Author manuscript, published in IEEE Transactions on Visualization and Computer Graphics, 24(1), pp. 195-204, 2018.
DOI: 10.1109/TVCG.2017.2744458

BiDots: Visual Exploration of Weighted Biclusters

Jian Zhao, Maoyuan Sun, Francine Chen, and Patrick Chiu

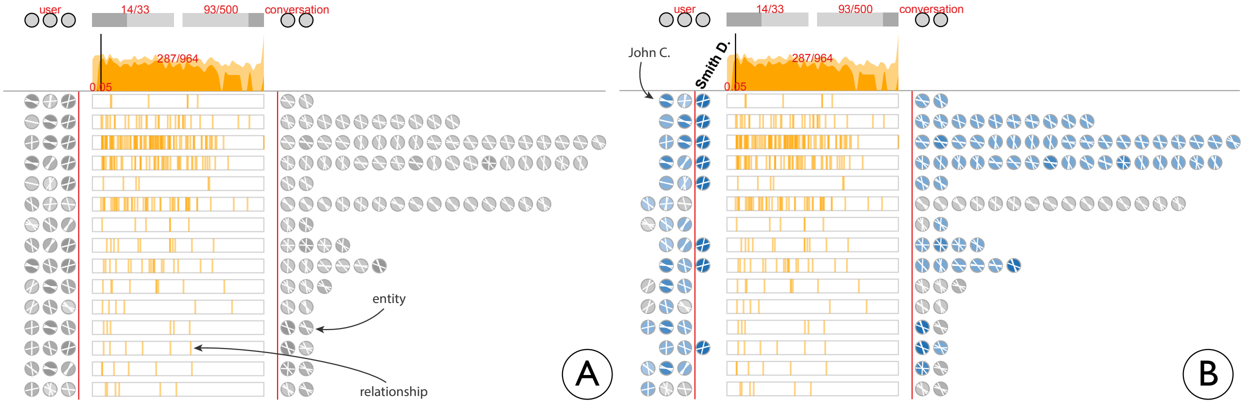


Fig. 1. An analyst uses BiDots to explore a biclustering result of user-conversation relationships mined from a large corpus of chat data. (A) The top part provides statistical information about the data, and the bottom part displays each bicluster in a row, where each unique entity is represented as a circle with a unique line pattern and weighted relationships between entities are represented as orange line strips, revealing distributions of the weights. (B) An analyst pins an entity in the user domain, `Smith D.`, which creates a separate column for the entity, allowing for easier examination of its memberships to all the biclusters.

Abstract—Discovering and analyzing biclusters, i.e., two sets of related entities with close relationships, is a critical task in many real-world applications, such as exploring entity co-occurrences in intelligence analysis, and studying gene expression in bio-informatics. While the output of biclustering techniques can offer some initial low-level insights, visual approaches are required on top of that due to the algorithmic output complexity.This paper proposes a visualization technique, called BiDots, that allows analysts to interactively explore biclusters over multiple domains. BiDots overcomes several limitations of existing bicluster visualizations by encoding biclusters in a more compact and cluster-driven manner. A set of handy interactions is incorporated to support flexible analysis of biclustering results. More importantly, BiDots addresses the cases of weighted biclusters, which has been underexploited in the literature. The design of BiDots is grounded by a set of analytical tasks derived from previous work. We demonstrate its usefulness and effectiveness for exploring computed biclusters with an investigative document analysis task, in which suspicious people and activities are identified from a text corpus.

Index Terms—Biclustering, coordinated relationship analysis, visual analytics.

1 Introduction

Biclustering, a fundamental data mining technique for coordinated relationship analysis, can detect groups of closely related entities with different types (i.e., *biclusters*); for example, the relationships formed by three people all visiting four places. Discovering and understanding such biclusters is essential in many real-world applications. For example, in intelligence analysis, connections between entities can be built upon co-occurrences of entities (e.g., person and place) in documents, and biclusters can lead to evidences of collusion [16]. Biologists explore biclusters between gene expression and interaction datasets to discover sets of genes/proteins that are commonly expressed or regulated conditions and species [24].

From relationships across two domains (i.e., entity types), a biclustering algorithm (e.g., LCM [34]) can compute biclusters, defined as complete bipartite graphs representing subsets of the relationships. While biclustering algorithms can offer analysts with some initial insights, the output can be large and complicated, (e.g., many biclusters containing numerous overlapping entities), and therefore impractical to explore in non-graphical form. Further, more than one pair of domains can be explored in many applications, which can produce *bicluster chains* (e.g., associating an organization-to-person bicluster and a person-to-place one). For better understanding of biclustering results, visualization techniques have been proven effective (e.g., [8,10,11,32]).

However, a critical aspect of biclusters—relationship weights—has been underexploited in existing visualizations of biclusters. In many real-world applications, relationships are often associated with real numbers, rather than just in binary form (i.e., exist or non-exist). For example, in bio-informatics, gene expression level is embedded in relationships between genes and conditions; and in intelligence analysis, connections between entities (e.g., person and organization) can contain likelihood values computed by text analytics techniques.

In addition, current visualizations have not effectively leveraged biclusters for coordinated relationship analysis, especially in the case of exploring multiple domains. Association of biclusters from two diverse relationships has only been expressed at the entity level (e.g., connecting identical entities with lines or aligning the exact same entity sets together [8, 32]). An analyst has to trace chains of entities, and may miss a bigger picture of potential bicluster chains. A higher-level and more flexible representation of results from different biclustering processes is thus demanded for better understanding of the data.

To address the above concerns, we propose BiDots, a visualization technique for analyzing weighted biclusters identified in data (Fig. 1).

- *Jian Zhao, Francine Chen, and Patrick Chiu are with FX Palo Alto Laboratory. E-mail: {zhao|chen|chiu}@fxpal.com.*
- *Maoyuan Sun is with University of Massachusetts Dartmouth. E-mail: smaoyuan@umassd.edu.*

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx

BiDots arranges biclusters in rows and visually encodes a bicluster, including its entities and relationships with weights, in a compact manner, which is scalable for showing a larger number of biclusters with many unique entities. A set of fluid interactions, such as filtering, ordering, and grouping, is provided to facilitate exploration and comparison of biclusters. The analysis of biclusters across multiple domains is powered by a novel bicluster matching and chaining algorithm, offering a flexible and holistic view of the data. Moreover, BiDots allows an analyst to interactively adjust thresholds on data relationship weights and parameters of biclustering algorithms to conduct biclustering in different settings and explore the results in an iterative manner.

The design of BiDots is grounded with a three-level framework of analytical tasks derived from the literature (e.g., [32, 33]) and our interviews with data mining experts. We demonstrate the effectiveness and usefulness of BiDots in text analytics with a usage scenario of investigative document analysis based on a benchmark dataset [15].

2 Background

In this section, we first introduce some background about biclustering, and then discuss related visualization techniques for biclusters and design guidelines for such visualizations.

2.1 Biclusters

From relationships between two domains or entity types (e.g., gene and condition, or organization and person), the goal of biclustering is to identify corresponding subsets of entities from each domain with the restriction that entities in each subset share similar attributes (e.g., genes behaving identically for gene-condition relationships in bio-informatics [24]). Compared to clustering, biclustering generalizes the idea of simultaneously finding coherent subsets from two domains. This is also the reason that biclusters serve as the basis for coordinated relationship analysis that is more general (e.g., insights discovered from a number of related biclusters or parts of a bicluster).

Computational results of this process are biclusters, each consisting of two sets of entities and their relationships. Formally, given two entity sets M and N and their relationships $rel(M,N)$ (that is a subset of the Cartesian product $M \times N$), we define a bicluster as (M',N'), where $M' \subseteq M$, $N' \subseteq N$, and $M' \times N' \subseteq rel(M,N)$. A bicluster is *closed* if adding any entities violates the above definition. The *size* of a bicluster is denoted as $|M| \times |N|$, where $|\cdot|$ represents the cardinality of a set.

For example, in Fig. 2, from the relationships between organization A_i and person B_j, a bicluster $(\{A_1,A_4,A_5\},\{B_3,B_5\})$ can be detected by an algorithm. This may indicate that both persons have connections with all the three organizations. The bicluster is closed because no relationships can be further added to increase its size (i.e., 3×2). In this paper, for simplicity, we refer to closed biclusters as biclusters.

Several algorithms have been proposed in the data mining community to discover closed biclusters from data (e.g., CHARM [39] and LCM [34]). One important parameter of these biclustering algorithms is *minimum support threshold*, which governs the minimum size of biclusters. Typically, a user has to set this parameter heuristically in real-world applications. Moreover, some of the algorithmically identified biclusters may *overlap* with each other. For example, in Fig. 2, there is another bicluster $(\{A_2,A_4,A_5\},\{B_3,B_4\})$, which shares entities A_4,A_5 and B_3 with the bicluster $(\{A_1,A_4,A_5\},\{B_3,B_5\})$.

