

# FpVAT: A Visual Analytic Tool for Supporting Frequent Pattern Mining

Carson Kai-Sang Leung<sup>\*</sup>  
Department of Computer Science  
The University of Manitoba  
Winnipeg, MB, Canada  
kleung@cs.umanitoba.ca

Christopher L. Carmichael  
Department of Computer Science  
The University of Manitoba  
Winnipeg, MB, Canada  
umcarmi1@cs.umanitoba.ca

## ABSTRACT

As frequent pattern mining plays an essential role in many knowledge discovery and data mining (KDD) tasks, numerous algorithms for finding frequent patterns have been proposed over the past 15 years. However, most of these algorithms return the mining results in the form of textual lists containing frequent patterns showing those frequently occurring sets of items. It is well known that “a picture is worth a thousand words”. The use of visual representation can enhance the user’s understanding of the inherent relations in a collection of frequent patterns. In this paper, we develop a simple yet useful *visual analytic tool for supporting frequent pattern mining* called *FpVAT*. Such a visual analytic tool consists of two modules: One module gives users an overview so that they can derive insight from a massive amount of raw data; another module enables users to perform analytical reasoning on the mining results via interactive visual interfaces so that users can detect the expected frequent patterns and discover the unexpected frequent patterns. As a visual analytic tool, our FpVAT is equipped with several interactive features for effective visual support in the data analysis and KDD process for various real-life applications.

## 1. INTRODUCTION

*Frequent pattern mining* [1; 16; 18; 23; 38] searches raw data for implicit, previously unknown, and potentially useful information in the form of frequent patterns. Here, *frequent patterns* refer to frequently occurring sets of items, which are also known as *frequent itemsets*. Examples of these frequent patterns include sets of frequently purchased merchandise, combinations of popular services requested by users, collections of Webpages frequently updated by users, groups of users who frequently edit the Webpages, travellers’ favourite ports of entry, sets of frequent callers, and parts of the building frequently visited by employees. In general, frequent pattern mining plays an essential role in many knowledge discovery and data mining (KDD) tasks such as the mining of association rules, correlation, sequences, episodes, maximal frequent patterns, as well as closed frequent patterns. Hence, frequent pattern mining is in demand in various real-life applications. The frequent patterns mined from raw data

<sup>\*</sup>Corresponding author: C.K.-S. Leung.

can answer many questions that help users make important decisions in various real-life situations. The following are some examples:

- Q1. Store managers may want to find out how frequently certain kinds of vegetables are purchased *individually* and how frequently they are purchased *together*?
- Q2. Botanists may want to discover which features or properties associated with edible mushroom are frequently observed?
- Q3. University administrators may want to know which popular elective courses are frequently taken together by students?

To help answer the above questions in these real-life situations, numerous frequent pattern mining algorithms have been proposed over the past 15 years. However, most of the algorithms return a collection of frequent patterns in *textual form* (e.g., a very long unsorted list of frequent patterns). However, presentation of a vast amount of the mining results in a conventional long list does not lead to ease of understanding. Consequently, users may not easily discover the knowledge and useful information that is embedded in the data. Showing a set of frequent patterns in *graphical form* can show the relations embedded in the data and help users understand the nature of the useful information and discovered knowledge. For example, let us look at Figure 1(a), which shows the first few frequent patterns in a very long unsorted list. The same set of frequent patterns can be more easily assimilated in graphical form as shown in Figure 1(b), from which the frequency information of the patterns can be easily read (e.g.,  $\{a, c\}$  occurs three times,  $\{c, e\}$  occurs twice, and  $\{a, d, e\}$  occurs once). Similarly, let us compare the first few transactions in a very long list of transactions in a database shown in Figure 1(c) with the same collection of transactions shown in Figure 1(d). The graphical form clearly reveals the presence and/or absence of domain items (e.g., item  $c$  is present in  $t_1$ , item  $d$  is absent from  $t_1$ ) and gives users insight about the raw data (e.g., item  $a$  occurs frequently, item  $b$  occurs rarely).

As graphical presentation of raw data and the mining results matches the power of the human visual and cognitive system, researchers have considered visual analytics [17; 27; 29; 30; 35; 37] and visualization techniques [10; 11; 32] to assist users in gaining insight into massive amounts of data or information. Existing visualization systems can be broadly divided into two categories. Systems in the first category like Spotfire [2], VisDB [12] and Polaris [33] were developed to visualize data. However, most of the systems in this cat-

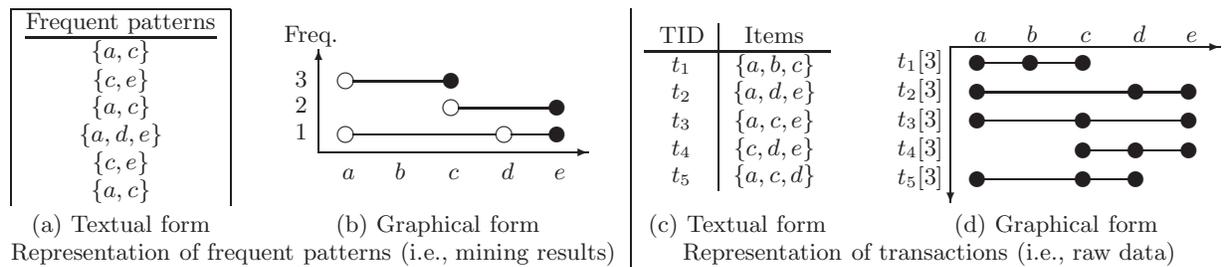


Figure 1: Representation of frequent patterns and transactions in both textual and graphical forms.

egory were not connected to any data mining algorithm, nor were they designed to display the mining results. Systems in the second category visualize the mining results, but the focus of many of these systems has been mainly on results such as clusters [14], decision trees [3] or association rules [5; 6]. However, not many systems in this category were designed for visualizing frequent patterns. Recently, some researchers have shown interests in visualizing frequent patterns. For example, Yang [36] developed a system that can visualize frequent patterns. However, his system was primarily designed to visualize association rules, and it does not scale very well in assisting users to immediately see certain useful information (such as exact frequencies) of a very large number of frequent patterns. As another example, Munzner et al. [26] presented a visualizer called PowerSet-Viewer (PSV), which provides users with guaranteed visibility of frequent patterns in the sense that the pixel representing a frequent pattern is guaranteed to be visible by highlighting such a pixel. However, multiple frequent patterns may be represented by the same pixel. As the third example, we previously proposed a visualization system—called FIsViz [21]—that aims to visualize frequent patterns. FIsViz represents each frequent pattern by a polyline in a two-dimensional space. The location of the polyline indicates the exact frequency of the pattern explicitly. As a result, FIsViz enables users to visualize the mining results (i.e., frequent patterns) for many real-life applications. However, in some other applications (especially, when the number of frequent patterns is huge), FIsViz may not scale very well. Users may require more effort to be able to clearly visualize frequent patterns. The problem is caused by the use of polylines for representing frequent patterns. To elaborate, the polylines can be bent and/or can cross over each other. This makes it difficult to distinguish one polyline (representing a frequent pattern) from another. For example, in Figure 2, how to distinguish the two frequent patterns  $\{a, c, d\}$  &  $\{b, c, e\}$  from another two patterns  $\{a, c, e\}$  &  $\{b, c, d\}$  if we did not use different thickness for the polylines?

