ParallelSpaces: Simultaneous Exploration of Feature and Data for Hypothesis Generation

Deokgun Park
Department of Computer Science
University of Maryland
College Park, MD, USA
intuinno@umd.edu

Jungu Choi
School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN, USA
Email: choi88@purdue.edu

Niklas Elmqvist
College of Information Studies
University of Maryland
College Park, MD, USA
elm@umd.edu

***Abstract*—We present ParallelSpaces, a novel method to explore bipartite datasets in both feature and data dimensions. This dyadic data is displayed as weighted bipartite graphs using scatterplots in two separated visual spaces, where each entity is positioned according to multi-dimensional properties of each entity or similarity in preferences. Selecting or navigating in one space is reflected in the other space, so that organic visual patterns can be formed to facilitate the characterization of underlying groupings. To aid visual pattern recognition we also overlay a contour plot based on kernel density estimation. We have implemented two instantiations of ParallelSpaces for (a) movie preferences, and (b) business reviews as web-based visualizations. To validate the method, we performed a qualitative user study involving eleven participants using these web-based tools to explore data and collect deep insights.**

***Keywords*-Multimodal graphs, multivariate graphs, social network analysis, kernel density estimation.**

I. Introduction

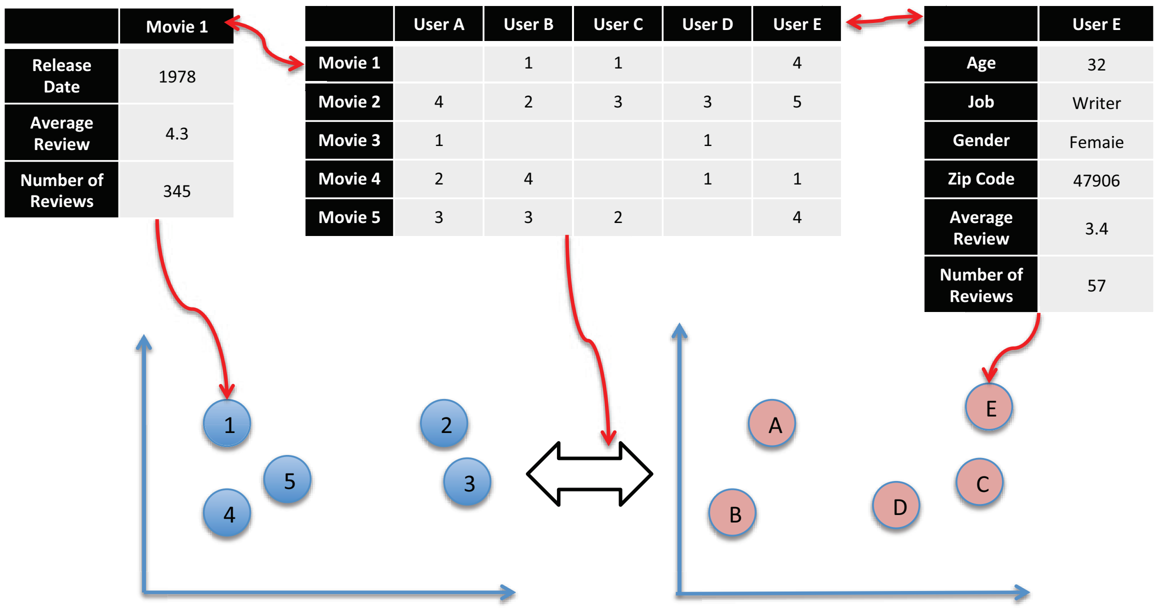
Bipartite graphs are common ways to represent content-actor relationships [13], such as movie ratings by users or e-commerce transaction between customers and products. For example, a movie fan may use a movie ratings dataset to discover interesting patterns to discuss with like-minded individuals in their social networks, whereas a market researcher may use it to find target segments of the market, such as “product A is favored by male engineers from the West Coast of ages 20 to 30.” While there are many statistical analyses to aid this process, establishing initial hypotheses remains challenging. In particular, the bipartite network graph nature of these datasets combined with the immense amount of data often becomes a barrier. Landesberger et al. [26] poses this as a future research challenge, where interactive feedback enables a hypothesis-insight-driven analytical process.

Even though there exist many statistical and computational tools to support this process, deriving such hypotheses in the first place is a creative, domain-specific, and culture-dependent process that requires human analysts. After the hypothesis has been formulated and tested, large-scale machine learning and statistical tools can streamline the validation process. While large data volumes, such as years of transaction records of a national retailer, can be managed by rapidly evolving technical advances in big data analytics, this is not true for the abilities of human analysts exploring the data to generate the initial hypotheses.

In this paper, we argue that the main barrier against effective adoption of big data machine learning methods is in interpreting their result. These methods often yield large coefficient vectors, which are difficult to map to high-level tasks such as selecting the target group for the next advertisement campaign, or finding major advantages and disadvantages of a company’s products compared to its competitors. To fill this gap, we propose a novel visualization technique for business transaction data called *ParallelSpaces*. ParallelSpaces visualizes the result of the statistical analysis in a user-friendly format. The visual design of ParallelSpaces is motivated by the fact that much analytic CRM data can be classified within two categories: qualitative and quantitative relations between and within the *customers* of a business as well as its *products* (and services).