Biclusters from different pairs of domains can be connected together to form bicluster chains based on shared entities. For example, the organization-to-person relationships can be appended with person-to-place relationships. This could provide insights such as a group of people who happened to visit a set of places also having connections within some common organizations (Fig. 2). The shared entity set between two biclusters can be identical (e.g., an organization-person bicluster has exactly the same people as a person-place bicluster). However, real-world scenarios can be difficult; often only some of the entities in the shared domain of two biclusters are common. Several advanced bicluster chain mining methods have been proposed in the data mining community (e.g., [38]). However, they are usually computationally expensive and do not offer visual exploration support.

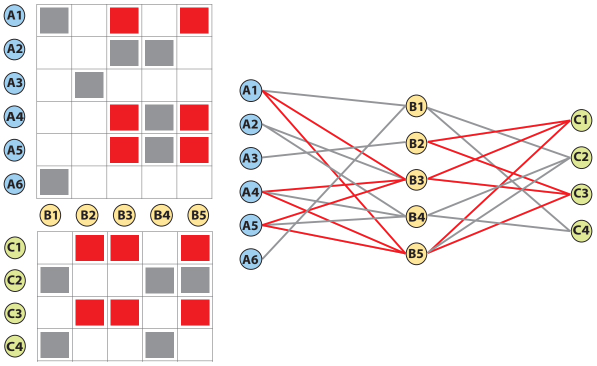


Fig. 2. An example of extracting biclusters from relationships between organizations A_i and person B_j as well as those between person B_j and places C_k, and further chaining the biclusters. Colored cells and lines indicate existing relationships respectively, revealing a bicluster-chain consisting of biclusters $(\{A_1,A_4,A_5\},\{B_3,B_5\})$ and $(\{B_2,B_3,B_5\},\{C_1,C_3\})$. However, only B_3 and B_5 are the common entities in the shared domain.

We address the above scenario of multi-domain bicluster exploration in BiDots by proposing an effective bicluster matching and chaining algorithm, which allows an analyst to interactively explore bicluster chains with different algorithm settings.

2.2 Visualization of Biclusters

Different approaches have been proposed for visualizing biclusters, which fall into three main categories: *entity*-driven, *relationship*-driven, and *cluster*-driven. Because entities and relationships are two essential components of biclusters, entity-driven and relationship-driven designs emphasize each of these two aspects. Cluster-driven design, on the other hand, seeks to focus on both aspects and further promotes biclusters as integral visualization objects. In the following, we review major modern visualization techniques developed on top of these concepts.

2.2.1 Entity-Driven Design

One popular entity-driven design displays entities in different domains as nodes in different lists and represents their relationships with links. A natural extension of this to multiple domains (i.e., bicluster chains) is Parallel Coordinates [17], similar to the concept in Fig. 2 (right).

Jigsaw [29] and parallel node-link bands [9] are classic examples of entity-driven design. It allows an analyst to select entities in different lists and only reveal links connecting all of them. To identify a bicluster, one needs to click on the entities of interests for multiple times and find two sets of entities with links between every entity on both sides. ConTour [25] and the system by Corput et al. [35], on the other hand, use visual cues characterizing entities to achieve similar goals, where links are hidden. However, the above entity-driven techniques do not visualize biclusters explicitly, relying on user interactions to further uncover them, which makes it difficult to discover biclusters or distinguish one bicluster from another. While ConTour further applies recursive nesting to ease bicluster differentiation, it requires many entity duplication and is not space-efficient.

A more generic visualization of the entity-driven design is a node-link diagram, which allows for a free layout of entities (nodes) in a 2D space rather than in linear lists. For example, PivotPath color-codes entities by type and place entities from three different domains in separate regions [6]. Entities of the shared domain (e.g., keywords) are shown in a list between two other domains (e.g., authors and articles) where entities are freely positioned. A similar method is employed by Pretorius et al. for multivariate graph exploration with the separation of source-nodes, edges, and target nodes [26]. PivotSlice allows a user to split a node-link diagram into a 2D separation (a tabular organization) in which each region represents entities with shared attributes [40]. However, similar to Jigsaw and Contour, these techniques require users to manually discover biclusters by tracing the links. Although BicOverlapper [28] explicitly shows boundaries of biclusters like a Venn diagram, it is limited due to visual clutter [2].

More importantly, none of the above techniques has explored weighted biclusters like BiDots. While visual encodings (e.g., color or line thickness) can be added to links to show weights, it is not visually salient due to the limitation of link thickness, thus still imposing difficulty for analysts to get a clear view of relationship weight information.

2.2.2 Relationship-Driven Design

The relationship-driven design focuses more on relationships between entities. One approach is to display data in matrices, similar to the concept shown in Fig. 2 (left). In this case, entities are displayed as rows and columns and their relationships are usually shown as squares in matrix cells at the corresponding locations, so that relationships are emphasized. This method has less visual clutter than using nodes and links in the entity-driven approach. It is commonly used in the domain of bio-informatics. Examples include BiVoc [10], Bicluster viewer [11], Expression Profiler [19], and BicOverlapper 2.0 [27].

However, one limitation of these techniques is that all biclusters cannot be revealed completely without constantly reordering rows and columns or duplicating some of them, because of entity overlaps among biclusters. Also, there may be large wasted space when there is a large number of unique entities (i.e., a sparse matrix).

For showing relationships in more than two domains, Scatterplot Matrix is one basic method [7]. Another way is to concatenate matrices of different pairs of domains together when they share entities [38, 41]. However, it is limited for displaying all possible bicluster chains due to the above issues of this relationship-driven approach.

Different from these techniques, BiDots encodes biclusters in a more compact form with the support of showing weighted relationships. Interaction techniques are provided for exploring entity overlaps across multiple biclusters. While still duplicating entities, BiDots allows for showing all bicluster chains in a flexible manner.

2.2.3 Cluster-Driven Design

By promoting biclusters visually, cluster-driven design attempts to combine the advantages of the above two approaches and further allow users to more easily explore and organize biclusters. Some are based on the notion of NodeTrix [12]. For example, Furby [30] and Bixplorer [8] represent each bicluster as a matrix and connect the same entities with links. Bicluster matrices can be freely positioned in a 2D space. These techniques present biclusters more explicitly than the previous two methods. Also, they can show all biclusters at once and are more space efficient for sparse data. On the other hand, BiSet [32], which is not based on NodeTrix [12], visually encodes biclusters as boxes between lists and employs edge bundling to reduce visual clutter (Fig. 3). Similarly, Matchmaker [22] and VisBricks [21] visualize clusters using extensive charts (e.g., heatmaps and parallel coordinates) and draw bundled links to indicate relationships between clusters. Liu et al. employ BiSet in a scenario of visualizing deep convolutional neural networks (CNNs), where links between layers have weights, but the weight information is not clearly encoded [23].

BiDots falls into this cluster-driven design by displaying each bicluster in a separate row and enabling users to manipulate them. However, the other approaches are less scalable when there are many biclusters or larger numbers of unique entities. Further, entity overlaps (or entity memberships to biclusters) are difficult to observe based on links, even when bundled (Fig. 3). This task can become harder for analyzing bicluster chains across multiple domains. Again, weighted biclusters have not been adequately explored in these works.

2.3 Design Guidelines of Visualizing Biclusters

The design of BiDots is largely affected by several design guidelines of exploring biclusters in the literature. A five-level design framework has been proposed by Sun et al. [33], including *entity*, *group*, *bicluster*, *chain*, and *schema* levels. Relationships in lower levels provide an essential support for exploring and understanding those in higher levels. For example, understanding biclusters is the basis of browsing and organizing bicluster chains [31]. Sun et al. further discussed the trade-off between entity-centric and relationship-centric designs as well as four types of explorations for identifying tasks in coordinated relationships analysis [32]. However, the aspect of weighted biclusters has been overlooked in these design guidelines. Together with the results of our interviews with domain experts, we extend these guidelines and propose three levels of analysis to guide the design of BiDots.

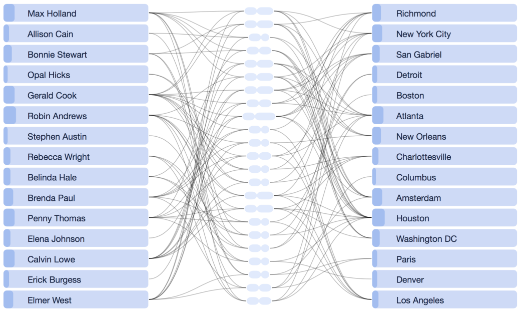


Fig. 3. An example of the BiSet technique [32].

