RadialViz: An Orientation-Free Frequent Pattern Visualizer

Carson Kai-Sang Leung[*] and Fan Jiang

Department of Computer Science, University of Manitoba, Canada
kleung@cs.umanitoba.ca

Abstract. Frequent pattern mining algorithms aim to find sets of frequently co-occurring items. Visual representation of the mining results is more comprehensible to users than the traditional long textual list of frequent patterns. Existing visualizers mostly show frequent patterns as graphs in a two-dimensional space with (x, y)-coordinates. Nowadays, in a collaborative environment, it is not uncommon for users to have face-to-face meetings when they show the graphs visualizing frequent patterns. In these situations, the viewing orientation of the graphs plays an important role as different orientations positively or negatively impact the graph legibility. A legible right-side-up graph to one user may become an illegible upside-down graph towards another user. In this paper, we propose a visualizer that uses a radial layout—which is orientation free—to show frequent patterns. Having such a visualizer is beneficial in the collaborative environment.

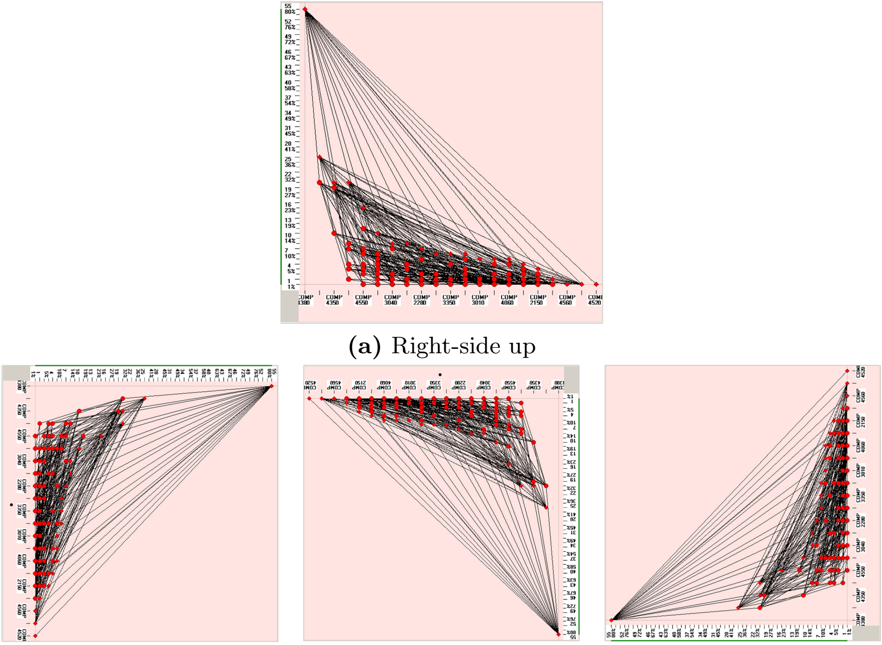
Keywords: Visual data mining, association analysis, frequent itemsets, human-machine interaction, pattern discovery.

1 Introduction

Frequent pattern mining [1] finds implicit, previously unknown, and potentially useful information in the form of sets of frequently co-occurring items or events (e.g., merchandises in a store, courses offered at a university). It plays an essential role in many knowledge discovery and data mining tasks. A common characteristic of these tasks is the identification of the frequencies of items, or sets of items, from datasets. For instance, a store manager may want to identify merchandise items that are frequently purchased together so as to place the items closer to each other (to reduce the distance required to travel by the shopper) or further apart (to encourage more purchase of items placed in between those frequently purchased ones). Similarly, a university administrator may want to know the collection of popular courses taken together by students in a semester (for lecture scheduling and exam scheduling). A book seller may want to recommend bundles of popular books to readers.

[*] Corresponding author.

P.-N. Tan et al. (Eds.): PAKDD 2012, Part II, LNAI 7302, pp. 322–334, 2012.



(a) Right-side up

(b) RHS (90° rot.) **(c)** Opposite side (180° rot.) **(d)** LHS (270° rot.)