ParallelSpaces, thus, creates dual side-by-side scatterplots and assigns separate 2D spaces to each such class of an entity. Each space uses a multivariate visualization of the entities in that class. Nodes are initially shown according to the similarity in the relationship with other spaces. Selections in one space are highlighted in the other space using *brushing* [15] based on the relationship between the items, thereby forming visual patterns in the views. The user can scan these patterns to gain an overview of the transaction data. Furthermore, scatterplots axes can be changed to enable exploration of multivariate properties of each node, such as customer demographic data or product properties. Figure 1 illustrates this basic concept.

To demonstrate the effectiveness of the ParallelSpaces visualization technique, we have built web-based prototype implementations for two separate datasets: (1) a movie ratings dataset called MovieLens dataset, and (2) Yelp business reviews. Figure 2 shows a screen image of the system. We used these prototypes in a qualitative user study where eleven participants were asked to explore the movie and business data in order to collect interesting findings. Our results highlight the utility of the ParallelSpaces method



	Movie 1
Release Date	1978
Average Review	4.3
Number of Reviews	345

	User A	User B	User C	User D	User E
Movie 1		1	1		4
Movie 2	4	2	3	3	5
Movie 3	1			1	
Movie 4	2	4		1	1
Movie 5	3	3	2		4

	User E
Age	32
Job	Writer
Gender	Femaie
Zip Code	47906
Average Review	3.4
Number of Reviews	57

Figure 1. Movie entities and user entities are represented as blue circles on the left and red circles on the right, respectively. The system uses the mapping matrix, middle, to brush and link the two spaces according to the user-defined criteria. Selecting users causes selection of movies they prefer and selecting a movie or movies leads to selection of users who give similar common ratings, vice versa. common ratings. Using axes rotation, the linked users and movie data can be further explored according to demographic criteria, shown in the right table, and the movie criteria, shown in the left table.

as well as our interaction techniques for the hypothesis generation step.

Our contributions are (1) the use of connected plots to show the results of the co-clustering, (2) the design of visual elements and interactions to enable exploration, and (3) an example system with a user study on the utility of ParallelSpaces to aid hypothesis generation.

II. Related Works

Our work intersects several research areas within the general areas of visualization and visual analytics:

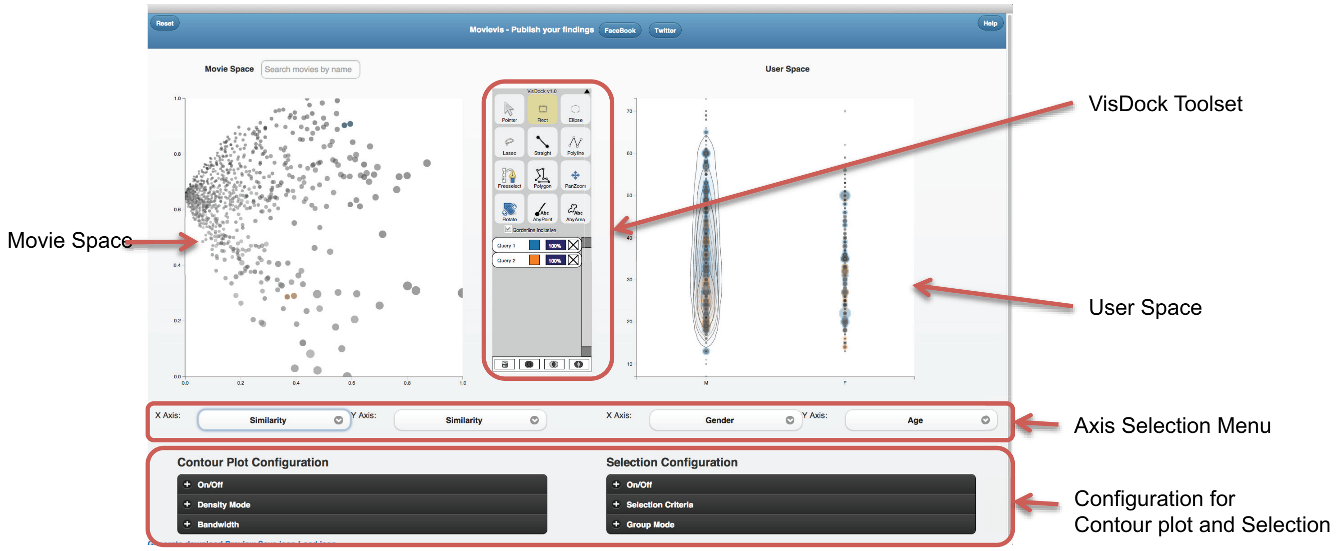
- **Bipartite graphs:** our data is graph-based and bimodal;
- **Multidimensional visualization:** our focus is on displaying multivariate data associated with graph vertices;
- **Machine learning:** use of mathematical and statistical modeling to extract data from multivariate datasets.

A. Bipartite Graphs

A *bipartite graph* (*bigraph*) is a graph $G = (V,E)$ whose vertices V can be partitioned into two independent sets (i.e., none of the vertices in the set are adjacent) T and U. The two vertex classes can be seen as two different types, or *modes*, of the graph, and can for example be colored using only two colors. A *weighted graph*, on the other hand, is a graph whose edges E have a weight w_i. This means that a *weighted bipartite graph* is a bipartite graph where the edges connecting the two sets have an associated weight.