Hence, some natural questions to ask are: Can we design a scalable system that helps users visualize frequent patterns effectively? Can we have an alternative representation

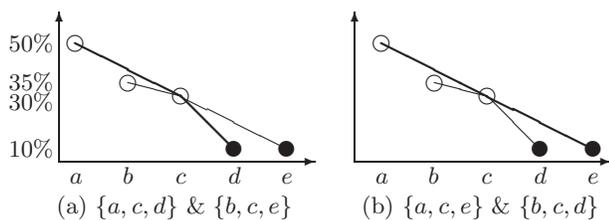


Figure 2: FIsViz uses polylines to show frequent patterns  $\{a, c, d\}$  &  $\{b, c, e\}$  and frequent patterns  $\{a, c, e\}$  &  $\{b, c, d\}$ .

that minimizes the bend and crossover of polylines? In response to these questions, we explored an alternative representation [22], which uses two half-screens to visualize the discovered knowledge about frequent patterns: One half of the screen showing all frequent patterns and another half showing their frequencies. Related to this alternative representation, some follow-up questions include: Can we use a full screen (instead of two half-screens) to interactively visualize the discovered knowledge? Can we visualize and analyze the raw data as well?

To answer the above questions, we propose in this paper a scalable visual analytic tool that uses a full screen for visualization. The tool allows users to effectively and interactively visualize raw data and the mined frequent patterns. In addition, the tool enhances the KDD process by providing answers to some important business questions (e.g., Q1–Q3 above). The **key contribution** of our work is a novel interactive and scalable *frequent pattern visual analytic tool*, called *FpVAT*, which provides users with effective visual support in the data analysis and KDD process. Specifically, *FpVAT* consists of two modules: (i) *RdViz* for *raw data visualization*, which in turn allows users to derive insight from the massive amount of raw data; (ii) *FpViz* for *frequent pattern visualization*, which enables users to detect the expected frequent patterns and discover the unexpected frequent patterns. Both modules use orthogonal graphs for visualizing raw data or the mining results. For example, *FpViz* provides users with clear and explicit depictions about frequent patterns that are embedded in the data of interest. In general, items within each frequent pattern are connected by a horizontal line in *FpViz*. Similarly, items in each transaction are also connected by a horizontal line in *RdViz*. Consequently, the bend and crossover of polylines are minimized. Furthermore, as a visual analytic tool, our *FpVAT* is equipped with several interactive visual features for effective analytical reasoning of raw data and the mining results for various real-life applications.

This paper, which is a revised and expanded version of our VAKD 2009 paper [19], is organized as follows. Next section describes related work. Then, in Section 3, we introduce one module of our proposed *FpVAT*—namely, the *RdViz* module, which helps users to visualize and analyze raw data. In Section 4, we present another module—namely, the *FpViz* module, which helps users to visualize and analyze the mining results (i.e., frequent patterns). Section 5 shows evaluation results. Finally, conclusions are given in Section 6.

## 2. RELATED WORK

Developing effective visualization or visual analytic systems for KDD has been the subject of many studies. This line of research can be divided into two general categories: (i) sys-

tems for visualizing raw data (e.g., Spotfire [2], VisDB [12], Polaris [33]) and (ii) those for visualizing data mining or analysis results. Many systems in the first category provide nice features to effectively arrange and display data in various forms. However, most of the systems were not connected to any data mining algorithm, let alone were they designed to display the mining results. In contrast, our proposed FpVAT allows users to visualize and analyze both the raw data and the mining results.

For systems in the second category, many of them focus on visualizing and/or analyzing the mining results other than frequent patterns (e.g., clusters [13; 14; 31], decision trees [3], temporal sequences [7]). In contrast, our proposed FpVAT was designed to allow users to visualize and analyze frequently occurring sets of items.

As for systems that visualize frequent patterns, Yang [36] designed a system mainly to visualize association rules—but can also be used to visualize frequent patterns—in a two-dimensional space consisting of many vertical axes. In his system, all domain items are sorted according to their frequencies and are evenly distributed along each vertical axis. A frequent pattern consisting of  $k$  items (i.e., a  $k$ -itemset) is then represented by a curve that extends from one vertical axis to another connecting  $k$  such axes. The thickness of the curve indicates the frequency of such a frequent pattern. However, such a representation suffers from the following problems: (i) The use of thickness only shows *relative* (but not *exact*) frequency of the patterns. Comparing the thickness of curves is not easy. (ii) Since items are sorted and *evenly* distributed along the axes, users only know some items are more frequent than the others, but cannot get a sense of how these items are related to each other in terms of their exact frequencies (e.g., whether item  $a$  is twice as frequent as, or just slightly more frequent than, item  $b$ ). In contrast, our proposed FpVAT provides users with *exact* frequency information.

Frequent Itemset Visualizer (FISViz) [21] is one of the recently developed visualizers. It was designed for visualizing frequent patterns. It represents a frequent pattern comprising  $k$  items (i.e., a  $k$ -itemset) by a polyline that connects  $k$  nodes (where each node represents an item in the  $k$ -itemset) in a two-dimensional space. The frequency of the  $i$ -th prefix of an itemset  $X$  is indicated by the position of the  $i$ -th node in the polyline representing  $X$ . For example, when  $X = \{a, c, d\}$  as shown in Figure 2(a), the frequencies of its prefixes  $\{a\}$  and  $\{a, c\}$  are respectively indicated by the  $y$ -positions of nodes  $a$  (i.e., 50%) and  $c$  (i.e., 30%) in the polyline. Similarly, the frequency of the itemset  $X = \{a, c, d\}$  is represented by the  $y$ -position of the node  $d$  (i.e., 10%) in that polyline. With such representation, slopes of different sectors of a polyline can vary. In other words, the polyline may be bent. Moreover, polylines representing different frequent patterns may cross each others. This makes it difficult for users to distinguish one sector of a polyline from another. In contrast, FpVAT uses horizontal lines to represent frequent patterns. As such, it avoids crossover of lines that represent frequent patterns.

### 3. RdViz: THE RAW DATA VISUALIZATION MODULE

Let us start presenting our proposed *Frequent pattern Visual Analytic Tool*, which aims to support frequent pattern

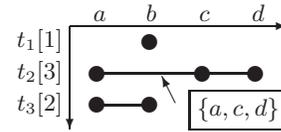


Figure 3: The RdViz module shows raw data (with visual clues and interactive features) in the form of transaction items.

mining. While it is important to visualize the output of the mining process (i.e., mining results in the form of frequent patterns), it is also important to visualize the input (i.e., raw data in the form of items within each transaction of the database) so as to derive insight or overview of the inputting data. In this section, we propose and present the *raw data visualization (RdViz)* module of FpVAT. This module shows raw data in a two-dimensional space. The  $x$ -axis shows the  $m$  domain items, and the  $y$ -axis shows the transaction IDs. Recall that, for frequent pattern mining, each transaction in the database consists of a set of items. RdViz represents each transaction containing  $k$  items by a horizontal line connecting  $k$  filled circles (i.e.,  $k$  discs). For example, suppose the first three transactions  $t_1, t_2$  &  $t_3$  of a database contain one item  $b$ , three items  $a, c$  &  $d$ , and two items  $a$  &  $b$ , respectively. Then, Figure 3 shows how our proposed RdViz module represents these transactions. With this representation, users can easily spot the presence or absence of items in each database transaction. For instance, the presence of a circle at  $(c, t_2)$ -location in Figure 3 implies that transaction  $t_2$  contains item  $c$ . Conversely, the absence of a circle from  $(b, t_2)$ -location implies that transaction  $t_2$  does not contain item  $b$ .