Fig. 1. Viewing frequent patterns with FIsViz [16] at different orientations

Over the past two decades, data mining researchers have designed and developed numerous frequent pattern mining algorithms. However, many of these algorithms have been focused on either functionality or efficiency. These algorithms usually return the mining results in textual form (e.g., a very long list of frequent patterns). Consequently, users may not easily comprehend the knowledge and useful information from the textual list. Conversely, visual representation of these patterns would be more comprehensible to users. However, not too many visualization tools have been developed to support frequent pattern mining. A common characteristic among the visualizers that were designed to support frequent pattern mining (e.g., FIsViz [16], PowerSetViewer [19]) is that they display the mined frequent patterns in a traditional two-dimensional rectangular space. For instance, FIsViz lists domain items on the x-axis, shows frequency values on the y-axis, and visualizes frequent patterns as polylines drawn on this two-dimensional rectangular space with (x, y)-coordinates. As such, the orientation of the graph displaying the patterns plays an important role in legibility of the graph. Consider a situation in which two users are facing each other and are discussing the frequent patterns shown on the graph (e.g., a marketing analyst was asked by a store manager, who sits on the opposite side of a table, to discuss the sets of merchandise items that are frequently purchased by shoppers). When showing frequent patterns in the graph as supporting evidence, it may be right-side up to the manager (e.g., as shown in Fig. 1(a)) but upside down to the analyst (e.g., as shown in Fig. 1(c), from which important information such as frequency is not easy to read) and vice versa.

To summarize, the users who face the unfavourable orientation may have difficulty in comprehending the frequent patterns shown on the graph. To improve the situations, we propose in this paper a visualizer—called RadialViz—that uses a radial layout to visualize frequent patterns. The *key contribution* of this paper is our radial visualizer that shows the discovered frequent patterns in an orientation-free environment.

This paper is organized as follows. The next section provides background and discusses related work. We propose our visualizer in Sect. 3. Evaluation results are presented in Sect. 4. Finally, we present the conclusions in Sect. 5.

2 Background and Related Work

Development of effective visualization systems for data mining has been the subject of many studies. This line of research can be sub-classified into two general categories: (i) systems for visualizing data (e.g., VisDB [12], independence diagrams [4], Polaris [21]) and (ii) systems for visualizing the mining results (e.g., systems that visualize decision trees [3], association rules [5,11], and clusters [10,14]).

Recently, some tools and techniques have been designed to visualize patterns involving sets of items or related co-occurring entities [6,15,17,18]. For example, Wong et al. [23] designed visualization tools for visualizing topic association rules and sequential patterns appearing in documents. Their visual tools are similar to parallel coordinates, in which keywords appear on the parallel coordinate axes in the y-direction and the sequential index (temporal or others) on the x-axis.

Similarly, Yang [24,25] designed a system mainly to visualize association rules (but can also be used to visualize frequent patterns) in a two-dimensional space consisting of parallel vertical axes. In his system, all domain itemset are sorted according to their frequencies and 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. As the frequency of such a pattern is indicated by the thickness of the curve, it is not easy to compare the frequencies of patterns.

PowerSetViewer (PSV) [19] 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, and PSV does not show the relationship between related frequent patterns (e.g., it is not easy for users to spot the prefix/extension relationship among patterns $\{a\}, \{a, b\}$ and $\{a, b, c\}$). Note that $\{a\}$ and $\{a, b\}$ are *prefixes* of $\{a, b, c\}$. Equivalently, $\{a, b, c\}$ is an *extension* of $\{a, b\}$, which is then an extension of $\{a\}$. For any k-itemset Z in a domain of m items, there are $k - 1$ non-empty prefixes (i.e., not counting the empty set and Z itself) of Z and at most $2^{m-k} - 1$ extensions of Z.

FIsViz [16] was proposed in PAKDD 2008 to visualize frequent k-itemsets as polylines connecting k nodes in a two-dimensional space with (x, y)-coordinates, in which domain items are listed on the x-axis and frequency values are indicated by the y-axis. The x-locations of all nodes in the polyline indicates the domain items contained in a frequent pattern Z, and the y-location of the rightmost node of a polyline for Z indicates the frequency of Z. As such, prefix/extension relationships can be observed by traversing along the polylines.

Nowadays, in a collaborative environment, it is not uncommon for collaborators to have face-and-face meetings. Partially due to the emerging of tabletop displays as an effective platform for collaboration, information is shared on the tabletop surface in the meetings. As such, orientation or view perspective cannot be neglected. Unlike a single-user environment (where orientation may not be an issue), object orientation becomes critical in a multi-user environment because not all users share a common perspective of the displayed information. As information is viewed from different positions, it may be perceived differently. A recent study [2] showed that user perception (e.g., legibility or readability) of a chart decreases when the chart is not oriented right-side up. Let us use FIsViz as an example. When frequent patterns showed by FIsViz are rotated 90° or 270° clockwise (as shown in Figs. 1(b) and 1(d)) corresponding to the guests who sit on the right-hand-side (RHS) or the left-hand-side (LHS) of the host, guests may encounter difficulties in quickly reading the information. It may take much longer when the charts are put upside down (as shown in Fig. 1(c)). Hence, although FIsViz visualizes frequent patterns, it is *not* orientation free.