Graphs in general are an active area of research, and is a core dataset for information visualization [20]. Multiple general graph visualizations exist [9]. Some tools and techniques are targeted specifically at bigraphs. Perhaps, the closest to our work is NetLens [13], which visualizes so-called "content-actor" networks using two side-by-side and coordinated views. This content-actor network model is essentially equivalent to bipartite graphs, except their model allows for intra-relationships (within-mode) to the same set. Furthermore, the interaction propagation from one mode to the other is similar to those in our ParallelSpaces work. However, NetLens was originally designed for publication data where the contents represent papers and actors represent authors. As a result, whereas NetLens has a complex interface with many different views and visual representations, ParallelSpaces uses two side-by-side scatterplots and simplifies the visual representation and interactions between them. Because the properties of entities are visible in scatterplots, making a query becomes selecting a region which will be easier for the users.

Another highly relevant work is semantic substrates [21], where graph nodes of different modes are partitioned into separate 2D regions on the visual space, often using an attribute-based layout such as time. The visualization suppresses edges between modes except for when a node is selected. ParallelSpaces similarly employs parallel 2D spaces to partition the two different sets of vertices in the bipartite graph, and also suppresses edges. However, the main difference is that ParallelSpaces puts more emphasis



X Axis: Similarity
Y Axis: Similarity
X Axis: Gender
Y Axis: Age

Figure 2. The MovieVis tool. Two groups in the movie space have been selected to compare corresponding user distribution. Two movies selected in the upper-center region—*One flew Over the Cuckoo's Nest* (1975) and *Amadeus* (1984)–and are shown in blue color. Another two movies selected in a lower-center region—*Phenomenon* (1996) and *Twister* (1996)—are shown in orange. The highlighted users are those who liked all both pairs of movies (because the group mode is set to "common"). Based on the user space axes—gender for the horizontal and age for the vertical—we can see that while the movie *One Flew Over the Cuckoo's Nest* and *Amadeus* were favored by male reviewers of all ages, the *Phenomenon* and *Twister* were liked by relatively younger male audiences.

on visualizing the multivariate attributes of the nodes, and is integrated with contour density plots to show how selections relate across spaces.

B. Multidimensional Visualization

Named as one of the classic information visualization data types [20], multidimensional data consist of multiple (more than three) dimensions and are often represented using data tables. Many systems for multidimensional visualization exist, including Tukey's PRIM-9 [25] system, Becker and Cleveland's trellis displays [3], Ward's XmdvTool [28], and the GGobi system [23]. ScatterDice [8] showed multidimensional visual exploration using scatterplot, where the users can interactively assign properties to axes. ParallelSpaces follows a similar approach, but extends the idea to multimodal datasets by juxtaposing two displays.

Creating multiple data views is rooted in linked graphs from more than 25 years of statistics [2], [3], and has often been combined with *brushing*. It is also a common strategy for dealing with multidimensional datasets in interactive visualization; examples of this practice include Mondrian [24], Improvise [29], and Tableau/Polaris [22]. The most common approach to organize multiple views is called *coordinated multiple views* (CMV) [1], [19] and simply juxtaposes views in the same visual space with *brushing* [2]—dynamic highlighting of items selected in one view in all other views—as the main coordination mechanism.

C. Machine Learning

Machine learning, data mining, and information retrieval are all research areas that, similar to visualization, are tackling sensemaking for big data. Many of the methods proposed from these domains are already extensively utilized in visualization and visual analytics. Arguably the most popular of such methods is cluster analysis [12] that uses the multivariate properties of data to find similar items so that they can be grouped together. This fits well with the concept of visual variables for visualization, where the position or location of a mark is its most salient visual feature [4]. In other words, visualizations of cluster analysis promote the understanding of latent classes in the data.

There exist many ways to extract visual coordinates from a multivariate dataset. Thus, techniques such as *dimensionality reduction* have long been an active area of research [5]. The challenge is that the process is an inherently lossy one. Self-organizing maps have been widely used as a tool for this purpose [14]. Another algorithm based on singular vector decomposition (SVD) tries to reduce the dimensionality to an underlying set of latent taste dimensions [10]. The reduced dimensionality represents "hidden themes" or "latent concepts" in the document, yielding the name Latent Semantic Indexing (LSI). A generalization of probabilistic LSI called Latent Dirichlet Allocation (LDA) [6] provides improved accuracy.

One of the applications of machine learning, where bipartite data is used, is *collaborative filtering* [18] for recommender systems. A *recommender system* [16] is an information filtering system designed to predict contents for a particular user based on their own past ratings and that of other like-minded individuals (collaborative filtering), as well as based on the characteristics of the content itself (content-based filtering). The data used for the former approach—collaborative filtering—is a dyadic dataset containing implicit ratings of the form "User A bought Content 1," or explicit ratings of the form "User A gave Content 1 a rating 4 out of 5." As it turns out, this type of dyadic data can be modeled as a weighted bipartite graph, where the two sets represent users and content, and the undirected edges between the sets are ratings that individual users applied to specific content. Iwata et al. [11] used latent semantic analysis methods to create scatterplot representations of extracted data. His scatterplot arranged the movies according to their similarity in ratings patterns of users. However, it is hard to see what each cluster means. To overcome this limitation, the ParallelSpaces tries to show the distribution of users who liked each cluster, in terms of their properties like age, gender or job. It will enable hypothetical labeling of each cluster.

