

Visualization of directed associations in e-commerce transaction data

Ming C. Hao, Umeshwar Dayal, Meichun Hsu,
Thomas Sprenger*, Markus H. Gross*
Hewlett Packard Research Laboratories, Palo Alto, CA.
(ming_hao, dayal, mhsu)@hpl.hp.com

Abstract. Many real-world e-commerce applications require the mining of large volumes of transaction data to extract marketing and sales information. This paper describes the Directed Association Visualization (DAV) system that visually associates product affinities and relationships for large volumes of e-commerce transaction data. DAV maps transaction data items and their relationships to vertices, edges, and positions on a visual spherical surface. DAV encompasses several innovative techniques (1) items are positioned according to their associations to show the strength of their relationships; (2) edges with arrows are used to represent the implication directions; (3) a mass-spring engine is integrated into a visual data mining platform to provide a self-organized graph. We have applied this system successfully to market basket analysis and e-customer profiling Internet applications.

1 Introduction

Market basket analysis has become a key success factor in e-commerce. Effective market basket analysis methods employ association [1, 4] and clustering [6] as methods of analyzing such data. E-commerce transactions often are comprised of several products (items) that are purchased together. An example of an association is that 85% of the people who buy a printer also buy paper. Understanding these relationships across hundreds of product lines and among millions of transactions provides visibility and predictability into product affinity purchasing behavior.

To date, there are many technologies that allow the visualization of associations for retail stores to make business decisions such as product recommendations, cross selling, and store shelf arrangements. As illustrated in Figure 1A, a common technique [7, 8], for visualizing associations is the matrix display. The matrix technique positions pairs of items on separate axes to visualize the strength of their relationships. The association visualizer from SGI MineSet [8] lays out the rules on a 3D grid landscape. Visual filtering and querying allow users to focus in on selected rules. However, to visualize millions of association rules, we have found that association matrixes are too restrictive. The number of rules shown at the same time needs to be pre-decided and can only be a small range of rules (10-20).

An alternative to the association matrix, as illustrated in Figure 1B, is to lay out associations on a graph. For example, LikeMinds [11] uses an individual purchase history to make suggestions to shoppers based on a graph. However, when the number of items grows large, the graph can quickly become cluttered with many interactions. Also, associated items may not be placed close together. The market analysis graph from Insights' Advizor [12] has achieved dramatic improvements by utilizing dynamic

*Presently with Department of Computer Science, Swiss Federal Institute of Technology, Zurich, Switzerland
sprenger@inf.ethz.ch

queries and presentations.

Besides the use for data associations, graph visualization methods have been very popular in information visualization. For instance, cone trees [17] and their hyperbolic projections [18, 19] are prominent examples for web and file system visualization. Eric [15] used fast graph layout to display various types of statistical data. A central approach to graph visualization are physics-based paradigms being exploited in [2], [9], or [10]. Recently, clustering algorithms improved performance and scalability of physics-based methods [20, 21].

In spite of the advances in the field, it is still difficult to mine and visualize customer's purchasing behavior from millions of Internet transactions. As the volume of e-commerce data grows and as the transaction data is integrated into offline data, new data visualization associations are required to extract useful and relevant information.

In this paper, we describe a system for visualizing associations of purchasing transactions for two and more items.

