it allows users to incrementally modify the matching results based on their knowledge and information needs.

Although the visual analysis reported here was limited to building and analyzing the full picture, several potentially

generalizable mechanism/components were produced. First, metric learning and feature selection can be used in the analysis of high-dimensional data. Second, the topic hierarchy generation method can be adopted by other visual text analytics tools to handle large data.

Our design also has some limitations. Although our graph matching algorithm and visualization method can handle any number of graphs, the number of corpora that can be visually compared is not large due to visual clutter and limited screen space. According to our interviews with experts, they can leverage TopicPanorama to analyze two or three topic graph matching results very well. It also works for four topic graphs though it takes longer to gain insight. It may fail to provide a better understanding for five or more topic graphs due to the limited display area and complex matching results. Previous experiments have found that approximately four objects can be tracked in visual comparison [34]. This conclusion is consistent with the feedback of our target users. They said they usually work on two or three corpora and seldom analyze four corpora in their work. Consequently, TopicPanorama works for most real-world applications. Another limitation is that not all the topics in the topic graph are meaningful. In our current implementation, we rank the topics and filter out the less important ones. A possible solution is to allow users to interactively modify topic mining results [52].

Future research will include the extension of interactive modification of matching results to topic mining results. The key is to study how to effectively combine the topic mining model with our graph matching algorithm. Another exciting avenue for the future is to design a suitable visualization for more than three corpora.

9 Acknowledgements

X. Wang, S. Liu, and J. Liu are supported by National Key Technologies R&D Program of China (No. 2015BAF23B03), the National Natural Science Foundation of China (No.s 61373070, 61272225, 61572274), and a Microsoft Research Fund (No. FY15-RES-OPP-112). J. Chen and J. Zhu are supported by the National Basic Research Program of China (No. 2013CB329403), National Natural Science Foundation of China (No.s 61322308, 61332007), a Microsoft Research Fund (No. FY14-RES-SPONSOR-111), and the National University Student Innovation Program.

References

[1] J. Zhang, Y. Song, C. Zhang, and S. Liu, "Evolutionary hierarchical dirichlet processes for multiple correlated time-varying corpora," in *KDD*, 2010, pp. 1079–1088.

[2] J. Chen, J. Zhu, Z. Wang, X. Zheng, and B. Zhang, "Scalable inference for logistic-normal topic models," in *NIPS*, 2013, pp. 2445–2453.

[3] K. Salomatin, Y. Yang, and A. Lad, "Multi-field correlated topic modeling," in *SDM*, 2009, pp. 628–637.

[4] S. Liu, X. Wang, J. Chen, J. Zhu, and B. Guo, "Topicpanorama: a full picture of relevant topics," in *IEEE VAST*, 2014, pp. 183–192.

[5] K. Riesen and H. Bunke, "Approximate graph edit distance computation by means of bipartite graph matching," *Image Vision Computing*, vol. 27, no. 7, pp. 950–959, 2009.

[6] G. A. Korsah, A. T. Stentz, and M. B. Dias, "The dynamic hungarian algorithm for the assignment problem with changing costs," Tech. Rep. CMU-RI-TR-07-27, 2007.

[7] J. Yan, Y. Tian, H. Zha, X. Yang, Y. Zhang, and S. M. Chu, "Joint optimization for consistent multiple graph matching," in *ICCV*, 2013, pp. 1649–1656.

[8] M. L. Williams, R. C. Wilson, and E. R. Hancock, "Multiple graph matching with bayesian inference," *Pattern Recognition Letters*, vol. 18, no. 11-13, pp. 1275–128, 1997.

[9] A. Solé-Ribalta and F. Serratosa, "Models and algorithms for computing the common labelling of a set of attributed graphs," *Computer Vision and Image Understanding*, vol. 115, no. 7, pp. 929–945, 2011.

[10] T. von Landesberger, A. Kuijper, T. Schreck, J. Kohlhammer, J. J. van Wijk, J.-D. Fekete, and D. W. Fellner, "Visual analysis of large graphs: State-of-the-art and future research challenges," *Computer graphics forum*, vol. 30, no. 6, pp. 1719–1749, 2011.

[11] M. Hascoët and P. Dragicevic, "Interactive graph matching and visual comparison of graphs and clustered graphs," in *AVI*, 2012, pp. 522–529.

[12] S. Liu, W. Cui, Y. Wu, and M. Liu, "A survey on information visualization: recent advances and challenges," *The Visual Computer*, pp. 1–21, 2014.

[13] G. Sun, Y. Wu, R. Liang, and S. Liu, "A survey of visual analytics techniques and applications: State-of-the-art research and future challenges," *Journal of Computer Science and Technology*, vol. 28, no. 5, pp. 852–867, 2013.

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

[15] S. Liu, M. X. Zhou, S. Pan, Y. Song, W. Qian, W. Cai, and X. Lian, "Tiara: Interactive, topic-based visual text summarization and analysis," *ACM TIST*, vol. 3, no. 2, pp. 25:1–25:28, 2012.