III. Data Analysis: Business Transactions

There are two kinds of datasets that characterize the majority of business intelligence data: quantitative and qualitative. We chose the two example datasets used in this paper for the purpose of representing both of these general types.

A quantitative dataset is mostly numeric, and an example is customer transaction records for a product. Such a dataset can be expressed as "customer A bought item B five times," or (A, B, 5). In this paper, these kinds of dataset are represented with the movie preference dataset called MovieLens.[1]

Even if not strictly a traditional business dataset, the movie dataset is adequate for the purposes of our paper for two reasons. First, there are no privacy issues, whereas transaction data from a real merchant can reveal the identity of customers and sensitive data related to medical or adult products. Our movie dataset has no such issues to begin with. Second, the movie preferences in our dataset are easily understandable without prerequisite knowledge and also generalizes to domain-specific business data. For instance, in the case of real transaction data, we cannot directly compare the preference based on the number of purchases if the product A and product B belongs different category.

Qualitative data is often more subjective in nature, such as customer reviews written for a product. Professional marketers try to understand the market responses by using reviews for their own product or for a competitors product to identify strengths, weaknesses, opportunities, and threats. However, sometimes the sheer number of reviews can be overwhelming. Methods such as topic modeling eases this burden by clustering documents based on their similarity. We argue that our Yelp datasets, which captures business reviews written by customers, represents such qualitative data. For example, the dataset allows for comparing good and bad Mexican or Asian restaurants based on these reviews. Again, the straightforward nature of this dataset demonstrates the ParallSpaces approach and generalizes easily to more specific qualitative business data.

IV. Task Analysis: Dyadic Data Exploration

Pirolli and Card [17] suggested a model for sensemaking, which can be used as reference for the hypothesis generation process. However given the specific forms of dataset in the context, the task of business intelligence analyst can be further specified as follows:

- **Search and filter:** Retrieve entities according to specific multivariate properties, such as age or rating range.
- **Data distribution:** Find the characteristics of selected entities in a multivariate dimension.
- **Finding similar entities:** Find entities that shows similar transaction patterns. Two definitions of similarity are possible. First, the properties of nodes can be similar. For example, users can be similar if they belong to the same age, gender, and geographical location group. Second, the nodes can be similar in their relations with the opposite parties. For example, two users can be similar if their buying patterns are similar.
- **Finding similar linked entities:** Find the related entities where relationship can be defined in the context of customer-product matrix. For example, in the context of the movie ratings dataset, given users, find the movies they gave more than 4 ratings. Also the relationship should be interactively adjustable. For example, the system should be able to find people who liked or disliked a certain items.
- **Estimate correlation:** Estimate the strength of relationship. For example, judging whether there is a strong correlation between the age of customers and the kind of movies they like.

As a hypothetical example to illustrate the use of these tasks, let us assume that a BI analyst is trying to find movies to recommend to a set of viewers. First he needs to select these viewers using *search and filter*. Then he may examine the property distribution of the selected moviegoers using *overview of property distribution*. Also the analyst may want to find users who show similar rating patterns with the selected target group using *identification of similar entities*. After identifying the similar users, the analyst may identify the movies these people like in common using *identification of related entities*. Finally, having the *ability to estimate the strength of correlation* helps the analyst to iteratively explore various options using information foraging models.

[1] http://www.grouplens.org/node/12

V. ParallelSpaces: Visual Design

ParallelSpaces is an interactive visualization technique for visualizing multimodal and multivariate data in dual juxtaposed spaces that each use mutually brushed visual representations (often scatterplots). In the section below we describe the visual design of the technique, including layout, position, size, color, brightness, and density plots.

A. Space Layout

A key observation from our bipartite graphs is that at its core, the graph can be split into two independent sets. For example, in the case of the movie preference data, the users and the movies form these two independent domains. However, because the sets do not overlap, we design a basic visual representation that consists of two parallel 2D spaces, one for each set. This design is similar to the separate content and actor spaces used in NetLens [13].

The bivariate graph closely connects nodes in one space to the other. The natural way to represent this is to support brushing and highlighting between the spaces (even if we, strictly speaking, are not brushing the same entity but connected entities).

Practically speaking, this means that selecting an entity in one space corresponds to selecting the connected entities in the other parallel space. For example, if we select a movie in the movie space, the users who liked the movie are selected (and highlighted) in the user space. Analogously, if a user is selected in the user space, the movies that received high ratings from that user can be selected in the movie space. Since we have relaxed the traditional constraint that brushing applies to the same item in different views, the underlying relationship is customizable. For example, a researcher may want to see which groups did not like a specific movie. In this case, the researcher can filter the relationship between the two spaces, making this pattern clearly visible.

The position of a visual mark is often the most salient feature in a visualization. In ParallelSpaces, any multidimensional property can be an axis. However, the relationship table is only visible when a user selects some entities. To make the relation between entities more clear, we also allowed the user to organize the 2D layout of points by their similarity. The more similar the entities are, the closer they will be placed.