#### 3.1 Scalability Issues

It is not uncommon that the database contains many transactions and/or many domain items in such a way that not all of the transactions and/or not all of the domain items may fit onto the screen. To handle this situation, RdViz provides users with (i) a *vertical scrollbar* for visualizing different transactions and (ii) a *horizontal scrollbar* for visualizing different domain items.

While the use of scrollbars enables users to visualize raw data from a very large database containing large numbers of transactions and domain items, users can only see—at any time instance—a small piece of a big picture. To allow users to view the big picture, RdViz provides users with an interactive *zooming feature*. It allows users to zoom in/out certain regions of the screen. When zooming out, users get an overview about the raw data on one screen; when zooming in, users get the details about the raw data fit onto multiple screens. Figure 6 shows a snapshot of the zoom-out view of a seasonal database, in which customers purchased a subset of items in a season and another subset in another season. To find the details about items purchased in a particular season, users can zoom in.

#### 3.2 Visual Clues and Interactive Features

The above representation of transactions by our proposed RdViz module gives users an overview so that they can gain insight from the raw data. In this section, we describe additional visual clues and interactive features of RdViz. While they are not essential, they provide user convenience.

ANNOTATION OF TRANSACTION LENGTH. Although counting the number of circles on a horizontal line representing a transaction in the zoom-in view gives its transaction length, our proposed RdViz provides users with a visual clue that allows them to easily find out the answer. Specifically, RdViz annotates each label on the  $y$ -axis (i.e., transaction ID) with a number representing the length of each transaction. For example, “ $t_2[3]$ ” in Figure 3 tells users that transaction  $t_2$  consists of 3 items.

DETAILS-ON-DEMAND. RdViz also provides users with interactive features like details-on-demand, which consists of techniques that provide more details whenever users request them. The key idea is that RdViz gives users an overview of the raw data and then allows users to interactively select parts of the data for which they request more details. When users hover the mouse over a horizontal line or a circle, RdViz shows the contents (i.e., items within that transaction represented by circles on that line). For instance, when a user hovers the mouse over the second line in Figure 3, RdViz gives the details of transaction  $t_2$  (i.e.,  $\{a, c, d\}$ ).

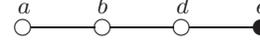
#### 4. FpViz: FREQUENT PATTERN VISUALIZATION MODULE

In this section, we present another module of our proposed FpVAT—namely, the *Frequent pattern Visualization (FpViz)* module. Here, FpViz is connected to a frequent pattern mining algorithm (e.g., FP-growth [9]), which finds frequent patterns from transaction database. Once frequent patterns are found, FpViz effectively displays them for data analysis. Note that FpViz is not confined to using FP-growth for frequent pattern mining. It can use some other frequent pattern mining algorithms (e.g., Apriori [1] for traditional frequent pattern mining, DCF [15] for constrained mining, UF-streaming [20] for stream mining, UF-growth [24] for uncertain data mining).

Like FIsViz [21], our proposed FpViz module also shows frequent patterns consisting of  $k$  items (i.e.,  $k$ -itemsets) in a two-dimensional space. The  $x$ -axis shows the  $m$  domain items. These items can be arranged in any order specified by users. For example, the user can arrange the items in (i) non-ascending frequency order, (ii) lexicographical order, or (iii) some other orders (e.g., put those items of interest—such as promotional items—on the left and less interesting items on the right side of the  $x$ -axis) for constrained mining. The  $y$ -axis shows the frequencies of the frequent patterns. Unlike FIsViz (which represents frequent patterns as polylines), the basic representation for our proposed FpViz module is an orthogonally laid out node-link diagram. According to graph aesthetics [34], reducing the number of edge crossings can improve the legibility of graphs. Similarly, assigning uniform lengths to edges and minimizing bends can enhance the legibility of the node-link diagram. Since the number of frequent patterns is potentially very large, a primary criterion in our design is to minimize edge crossings and bends. We, therefore, adopted an orthogonal layout mechanism that preserves edge crossings to a minimum. Bends occur only at  $0^\circ$  or  $90^\circ$  angles. As a result, FpViz minimizes crossings, facilitating legibility and visual comprehension.

Like RdViz (which represents items within each transaction as filled circles connected by a horizontal line), our proposed FpViz represents  $k$  items within each frequent pattern  $X$  as

$k$  circles connected by a horizontal line with the last circle filled. For example, the 4-itemset  $\{a, b, d, e\}$  is represented by a horizontal line connecting four circles (where each circle represents an item), as follows:

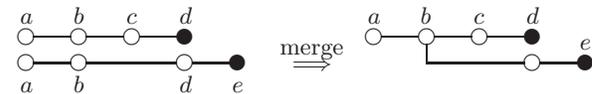


Note that the filled circle (i.e., disc) represents the last item (according to the item order  $\mathcal{R}$ ) in the frequent pattern. For singletons (i.e., 1-itemsets), they are represented by just filled circles in FpViz. For example, the singleton  $\{e\}$  is represented as:



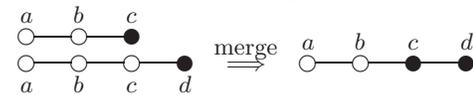
#### 4.1 Showing Frequencies of Multiple Frequent Patterns: Merging, Collapsing, and Expanding Horizontal Lines

With the above representation, the frequency of a frequent pattern consisting of  $k$  items (which is represented by a horizontal line connecting  $k$  circles with the last circle filled) is indicated by the  $y$ -position of the filled circle. This way of showing the frequencies work reasonable well when each frequent pattern has a distinct frequency (i.e., at most one horizontal line for each frequency value—value of the  $y$ -position). However, in many real-life situations, it is not uncommon that multiple frequent patterns happen to have the same frequency. In these situations, we apply *compression techniques* to our proposed FpViz: If the two frequent patterns  $X$  and  $Y$  of the same frequency share the same prefix, then their common prefix is merged. The suffixes of  $X$  and  $Y$  are then branching out from the last item of the common prefix. For example, if frequent patterns  $\{a, b, c, d\}$  and  $\{a, b, d, e\}$  (which share the same prefix  $\{a, b\}$ ) are of the same frequency, they can be represented as follows:



Here,  $\{c, d\}$  and  $\{d, e\}$  are two branches of the common prefix  $\{a, b\}$ .

A special case of the merge occurs when a suffix of  $Y$  is branching out from the last item of  $X$  (i.e.,  $X$  is a prefix of  $Y$ ). In this case, the two horizontal lines representing the two frequent patterns  $X$  and  $Y$  would be merged into one line. For example, for frequent patterns  $\{a, b, c\}$  and  $\{a, b, c, d\}$ , the former is a prefix of the latter. Hence, these two frequent patterns can be merged to form the following:



Here, the filled circle  $d$  indicates the last item of the frequent pattern  $\{a, b, c, d\}$ , whereas the filled circle  $c$  indicates the last item of the prefix  $\{a, b, c\}$ . Note that this merge helps reduce the number of horizontal lines to be drawn (i.e., reduce the amount of vertical space required for displaying the frequencies of all the frequent patterns).