3 Analytical Tasks

To guide the design of BiDots, we conducted an informal one-hour interview with two domain experts, who are researchers in the fields of data mining, machine learning, and data analytics (over twenty years of experience). They only have general knowledge in data visualization. During the interview, we aimed to understand how biclustering is used in their everyday research of solving practical problems. We then further coordinated our findings from the interview with design rationales proposed in the literature (e.g., [33]). In the following, we identify three levels of analysis when exploring weighted biclusters based on outputs of biclustering algorithms.

T1. Topological analysis. Since the underlying data model in which biclusters are mined is a graph (more precisely, a weighted bipartite graph for a relationship space with one pair of domains), some analytical tasks related to graph visualization (e.g., [20]) are relevant here. More particularly, tasks on exploring the graph structure—topological analysis—can be valuable. We further classify the topological analysis tasks into two levels.

- *Low-level tasks.* Low-level analysis is related to the access of basic topological information in the bipartite graph. For example, questions may include: how are entities connected, what is the weight of a specific relationship, what are the entities in the other domain and in other biclusters connected to a specific entity?
- *High-level tasks.* High-level analysis, while still based on the topology, addresses the integration of information. For example, questions may include: what is the overall distribution of a bicluster's relationship weights, which biclusters does a specific entity belong to, and what are the most commonly shared entities across specific biclusters?

T2. Semantic analysis. Beyond graph structures, semantic analysis concerns higher-level correlations and connections between biclusters in the entire dataset.

- *Ordering and filtering.* An analyst's goals may change dynamically over the course of analysis, through interactively manipulating visual arrangement and representation of biclusters. Ordering biclusters based on average relationship weight, size, and other quantities can help discover biclusters with extreme attributes. Moreover, an analyst may need to filter out less prominent entities and relationships to reduce the visual complexity at appropriate analysis stages.
- *Similarity.* Similarity analysis of biclusters could result in better sense making of the data. For example, grouping biclusters with similar sets of entities or relationship weight distributions may lead to deeper insights based on larger (but more loosely connected) biclusters. Matching similar biclusters with shared domains to construct bicluster chains

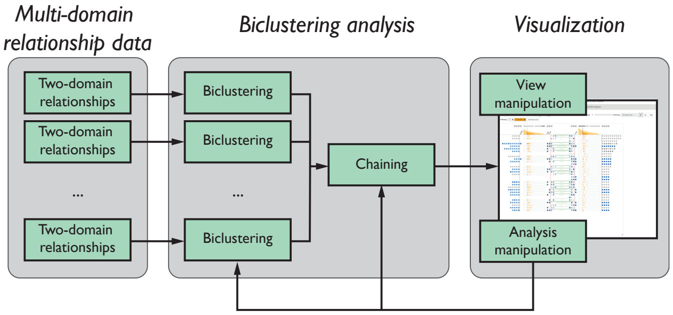


Fig. 4. An overview of BiDots.

may facilitate the discovery of hidden connections in various perspectives.

T3. Interactive analysis. As mentioned earlier, several parameters of biclustering algorithms such as the minimum support threshold are empirically determined by an analyst during the analysis. In the case of exploring weighted biclusters, pruning of lower weighted relationships before biclustering is an essential task for obtaining more suitable and manageable results. However, an analyst needs to iteratively try different values to tune these parameters until satisfied, requiring the support of interactively adjustable algorithm parameters in the system.

In summary, the topological analysis is to understand the *structure* of data; the semantic analysis focuses on the *meaning* of the data; and the interactive analysis adjusts the *scope* of data exploration.

The above three-level analytical tasks correspond to the five-level design framework proposed by Sun et al. [33], although it does not discuss weighted biclusters. In particular, the topological analysis is related to entity, group, and bicluster levels; the semantic analysis focuses on bicluster and chain levels; and the interactive analysis deals with tasks at the schema level.

Moreover, the topological analysis touches the four types of exploration discussed by Sun et al. [32], and the semantic analysis extends their from-relationship-to-relationship exploration (relationship here means bicluster) by introducing similarity analysis. Further, the interactive analysis addresses a fifth type of exploration that has been under-emphasized.

4 System Overview

Guided by the aforementioned analytical tasks, we designed and developed BiDots, an interactive visualization for exploring weighted biclusters from data mining results.

Fig. 4 shows an overview of the system architecture. For relationships between every pair of domains that could be interesting, the system performs biclustering on each pair. We employ LCM [34] to extract biclusters, however, any other biclustering algorithm can be easily integrated. Then, the biclusters from different biclustering processes are combined to construct bicluster chains which serve as the input to visualize. Next, an analyst can leverage the visualization to discover insights from biclusters by interactively manipulating the visual views and adjusting the biclustering parameters to generate new inputs for visualization.

5 BiDots Technique

In this section, we describe the BiDots technique in detail. We start by introducing the basic visual representation of biclusters and rich interactions available in BiDots. Then, we discuss the aspects regarding bicluster chains and BiDots's interactive analysis support.

5.1 Basic Visual Encoding

Based on the spirit of the cluster-driven design, BiDots displays all biclusters explicitly as visual objects because it has been shown to be more effective than the entity- and relationship-based designs [8, 32]. Fig. 1A illustrates the basic visual encoding in BiDots, which shows biclustering results for user-conversation relationships (i.e., which users participate in which conversations at what levels of involvement) mined from a large corpus of chatting data. In addition to showing biclusters, it also provides some statistical information of the entire data at the top.

Bicluster encoding. A bicluster, which consists of two sets of entities and their weighted relationships, is displayed in a row . The relationships are shown as small semi-transparent (since they may overlap) orange lines in a rectangle in the middle; and the two sets of entities are shown as gray circles with line patterns to help differentiate entities, each set on one side.

Each weighted *relationship* in a bicluster is represented by the position of a line, e.g., from minimum to maximum values, left to right. Thus, this encoding indicates a distribution of the relationship weights associated with the bicluster. For example, the first row in Fig. 1A represents a 3-by-2 bicluster, and some of its six relationships have similar weights that overlap with each other.

For *entities*, the line patterns on the circles, to some extent, represent their identity (i.e., the same entity must have the same pattern). To generate a unique pattern for each entity, we first hash-map an entity's ID to a 32-bit number and then compute the degrees of three lines from different portions of the hashed number. The circle color density of an entity is mapped to the number of biclusters that the entity belongs to (i.e., bicluster membership). The darker the color is, the more biclusters the entity is in. For example, in the first row of Fig. 1A, the first and third entities on the left have darker colors, implying that they are shared by many other biclusters (T1). This can help an analyst choose which entities to explore further, since previous study has found that users tended to check high-frequency entities in biclusters [31].

Dataset overview. BiDots also provides an overview of the biclustering result compared to the original input consumed by the biclustering algorithm as well as some indicators for the algorithm parameters. At the top, the circles on both sides

represent the minimum entity size requirement for the biclustering. For example, in Fig. 1A, an analyst wants to find biclusters whose sizes are at least 3-by-2. The two bars in the middle

indicate the percentage of entities included in the biclustering result with respect to the total numbers of entities in the data, where the darker portion represents the percentage of entities included. Actual numbers are also displayed in red ink. At the bottom of this overview, two overlaid area charts present the distributions of all the relationships in the data (lighter orange) and all the relationships included in the biclusters (darker orange)

. The charts share the same relationship weight axis with the rectangles of the biclusters' relationships below (i.e., from 0 to 1, left to right). In addition, a black vertical line indicates a threshold that an analyst sets for ignoring relationships with weights below that value (e.g., 0.05).

Design alternatives. To represent the uniqueness of entities, other types of textures/glyphs, or color hues, or simply drawing IDs could be used. We choose line patterns because they are more scalable than color hues in terms of the visual variable length (i.e., the available number of bins to properly encode a variable) [4], and it may interfere with color density that is used for encoding cluster membership counts. Moreover, displaying ID strings may take bigger space when they are long. However, when the number of unique entities is not large, color hues might be a better choice, which provides a clearer overview. It is left for future work to investigate which visual encodings are more effective and study the perceptual properties of the encodings based on prior work [14, 36]. For visualizing weighted relationships in a bicluster, other visualizations of data distributions, such as histogram, box-plot, and jittered strip plot, can be easily applied to the current visualization. We choose the current design because a histogram or box-plot aggregates data, preventing an analyst from selecting and viewing individual relationships. Also, while a jittered strip plot reduces overlap, it visualizes a relationship with a tiny dot, making interaction difficult. But these designs can be easily integrated as alternatives.

5.2 Interaction

BiDots offers a range of fluid interactions to assist an analyst with making sense of the biclustering results and discovering insights in coordinated relationship analysis.