Given that each user has ratings over m possible movies, each user is represented as an m-vector. Similarly, each movie is an n-vector representing users and their ratings. Finding a position for each entity in a 2D space thus becomes a projection (or dimensionality reduction) problem where m- or n-dimensional vectors are projected onto a two-dimensional space. Naturally, there are many approaches to achieving this goal: principal component analysis (PCA), multidimensional scaling (MDS), singular vector decomposition (SVD), and probabilistic Latent Semantic Analysis (pLSA) are some of the choices. In our implementation, we choose PCA as our solution, but other alternatives—even hybrid ones—are possible and within context of the overall ParallelSpaces method.

Entity	Feature	Visual Variable
Movie	Number of ratings	Size
User	Number of ratings	Size
Movie	Average ratings	Opacity + brightness
User	Average ratings	Opacity + brightness

Table I
Selection of salience features and the matching visual variables for ParallelSpaces in the MovieVis prototype implementation.

This similarity positioning feature provides starting point for the analysis, because the meaning of each cluster can mean a market segment, which shares similar preference patterns. For example, selecting a region in the movie space at similarity axis, the people who liked the movies can be selected. Further, by interactively changing axes of the user space, we can explore if there are particular patterns in age, gender, job or location dimensions.

Because the number of nodes can be high, we need to differentiate the visibility of nodes according to their importance. In our MovieVis implementation of ParallelSpaces for movie preference, we selected the setup listed in Table I to represent salience.

In ParallelSpaces, we use colors to represent set memberships when highlighting items. Also transparency is applied, so that when an item is part of two or more selections, the colors are mixed to represent its memberships. The benefit of this approach is that it does not change the size of the mark. However, the drawback is that color transparency and blending are more difficult to perceive, particularly for many selections.

B. Showing Distribution: Contour Plot

Perceiving the distribution and density of a large number of visual points in a substrate is negatively affected by both scale and overplotting (which leads to occlusion). Meanwhile, being able to assess the distribution of a group of entites corresponding to a brushed selection in another space is an important analytical task. There exist several different approaches to address this problem, such as:

- Smaller marks, yielding less overplotting;
- Transparency, to mitigate occlusion;
- Contour plots, to represent the density pattern; or
- 3D mesh gradient, to characterize the distribution.

Smaller marks may affect user interaction because the users will have difficulties in selecting them. This can be particularly problematic for touch-based tablets or mobile phones with screens. Furthermore, transparency is already assigned to data salience.

Our design choice is therefore to dynamically construct contour plots for each visualization space to show data density. More specifically, we use *kernel density estimation* (KDE) to smooth and quantify the underlying group of points. Basically, the idea is to construct visual representations of KDE clusters around the selected group of points to communicate their distribution. While a 3D mesh representation may also have been useful, we prefer to choose visual representations that fit within our 2D visual design.

KDE algorithms generally have two tunable factors: the *bandwidth* and the *kernel*. The bandwith determines the size of each kernel, which indirectly yields the degree of smoothness of the resulting image. If it is low, it may cause noisy patterns, which are hard to identify. When it is too high, on the other hand, it can create a distribution pattern that is too smooth and carries little meaningful information. Previous work shows that the optimal bandwidth can be determined by signal characteristics [27]. In our work, the user can interactively change the bandwidth. For bivariate KDE, the kernel parameter can also have an effect on the accuracy.

In our MovieVis prototype for ParallelSpaces, we support two types of contour plots (Figure 3):

- **Density mode:** Show the density of selected entities in the particular space (the common approach). All kernels will have the same height, regardless of their weight (i.e., movie rating).
- **Amplitude mode:** Modulate the kernel height by the corresponding entity ratings, causing higher values in areas with high ratings and less influence from areas with lower ratings. While conveying more than just point density, this approach has the drawback of confounding density with weights.

If the analyst merely wants to see which movies a user or group of users rated, the density contour plot can provide that information. However, the amplitude contour plot will also show information on the individual ratings that the selected users gave to these movies. Comparing the two plots may yield interesting new insights.

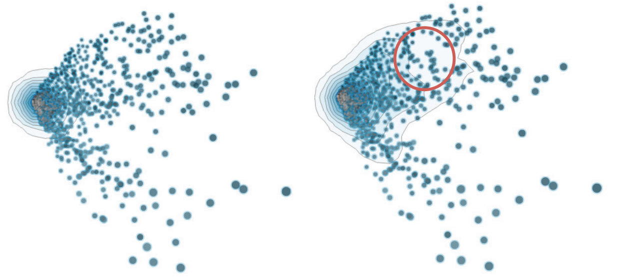


Figure 3. Density (left) and amplitude (right) contour plots for all the movies rated by a male educator (age 47). The selection criteria was every movie he liked. By comparing density mode KDE and amplitude mode KDE, we can spot the area where the users especially liked over the area the users have transaction records.In this example, the red circled area will contain the movies he rated more highly.

VI. ParallelSpaces: Interaction Design

ParallelSpaces relies heavily on interaction to support visual exploration. Below we review our interaction design.

A. Selection

One of the most frequent tasks of visual analytics is comparing patterns between multiple entities. To support this process, an ordinal color is given to each selection to show which items belong to the selection. Selections can consist of one or multiple entities defined by an enclosing border. A lasso tool allows selecting multiple entities. Hovering over an item shows a tooltip with the movie title and a link to the IMDb page, where more information is available.

To support finding particular movies and users, we provide a search toolbar with autocomplete support. When the user is looking for a specific movie, he or she can type a few words to find it. Selecting a movie from the search bar is equivalent to clicking it.