When the number of mined frequent patterns is not huge, the merging of patterns with their prefixes having the same frequencies (e.g., the case for  $\{a, b, c\}$  and  $\{a, b, c, d\}$ ) reduces the amount of vertical space required. However, when the number of mined frequent patterns is huge, we may still run out of vertical space to fit all horizontal lines representing all the mined frequent patterns—even when merging is applied. Hence, we need to apply further compression

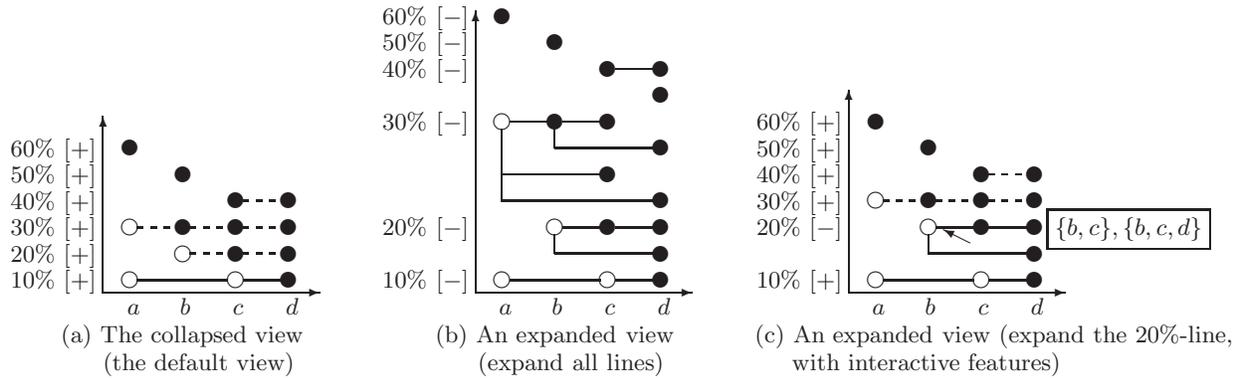
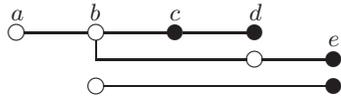


Figure 4: Expanded and collapsed views for visualizing frequent patterns with our proposed FpViz.

technique as follows. To reduce the amount of space required in the  $y$ -direction, if multiple frequent patterns (say,  $m$  frequent patterns represented by  $m'$  horizontal solid lines, where  $m' \leq m$ ) have the same frequency, they are projected or collapsed into *one horizontal dashed line* (instead of  $m'$  solid lines). For instance, frequent patterns  $\{a, b, c, d\}$ ,  $\{a, b, c, e\}$  and  $\{b, e\}$  are of the same frequency:



These  $m = 4$  frequent patterns (represented by  $m' = 3$  horizontal solid lines) are collapsed into one horizontal dashed line, as shown below:



By so doing, each existing frequency value would be represented by one—and only one—dashed horizontal line. For example, Figure 4(b) shows  $m = 14$  frequent patterns represented by three disjointed filled circles  $\{a\}$ ,  $\{b\}$  &  $\{d\}$  and  $m' = 8$  horizontal lines for other 11 frequent patterns. Figure 4(a) shows how these  $m' = 8$  horizontal solid lines are collapsed into four lines by using our proposed FpViz module. The resulting view shows two disjointed filled circles and four lines, which represent  $m = 14$  frequent patterns having  $2 + 4 = 6$  distinct frequencies.

Here, FpViz uses a dashed line (instead of a solid line) to represent the result of collapsing multiple horizontal solid lines. Circles on the dashed line only indicate their presence in some frequent patterns having such a frequency value represented by the dashed line. Note that there is no guarantee (although it is possible) that all items represented by circles on the dashed line appear together in the same frequent pattern. In other words, a solid line connecting  $k$  circles represents a frequent  $k$ -itemset (in which all  $k$  items appear together within the same frequent pattern), whereas a dashed line connecting  $k$  circles only indicates the presence of each of the  $k$  items appear in some frequent patterns of a particular frequency value (i.e., not necessary in the same frequent pattern). For example, in Figure 4(a), the dashed line at frequency=30% indicates the presence of items  $a, b, c$  &  $d$  in some frequent patterns of frequency=30%. Similarly, the dashed line at frequency=20% indicates the presence of items  $b, c$  &  $d$  in some frequent patterns of frequency=20%. Since these two lines are dashed, there is no guarantee (although it is possible) that patterns  $\{a, b, c, d\}$  and  $\{b, c, d\}$  are present at frequency=30% and 20% respectively. In this

illustrative example,  $\{b, c, d\}$  is present (at frequency=20%) but  $\{a, b, c, d\}$  is not.

Our proposed FpViz normally shows frequent patterns in the (default) *collapsed view* so as to reduce the amount of vertical space required for displaying all patterns. As this collapsed view may hide some details, FpViz provides users with the option to interactively expand any lines that are interesting by clicking the  $[+]$  buttons. By so doing, users would be able to obtain all the details. As an example, when the user clicks the  $[+]$  button for frequency=20% (as shown in Figure 4(b)), FpViz expands the horizontal dashed line representing frequent patterns of frequency=20%. Consequently, the user obtains the expanded view as presented in Figure 4(c), which shows frequent patterns  $\{b, c\}$ ,  $\{b, c, d\}$  &  $\{b, d\}$ . Note that, on the one hand, if a horizontal line is dashed (e.g., frequency=20%), clicking the  $[+]$  button gives an expanded view. On the other hand, if a horizontal line is solid (e.g., frequency=10%), clicking the  $[+]$  button gives an expanded view that is identical to its collapsed view. Moreover, users are not confined to clicking only one  $[+]$  button, they could click all  $[+]$  buttons to obtain an expanded view as shown in Figure 4(b).

With this representation of frequent patterns and their frequencies in our proposed FpViz module, users can observe the following from the default collapsed view.

*Observation 1.* By default, FpViz arranges the domain items in non-ascending frequency order. As a result, the most frequently occurring item (with the highest frequency) appears on the left side and the least frequently occurring one appears on the right side. In other words, users can easily gain insight about the frequency ranking of all the domain items by walking along the  $x$ -axis. For example, we observed from Figure 4(a) that item  $a$  is the most frequent domain item, which is followed by items  $b$  and  $c$ , and item  $d$  is the least frequent domain item. (It is important to note that, users are not confined to this ordering; they can choose other ordering  $\mathcal{R}$  to arrange all items in the domain.)

*Observation 2.* The frequency of any subset of a frequent pattern  $X$  is guaranteed to be higher than or equal to that of  $X$ . Hence, the disjointed filled circle representing any singleton subset of  $X$  (or the horizontal line representing any non-singleton subset of  $X$ ) is guaranteed to appear on or above the horizontal line representing  $X$ . For example, let us consider  $\{c, d\}$ , which is a subset of frequent pattern  $\{a, c, d\}$ . We observed from Figure 4(a) that the frequency of the subset  $\{c, d\} = 40\%$ , which is higher than that of

pattern  $\{a, c, d\}$  (10%). Similarly, we observed that the frequency of both  $\{b, c\}$  and  $\{b, c, d\}$  are the same (of 20%). Conversely, the frequency of any superset of a frequent pattern  $X$  is guaranteed to be lower than or equal to that of  $X$ . Hence, the horizontal line representing any superset of  $X$  is guaranteed to appear on or below the horizontal line representing  $X$ . For example, we observed from Figure 4(a) that the frequency of the superset  $(\{a, c, d\}) = 10\%$ , which is lower than that of  $\{c, d\}$  (40%). The frequency of both  $\{b, c, d\}$  and  $\{b, c\}$  are the same.