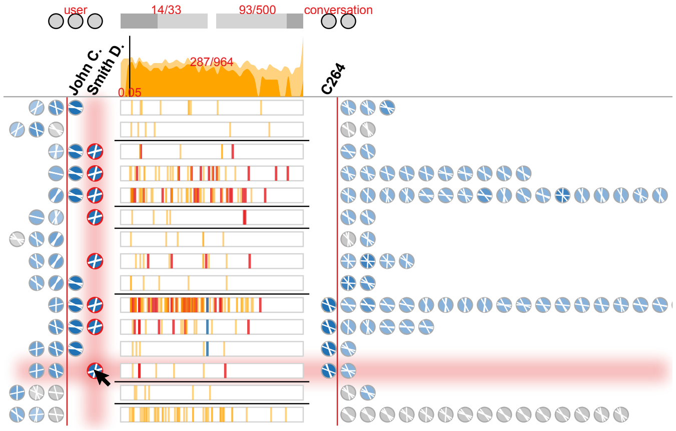


Fig. 5. An analyst pins Smith D. and John C. in the user domain and C264 in the conversation domain, and further groups the biclusters based on their similarities. As Smith D. is hovered over in one bicluster, the same entity in all other biclusters as well as their associated relationships are highlighted to reveal relevancy.

Pinning. When an analyst becomes interested in a particular entity, she can bring the entity out (pinning) to form an individual column that better displays its memberships to all the biclusters. For example, in Fig. 1B, Smith D., one of the darker gray entities in Fig. 1A is pinned by an analyst, as it belongs to many of the biclusters. After pinning, the entity is in a separate column outside the red border line, and an analyst can clearly see which biclusters contain this entity.

Further, all the biclusters containing the pinned entity are *deemed* to be of interest by an analyst. Thus, all entities belonging to these biclusters are colored differently in blue as opposed to the rest that are in gray. Similarly, the color density of each entity is mapped to the membership count of that entity in all the deemed-interesting biclusters. Usually, due to complicated entity overlaps across biclusters, these entities may appear in biclusters other than the ones where the pinned entity presents. This visual encoding can help an analyst further identify what is the next most-shared entity in the biclusters related to the pinned entity (T1). For example, in Fig. 1B, the first entity, John C., in the user domain of the first bicluster is in darker blue, indicating that it appears most frequently in conversations where Smith D. participates in. But John C. may appear in other biclusters as well. As shown in Fig. 5, an analyst can then pin this entity and the color coding updates accordingly based on the new set of deemed-interesting biclusters (i.e., biclusters containing both Smith D. and John C.). Similarly, an analyst can explore the other domain on the right, conversation, by adding entities of interest.

The above design allows for on-demand browsing of entities and their associations to biclusters, and guiding an analyst to select important entities (i.e., common shared ones) during the exploration. Instead of showing all entities in a matrix-like representation that reveals all their bicluster memberships (which equals to pinning all entities), this approach is more scalable; otherwise, a very large visualization could be generated. For convenience, an analyst can pin all the entities in one bicluster at a time using a modifier key in BiDots. Moreover, this flexible pinning interaction helps an analyst to interactively compare or identify biclusters based on a set of entities that are of interest, (e.g., discovering biclusters containing all the concerned entities or sharing some common entities).

Visual linking. In addition to pinning, some visual linking features are available in BiDots to unveil the topological associations between entities, relationships, and biclusters, in a quick and easy manner (T1). For example, hovering over an entity highlights the same entity in other biclusters and the relationships connected to the entity in each bicluster (Fig. 5). At the same time, two red ribbons emphasize the row and column of the hovered-over entity to ease the exploration. Similarly, hovering over a relationship reveals associated entities and the same relationship in other biclusters.

Ordering. BiDots supports ordering biclusters (vertically) in a number of ways in order to facilitate an analyst with finding biclusters with extreme attributes (T2). In our current implementation, sorting by bicluster size, average relationship weight, and relationship weight variance, either in increasing or decreasing order, is provided. However, other criteria can be easily built-in. In addition, an analyst can order the biclusters by the membership of a particular pinned entity. For example, in Fig. 1B, clicking the entity label Smith D. pushes all the associated biclusters to the top.

Filtering. An analyst sometimes may not find everything needed in the biclustering results, for example, some trivial relationships that appear everywhere in the data may mask the needed information. To reduce the visual complexity and hide unnecessary information, BiDots allows for filtering out specific entities or relationships based on an analyst's selection (T2). In particular, filtered entities are shown as hollow circles ○ and their connected relationships are shown in semi-transparent gray ▯.

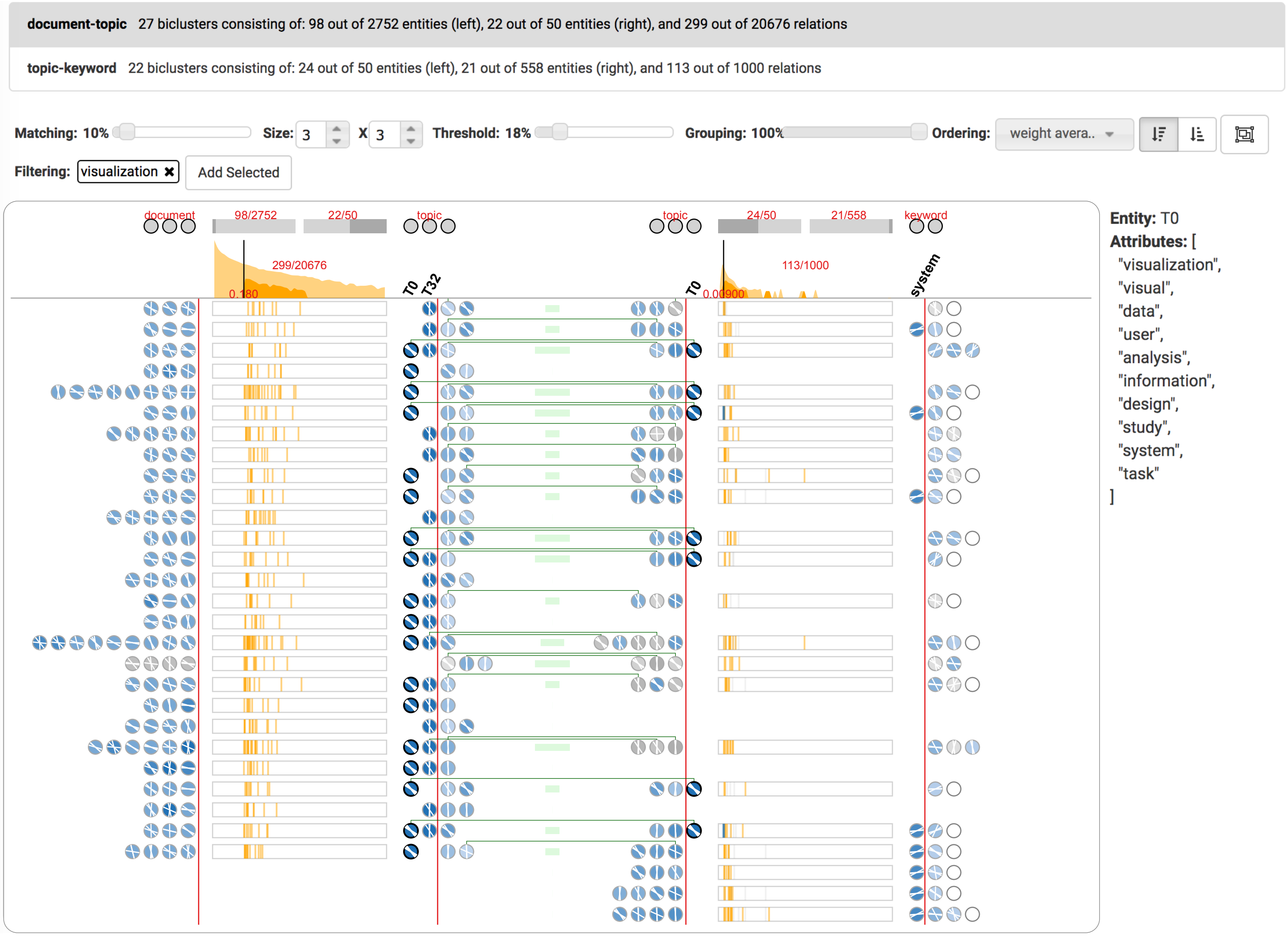
Grouping. To support similarity analysis (T2), BiDots provides an interactive grouping function, in which similar biclusters are placed together in the visualization. To measure how close two biclusters are, we employ a metric consisting of three components: similarities of entity sets in two domains, and similarity of relationships. Entity set similarity is measured based on the Jaccard distance (a common distance metric for discrete sets [5]), and relationship similarity is based on the weighted Jaccard distance (an extension to the Jaccard distance for continuous variables [5]). The overall similarity of two biclusters is a weighted sum of the above three quantities (i.e., two entity set similarities, and one relationship similarity). Equal weights are used in this implementation. Based on the similarity measure, we employ an agglomerative clustering approach, with a distance threshold interactively set by an analyst (Fig. 6), to group biclusters. This method does not require a predefined number of groups [37]. Fig. 5 indicates the grouped biclusters where bicluster groups are separated by black lines. An analyst can further order these groups based on the average bicluster size of a group, relationship weight, etc. Also, biclusters within each group can be ordered individually with these criteria.