Because we regard each space independently, there are two modes of selection. When items are selected in the movie space, movies are selected and the selection propagates to the user space based on their relation. This is movie mode selection. Similarly, when users are selected in the user space, the corresponding movies will highlighted in the movie space. This is accordingly called user mode selection. Selection modes are simply switched by clicking in the opposite space. In the case of movie mode, selecting another movie will add the selected movie to the selection queue to enable the comparison of the visual pattern with previously selected movies.

B. Relationship between Spaces

As the relationship between the parallel spaces is customizable, we provided a simple range slider to adjust the relationship to investigate. For example, when the range slider is in the 4 to 5 range, selecting a movie entity will highlight all the users who gave that particular movie 4 to 5 ratings. However, when the range slider is in the 0 to 1 range, selecting a user entity will highlight all movies, which that particular user gave 0 to 1 ratings.

Our implementation also supports standard navigation techniques such as zooming and panning using mouse wheel and dragging. To reduce the effect of overplotting, we applied semantic zooming, where the points become smaller when zoomed in. We also use animated transitions to maintain object constancy in the display and allow the user to easily perceive state changes. This is particularly important for the axis rotation, where points change position.

VII. Implementation Notes

We have implemented two prototype instantiations of the ParallelSpaces techniques: MovieVis, for movie ratings using the MovieLens 100k dataset, and YelpVis, for business reviews from Yelp.com. In the case of YelpVis, the

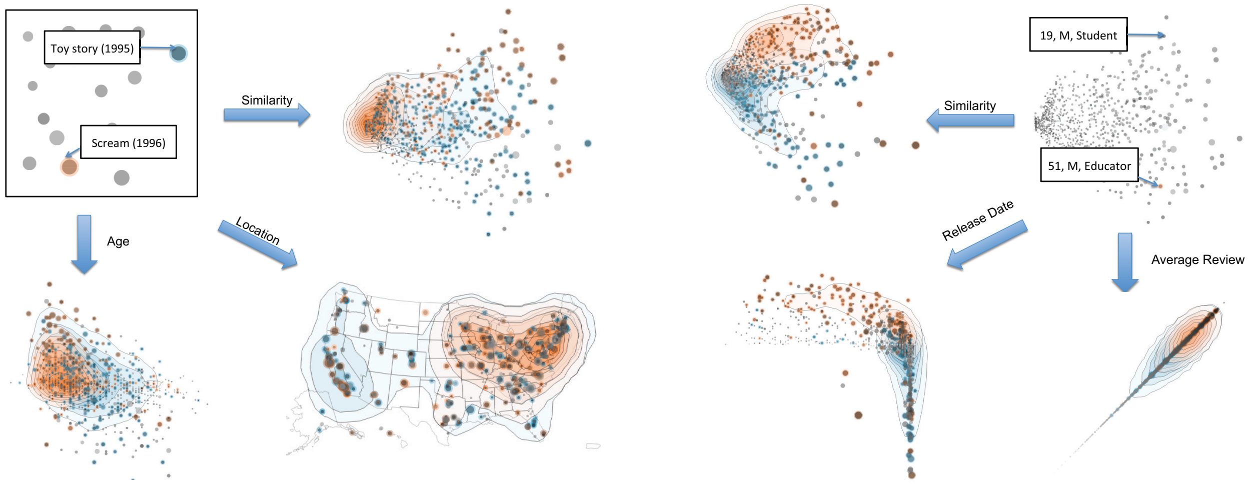


Figure 4. On the left, we compare two movies, *Toy Story* (1995), in blue, and *Scream* (1996), in orange, according to the age, location and similarity criteria for users. Some notable observations are while the former is liked all around the U.S. by any age groups the latter is mostly popular in the eastern part and within a younger generation. On the right, we compare two users, a 19-year-old male student, in blue, and a 51-year-old male educator, in orange according to the average, release date, and similarity criteria for movies. We observe that the older user tends to rate older films highly. In addition, his average review tends to conform to the average ratings patterns of all users while the younger user seems to deviate from it.

relationship between words and business was the number of occurrence of the words for particular business. The word space contains words like 'fantastic', 'good', or 'bad' for restaurants and the frequencies at which certain words' appear vary for different restaurants. The rationale is that users can easily discover the patterns in the reviews of restaurants using a set of such words. Both prototypes were built as web-based JavaScript and SVG applications using the D3 visualization toolkit [7]. We use the VisDock[2] library (also JavaScript) for advanced cross-cutting interaction support for selection, query management, and annotation. An interactive demonstration of the MovieVis prototype can be seen at http://vistalk.herokuapp.com/movievis/, and the YelpVis prototype is available at http://vistalk.herokuapp.com/yelpvis/.

VIII. Usage Example

We give a usage scenario to explain how the ParallelSpaces tool can help someone with forming an initial hypothesis about the dataset. Let's say a market researcher uses MovieVis to study the preference data of two movies *Scream* (1996) and *Toy Story* (1995). She selects these two movies in the movie space using the search option provided by MovieVis. This visualizes the preference data on the user space with both axes set to similarity by default. MovieVis provides a drop-down menu to set the axes in the user space to one of the seven quantities: Similarity, Age, Job, Location, Gender, Average Review, and Number of Reviews. She selects "Location" as the X-axis in the user space to display the users on a geographical map of United States. Figure 4 shows the visualization after applying the settings above to the user space. She observes from the contour plots in the user space that, while *Scream* is highly rated by users on the East Coast, *Toy Story* is highly rated by users all around the United States. Thus, she changes the Y-axis to "Age" while leaving the X-axis to the default setting, yielding the visualization in the bottom left of Figure 4. She then observes that while *Scream* is highly rated by users of age groups 15 to 30, *Toy Story* is highly rated by users of all age groups.