The aforementioned study [2] suggested that the legibility or readability can be improved by using radial charts. The sunburst technique [20,22] is a space-filling visualization that uses a *radial* layout (i.e., a ring chart, a multilevel pie, or concentric circles) [7] to offer an explicit portrayal of a hierarchical structure. Specifically, items in a hierarchy are laid out radially in sunburst. The root/top of the hierarchy is put at the center, and deeper/leaf levels are put farther away from the center (i.e., with the hierarchy moving outward from the center). Each hierarchical level forms a "block arc" or pie segment. An inner block arc (or a pie segment of an inner ring) bears a hierarchical relationship to those outer block arcs (or pie segments of an outer ring) which lie within the angular sweep of the parent arc. The arc length (and thus the central angle and area) of a block arc is usually proportional to the quantitative values associated with that arc. Fig. 2 shows how sunburst visualizes a hierarchical structure of customers. From this figure, we observe the following properties of hierarchical data represented by sunburst:

P1. All children of a node in the hierarchy are disjoint.
P2. The quantitative value associated with a parent node is higher than or equal to the *sum* of quantitative values associated with *all* its child nodes.
P3. Given P1 and P2, the quantitative value associated with a parent node is higher than or equal to the quantitative value associated with *each* of its child nodes.

Data in the hierarchical structure of customers in Fig. 2 possess the above three properties. For instance, a customer is either a member or a non-member. A member is either a gold, silver, or bronze member. Here, the quantitative value is

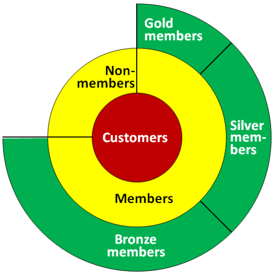


Fig. 2. An example of sunburst [20,22]

the count. The total customer count is the sum of the total numbers of members and non-members. The total membership count is the sum of the membership counts of all gold, silver, and bronze members.

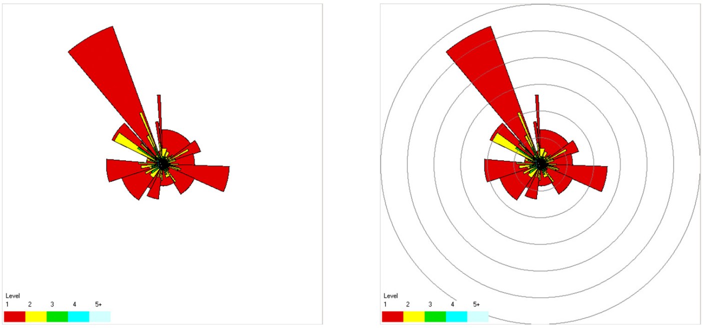
FP-Viz [13] uses a radial layout for visualizing tree-based frequent pattern mining. Specifically, FP-Viz visualizes the FP-tree [9] used in the mining process. As (i) all children of a node in an FP-tree are disjoint and (ii) the support value of a parent node is higher than or equal to the *sum* of *all* support values of its child nodes, then (iii) the support value of a parent node is higher than or equal to that of *each* of its child nodes. In other words, when FP-VIz visualizes the database transactions in the radial layout, the database transactions captured in the FP-tree satisfies Properties P1–P3 above. However, FP-Viz does *not* directly visualize frequent patterns, which need to be mined from the FP-tree. Moreover, a pattern Z may be embedded in multiple paths of an FP-tree (e.g., $\{b, e\}$ may be contained in paths representing transactions $t_i = \{a, b, c, d, e, f\}$ and $t_j = \{b, c, e, g\}$), and thus appears in different block arcs in FP-Viz. Consequently, it may not be easy to directly read the frequency of Z.

3 RadialViz: Our Proposed Visualizer

Recall from the previous section that FIsViz visualizes frequent patterns as polylines in a two-dimensional rectangular space, but FIsViz is not orientation free. In contrast, FP-Viz is orientation-free with a radial layout, but it shows the contents of an FP-tree (i.e., database transactions to be mined) instead of directly showing frequent patterns (i.e., the results mined from the FP-tree). In this section, we propose a visualizer—called ***RadialViz***—to use a radial layout (which is orientation free) to directly show frequent patterns and their relationships (e.g., prefix/extension relationships).

Visualizing the hierarchical structure of frequent patterns (and their prefix/extension relationships) in a radial layout is challenging because frquent patterns in the prefix/extension hierarchy does *not* satisfy Properties P1 and P2. We observe the following with *frequency* being the quantitative value.