[16] M. Dörk, D. M. Gruen, C. Williamson, and M. S. T. Carpendale, "A visual backchannel for large-scale events," *IEEE TVCG*, vol. 16, no. 6, pp. 1129–1138, 2010.

[17] W. Dou, X. Wang, R. Chang, and W. Ribarsky, "Paralleltopics: A probabilistic approach to exploring document collections," in *IEEE VAST*, 2011, pp. 231–240.

[18] W. Cui, S. Liu, L. Tan, C. Shi, Y. Song, Z. Gao, H. Qu, and X. Tong, "Textflow: Towards better understanding of evolving topics in text," *IEEE TVCG*, vol. 17, no. 12, pp. 2412–2421, 2011.

[19] W. Cui, S. Liu, Z. Wu, and H. Wei, "How hierarchical topics evolve in large text corpora," *IEEE TVCG*, vol. 20, no. 12, pp. 2281–2290, 2014.

[20] S. Gad, W. Javed, S. Ghani, N. Elmqvist, T. Ewing, K. N. Hampton, and N. Ramakrishnan, "Themedelta: Dynamic segmentations over temporal topic models," *IEEE TVCG*, vol. 21, no. 5, pp. 672–685, 2015.

[21] G. Sun, Y. Wu, S. Liu, T.-Q. Peng, J. J. H. Zhu, and R. Liang, "EvoRiver: Visual analysis of topic coopetition on social media," *IEEE TVCG*, vol. 20, no. 12, pp. 1753–1762, 2014.

[22] Y. Wu, S. Liu, K. Yan, M. Liu, and F. Wu, "Opinionflow: Visual analysis of opinion diffusion on social media," *IEEE TVCG*, vol. 20, no. 12, pp. 1763–1772, 2014.

[23] P. Xu, Y. Wu, E. Wei, T.-Q. Peng, S. Liu, J. J. H. Zhu, and H. Qu, "Visual analysis of topic competition on social media," *IEEE TVCG*, vol. 19, no. 12, pp. 2012–2021, 2013.

[24] J. Zhao, N. Cao, Z. Wen, Y. Song, Y.-R. Lin, and C. Collins, "Fluxflow: Visual analysis of anomalous information spreading on social media," *IEEE TVCG*, vol. 20, no. 12, pp. 1773–1782, 2014.

[25] A. J.-B. Chaney and D. M. Blei, "Visualizing topic models." in *ICWSM*, 2012, pp. 419–422.

[26] W. Dou, L. Yu, X. Wang, Z. Ma, and W. Ribarsky, "Hierarchicaltopics: Visually exploring large text collections using topic hierarchies," *IEEE TVCG*, vol. 19, no. 12, pp. 2002–2011, 2013.

[27] E. Alexander, J. Kohlmann, R. Valenza, M. Witmore, and M. Gleicher, "Serendip: Topic model-driven visual exploration of text corpora," in *IEEE VAST*, 2014, pp. 173–182.

[28] N. Cao, J. Sun, Y.-R. Lin, D. Gotz, S. Liu, and H. Qu, "Facetatlas: Multifaceted visualization for rich text corpora," *IEEE TVCG*, vol. 16, no. 6, pp. 1172–1181, 2010.

[29] N. Cao, D. Gotz, J. Sun, Y.-R. Lin, and H. Qu, "Solarmap: Multifaceted visual analytics for topic exploration," in *IEEE ICDM*, 2011, pp. 101–110.

[30] Y. Lu, M. Steptoe, S. Burke, H. Wang, J. Tsai, H. Davulcu, D. Montgomery, S. Corman, and R. Maciejewski, "Exploring evolving media discourse through event cueing," *IEEE TVCG*, vol. 22, no. 1, pp. 220–229, Jan 2016.

[31] S. Liu, Y. Chen, H. Wei, J. Yang, and K. Zhou, "Exploring topical lead-lag across corpora," *IEEE TKDE*, vol. 27, no. 1, pp. 115–129, 2015.

[32] D. Oelke, H. Strobelt, C. Rohrdantz, I. Gurevych, and O. Deussen, "Comparative exploration of document collections: a visual analytics approach," *Computer Graphics Forum*, vol. 33, no. 3, pp. 201–210, 2014.

[33] Y. Wu, T. Provan, F. Wei, S. Liu, and K.-L. Ma, "Semantic-preserving word clouds by seam carving," in *Computer Graphics Forum*, vol. 30, no. 3, 2011, pp. 741–750.

[34] S. Yantis, "Multielement visual tracking: Attention and perceptual organization," *Cognitive psychology*, vol. 24, no. 3, pp. 295–340, 1992.

[35] X. Wang, S. Liu, Y. Song, and B. Guo, "Mining evolutionary multi-branch trees from text streams," in *KDD*, 2013, pp. 722–730.

[36] P. Jain, B. Kulis, I. S. Dhillon, and K. Grauman, "Online metric learning and fast similarity search," in *NIPS*, 2009, pp. 761–768.

[37] A. C. Pocock, "Feature selection via joint likelihood," Ph.D. dissertation, University of Manchester, 2012.