IX. Qualitative User Study

The primary purpose of ParallelSpaces is to aid in generating initial hypotheses for weighted bivariate graphs. We conducted a qualitative user study to evaluate whether the system achieves this purpose.

A. Method

We recruited 11 (8 male, 3 female) paid participants to use the MovieVis and YelpVis systems for 20 minutes each. All participants were university students, and the average age was 26, ranging from 20 to 34. Prior to using the systems, participants were given 10 minutes of training in using the tools. During the exploration (two sessions of 20 minutes), they were encouraged to write comments about their findings using an annotation feature embedded in the tools. After completing the exploration sessions, the users were asked to evaluate their experience in terms of usefulness, enjoyability, and ease of use. We also collected subjective free-form feedback (comments and notes) as well as basic demographic and technical information about the participants. A full user study session lasted approximately one hour (10 minutes of training, two 20-minute sessions for exploration, and 10 minutes for the post-test survey).

[2] https://github.com/VisDockHub/NewVisDock

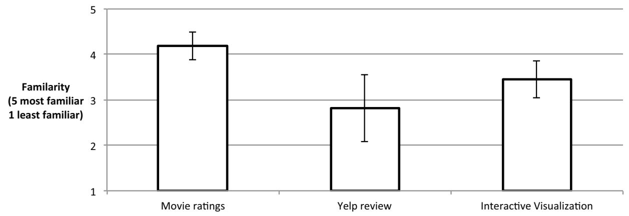


Figure 5. The demographic survey shows that our participants were quite familiar with movie ratings, while their knowledge of Yelp business reviews was on average lower and with higher variation. Participant expertise for interactive visualization was also diverse.

B. Results

Figure 5 shows the demographic survey data for our participants. In general, all 11 of our participants were able to understand the MovieVis and YelpVis tools and to independently perform data exploration using them. In total, participants wrote 71 comments for MovieVis and 52 comments for YelpVis using the embedded annotation mechanism in the tools, yielding an average of 6.5 (s.d. 4.8) and 4.7 (s.d. 3.8) comments per participant, respectively. The overall feedback for the tools was generally positive, but participants provided many specific points of improvement and criticism.

Figure 6 shows the post-study survey ratings on efficiency, ease of use, and enjoyability. The ratings for YelpVis were lower than for MovieVis. One explanation might be that the participants' prior interest and knowledge of the datasets was lower for Yelp business reviews than for movies (Figure 5). This is supported by the fact that of the 11 participants, the five with low familiarity with Yelp reviews also gave significantly lower subjective ratings than the remaining six who were familiar with Yelp reviews. Interestingly, that same group of five gave MovieVis higher scores.

In the treatment below, we analyze our qualitative results from the study based on three basic aspects: efficiency (the perceived usefulness of the tools), enjoyability and motivation (how well the tool guided and motivated the participants), and ease of use (usability or conceptual barriers hindering the exploration). We also discuss several points of improvement that were raised by participants.

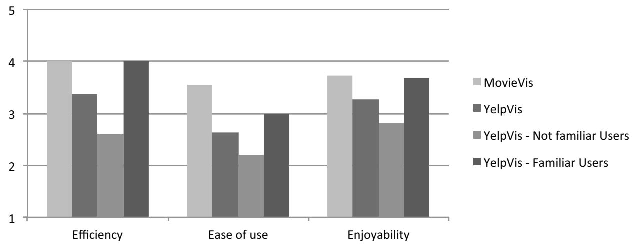


Figure 6. Subjective ratings for the MovieVis and YelpVis tools for the qualitative user study.

Efficiency: Most participants expressed very positive feedback on the efficiency of ParallelSpaces in terms of general usefulness and utility. MovieVis, in particular, was preferred as highly useful, presumably due to the familiarity and interest bias of the datasets as discussed above. Several of the comments were expressly derived from advanced features of the system. For example, one participant stated *"So Matilda and Contact are both good movies and both liked by a lot of people from all ages, but they have a 'far' similarity because Contact has way more reviews than Matilda and [is] closer to movies [...] like Star Wars..."* The same participant used selections and graphical axes to speculate how the number of reviews affect the similarity metric in the visualization, and also suggested a fragmentation in the audience of these two movies that corresponds to their different genres.

Enjoyability and Motivation: Motivational factors play an important role in collective intelligence systems, which rely on the voluntary efforts of individual users. In the feedback from participants, several people provided positive feedback, such as one participant noting that he did not notice how 20 minutes had passed already, and another requesting the URL of the tool to continue exploring after the study. However, a few participants did not seem to enjoy the experience even if this was not clear from their verbal or written feedback. We speculate that this is due to the relatively high analytic and conceptual thresholds in using ParallelSpaces effectively; one participant underscored this by stating that *"as a geek, I would like to play with this, but it is not for non-geeks."*

Ease of Use: The score for the ease of use was of the lowest of the three. Several participants were concerned about the usability of the system, in particular for understanding the word business relationships in YelpVis. The stopwords for the general query was not adequate for YelpVis and resulted in many frequent words with little meaning, such as *go* and *place*.