1^-. Not all extensions of a frequent pattern Z are disjoint. In fact, extensions of Z are usually overlapping (e.g., as two extensions of $\{a, b\}$, both $\{a, b, c\}$ and $\{a, b, d\}$ are overlapping).



(a) RadialViz

(b) RadialViz with radial grid

Fig. 3. RadialViz shows the same set of frequent patterns as in Fig. 1

2^-. The frequency of a frequent pattern Z is *not* necessarily higher than or equal to the sum of frequencies of *all* extensions of Z.

3^+. Fortunately, the frequency of a frequent pattern Z is still higher than or equal to the frequency of *each* extension of Z.

We design RadialViz for visualizing frequent patterns based on these observations.

3.1 Basic Representation of Frequent Patterns in RadialViz

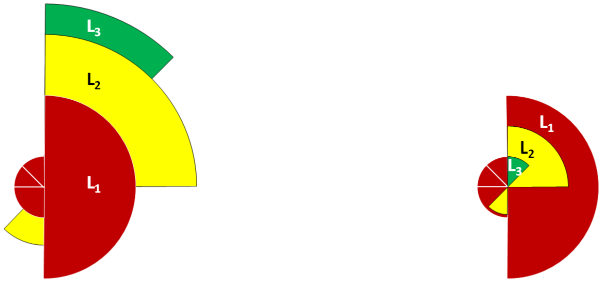
Recall that both sunburst and FP-Viz divide the central angle according to quantitative values associated with the nodes in a hierarchy. This works well for the hierarchy that satisfies Properties P1 and P2 (e.g., hierarchy in an FP-tree), in which (i) child nodes of a parent node are disjoint and (ii) the quantitative values associated with parent nodes are bounded below by the sum of quantitative values associated with their child nodes (e.g., the support value of a parent node in a tree path representing a set of similar transactions is at least the sum of support values of all its child nodes in an FP-tree). The central angle for each parent node is then subdivided according to the quantitative values associated with its child nodes. However, these two properties do not hold for visualization of frequent patterns as noted in Observations 1^- and 2^-. For example, if frequency of $\{a, b\}$ is 10, then individual frequency of its extension $\{a, b, c\}$ or $\{a, b, d\}$ is at most 10. However, their sum can range from 0 to 20. If the sum were above 10, then how can we represent $\{a, b, c\}$ and $\{a, b, d\}$ radiating from the sector or block arc representing $\{a, b\}$? A naive solution is to overlap the areas for these two extensions. This works for this particular example. However, what if there are multiple extensions of $\{a, b\}$ (e.g., for a domain of 100 items, there are potentially $2^{98} - 1 \approx 3 \times 10^{29}$ extensions of $\{a, b\}$ including potentially 98 immediate extensions of $\{a, b\}$). It is unclear how to overlap these 98 extensions so that the outcome is still comprehensible to users. As such, we cannot divide the central angle according to the frequency of a frequent pattern. This leads to two questions: (i) How to represent the frequency of a pattern, which plays an important role in frequent pattern mining? (ii) How to divide the central angle of the radial layout?

Representation of Frequency of a Frequent Pattern. To answer the first question, our *RadialViz uses radius* (instead of the central angle, sector area, or arc length) *to represent the frequency information.* By doing so, users can easily infer the frequency distribution of all frequent patterns. For example, one can easily spot from Fig. 3 the most popular course (with the high frequency) as it is indicated by the sector with the longest radius (on the upper left portion of the graph). Moreover, patterns with the same frequency have the same radius.

Representation of Cardinality of a Frequent Pattern. Representing frequency of patterns by radius may lead to the following question. In both sunburst and FP-Viz, each level of the hierarchy forms a ring or block arc. Here, when visualizing frequent patterns in RadialViz, each level of the hierarchy represents the cardinality k of k-itemsets. Given that RadialViz represents frequencies of patterns by radius, patterns of the same cardinality may not necessarily form a ring or block arc with the same radius from the center. The block arc for a k-itemset (e.g., with frequency=55) may appear much further away from the center than that for another k-itemset (e.g., with frequency=5). Since the block arcs for $(k+1)$-itemset extensions of a k-itemset Z is shown to be radiating from the block arc for Z, users can count the number of levels of block arcs to determine the cardinality of Z.

For user convenience, *RadialViz uses colour to represent the cardinality of frequent patterns.* By doing so, users can directly get the cardinality without counting multiple block arcs, each representing a cardinality level in the hierarchy. See Fig. 3, in which the colour bar at the bottom indicates the cardinality (e.g., from red indicating the minimum cardinality of 1 to light blue indicating the cardinality of 5^+ for the illustrative student database).