5.3 Integration of Multiple Biclustering Results

In many real-world applications, an analyst needs to explore biclusters across more than two domains and associate biclusters with shared domains to form chains (as in the example shown in Fig. 2), which can reveal deeper insights in the data (T2). However, combining the results of multiple biclustering processes is not an easy task, because entity sets in the common domain of two biclusters may not always match exactly. Only showing bicluster chains with exactly matched entity sets in common domains may restrict the discovery of hidden connections.

To address this issue, we develop an algorithm in BiDots to combine biclusters computed from different pairs of domains, containing two parts: *matching* and *chaining*. First, the algorithm aims to generate bicluster pairs in a greedy approach from biclusters with shared domains, based on similarities between them (see Algorithm 1). Then, the algorithm connects the matched bicluster pairs into longer bicluster chains (see Algorithm 2). The matching part implements one-to-one matching, and more complicated methods are left for future studies. In BiDots, an analyst can interactively adjust the matching threshold to explore bicluster chains with different minimum matching strengths, however, chains cannot be interactively modified. Indirectly, the matching threshold determines how many biclusters can be included into a bicluster chain and how many chains can be generated.

Fig. 6 shows a visualization of chained biclustering results of document-topic (left) and topic-keywords (right) relationships where topics were computed based on LDA topic modeling [3] over abstracts of a publication corpus of VIS conferences [18]. The common domain—topic—is used for matching the biclusters on both sides and forming bicluster chains. This allows an analyst to explore topics that frequently appear in sets of documents and further trace the topics to obtain keywords that frequently co-exist in topics. For example, in Fig. 6, T0 (selected and pinned) and T32 (pinned) are two frequently mentioned topics; for T0, visualization is one of the most common



document-topic 27 biclusters consisting of: 98 out of 2752 entities (left), 22 out of 50 entities (right), and 299 out of 20676 relations

topic-keyword 22 biclusters consisting of: 24 out of 50 entities (left), 21 out of 558 entities (right), and 113 out of 1000 relations

Matching: 10% Size: 3 X 3 Threshold: 18% Grouping: 100% Ordering: weight avera..

Filtering: visualization ✖ Add Selected

Entity: T0
Attributes: [
"visualization",
"visual",
"data",
"user",
"analysis",
"information",
"design",
"study",
"system",
"task"
]

Fig. 6. The full user interface of BiDots. The visualization displays bicluster chains computed from the results of two biclustering processes on the document-to-topic and topic-to-keyword relationships respectively. The relationships are mined from abstracts of a publication corpus of VIS conferences (from 1990 to 2015) [18] using LDA topic modeling.

Algorithm 1: Bicluster matching

```
Data: Two lists of biclusters $A = \{a_i\}$ and $B = \{b_j\}$.
Result: A list of matched bicluster pairs $P = \{(a_i, b_j)\}$ where $a_i \in A$ or $nil$ and
        $b_j \in B$ or $nil$.
$S \leftarrow \{s_{ij}\}$ where $s_{ij} = \texttt{Similarity}(a_i, b_j)$;
sort $S$ in descending order;
$A' \leftarrow \{\}, B' \leftarrow \{\}, P \leftarrow \{\}$;
foreach $s_{ij} \in S$ do
    if $s_{ij} < threshold$ then
        break;
    if $i \notin A'$ and $j \notin B'$ then                  // a matched bicluster pair
        add $(a_i, b_j)$ to $P$;
        add $i$ to $A'$, add $j$ to $B'$;
foreach $a_i \in A$ do                                     // unmatched biclusters
    if $i \notin A'$ then
        add $(a_i, nil)$ to $P$;
foreach $b_j \in B$ do                                     // unmatched biclusters
    if $j \notin B'$ then
        add $(nil, b_j)$ to $P$;
```

keywords, which is filtered out, and the next most common keyword is `system` (pinned).

The visual representation of bicluster chains is similar to that of biclusters described earlier. More particularly, a bicluster chain is shown in a row, and entities in the shared domain are connected with lines . An empty space in a row indicates that there is no bicluster matched in a chain for that pair of domains (or relationship space). Also, the matching score of two biclusters is indicated by the length of a green bar between them.

Algorithm 2: Bicluster chaining

```
Data: A list of bicluster pair sets $X = \{P_i, i = 1 \dots n\}$ where $P_i = \{(a_k^i, b_k^i)\}$.
Result: A list of bicluster chains $Y = \{C_j\}$ where $C_j = (c_1^j, c_2^j, \dots, c_{n+1}^j)$ and
        $(c_i^j, c_{i+1}^j) \in P_i$ or $(nil, nil)$.
$Y \leftarrow P_1$;
for $i = 2 \dots n$ do
    foreach $(a_k^i, b_k^i) \in P_i$ do
        if $a_k^i = nil$ then     // add placeholders when starting a chain
            add $(nil_1, nil_2, \dots, nil_i, b_k^i)$ to $Y$;
        else
            find $C_j \in Y$ where $c_i^j = a_k^i$;
            append $b_k^i$ to the end of $C_j$;
    foreach $C_j \in Y$ do          // add placeholders for ending chains
        if Length($C_j$) $< i + 1$ then
            append $nil$ to the end of $C_j$;
```

Similar to manipulating one set of biclusters, an analyst can order the bicluster chains by different quantities of biclusters (e.g., size, average relationship length, etc.) from any of relationship spaces. For example, Fig. 6 orders bicluster chains by average relationship weight of the document-topic relationships (left), so gaps are observed on the topic-keyword relationships (right) indicating that those chains only have one bicluster.

5.4 Interactive Configuration of Biclustering Algorithms

Performing real-time interactive analysis of coordinated relationship is essential in practical applications, allowing an analyst to iteratively try different biclustering parameters and consolidate findings (T3). BiDots supports this with a toolbar area on the user interface of the whole system (Fig. 6). An analyst can adjust the biclustering process in real-time on each pair of domains individually with different algorithm parameters, such as the minimum weight threshold of relationships and the minimum size of biclusters. In addition, some of the interactions mentioned earlier can be operated through this toolbar area, such as setting thresholds for bicluster grouping and matching and for filtering.

6 Usage Scenario

To demonstrate the usefulness and effectiveness of BiDots in exploring biclusters, we describe an investigative document analysis scenario with the Atlantic Storm dataset [15]. The corpus contains 111 fictional intelligence reports which can be used to discover a hidden plot of terrorists transferring biological agents to the USA. We extract 716 entities using NLTK [1] (e.g., people, organizations, places, etc.) from the reports, and compute relationship weights between entities based on the numbers of co-occurrences and normalized word distances in text (the higher the weights, the stronger the relationship that two entities have). We employ LCM [34] for discovering biclusters from the relationship spaces. To support the analysis of this dataset, we develop a customized text view which can load corresponding reports of a bicluster and highlight entities in the text (Fig. 7A).

Imagine that Emma, an investigator, is asked to uncover the potential threats and key players from the large collection of reports. She launches BiDots by first loading the organization, person, and place domains. In total, there are 127 organizations, 216 people, and 254 places; and 715 organization-to-person and 1893 person-to-place relationships, discovered from the reports.

Emma initially sets the minimum bicluster size to 2-by-2 for both domain pairs, however, too many biclusters are extracted in the person-to-place relationships, which is overwhelming. Thus, she increases the minimum relationship weight, ignoring person-to-place relationships with lower weights (i.e., less co-occurrences) (T3). With the new settings, BiDots displays 50 biclusters for organization and person domain pairs, and 25 biclusters for person and place domain pairs.

Quick identification of key entities and relationships. Emma starts her investigation by looking into the organizations. By observing the color density of entities, she finds that the two most common organizations shared across all biclusters are `FBI` and `CIA` (i.e., with darker gray), which is trivial since they appear everywhere because of the report sources (T1). Thus, she filters out these two organizations and discovers that the next commonly-shared entity is `Al Qaeda` (T2). She then pins this organization, forming an extra column of it, and observes that the darkest blue organization in the rest is `Central Russia Airlines` (T1). This means that `Central Russia Airlines` appears most frequently in all the biclusters that contain `Al Qaeda`. So she pins `Central Russia Airlines` as well.

Next, Emma shifts her attention to the person domain. Very easily, from the dark blue entities, she identifies `Safrygin` and `Pyotr Safrygin` are the most shared people in the biclusters consisting of the above two organizations (T1). Reading some of the reports reveals that `Safrygin` who works at `Central Russia Airlines` has contact to `Al Qaeda`. In fact, `Safrygin` and `Pyotr Safrygin` are the same person but the name-entity detection algorithm did not associate them. Thus, Emma pins these two entities in the person domain and clicks the headers to reorder biclusters by bringing relevant ones to top of the list (Fig. 7B) (T1). She then browses a few reports in these biclusters, and obtains information that both `Safrygin` and another suspicious person `Bugarov`, who is a bio-weapon scientist, were formerly employed by a Russian institute called `VECTOR` (Fig. 7A). From the entities of these biclusters, Emma pins `Bugarov` and `VECTOR` for further exploration.