*Observation 3.* The highest frequency value for each item  $u$  is the frequency of the singleton  $\{u\}$ . For example, the highest frequency value of item  $b$  is 50%, which is the frequency of the singleton  $\{b\}$ .

*Observation 4.* If a horizontal line is solid, then we do not need to expand it because a solid line is an indication of an existing frequent pattern (i.e., not a result of collapsing multiple lines). In such case, the expanded view is identical to its collapsed view (cf. the 10%-line in Figure 4(a) and Figure 4(b)).

*Observation 5.* For a horizontal line that involves *only two* items  $u$  &  $v$ , if the line is solid, then (i) the last circle must be filled and (ii) the itemset containing these two items (i.e.,  $\{u, v\}$ ) is frequent at that frequency =  $y\%$ . Moreover, if such a solid line starts with a filled circle, then this line gives additional information that (iii) the 2-itemset  $\{u, v\}$  and its singleton prefix  $\{u\}$  are of the same frequency =  $y\%$ . If the line is dashed (instead of solid), then it tells users that (iv) the singleton itemset containing the last item (i.e.,  $\{v\}$ ) is also frequent at that frequency =  $y\%$ . The 40%-line in Figure 4(a) is an example that frequent patterns  $\{c, d\}$ ,  $\{c\}$  &  $\{d\}$  are of the same frequency of 40%.

*Observation 6.* For a horizontal (solid or dashed) line representing frequency= $y\%$ , if the first circle (representing item  $u$ ) is filled, then the singleton  $\{u\}$  has frequency of  $y\%$ . The 50%-line and 40%-line in Figure 4(a) show two examples that frequencies of  $\{b\}$  &  $\{c\}$  are 50% and 40% respectively.

*Observation 7.* For other horizontal dashed lines (e.g., involve more than two items, first circle is hollow), FpViz provides information about the *absence* of any items from frequent patterns of that frequency. For example, since item  $a$  does not appear in the 20%-line,  $a$  is guaranteed not to appear in any patterns of that frequency. However, one may not be able to easily determine the contents of the frequent patterns represented by this line. For example, the 20%-line in Figure 4(a) could represent frequent pattern  $\{b, c\}$  with any of the frequent patterns  $\{b, c, d\}$ ,  $\{b, d\}$  &  $\{c, d\}$ . (This explains why FpViz provides users with  $[+]$  buttons for interactively expanding the lines to clearly obtain detailed information.)

In addition, users can also observe the following when they interactively click one or more  $[+]$  buttons to expand horizontal dashed lines of their interest.

*Observation 8.* After expanding a horizontal dashed line, users can obtain the cardinality of any frequent pattern by counting the number of circles on an expanded solid line (from the leftmost circle to a filled one) representing such

a pattern. For example, we can expand the 20%-line in Figure 4(b) to get Figure 4(c), from which we can easily determine that the cardinality of  $\{b, c\}$  is 2 because there are 2 circles from the leftmost one to the first filled one along the 20%-line shown in Figure 4(c). While users can count the number of circles to determine the cardinality of a frequent pattern, FpViz provides the feature called *query on cardinality* for user convenience. See Section 4.3.

*Observation 9.* If the circle representing the last item of any prefix  $X$  of a frequent pattern  $X'$  is hollow, then the frequency of  $X$  is different from that of  $X'$ . For example, we observed from Figure 4(c) that the frequent pattern  $\{b, c\}$  does not have the same frequency as its prefix  $\{b\}$  (20% vs. 50%) because the circle representing  $b$  in  $\{b, c\}$  is hollow.

*Observation 10.* If there are more than one filled circle on an expanded solid line representing a frequent pattern  $X$ , then at least one of its prefix have the same frequency. For example, as shown in Figure 4(c), frequent patterns  $\{b, c, d\}$  and its prefix  $\{b, c\}$  have the same frequency of 20%.

## 4.2 Scalability Issues

Recall that FpViz applies the compression technique for merging several horizontal lines that represent frequent patterns of the same frequency into one line. To a further extent, if the number of distinct frequencies in the mining results exceeds the number of vertical pixels (or the number of horizontal lines allocated for the space), we can apply the compression technique further. To elaborate, in addition to compressing the frequent patterns of the same frequency (e.g.,  $sup=72\%$ ), we can also compress those frequent patterns with the same frequency range (e.g.,  $sup \in [70\%, 75\%]$ ). The lesser the allocated space, the broader would be the frequency range to be used in compression.

Besides compressing/merging several horizontal lines (which represent several frequent patterns) into a single line, FpViz can also provide users with some alternative options. For example, we can allow users to pick an option to display and visualize only those *closed frequent patterns* [28] or *maximal frequent patterns* [4] (instead of showing all frequent patterns). A frequent pattern  $X$  is *closed* if there does not exist any proper superset of  $X$  having the *same frequency as  $X$* , and a frequent pattern  $Y$  is *maximal* if there does not exist any proper superset of  $Y$  that is also *frequent*. By visualizing only closed or maximal frequent patterns, FpViz can reduce the number of horizontal lines that need to be shown.

While the above dealt with the scalability issue for the  $y$ -direction (i.e., scalable for large range of frequencies), we discuss here the scalability issue for the  $x$ -direction. More specifically, what if the number of items exceeds the number of pixels allocated for the  $x$ -direction? The challenge is that the  $y$ -direction contains frequency information (which are numerical data) but the  $x$ -direction contains items (which are categorical data). In FpViz, we can apply the following two techniques. First, if items are arranged in decreasing frequency order, then we can group several adjacent items together into a “mega”-item. For example, “mega”-item<sub>1</sub> represents all the domain items having the top 10 frequencies, “mega”-item<sub>2</sub> represents all the domain items having the next 10 highest frequencies, etc. Alternatively, we can group items according to some hierarchy or taxonomy. For example, we could group items “apples”, “bananas” & “cherries”

into a mega-item “fruits”. Similarly, we could also group “donuts” & “egg-tarts” into a mega-item “snack”.

### 4.3 Visual Clues and Interactive Features

The above representation of FpViz allows users to gain insight about the overview distribution of data mining and analysis results in the form of frequent patterns (in the default collapsed view). It also provides users with some relevant details (in the expanded view). In this section, we describe some additional visual clues and interactive features of our proposed FpViz module. While these features are not essential, they provide user convenience.

**QUERY ON FREQUENCY.** With FpViz, users can easily find all *frequent patterns* (i.e., with frequencies exceeding the user-specified minimum frequency threshold *minsup*) by ignoring everything that lies below the “threshold line”  $y = \text{minsup}$  (i.e., ignoring the lower portion of the graph). To a further extent, the representation of frequent patterns in FpViz leads to effective *interactive mining*. To elaborate, with FpViz, users can see what (and how many) frequent patterns are above a certain frequency. Based on this information, users can freely adjust *minsup* by moving the slider (see the vertical bar on the left side in Figure 5(b))—which controls *minsup*—up and down along the  $y$ -axis to find an appropriate value for *minsup*.

**QUERY ON CARDINALITY.** FpViz allows users to pose a query on cardinality, and it only shows frequent patterns of the user-specified cardinality  $k$ . Moreover, FpViz also allows users to pose a *range query on cardinality* so that only those frequent patterns with cardinality  $k$  within the user-specified range  $[k_{min}, k_{max}]$  are drawn.

**QUERY ON FREQUENT PATTERNS.** FpViz also allows users to interactively select certain domain items of interest (e.g., promotional items in a store) and to pose queries on frequent patterns.