Representation of a Frequent Pattern. Recall that RadialViz uses colour to represent the cardinality of frequent patterns and uses radius to represent the frequency. Based on Observation 3^+, we know that the frequency of any extension of a frequent k-itemset Z is bounded above by the frequency of Z. This implies that the radii of block arcs for extensions of Z is bounded above by the radius of block arc for Z. Hence, we do not have to put the block arcs for $(k+1)$-itemset extensions of Z radiating from and *outside* the block arc for Z (as shown in Fig. 4(a)) so as to avoid having a radial graph spanning too far from the center. Instead, RadialViz stacks the block arcs or sectors for $(k+1)$-itemset extensions of Z on *top* of the block arc or sector for Z. By doing so, *RadialViz represents each frequent pattern by a sector radiating from the center.* There are several advantages of this representation of frequent patterns by RadialViz. First, the span of the radial graph is bounded above by the maximum radius of all singletons (i.e., 1-itemsets). See Fig. 4(b). Second, the prefix/extension relationship can then be represented through *containment* (i.e., sectors for the extensions of Z are contained in the sector for Z). Third, it is much easier to



(a) Radiating out level-by-level **(b)** Stacking up levels (as in RadialViz)

Fig. 4. Representation of frequent patterns

spot the change of frequency between Z and its extensions because sectors (for Z and its extensions) are all radiating from the same center.

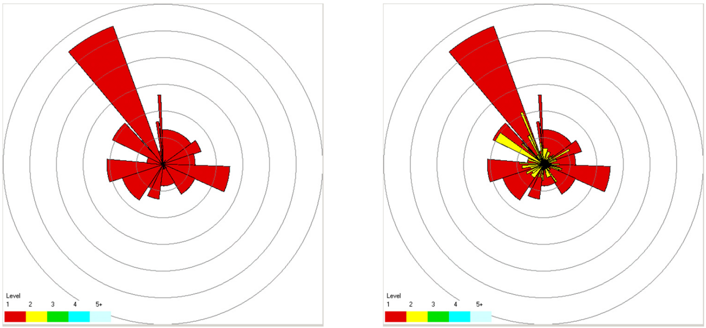
Next, let us answer the earlier question on how to divide the central angle of the radial layout. A naïve approach is to divide the central 360° angle among all singletons, and then recursively subdivide the angle associated for each k-itemset among their immediate $(k + 1)$-itemset extensions. However, a potential problem associated with this naïve approach is that some sectors may be very dense (due to the large number of frequent pattern extensions) while others may be very sparse (due to the small number of frequent pattern extensions). Hence, our RadialViz uses a different approach. Instead, *RadialViz divides the central 360° angle into p sectors, and each sector represents one of the p mined frequent patterns.*

3.2 Other Features and Observations on RadialViz

In the previous section, we introduced some essential features of our RadialViz. In this section, let us present some optional features of RadialViz.

Frequency of Frequent Patterns. Recall that RadialViz uses radius to represent frequency. Sectors with long radii represent frequent patterns with high frequencies, while sectors with short radii represent less frequent patterns. In many real-life applications, users need to compare frequencies of different patterns. *RadialViz provides users with the radial gridline* so that users can easily read off the frequency of different patterns and compare among them. For example, each ring formed by the radial gridline indicates a frequency increment of 10 in Fig. 3(b). So, users can easily learn that the enrolments of the two most popular courses are 55 and 26. There are also two courses with the same enrolment of 21.

Moreover, *RadialViz also provides users details-on-demand by allowing them to hover the mouse over a sector* to get a small box showing the frequency of the corresponding frequent pattern represented by the sector. For example, the longest red sector on the upper left indicates that the corresponding course was taken by most students. When users hover the mouse over such a long red sector, the small textbox appears and explicitly shows “55” as the exact frequency for that course.



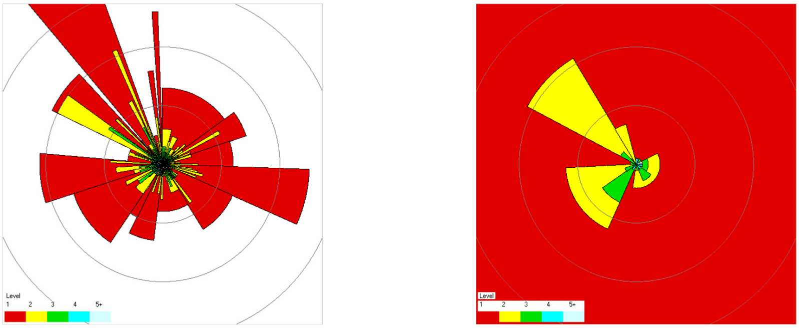
(a) Only frequent 1-itemsets **(b)** Frequent 1- and 2-itemsets