Furthermore, the concept of similarity was not well-understood for some participants. They frequently relied only on the other concrete axes such as age, occupation, and average rating. In addition, participants rarely used the contour plot in the user study, and even those that did expressed confusion on its meaning, suggesting that this functionality could be better integrated into the tool.

X. Conclusion and Future Work

In summary, this work presents a novel visualization technique called ParallelSpaces designed for business transaction data often used for generating initial hypotheses for business intelligence. We also reported on two concrete instantiations of ParallelSpaces as web-based visualizations designed for casual end-users as well as results from a qualitative evaluation investigating the utility of the technique for users to aid the hypothesis generation. Our future work will continue to explore visual mechanisms for business intelligence analytics, focusing in particular on the type of business transaction data studied here.

References

[1] M. Q. W. Baldonado, A. Woodruff, and A. Kuchinsky. Guidelines for using multiple views in information visualization. In *Proceedings of the ACM Conference on Advanced Visual Interfaces*, pages 110–119, 2000.

[2] R. A. Becker and W. S. Cleveland. Brushing scatterplots. *Technometrics*, 29(2):127–142, 1987.

[3] R. A. Becker, W. S. Cleveland, and M.-J. Shyu. The visual design and control of trellis display. *Journal of Computational and Graphical Statistics*, 5(2):123–155, 1996.

[4] J. Bertin. *Semiology of graphics*. University of Wisconsin Press, 1983.

[5] D. Billsus and M. J. Pazzani. Learning collaborative information filters. In *Proceedings of the International Conference on Machine Learning*, pages 46–54, 1998.

[6] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, Mar. 2003.

[7] M. Bostock, V. Ogievetsky, and J. Heer. D3: data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309, 2011.

[8] 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.

[9] I. Herman, G. Melançon, and M. S. Marshall. Graph visualization and navigation in information visualization: A survey. *IEEE Transactions on Visualization and Computer Graphics*, 6(1):24–43, 2000.

[10] T. Hofmann. Latent semantic models for collaborative filtering. *ACM Transactions on Information Systems*, 22(1):89–115, Jan. 2004.

[11] T. Iwata, T. Yamada, and N. Ueda. Probabilistic latent semantic visualization: topic model for visualizing documents. In *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining*, pages 363–371, 2008.

[12] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Computer Surveys*, 31(3):264–323, Sept. 1999.

[13] H. Kang, C. Plaisant, B. Lee, and B. B. Bederson. NetLens: iterative exploration of content-actor network data. *Information Visualization*, 6(1):18–31, 2007.

[14] S. Kaski, T. Honkela, K. Lagus, and T. Kohonen. WEBSOM - self-organizing maps of document collections. *Neurocomputing*, 21:101–117, 1998.

[15] D. A. Keim. Information visualization and visual data mining. *IEEE Transaction on Visualization and Computer Graphics*, 8(1):1–8, 2002.

[16] J. Konstan and J. Riedl. Recommender systems: from algorithms to user experience. *User Modeling and User-Adapted Interaction*, 22:101–123, 2012.

[17] P. Pirolli and S. Card. The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. In *Proceedings of International Conference on Intelligence Analysis*, volume 5, pages 2–4, 2005.

[18] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl. GroupLens: an open architecture for collaborative filtering of netnews. In *Proceedings of the ACM Conference on Computer-Supported Cooperative Work*, pages 175–186, 1994.

[19] J. C. Roberts. State of the art: Coordinated & multiple views in exploratory visualization. In *Proceedings of the International Conference on Coordinated Multiple Views in Exploratory Visualization*, 2007.

[20] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings of the IEEE Symposium on Visual Languages*, pages 336–343, 1996.

[21] B. Shneiderman and A. Aris. Network visualization by semantic substrates. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):733–740, 2006.

[22] C. Stolte, D. Tang, and P. Hanrahan. Polaris: A system for query, analysis, and visualization of multidimensional relational databases. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):52–65, 2002.

[23] D. F. Swayne, D. T. Lang, A. Buja, and D. Cook. GGobi: evolving from XGobi into an extensible framework for interactive data visualization. *Computational Statistics & Data Analysis*, 43(4):423–444, 2003.

[24] M. Theus and S. Urbanek. *Interactive Graphics for Data Analysis: Principles and Examples*. Chapman & Hall/CRC, 2008.

[25] J. W. Tukey, M. A. Fisherkeller, and J. H. Friedman. PRIM-9: An interactive multi-dimensional data display and analysis system. In W. S. Cleveland and M. E. McGill, editors, *Dynamic Graphics for Statistics*, pages 111–120. Wadsworth & Brooks/Cole, 1988.

[26] T. von Landesberger, A. Kuijper, T. 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*, 30(6):1719–1749, 2011.

[27] M. P. Wand and M. C. Jones. Comparison of smoothing parameterizations in bivariate kernel density estimation. *Journal of the American Statistical Association*, 88(422):520–528, 1993.

[28] M. O. Ward. XmdvTool: Integrating multiple methods for visualizing multivariate data. In *Proceedings of the IEEE Conference on Visualization*, pages 326–333, 1994.

[29] C. Weaver. Building highly-coordinated visualizations in Improvise. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 159–166, 2004.