**QUERY ON RELATIONSHIPS AMONG FREQUENT PATTERNS.** Recall that users can easily find the prefixes or extensions of a frequent pattern that share the same frequencies. However, sometimes prefixes or extensions (or more general case, subsets or supersets) of a frequent pattern  $X$  may have different frequencies than that of  $X$ . Hence, FpViz provides interactive features to highlight these subsets or supersets (prefixes or extensions) of a pattern of user interests. Highlighting these subsets and supersets of a pattern is very helpful in computing the confidence of association rules and forming rules with high confidence values. To elaborate, once users observe the frequencies of a frequent pattern  $X$  and its superset  $(X \cup Y)$ , they can compute the confidence of association rule  $X \Rightarrow Y$  using  $\frac{\text{frequency}(X \cup Y)}{\text{frequency}(X)}$ .

**DETAILS-ON-DEMAND.** FpViz provides users with the interactive feature of details-on-demand, which consists of techniques that provide more details whenever users request them. The key idea is that FpViz gives users an overview of the entire set of frequent patterns in an expanded view and then allows users to interactively select parts of the patterns for which they request more details. When users hover the mouse over a horizontal *solid* line, FpViz shows the contents (i.e., items within that frequent pattern represented by circles on that solid line). For instance, when a user hovers the mouse over the first solid line for frequency=20% in Figure 4(c), FpViz gives the details of frequent patterns represented by such a solid line (i.e.,  $\{b, c\}$ ,  $\{b, c, d\}$ ). Note that

FpViz does not provide users with details-on-demand when the mouse is hovered over a *dashed* line in a collapsed view because a dashed line indicates that it is a result of collapsing multiple solid lines. Users should instead click the [+]  
button to expand that line and then explore its contents.

**GUARANTEED VISIBILITY.** Our proposed FpViz allows users to specify their preference on visualization of frequent patterns. For example, if users are interested in finding those patterns containing fruits, then FpViz ensures that all corresponding horizontal lines are clearly visible.

## 5. EVALUATION RESULTS

In this section, we show our results on evaluating the proposed FpVAT. Here, we conducted five sets of evaluation tests. In the first set, we tested functionality of our FpVAT by showing how it can be applicable in various scenarios. In the second set, we tested performance of our FpVAT. In the next two sets, we evaluated readability and interactivity of FpVAT. In the last set, we demonstrated the applicability of FpVAT in several applications.

### 5.1 Evaluating the Functionality

In the first set of evaluation tests, we compared our proposed FpVAT with existing visualization systems. Recall from Section 1 that many existing systems either visualize raw data or the mining results, but not both. For systems that visualize the mining results, most of them were not designed to visualize frequent patterns. Among all those that visualize frequent patterns, we picked FIsViz [21] for comparison as we have easy access to FIsViz. Here, we considered many different real-life scenarios. For each scenario, we determined whether these systems can handle the scenarios. If so, we examined how these systems displayed the mining results. The evaluation results show that our FpViz was as effective as FIsViz in all these scenarios. A few samples of these scenarios are shown below.

*Q1(a) What kinds of vegetables are frequently purchased by customers?* Frequently purchased vegetables are patterns with high frequency. With FIsViz, as polylines representing itemsets may cross each other, users may *not* be able to easily see the itemsets of high frequency if they are in the dense or clustered area of the display. In contrast, our FpViz shows all frequent patterns by horizontal lines, which are easily visible and never cross any other horizontal lines. Let us compare Figure 5(a) with Figure 5(b). The former is a snapshot of FIsViz [21] and the latter is a snapshot of the FpViz module presented in Section 4. These two snapshots both show the same set of frequent patterns.

*Q1(b) How frequently are the vegetables purchased individually?* Depending on the density of the display in FIsViz, the frequencies of some itemsets may *not* be visible if they are in the dense or clustered areas of the display. In contrast, users can easily obtain the frequencies of patterns from the RdViz module of our proposed FpVAT.

*Q1(c) How frequently are the vegetables purchased together?* Again, users can easily obtain the frequencies of patterns from the FpViz module of our proposed FpVAT because there is no line crossing.

*Q1(d) What kinds of vegetables that are frequently purchased together with carrots?* Both FIsViz and FpViz provide users with a feature of handling queries on frequent patterns containing some specific items (in this scenario, carrots).

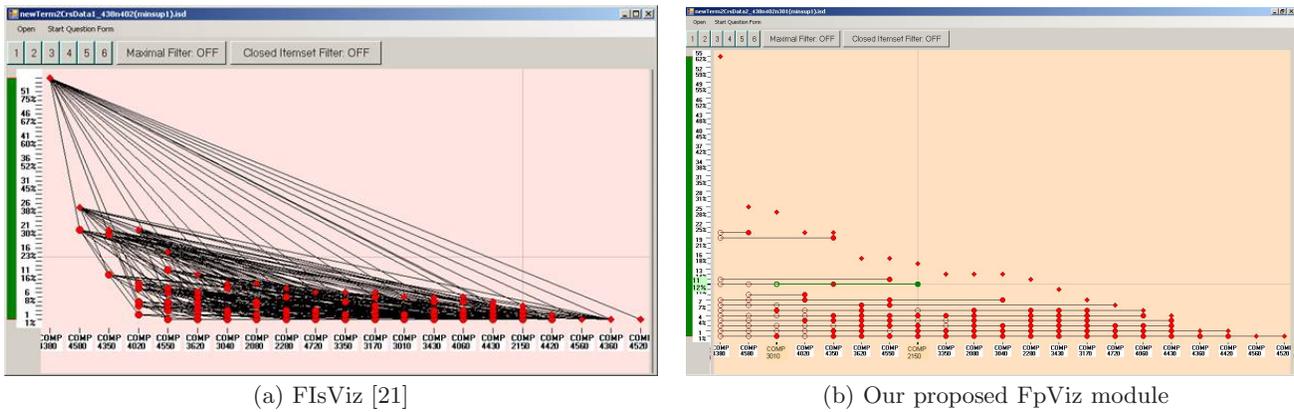


Figure 5: Two visualizers showing the same set of frequent patterns mined from a student-course DB.

We observed from all scenarios (including the above four samples) that our proposed FpVAT either retained the existing features of FIsViz (e.g., for Q1(d)) or provided additional improvements over FIsViz (e.g., for Q1(a)–(c)).

### 5.2 Evaluating the Performance

In the performance test, we used (i) several IBM synthetic datasets [1], (ii) some real-life databases (e.g., mushroom dataset) from UC Irvine Machine Learning Depository, (iii) some CNN documents, and (iv) a student-course database for our university. The results produced are consistent. In the first experiment, we varied the size of the database. We measured the time to (i) construct the display layout for database transactions (by using our proposed RdViz), (ii) mine frequent patterns (by using FP-growth [9]) and (iii) construct the display layout for the mined patterns (by using our proposed FpViz). The results showed that the runtime (which includes CPU and I/Os) increased linearly with the number of transactions in the database. In the second experiment, we varied the number of items in the domain. The results showed that the runtime increased when the number of domain items increased. In the third experiment, we varied the user-defined frequency threshold. When the threshold increased, we observed that the number of frequent patterns that satisfy the threshold (i.e., frequent patterns to be displayed by FpViz) decreased, which in turn led to a decrease in runtime. The number of transactions displayed by RdViz remained unchanged as it is independent of the threshold.