Combination of insights from multiple relationships and domains. Tracing `Safrygin` and `Bugarov` through bicluster chains (by following the green lines), Emma decides to investigate the person-to-place relationships (T1). Using similar interactions, i.e., pinning the entities and observing the darker blue ones, she identifies `Moscow` and `Havana` as frequently mentioned places in the biclusters containing `Safrygin` and `Bugarov`, which further confirms their connections in Russia and reveal potential suspicious activities in Cuba (Fig. 7C) (T1). Also, the relationship weights between `Bugarov` and `Moscow` and `Havana` are relatively high. She then reads more information in the reports of the biclusters that `Bugarov` has been seen frequently traveling between Russia and Cuba.

Emma then applies grouping to the biclusters with a threshold of 0.9, and immediately finds that biclusters containing `Safrygin`, `Bugarov`, and `Moscow` are assigned to the same group, further indicating their close relationships (Fig. 7D) (T2 and T3). Next, Emma's attention is drawn by another group that contains biclusters of `Havana` and a few other entities in darker blue. Soon she discovers a group of three people, `Arze`, `Escalante`, and `Morales`, which also connect back to entities in the organization-person biclusters via the chains (T1 and T2). However, they have no overlap with the formerly pinned organizations, which surprises Emma about how deep it hides. Emma pins another darker blue place entity `Cuba` and removes `Moscow` in order to investigate suspicious activities there. She quickly discovers `Santo Domingo` from one of the most shared entities, and the associated reports mention they were seen together in a park in `Santo Domingo` (Fig. 7E) (T1).

Deeper and broader analysis of biclusters. Emma suspects that there is more, but not much information can be found in the reports. To conduct further investigation, she lowers the relationship weight threshold, therefore many more biclusters (70 in total) are extracted from the person-to-place relationships (T3). Emma then sorts the biclusters based on their average relationship weight (T2). As expected, the pinned entities (e.g., `Arze`, `Escalante`, etc.) appear in many of the higher-weight biclusters. However, the one with the highest average relationship weight does not contain any of them (Fig. 7E). By exploring the relationship weight distribution, Emma finds that the highest weight is between `Hanif` and `Nassau`, indicating that they co-occur in many reports (T1). So Emma selects the bicluster to read the associated reports, and discovers that a suspicious person `Sufaat` frequently runs between `Nassau` and `Santo Domingo` and was seen helping another person `Hanif` transfer a "packet" labelled in Russian to the USA (Fig. 7F).

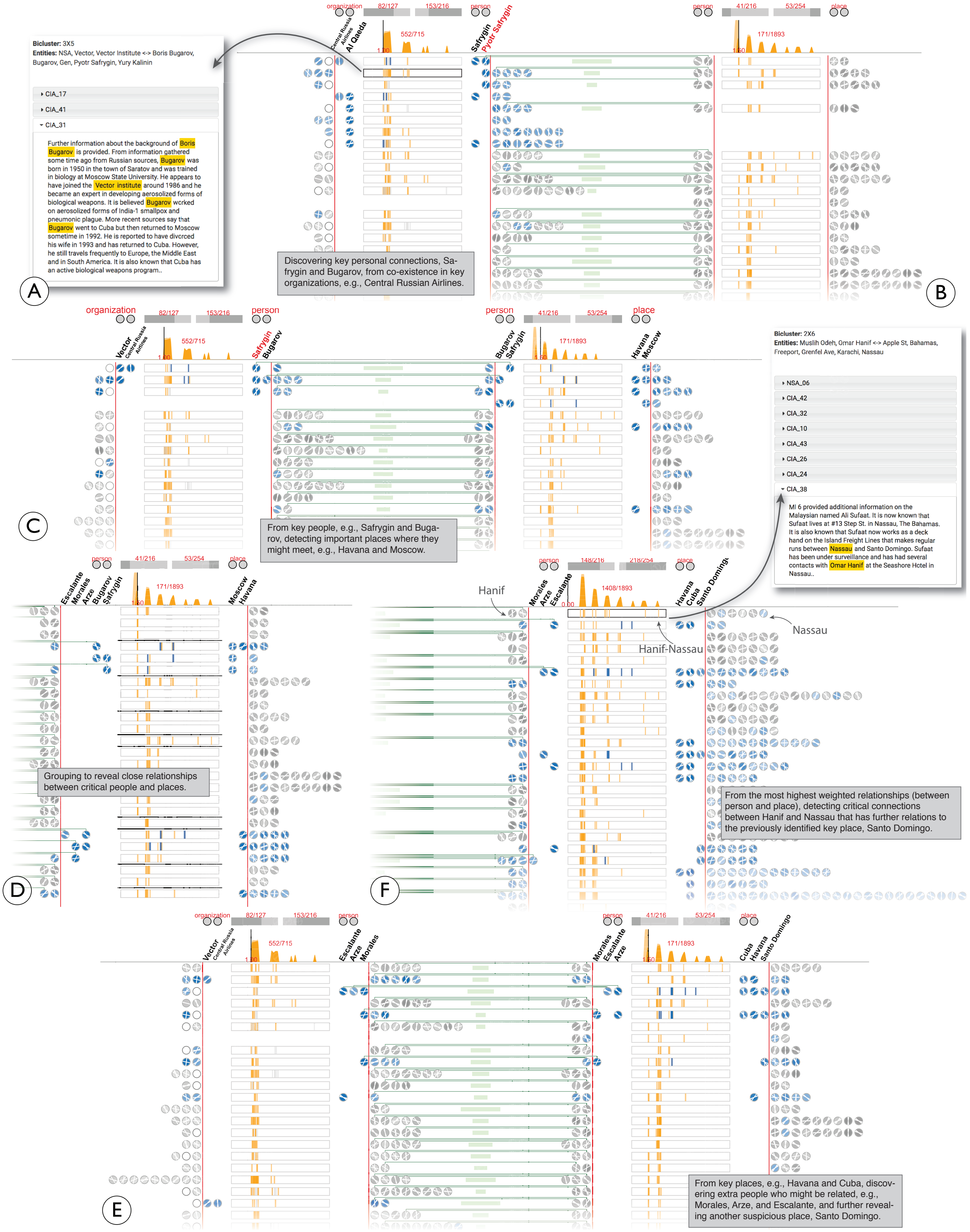
To connect the dots, Emma pins `Nassau`, clicks the labels of `Nassau` and `Santo Domingo` to bring corresponding biclusters to the top of the list, and thus identifies a bicluster that contains both place entities (T1). She then reads the reports; one is regarding a phone call between `Escalante` and `Morales` about passing some "medical supplies" to a guy named `Sufaat`.

Hypothesis generation. Therefore, Emma reaches the conclusion of her investigation: `Safrygin` and `Bugarov` help `Al Qaeda` ship biological agents from Russia to South America, and furthermore, with the assistance of a group of local people to transfer them to the USA for terrorist activities.

7 Initial Expert Feedback

To better understand the usefulness of BiDots, we carried out an interview with the experts who helped us to develop the analytical tasks (Sect. 3). During the session, we first demonstrated BiDots and then asked the experts to use it to explore biclusters extracted from the VIS publication dataset (Fig. 6), since the data scenario is easily understood, while they are from non-visualization research fields. We conducted observations and solicited feedback from the experts.

Overall, the experts highly appreciated BiDots, especially the compelling visual design and interactivity, commenting *"It allows me to easily see the correlations between biclusters, such as entity overlaps. I can follow some entities to extract deeper information from the results. In the past, I have had to manually examine the outputs."* They successfully identified popular, frequently co-occurring topics and keywords in the publication corpus as well as document groups that are coherent in content. The experts enjoyed the similarity analysis features, such as grouping, mentioning that *"This is critical because the computed biclusters only reveal partial aspects in coordinated relationship analysis. There are entities we want to include or not. You need to combine a couple of closely related biclusters."* They further suggested that *"ordering biclusters based on similarity to a selected one"* would be a helpful function to add. Another aspect that the experts liked was the interactive features of adjusting different parameters for



Bicluster: 3X5
Entities: NSA, Vector, Vector Institute <-> Boris Bugarov, Bugarov, Gen, Pyotr Safrygin, Yury Kalinin

- CIA_17
- CIA_41
- CIA_31

Further information about the background of Boris Bugarov is provided. From information gathered some time ago from Russian sources, Bugarov was born in 1950 in the town of Saratov and was trained in biology at Moscow State University. He appears to have joined the Vector institute around 1986 and he became an expert in developing aerosolized forms of biological weapons. It is believed Bugarov worked on aerosolized forms of India-1 smallpox and pneumonic plague. More recent sources say that Bugarov went to Cuba but then returned to Moscow sometime in 1992. He is reported to have divorced his wife in 1993 and has returned to Cuba. However, he still travels frequently to Europe, the Middle East and in South America. It is also known that Cuba has an active biological weapons program..