[38] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *JMLR*, vol. 3, pp. 1157–1182, Mar. 2003.

[39] P. McLachlan, T. Munzner, E. Koutsofios, and S. North, "Liverac: interactive visual exploration of system management time-series data," in *CHI*, 2008, pp. 1483–1492.

[40] B. Alper, B. Bach, N. H. Riche, T. Isenberg, and J.-D. Fekete, "Weighted graph comparison techniques for brain connectivity analysis," in *CHI*, 2013, pp. 483–492.

[41] Y. Wu, F. Wei, S. Liu, N. Au, W. Cui, H. Zhou, and H. Qu, "Opinionseer: Interactive visualization of hotel customer feedback," *IEEE TVCG*, vol. 16, no. 6, pp. 1109–1118, 2010.

[42] Y. Wu, G.-X. Yuan, and K.-L. Ma, "Visualizing flow of uncertainty through analytical processes," *IEEE TVCG*, vol. 18, no. 12, pp. 2526–2535, 2012.

[43] A. M. MacEachren, R. E. Roth, J. O'Brien, B. Li, D. Swingley, and M. Gahegan, "Visual semiotics & uncertainty visualization: An empirical study," *IEEE TVCG*, vol. 18, no. 12, pp. 2496–2505, 2012.

[44] T. Kamada and S. Kawai, "An algorithm for drawing general undirected graphs," *Information processing letters*, vol. 31, no. 1, pp. 7–15, 1989.

[45] K. Misue, P. Eades, W. Lai, and K. Sugiyama, "Layout adjustment and the mental map," *Journal of visual languages and computing*, vol. 6, no. 2, pp. 183–210, 1995.

[46] M. Balzer and O. Deussen, "Voronoi treemaps," in *IEEE InfoVis*, 2005, pp. 49–56.

[47] O. D. Lampe and H. Hauser, "Interactive visualization of streaming data with kernel density estimation," in *IEEE PacificVis*, 2011, pp. 171–178.

[48] B. Bach, E. Pietriga, and J.-D. Fekete, "Graphdiaries: animated transitions andtemporal navigation for dynamic networks," *IEEE TVCG*, vol. 20, no. 5, pp. 740–754, 2014.

[49] A. Endert, P. Fiaux, and C. North, "Semantic interaction for visual text analytics," in *ACM SIGCHI*, 2012, pp. 473–482.

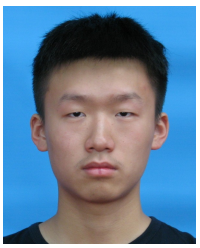
[50] W. Willett, J. Heer, and M. Agrawala, "Scented widgets: Improving navigation cues with embedded visualizations," *IEEE TVCG*, vol. 13, no. 6, pp. 1129–1136, Nov. 2007.

[51] "Boardreader," http://www.boardreader.com, Aug. 2015.

[52] J. Choo, C. Lee, C. K. Reddy, and H. Park, "Utopian: User-driven topic modeling based on interactive nonnegative matrix factorization," *IEEE TVCG*, vol. 19, no. 12, pp. 1992–2001, 2013.

Xiting Wang is a PhD candidate at Tsinghua University, China. Her research interests include visual text analytics and text mining. She received a BS degree in Electronics Engineering from Tsinghua University.

Shixia Liu is an associate professor at Tsinghua University, China. Her research interests include visual text analytics, visual social analytics, and graph visualization. Before joining Tsinghua University, she worked as a research staff member at IBM China Research Lab and a lead researcher at Microsoft Research Asia. She received a B.S. and M.S. in Computational Mathematics from Harbin Institute of Technology, a Ph.D. in Computer Science from Tsinghua University. She is an associate editor of IEEE TVCG.



Junlin Liu is currently an undergraduate at School of Software, Tsinghua University. He won a gold medal in National Olympiad in Informatics and was a member of national training team for International Olympiad in Informatics.

Jianfei Chen is a PhD candidate at Tsinghua University, China, where he received his BS degree in Computer Science from. His research interests include large scale machine learning and topic modeling.

Jun Zhu received his BS and PhD degrees from the Department of Computer Science and Technology in Tsinghua University, China, where he is currently an associate professor. He was a project scientist and postdoctoral fellow in the Machine Learning Department, Carnegie Mellon University. His research interests are primarily on machine learning, Bayesian methods, and large-scale algorithms. He was selected as one of the "AI's 10 to Watch" by IEEE Intelligent Systems in 2013. He is an associate editor of IEEE Trans. on PAMI.

Baining Guo is the Assistant Managing Director of Microsoft Research Asia, where he also serves as the head of the graphics lab. Prior to joining Microsoft in 1999, Dr. Guo was a senior staff researcher with the Microcomputer Research Labs of Intel Corporation in Santa Clara, California. Dr. Guo received Ph.D. and M.S. from Cornell University and B.S. from Beijing University. His research interests include computer graphics and visualization, in the areas of texture and reflectance modeling, texture mapping, translucent surface appearance, real-time rendering, and geometry modeling.