Fig. 5. Our proposed RadialViz showing frequent patterns of the first k levels/cardinalty (where k = (a) 1 and (b) 2) in Fig. 3

Ordering of Items in Patterns. Recall that FIsViz arranges domain items in descending frequency order. Such an ordering would be helpful if users just want to find out items with the highest or lowest frequencies. However, in many other real-life applications, it is more common for users to look up the frequency of some particular patterns of interest. In those applications, having the domain items arranged in descending frequency order means that users need to perform a linear scan for the interesting patterns. Hence, to facilitate easy lookup of frequent patterns, *RadialViz arranges items clockwise in some user-specified order* (e.g., alphabetical order). With such an arrangement, users can easily locate the patterns of interest. For instance, Figs. 3 and 5 show second- to fourth-year courses (arranged by course number clockwise from the 12 o'clock position). Users can infer some knowledge like "the most popular course is a fourth-year course". Moreover, knowing that the most popular course is COMP 4380 (with longest radius), if users want to find COMP 4350, then they only need to search in a counterclockwise direction for a sector very close to COMP 4380. Furthermore, with this item arrangement, users can still easily spot the patterns with the highest or lowest frequencies. The reason is that, as RadialViz uses radius to show frequency, patterns with the highest and lowest frequencies would have the longest and shortest radii, respectively.

Patterns of Some Specific Cardinality. Recall that RadialViz uses colour to represent the cardinality k of frequent k-itemsets. Frequent patterns of the same cardinality are represented by the same colour. Moreover, in many real-life applications, it is uncommon to find frequent patterns of certain cardinality. Hence, *RadialViz allows users to specify which cardinality levels to be displayed.* For instance, Fig. 5(a) shows only frequent patterns of cardinality 1 (i.e., 1-itemsets), whereas Fig. 5(b) shows only frequent patterns of both cardinalities 1 and 2 (i.e., 1- and 2-itemsets).

Zoom-In and Zoom-Out. When RadialViz shows all frequent patterns (as in Fig. 3), it gives users an overview about the distribution of all frequent patterns.



(a) Zooms in to the sector of interest **(b)** Zooms in to drilled-in sector of interest

Fig. 6. Our proposed RadialViz (a) zooms in and (b) drills in to the sector of interest in Fig. 3

As some sectors are small, RadialViz provides users with interactive features to *zoom in* and *zoom out* so that users can obtain information of the granularity level of their interest. See Fig. 6(a) for a zoom-in view.

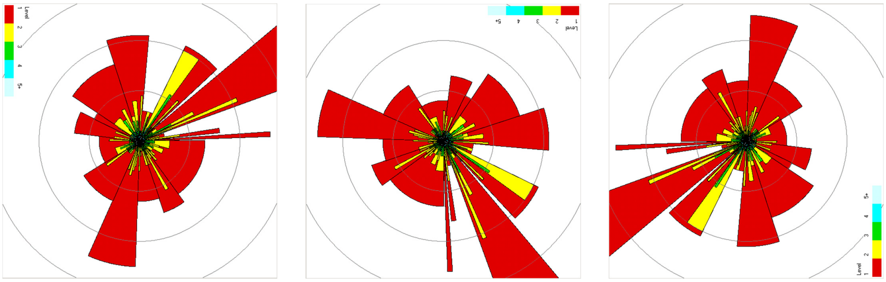
Drill-In. Moreover, RadialViz also provides users with interactive features to *drill in* some specific area of interest. The key difference between zoom-in and drill-in is that the former just magnifies the sector of interest (i.e., same layout) whereas the latter redraws the sector of interest. To get a close-up of the drilled-in image, RadialViz allows users to zoom in to the sector of interest in this drilled-in image. See Fig. 6(b) for the zoom-in view when we drilled in to details of COMP 4380. Note that combination of zoom-in and drill-in features is useful when dealing with large amounts of data.

4 Evaluation

In this section, we show our results on evaluating our proposed RadialViz. Here, we compare functionality and performance of our RadialViz with some existing systems (e.g., FIsViz [16] from PAKDD 2008). We conducted two sets of evaluation tests. In the first set, we tested functionality of our RadialViz by showing how it can be applied to various scenarios or real-life applications. In the second set, we tested performance of our RadialViz.

In terms of functionality, we considered many different real-life scenarios. We determined whether RadialViz can handle each scenarios. If so, we examined how it displays the mining results. The evaluation results show that RadialViz was effective in all these scenarios. A few samples of these scenarios are shown below:

Q1. Which course has the highest enrolment?
Q2. Which is the most frequent 2-itemset extensions of COMP 4380 and the most frequent 3-itemset extensions of COMP 4380?