Discovering key personal connections, Safrygin and Bugarov, from co-existence in key organizations, e.g., Central Russian Airlines.

Bicluster: 2X6
Entities: Muslih Odeh, Omar Hanif <-> Apple St, Bahamas, Freeport, Grenfel Ave, Karachi, Nassau

- NSA_06
- CIA_42
- CIA_32
- CIA_10
- CIA_43
- CIA_26
- CIA_24
- CIA_38

MI 6 provided additional information on the Malaysian named Ali Sufaat. It is now known that Sufaat lives at #13 Step St. in Nassau, The Bahamas. It is also known that Sufaat now works as a deck hand on the Island Freight Lines that makes regular runs between Nassau and Santo Domingo. Sufaat has been under surveillance and has had several contacts with Omar Hanif at the Seashore Hotel in Nassau..

From key people, e.g., Safrygin and Bugarov, detecting important places where they might meet, e.g., Havana and Moscow.

Grouping to reveal close relationships between critical people and places.

From the most highest weighted relationships (between person and place), detecting critical connections between Hanif and Nassau that has further relations to the previously identified key place, Santo Domingo.

From key places, e.g., Havana and Cuba, discovering extra people who might be related, e.g., Morales, Arze, and Escalante, and further revealing another suspicious place, Santo Domingo.

Fig. 7. An intelligence officer uses BiDots to investigate terrorist activities described in the Atlantic Storm dataset [15]. After extracting entities from the data and building relationships based co-occurrence, biclusters of organization-to-person, and person-to-places relationships are mined and further connected as bicluster chains for complex coordinated relationship analysis of the three concerned domains.

biclustering, matching, and grouping. They said that it streamlined their data exploration process by allowing for efficiently trying different threshold settings.

The interview also revealed a couple of places to improve BiDots. Initially our experts were confused that different circles might represent the same entity in different rows, although the line patterns were helpful. The level of confusion decreased gradually as they started to manipulate the visualization by hovering over and pinning entities. The experts also commented that it was not easy to see details of a specific relationship (e.g., the entities to which it is connected) unless they moused over it. Moreover, the experts demanded a searching function for entities, relationships, etc., to facilitate data exploration. Limitations and trade-offs of the BiDots design is further discussed in the next section.

8 Discussion

In this section, we discuss the scalability and limitations of BiDots and lay out several promising directions for generalizing the design.

8.1 Scalability and Comparison to Existing Techniques

Overall, BiDots is more compact than existing popular approaches utilizing a cluster-driven design, such as Bixplorer [8] and BiSet [32] (Fig. 3). Bixplorer may generate a much larger visualization compared to BiDots because every bicluster is represented as a 2D matrix that is not space effective. Compared to BiSet, BiDots is more scalable if the numbers of unique entities in the two domains become larger, (i.e., the number of biclusters in the data is sparse). In this case, BiSet could have two large vertical lists of entities, whereas the visual size of BiDots equals the number of biclusters, which could be much smaller. In the situation of dense data (e.g., the number of biclusters is smaller than the number of entities), the visual sizes of BiDots and BiSet are the same because they both display biclusters in the center.

However, BiDots is not without scalability issues. First, it might not be effective for a very large number of biclusters, although the ordering and grouping functions can facilitate an analyst's exploration. An extension would be to add support for aggregating multiple biclusters based on their similarity. Moreover, if the size of biclusters grows (i.e., each containing many entities), the visualization potentially becomes large because rows (biclusters) can be very wide. In this case, BiSet [32], Jigsaw [29], and similar list-based design would be more effective, although the links representing relationships may be cluttered. For BiDots, aggregating multiple entities into one or employing more condensed visual encoding (e.g., pixel-based techniques) may address this issue. Larger bicluster sizes also raise a problem for showing bicluster chains, as the lines connecting the same entity in a shared domain would be crowded. In this case, edge bundling [13] can be applied to reduce visual complexity. A third aspect related to scalability is the number of domains, in which Bixplorer [8] and similar matrix-based design would be more scalable than BiDots or BiSet. However, normally in practice, people do not explore too many domains at the same time.

8.2 Limitations and Trade-Offs

Similar to the drawbacks of many other bicluster visualizations (e.g., [8]), entities are duplicated, which may confuse an analyst. While we generate a random line pattern for each entity, the patterns may not be easily distinguished and pattern collision does exist when the number of unique entities increases. Pinning in BiDots temporally resolves this problem but it requires space and user interaction, and in some sense, it helps an analyst to compare biclusters with a fixed common entity set. It is not practical to pin all the entities (which essentially generates a visualization in a similar vein to the matrix-based approaches [10, 11,27]) when there are many unique entities. This mirrors the design trade-off between entity-centric and relationship-centric approaches discussed by Sun et al. [32]. As an extension to BiDots, allowing an analyst to bookmark certain objects, for example, assigning a specific color to an entity or relationship, may help an analyst's short-term memory during data exploration.

Moreover, in BiDots, weighted relationships are shown as line strips to indicate their distributions in a bicluster. This visual encoding is minimal and effective, but user interactions (e.g., mouse hovering) are needed to find out which two entities are connected by a specific relationship or vice versa. Matrix-based visualizations, such as [8], can better display the relationships with matrix cells, but requires much more screen real estate. The line strips also reveal similarity of biclusters at the relationship level. Thus, there is a trade-off between collapsing all the relationships into a one-dimensional form and keeping them in two dimensions.

Lastly, the coordinated relationship analysis based on complete biclusters implemented in BiDots may overlook some outliers or less frequent patterns. But, on the other hand, presenting too many results (which may be important or not) to an analyst could easily overwhelm them. It is left for future work to incorporate other forms of biclustering (e.g., discovering partial biclusters) or relationship analysis into BiDots.

8.3 Generalization of BiDots

While we describe specific visual encodings of entities and relationships in this paper, the BiDots design can be used as a generic visualization framework for coordinated relationship analysis. Slight changes can be applied to the visual representation for adaptation to different applications, while the visual layout and user interactions can remain the same. For example, the size of an entity circle can be used to represent the number of biclusters that the entity belongs to, rather than color density. The entities need not be shown as circles, and could instead be other visual objects. Similarly, relationships can be visualized with other types of charts showing distributions.

Moreover, although our focus in this paper is to address the scenario of weighted biclusters, which is underexploited in the literature of bicluster visualization, BiDots can be extended to support the scenario of binary relationships (i.e., existing or non-existing). In that case, the relationship plot between entities would reduce into one line, because relationship weights are identical (e.g., 1.0) in all biclusters. Thus, it is redundant to draw relationships. However, the layout method, interactions (e.g., pinning, ordering, grouping, etc.), and construction of bicluster chains in BiDots are applicable to empower users for the exploration of non-weighted biclusters, without loosing any of the positive features in BiDots, such as compactness.

A third aspect is that BiDots can be applied to the analysis of biclusters in other application domains, such as bio-informatics, although we use text analytics as examples throughout the paper. It can be used for the cases of partial biclusters (i.e., not complete bipartite graphs), when proper analysis methods are integrated. This is actually equivalent to the case where some relationships or entities are filtered out in BiDots. However, additional visual design experiments are needed to distinguish partial and closed biclusters as well as indicating "missing" relationships and corresponding entities.

9 Conclusion and Future Work

We have described a visualization technique, BiDots, for performing coordinated relationship analysis based on biclustering results. BiDots supports the exploration of weighted relationships which has not been emphasized in the bicluster visualization literature. The proposed visualization is more compact and scalable, and provides a set of fluid interactions, such as pinning, ordering, filtering, and grouping, to facilitate a user's exploration. A greedy bicluster chaining algorithm has been proposed for combining biclustering results of relationships from different pairs of domains in a dynamic manner. We have also outlined a set of analytical tasks that ground our design of BiDots, based on the literature and expert user interviews. Finally, a usage scenario with a benchmark investigative analysis dataset [15] has been described to demonstrate the usefulness and effectiveness of BiDots.

There are several promising directions for future research. First, we plan to conduct quantitative user studies to further evaluate BiDots to complement the qualitative study in this paper. We would also like to experiment with different visual encodings of entities and relationships under the general design framework of BiDots for situating the technique into different application domains. Lastly, we want to enhance the scalability of BiDots by incorporating the ideas discussed earlier, such as visual aggregations of biclusters and entities.

References

[1] NLTK. http://www.nltk.org/, accessed in 2017.