### 5.3 Evaluating the Readability of Raw Data and Mining Results

To assess the effectiveness of conveying frequent pattern relationships, we carried out a user evaluation with FpVAT. The evaluation was primarily case-based, within which several types of users were required to solve many different questions based on the visualizations of a given dataset (e.g., a database containing information about courses taken by students (see Figure 5)). Therefore, the scenario was that users need to identify a set of relationships and make decisions based on their observations.

We recruited 24 participants and separated them into two groups: (i) those who have data mining background and (ii) those who do not. None of the participants (regardless which of the two groups) was exposed to any form of

visualization for frequent patterns—including our proposed FpVAT.

To test the expressiveness of our visualization, we formulated two types of questions: multiple choices and those open-end ones that require participants to perform some level of analytical reasoning with the visualization. Sample questions include the following:

1. Which course is most frequently taken (i.e., course with highest enrolment)?
2. Which course is the next/second most frequently taken (i.e., course with second highest enrolment)?
3. Which two courses are most frequently taken?
4. What three courses are taken together by exactly four students?
5. How would you use FpVAT to make any changes in terms of how upper-level courses are offered and/or distributed?
6. If you were to reduce the offerings of an upper-level course, which one would you select and why (i.e., how did FpVAT lead to your conclusions)?
7. If you were to schedule exams of these courses, which pair of courses would you avoid scheduling on the same day?

We began the evaluation by presenting our FpVAT (especially, FpViz) and asking the participants to explore it at their own will. We did not give them any information regarding what the symbols and representations meant in the visualization. We first questioned them on what they were able to identify. Evaluation results showed that all the participants were able to identify the basic meaning behind the representations (e.g., that frequency was assigned to the *y*-axis, and courses to the *x*-axis). Participants were also able to identify the most frequently taken courses, without having us to tell them the answer.

Afterwards, we gave the participants detailed information on how to read the graphs and what the various lines (solid vs. dashed) and circles (hollow vs. filled) meant. This information was then followed by a set of questions that queried into the participants’ ability to simply read the graph, with a set of close-ended questions. Evaluation results showed that a majority of the participants were able to correctly answer most—and some even correctly answer all—of the questions. Statistically, the average accuracy rate was above 80% and three participants obtained an accuracy rate of 100%.

Several interesting observations were found at the post evaluation interview. For example, the participants told us that, while we gave the participants a multiple choice list to help them identify the answers, they did not use the multiple choice questions to guide their selection. Instead, their curiosity in understanding the visualizations led them to answer the questions by looking at the graphs first, and then confirming their answer with one of the choices provided. Moreover, the results were similar in both participant groups (with or without data mining background). This showed that the easy readability of FpVAT. This also suggested that, with very little training, participants felt comfortable to use the visualizations. Furthermore, they were able to quickly assimilate the representations, to the point at which they were able to answer all questions adequately.

Finally, we asked the participants a set of open-ended questions. Each participant completed the evaluation separately (i.e., no discussion among the participants). The results showed that the participants were able to make the best use of visualization for answering these questions.

#### 5.4 Evaluating the Benefits of the Interactive Features

We also evaluated the effectiveness of the interactivity of our FpVAT on various datasets mentioned above. We divided a set of 24 participants into two groups. The first group performed some tasks using *only* the essential features (e.g., without using any interactive features described in Section 4.3), and then used both the essential and the interactive features to answer some similar but not identical tasks. The second group did so in the reverse order (i.e., with interactive features and then without interactive features). By so doing, we avoided measuring the unwanted effect of learning (i.e., participants may learn from the first set of tasks). We observed the following from the results: (i) The participants were able to correctly answer all the questions using the interactive features. (ii) Most participants were able to do so without using the interactive features, but required much longer time. On average, participants took about 5 minutes to complete all the questions when using the interactive features, but took longer than 12 minutes to complete all the questions when not using any interactive feature. This indicated that, while interactive features are not essential, they provide convenience to users (and saved their time). Hence, it is beneficial to use the interactive features provided by FpVAT.

#### 5.5 Evaluating the Applicability of FpVAT

In this section, we very briefly demonstrated how our proposed FpVAT can be applicable in various applications. For example, Figure 6 shows how RdViz represents the raw data and gives users an insight about the distribution of a seasonal database [25].

Moreover, we also applied FpVAT to the VAST 2008 Mini Challenge on Cell Phone Call Records. Our RdViz module showed the raw data (i.e., 9,835 transactions), from which users could derive insight. The FpViz module revealed that Caller 200 made 23 calls and received 24 calls for a total of 47 calls. As Caller 200 (Ferdinando Catalano) was of interest to users (say, investigators), users applied an interactive feature of our FpVAT to select only those transactions related to Caller 200. As a result, FpViz displayed only those frequently occurring pairs (e.g., calls made between Caller 200

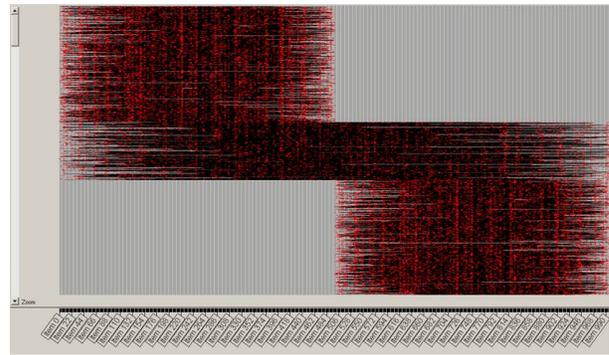


Figure 6: A zoom-out view of applying FpVAT (specifically, the RdViz module) to visualize raw data (transactions) in a seasonal database.

and his close relatives and friends). It revealed that Caller 200 communicated most frequently with Caller 5 (who was highly likely to be Ferdinando’s brother, Estaban). Next few frequent callers (in terms of the number of calls) were Callers 1, 2 and 3 (with nine, eight and eight calls), who were likely to be David Vidro, Juan Vidro and Jorge Vidro. To learn more details, we divided the raw data according to the date. With the interactive features, our FpVAT then showed the details of communication between Caller 200 and others. For example, in Day 1, Callers 5, 1, 2 and 3 called Caller 200, who also called Callers 5 and 1. In Day 2, Callers 5, 1, 2 and 3 made calls to (and received calls from) Caller 200, who also called Caller 137. For the next five days, Caller 200 only communicated with some of Callers 5, 1, 2 and 3. It is interesting to observe that Caller 200 did not make or receive any calls in Day 8. In Days 9 and 10, communication between Caller 200 and others was greatly reduced and it revealed that Caller 200 mostly called Caller 97. As frequent pattern mining is a fundamental step of various KDD tasks. Based on the information displayed by FpVAT, users can then take appropriate action (e.g., construct a social network graph between Caller 200 & his callers and discover interesting knowledge from it).

## 6. CONCLUSIONS

Many frequent pattern mining algorithms return a collection of the data analysis results in the form of a textual list of frequent patterns. This list can be very long and difficult to comprehend. Since “a picture is worth a thousand words”, it is desirable to have visualization systems. However, many existing visualization systems were not designed to show frequent patterns. To improve this situation, we proposed and developed a simple yet useful *frequent pattern visual analytic tool* called *FpVAT*, which provides users with explicit and easily-visible information among the input (i.e., raw data) and output (i.e., mining results in the forms of frequent patterns) of the frequent pattern mining process. Among the two modules in FpVAT, the RdViz module presents items in each transaction (i.e., raw data) as circles connected by a horizontal line. The FpViz module represents frequent patterns as horizontal lines in a two-dimensional graph. If multiple patterns have the same frequency, the corresponding lines representing these patterns are collapsed into one line. With this compression technique, FpViz allows the user to expand some portion (or all) of the “collapsed” mining re-

sults for data or result exploration. Moreover, both RdViz and FpViz also provide users with additional visual clues and interactive features for user convenience. Evaluation results showed the effectiveness of FpVAT in terms of functionality, performance, readability, interactivity, and applicability to some real-life situations. FpVAT provides users with visual support for visual analytics as well as knowledge discovery and data mining (KDD).