(a) RHS (90° rot.) **(b)** Opposite side (180° rot.) **(c)** LHS (270° rot.)

Fig. 7. Viewing the frequent patterns in Fig. 6 with RadialViz at different orientations

Q3. How many frequent 2-itemset extensions of COMP 4380 and the most frequent 3-itemset extensions of COMP 4380 enrolled by more than 5 students?

Q4. What is the highest cardinality for the frequent extensions of COMP 4380?

Q5. Which frequent course pairs have the same frequency?

Recall from Fig. 1 the snapshots of different orientations of FIsViz. See Fig. 7 for snapshots of different orientations of RadialViz besides the right-side up view shown in Fig. 3.

With RadialViz, we easily located the course with highest enrolment (i.e., the sector with longest radius) regardless of the orientation. Although we spotted the same information from FIsViz when frequent patterns are shown right-side up, it took a bit longer time for other orientations of FIsViz.

To answer Q2 and Q3, we drilled in COMP 4380 and then zoomed in to the center (as shown in Fig. 6(b)). With the colour bar, we easily spotted the frequent 2-itemset and 3-itemset extensions (i.e., longest yellow and green sectors, respectively) of COMP 4380 from RadialViz regardless of the orientation. By hovering the mouse over the corresponding sectors, the course labels were revealed and answers were obtained (i.e., {COMP 4380, COMP 4580} with 21 students enrolled, and {COMP 4380, COMP 4550, COMP 4720} with 7 students enrolled). Similarly, we counted the number of yellow and green sectors with radius ≥ 5, and we got three course pairs and one course triplet that satisfy the enrolment condition. In contrast, for FIsViz, we needed to traverse all polylines going out from COMP 4380. As many of these polylines were bent and overlapping, it was not easy to trace and count each polyline. The situation was worsened when the graphs were not right-side up.

Similarly, for Q4, we just needed to look for the sector with colour representing the highest cardinality from the figure. In this case, it was light blue indicating four courses. For Q5, we easily spotted from Fig. 3(b) that COMP 4020 and 4350 have the same enrolment of 21. Answering these two questions in FIsViz again required traversal of those bent and overlapping polylines.

In terms of performance, we varied the size of databases. The results showed that the runtime (which includes CPU and I/Os) increased almost linearly with the number of transactions in the database. We also varied the number of items in the domain, and the results showed that the runtime increased when the number of domain items increased. Moreover, when the user-defined frequency threshold *minsup* increased, the number of itemsets that satisfy the threshold (i.e., itemsets to be displayed) decreased, which in turn leads to a decrease in runtime. As ongoing work, we are conducting more extensive experimental evaluation.

5 Conclusions

In this paper, we proposed a frequent pattern visualization system, called *RadialViz*, which enables users to visualize the mined frequent patterns. RadialViz represents k-itemsets using a radial layout (which is orientation free) and in a hierarchical fashion (so that extensions of a pattern Z are contained within the sector representing Z). Patterns of the same cardinality have the same colour, and patterns of different cardinalities have different colours. Since RadialViz uses radius to indicate the frequencies of patterns, users can easily observe the frequency distribution of all the patterns. Patterns having similar radius have similar frequencies. With interactive features (e.g., mouse hover, zoom-in, drill-in), users can easily explore patterns of interest. Evaluation results showed the effectiveness of RadialViz. Our proposed system helps users to answer many questions for real-life applications, and thus assist them in making appropriate business intelligence (BI) decisions, especially in face-to-face tabletop collaborative environments.

Acknowledgement. This project is partially supported by NSERC (Canada) and University of Manitoba.

References

1. Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., Verkamo, A.I.: Fast discovery of association rules. In: Advances in Knowledge Discovery and Data Mining, ch. 6. AAAI/MIT Press (1995)
2. Alallah, F., Jin, D., Irani, P.: OA-graphs: orientation agnostic graphs for improving the legibility of charts on horizontal displays. In: ITS 2010, pp. 211–220 (2010)
3. Ankerst, M., Elsen, C., Ester, M., Kriegel, H.-P.: Visual classification: an interactive approach to decision tree construction. In: ACM KDD 1999, pp. 392–396 (1999)
4. Berchtold, S., Jagadish, H.V., Ross, K.A.: Independence diagrams: a technique for visual data mining. In: KDD 1998, pp. 139–143 (1998)
5. Blanchard, J., Guillet, F., Briand, H.: Interactive visual exploration of association rules with rule-focusing methodology. KAIS 13(1), 43–75 (2007)
6. Carmichael, C.L., Hayduk, Y., Leung, C.K.-S.: Visually contrast two collections of frequent patterns. In: IEEE ICDM Workshops 2011, pp. 1128–1135 (2011)
7. Di Caro, L., Frias-Martinez, V., Frias-Martinez, E.: Analyzing the Role of Dimension Arrangement for Data Visualization in Radviz. In: Zaki, M.J., Yu, J.X., Ravindran, B., Pudi, V. (eds.) PAKDD 2010, Part II. LNCS (LNAI), vol. 6119, pp. 125–132. Springer, Heidelberg (2010)