[2] B. Alsallakh, L. Micallef, W. Aigner, H. Hauser, S. Miksch, and P. Rodgers. The state-of-the-art of set visualization. *Computer Graphics Forum*, 35(1):234–260, 2016. doi: 10.1111/cgf.12722

[3] D. M. Blei. Probabilistic topic models. *Communications of the ACM*, 55(4):77–84, 2012. doi: 10.1145/2133806.2133826

[4] S. Carpendale. Considering visual variables as a basis for information visualisation. Technical report, University of Calgary, 2003.

[5] F. Chierichetti, R. Kumar, S. Pandey, and S. Vassilvitskii. Finding the jaccard median. In *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 293–311, 2010.

[6] M. Dörk, N. H. Riche, G. Ramos, and S. Dumais. Pivotpaths: Strolling through faceted information spaces. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2709–2718, 2012. doi: 10.1109/TVCG.2012.252

[7] N. Elmqvist, P. Dragicevic, and J. D. Fekete. Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1539–1148, 2008. doi: 10.1109/TVCG.2008.153

[8] P. Fiaux, M. Sun, L. Bradel, C. North, N. Ramakrishnan, and A. Endert. Bixplorer: Visual analytics with biclusters. *Computer*, (8):90–94, 2013. doi: 10.1109/MC.2013.269

[9] S. Ghani, B. C. Kwon, S. Lee, J. S. Yi, and N. Elmqvist. Visual analytics for multimodal social network analysis: A design study with social scientists. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2032–2041, Dec 2013. doi: 10.1109/TVCG.2013.223

[10] G. A. Grothaus, A. Mufti, and T. Murali. Automatic layout and visualization of biclusters. *Algorithms for Molecular Biology*, 1(1):1–15, 2006. doi: 10.1186/1748-7188-1-15

[11] J. Heinrich, R. Seifert, M. Burch, and D. Weiskopf. Bicluster viewer: A visualization tool for analyzing gene expression data. In G. Bebis, R. Boyle, B. Parvin, D. Koracin, S. Wang, K. Kyungnam, B. Benes, K. Moreland, C. Borst, S. DiVerdi, C. Yi-Jen, and J. Ming, eds., *Proceedings of the International Symposium Advances in Visual Computing*, pp. 641–652. Springer Berlin Heidelberg, 2011. doi: 10.1007/978-3-642-24028-7_59

[12] N. Henry, J.-D. Fekete, and M. McGuffin. NodeTrix: a hybrid visualization of social networks. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1302–1309, 2007. doi: 10.1109/tvcg.2007.70582

[13] D. Holten and J. J. Van Wijk. Force-directed edge bundling for graph visualization. *Computer Graphics Forum*, 28(3):983–990, 2009. doi: 10.1111/j.1467-8659.2009.01450.x

[14] D. Holten, J. J. V. Wijk, and J.-B. Martens. A perceptually based spectral model for isotropic textures. *ACM Transactions on Applied Perception*, 3(4):376–398, 2006. doi: 10.1145/1190036.1190039

[15] F. Hughes. Discovery, proof, choice: The art and science of the process of intelligence analysis, case study 6, "all fall down". 2005.

[16] F. Hughes and D. Schum. Discovery, proof, choice: the art and science of the process of intelligence analysis-preparing for the future of intelligence analysis. *Washington, DC: Joint Military Intelligence College*, 2003.

[17] A. Inselberg and B. Dimsdale. Parallel coordinates: A tool for visualizing multi-dimensional geometry. In *Proceedings of the 1st Conference on Visualization*, pp. 361–378, 1990.

[18] P. Isenberg, F. Heimerl, S. Koch, T. Isenberg, P. Xu, C. D. Stolper, M. M. Sedlmair, J. Chen, T. Möller, and J. Stasko. http://www.vispubdata.org/site/vispubdata/, accessed in 2017.

[19] M. Kapushesky, P. Kemmeren, A. C. Culhane, S. Durinck, J. Ihmels, C. Korner, M. Kull, A. Torrente, U. Sarkans, J. Vilo, and A. Brazma. Expression profiler: next generation–an online platform for analysis of microarray data. *Nucleic Acids Research*, 32:W465–W470, 2004. doi: 10.1093/nar/gkh470

[20] B. Lee, C. Plaisant, C. S. Parr, J.-D. Fekete, and N. Henry. Task taxonomy for graph visualization. In *Proceedings of the AVI workshop on BEyond time and errors novel evaluation methods for information visualization - BELIV*, 2006. doi: 10.1145/1168149.1168168

[21] A. Lex, H.-J. Schulz, M. Streit, C. Partl, and D. Schmalstieg. VisBricks: Multiform visualization of large, inhomogeneous data. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2291–2300, 2011. doi: 10.1109/tvcg.2011.250

[22] A. Lex, M. Streit, C. Partl, K. Kashofer, and D. Schmalstieg. Comparative analysis of multidimensional, quantitative data. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1027–1035, 2010. doi: 10.1109/tvcg.2010.138

[23] M. Liu, J. Shi, Z. Li, C. Li, J. Zhu, and S. Liu. Towards better analysis of deep convolutional neural networks. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):91–100, 2017. doi: 10.1109/tvcg.2016.2598831

[24] S. C. Madeira and A. L. Oliveira. Biclustering algorithms for biological data analysis: A survey. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 1(1):24–45, Jan. 2004. doi: 10.1109/TCBB.2004.2

[25] C. Partl, A. Lex, M. Streit, H. Strobelt, A.-M. Wassermann, H. Pfister, and D. Schmalstieg. Contour: Data-driven exploration of multi-relational datasets for drug discovery. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1883–1892, 2014. doi: 10.1109/TVCG.2014.2346752

[26] A. J. Pretorius and J. J. van Wijk. Visual inspection of multivariate graphs. *Computer Graphics Forum*, 27(3):967–974, 2008. doi: 10.1111/j.1467-8659.2008.01231.x

[27] R. Santamaria, R. Theron, and L. Quintales. BicOverlapper 2.0: visual analysis for gene expression. *Bioinformatics*, 30(12):1785–1786, 2014. doi: 10.1093/bioinformatics/btu120

[28] R. Santamara, R. Thern, and L. Quintales. Bicoverlapper: A tool for bicluster visualization. *Bioinformatics*, 24(9):1212, 2008. doi: 10.1093/bioinformatics/btn076

[29] J. Stasko, C. Görg, and Z. Liu. Jigsaw: Supporting investigative analysis through interactive visualization. *Information visualization*, 7(2):118–132, 2008. doi: 10.1057/palgrave.ivs.9500180

[30] M. Streit, S. Gratzl, M. Gillhofer, A. Mayr, A. Mitterecker, and S. Hochreiter. Furby: fuzzy force-directed bicluster visualization. *BMC Bioinformatics*, 15(Suppl 6):S4, 2014. doi: 10.1186/1471-2105-15-s6-s4

[31] M. Sun, L. Bradel, C. L. North, and N. Ramakrishnan. The role of interactive biclusters in sensemaking. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1559–1562, 2014. doi: 10.1145/2556288.2557337

[32] M. Sun, P. Mi, C. North, and N. Ramakrishnan. Biset: Semantic edge bundling with biclusters for sensemaking. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):310–319, 2016. doi: 10.1109/TVCG.2015.2467813

[33] M. Sun, C. North, and N. Ramakrishnan. A five-level design framework for bicluster visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1713–1722, 2014. doi: 10.1109/TVCG.2014.2346665

[34] T. Uno, T. Asai, Y. Uchida, and H. Arimura. An efficient algorithm for enumerating closed patterns in transaction databases. In *International Conference on Discovery Science*, pp. 16–31, 2004. doi: 10.1007/978-3-540-30214-8_2

[35] P. van der Corput, J. Arends, and J. J. van Wijk. Visualization of medicine prescription behavior. *Computer Graphics Forum*, 33(3):161–170, 2014. doi: 10.1111/cgf.12372

[36] C. Ware. *Information Visualization: Perception for Design*. Morgan Kaufmann, 2004.

[37] I. H. Witten, E. Frank, and M. A. Hall. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, 3rd ed., 2011.

[38] H. Wu, J. Vreeken, N. Tatti, and N. Ramakrishnan. Uncovering the plot: detecting surprising coalitions of entities in multi-relational schemas. *Data Mining and Knowledge Discovery*, 28(5-6):1398–1428, 2014. doi: 10.1007/s10618-014-0370-1

[39] M. J. Zaki and C.-J. Hsiao. Efficient algorithms for mining closed itemsets and their lattice structure. *IEEE Transactions on Knowledge and Data Engineering*, 17(4):462–478, 2005. doi: 10.1109/TKDE.2005.60

[40] J. Zhao, C. Collins, F. Chevalier, and R. Balakrishnan. Interactive exploration of implicit and explicit relations in faceted datasets. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2080–2089, 2013. doi: 10.1109/tvcg.2013.167

[41] J. Zhao, Z. Liu, M. Dontcheva, A. Hertzmann, and A. Wilson. Matrixwave: Visual comparison of event sequence data. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, 2015. doi: 10.1145/2702123.2702419