## 7. ACKNOWLEDGEMENTS

This project is partially supported by NSERC (Canada) in the form of research grants.

## 8. REFERENCES

- [1] R. Agrawal & R. Srikant. Fast algorithms for mining association rules in large databases. In *Proc. VLDB 1994*, pp. 487–499.
- [2] C. Ahlberg. Spotfire: An information exploration environment. *SIGMOD Record*, **25**(4), pp. 25–29, 1996.
- [3] M. Ankerst, C. Elsen, M. Ester & H.-P. Kriegel. Visual classification: An interactive approach to decision tree construction. In *Proc. KDD 1999*, pp. 392–396.
- [4] R.J. Bayardo Jr. Efficiently mining long patterns from databases. In *Proc. ACM SIGMOD 1998*, pp. 85–93.
- [5] J. Blanchard, F. Guillet & H. Briand. Interactive visual exploration of association rules with rule-focusing methodology. *KAIS*, **13**(1), pp. 43–75, 2007.
- [6] C. Brunk, J. Kelly & R. Kohavi. MineSet: An integrated system for data mining. In *Proc. KDD 1997*, pp. 135–138.
- [7] S.-M. Chan, L. Xiao, J. Gerth & P. Hanrahan. Maintaining interactivity while exploring massive time series. In *Proc. IEEE VAST 2008*, pp. 59–66.
- [8] C.H. Chih & D.S. Parker. The persuasive phase of visualization. In *Proc. KDD 2008*, pp. 884–892.
- [9] J. Han, J. Pei, Y. Yin & R. Mao. Mining frequent patterns without candidate generation: A frequent-pattern tree approach. *Data Mining and Knowledge Discovery*, **8**(1), pp. 53–87, 2004.
- [10] T. Iwata, T. Yamada & N. Ueda. Probabilistic latent semantic visualization: Topic model for visualizing documents. In *Proc. KDD 2008*, pp. 363–371.
- [11] D.A. Keim. Information visualization and visual data mining. *IEEE TVCG*, **8**(1), pp. 1–8, 2002.
- [12] D.A. Keim & H.-P. Kriegel. Visualization techniques for mining large databases: A comparison. *IEEE TKDE*, **8**(6), pp. 923–938, 1996.
- [13] D. Keim & J. Schneidewind (eds.) Special issue on visual analytics. *SIGKDD Explorations*, **9**(2), 2007.
- [14] Y. Koren & D. Harel. A two-way visualization method for clustered data. In *Proc. KDD 2003*, pp. 589–594.
- [15] L.V.S. Lakshmanan, C.K.-S. Leung & R.T. Ng. Efficient dynamic mining of constrained frequent sets. *ACM TODS*, **28**(4), pp. 337–389, 2003.
- [16] L.V.S. Lakshmanan, C.K.-S. Leung & R.T. Ng. The segment support map: Scalable mining of frequent itemsets. *SIGKDD Explorations*, **2**(2), pp. 21–27, 2000.
- [17] H. Lam, D. Russell, D. Tang & T. Munzner. Session viewer: Visual exploratory analysis of web session logs. In *Proc. IEEE VAST 2007*, pp. 147–154.
- [18] C.K.-S. Leung. Frequent itemset mining with constraints. In *Encyclopedia of Database Systems*, pp. 1179–1183, Springer, 2009.
- [19] C.K.-S. Leung & C.L. Carmichael. FpViz: A visualizer for frequent pattern mining. In *Proc. VAKD 2009*, pp. 30–39.
- [20] C.K.-S. Leung & B. Hao. Mining of frequent itemsets from streams of uncertain data. In *Proc. IEEE ICDE 2009*, pp. 1663–1670.
- [21] C.K.-S. Leung, P.P. Irani & C.L. Carmichael. FIsViz: A frequent itemset visualizer. In *Proc. PAKDD 2008*, pp. 644–652.
- [22] C.K.-S. Leung, P.P. Irani & C.L. Carmichael. WiFIsViz: Effective visualization of frequent itemsets. In *Proc. IEEE ICDM 2008*, pp. 875–880.
- [23] C.K.-S. Leung, L.V.S. Lakshmanan & R.T. Ng. Exploiting succinct constraints using FP-trees. *SIGKDD Explorations*, **4**(1), pp. 40–49, 2002.
- [24] C.K.-S. Leung, M.A.F. Mateo & D.A. Brajczuk. A tree-based approach for frequent pattern mining from uncertain data. In *Proc. PAKDD 2008*, pp. 653–661.
- [25] C.K.-S. Leung, R.T. Ng & H. Mannila. OSSM: A segmentation approach to optimize frequency counting. In *Proc. IEEE ICDE 2002*, pp. 583–592.
- [26] T. Munzner, Q. Kong, R.T. Ng, J. Lee, J. Klawe, D. Radulovic & C.K.-S. Leung. Visual mining of power sets with large alphabets. Tech. rep. UBC CS TR-2005-25, Dept. Computer Science, UBC, Canada, 2005.
- [27] D. Oelke, P. Bak, D.A. Keim, M. Last & G. Danon. Visual evaluation of text features for document summarization and analysis. In *Proc. IEEE VAST 2008*, pp. 75–82.
- [28] N. Pasquier, Y. Bastide, R. Taouil & L. Lakhal. Discovering frequent closed itemsets for association rules. In *Proc. ICDT 1999*, pp. 398–416.
- [29] K. Puolamäki (ed.) *Proc. VAKD 2009*.
- [30] J. Scholtz. Beyond usability: Evaluation aspects of visual analytic environments. In *Proc. IEEE VAST 2006*, pp. 145–150.
- [31] T. Schreck, J. Bernard, T. Tekusova & J. Kohlhammer. Visual cluster analysis of trajectory data with interactive Kohonen Maps. In *Proc. IEEE VAST 2008*, pp. 3–10.
- [32] R. Spence. *Information Visualization: Design for Interaction*, 2e. Prentice Hall, 2007.
- [33] C. Stolte, D. Tang & P. Hanrahan. Query, analysis, and visualization of hierarchically structured data using Polaris. In *Proc. KDD 2002*, pp. 112–122.
- [34] C. Ware, H. Purchase, L. Colpoys & M. McGill. Cognitive measurements of graph aesthetics. *Information Visualization*, **1**(2), pp. 103–110, 2002.
- [35] P.C. Wong & J. Thomas. Visual analytics. *IEEE CG&A*, **24**(5), pp. 20–21, 2004.
- [36] L. Yang. Pruning and visualizing generalized association rules in parallel coordinates. *IEEE TKDE*, **17**(1), pp. 60–70, 2005.
- [37] X. Yang, S. Asur, S. Parthasarathy & S. Mehta. A visual-analytic toolkit for dynamic interaction graphs. In *Proc. KDD 2008*, pp. 1016–1024.
- [38] J. Yuan, Y. Wu & M. Yang. From frequent itemsets to semantically meaningful visual patterns. In *Proc. KDD 2007*, pp. 864–873.