8. Grinstein, G., Plaisant, C., Laskowski, S., O'Connell, T., Scholtz, J., Whiting, M.: VAST 2008 challenge: introducing mini-challenges. In: IEEE VAST 2008, pp. 195–196 (2008)
9. Han, J., Pei, J., Yin, Y.: Mining frequent patterns without candidate generation. In: ACM SIGMOD 2000, pp. 1–12 (2000)
10. Hassan, M. R., Ramamohanarao, K., Karmakar, C., Hossain, M.M., Bailey, J.: A Novel Scalable Multi-class ROC for Effective Visualization and Computation. In: Zaki, M.J., Yu, J.X., Ravindran, B., Pudi, V. (eds.) PAKDD 2010, Part I. LNCS (LNAI), vol. 6118, pp. 107–120. Springer, Heidelberg (2010)
11. Hofmann, H., Siebes, A.P.J.M., Wilhelm, A.F.X.: Visualizing association rules with interactive mosaic plots. In: ACM KDD 2000, pp. 227–235 (2000)
12. Keim, D.A., Kriegel, H.-P.: Visualization techniques for mining large databases: a comparison. IEEE TKDE 8(6), 923–938 (1996)
13. Keim, D.A., Schneidewind, J., Sips, M.: FP-Viz: visual frequent pattern mining. IEEE InfoVis 2005 Poster (2005)
14. Koren, Y., Harel, D.: A two-way visualization method for clustered data. In: ACM KDD 2003, pp. 589–594 (2003)
15. Leung, C.K.-S., Carmichael, C.L.: FpVAT: a visual analytic tool for supporting frequent pattern mining. ACM SIGKDD Explorations 11(2), 39–48 (2009)
16. Leung, C.K.-S., Irani, P.P., Carmichael, C.L.: FIsViz: a frequent itemset visualizer. In: Washio, T., Suzuki, E., Ting, K.M., Inokuchi, A. (eds.) PAKDD 2008. LNCS (LNAI), vol. 5012, pp. 644–652. Springer, Heidelberg (2008)
17. Leung, C.K.-S., Irani, P.P., Carmichael, C.L.: WiFIsViz: effective visualization of frequent itemsets. In: IEEE ICDM 2008, pp. 875–880 (2008)
18. Leung, C.K.-S., Jiang, F., Irani, P.P.: FpMapViz: a space-filling visualization for frequent patterns. In: IEEE ICDM Workshops 2011, pp. 804–811 (2011)
19. Munzner, T., Kong, Q., Ng, R.T., Lee, J., Klawe, J., Radulovic, D., Leung, C.K.: Visual mining of power sets with large alphabets. Technical report TR-2005-25, UBC, Canada (2005)
20. Stasko, J., Zhang, E.: Focus+context display and navigation techniques for enhancing radial, space-filling hierarchy visualizations. In: IEEE InfoVis 2000, pp. 57–65 (2000)
21. Stolte, C., Tang, D., Hanrahan, P.: Query, analysis, and visualization of hierarchically structured data using Polaris. In: ACM KDD 2002, pp. 112–122 (2002)
22. Tobiasz, M., Isenberg, P., Carpendale, S.: Lark: coordinating co-located collaboration with information visualization. IEEE TVCG 15(6), 1065–1072 (2009)
23. Wong, P.C., Cowley, W., Foote, H., Jurrus, E., Thomas, J.: Visualizing sequential patterns for text mining. In: IEEE InfoVis 2000, pp. 105–111 (2000)
24. Yang, L.: Visualizing Frequent Itemsets, Association Rules, and Sequential Patterns in Parallel Coordinates. In: Kumar, V., Gavrilova, M.L., Tan, C.J.K., L'Ecuyer, P. (eds.) ICCSA 2003, Part I. LNCS, vol. 2667, pp. 21–30. Springer, Heidelberg (2003)
25. Yang, L.: Pruning and visualizing generalized association rules in parallel coordinates. IEEE TKDE 17(1), 60–70 (2005)