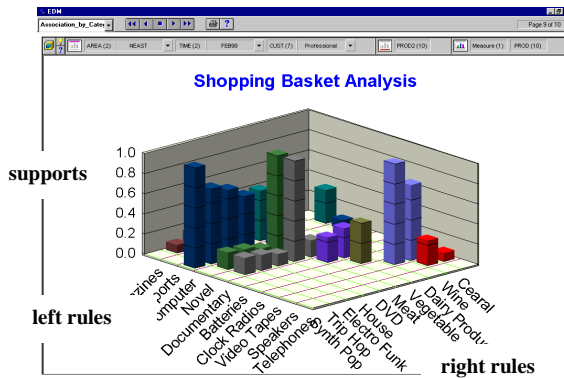


Figure 1A: A Matrix Technique

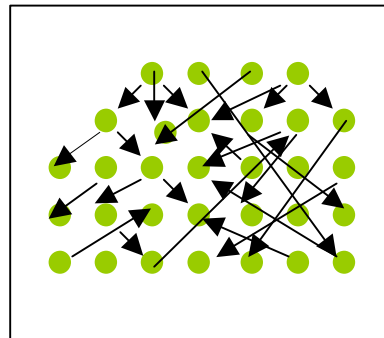


Figure 1B: A Graph – Based Technique

2 Our approach

At HP Laboratories, we have devised a “Directed Association Visualization (DAV)” 3D system. To meet the needs mentioned previously, we attempt to visualize the internal relationships and implications between large volumes of transaction data. DAV maps the transaction items and relationships to vertices and positions on a visual spherical surface. DAV uses weighted edges with arrows to represent association directions and levels. In addition, DAV employs dynamic aggregation and hierarchical link lists to enhance the scalability.

DAV integrates a well-known physics-based “Mass Spring” visualization system [2,5,9,10,14] into a visual mining platform [3]. DAV uses a sphere layout to place the most tightly related item in the center and all others around. The most tightly related item is the item with the highest correlation with other item. With various association algorithms, DAV provides a self-organized graph. It consists of the following:

1. The “distance” between each pair of items represents *support*.
2. A “directed edge” represents the direction of the association. The color of the edge is used to represent the *confidence level*.
3. A “cluster” is used to wrap around highly related items using an ellipsoidal surface.

3 Component architecture

DAV is built on a Java-based client-server model. Its architecture contains four basic components - initialization, relaxation, direction, and clustering.

Figure 2 illustrates the overall architecture of DAV. Each of the above components is described in the following sections.

3.1 Directed association definition

In order to better understand what follows, we start with a few definitions:

An association rule is of the form $X \rightarrow Y$ where X and Y are sets of items. X is called the antecedent and Y the consequence of the rule. The strength of a rule is expressed by two factors: *support and confidence*. The support of rule $X \rightarrow Y$ is the frequency of occurrence of $X \cup Y$ in all transactions, i.e. support of $X \cup Y$ is defined as the ratio of the number of transactions in which X and Y occurs to the total number of transactions. The confidence of rule $X \rightarrow Y$ is the probability that if a transaction contains the antecedent, then it also contains the consequent, i.e. the ratio of the number of transactions that contain $X \cup Y$ to the number of transactions that contain X . Thus if 85% of the customers who bought printer also bought paper, and only 10% of all the customers bought both, then the association rule has confidence 85% and support 10%. The association direction is from the printer to the paper.

3.2 Initialization

DAV arranges items extracted from the web transaction data in a spherical surface. Items are represented as vertices. The transaction data is described as the following:

$$\begin{aligned} & \text{Transactions } \{T_1, T_2, \dots, T_n\} \\ & \text{Products } \{P_1, \dots, P_m\} \\ & \text{Transaction } T_i = \{P_1, \dots, P_{mi}\} \quad i = [1..n] \end{aligned}$$

The initial positions of items on the spherical surface could be at random. To avoid random pre-clustering, DAV distributed items equally on a sphere. The computation of equally spaced positions is based on a Poisson Disc Sampling [13] for approximation. After the computation of those positions, the most tightly related item is in the center and others are evenly distributed around. The tightness of an item is the sum of all supports to its directly adjacent items.

3.3 Relaxation

DAV constructs a frequency (support) matrix F . This matrix defines the stiffness of the spring attached to a pair of items. The strength of the relationship between items is represented by the stiffness of the spring. Each element contains the frequency of

occurrence of the association in all transactions (normalized relative to the most frequent item).

$$F = \begin{bmatrix} f_{11} & & & \\ f_{1i} & f_{2i} & f_{ii} & \\ \dots & & & \\ f_{1n} & \dots & & f_{nn} \end{bmatrix}$$

$$f_{ij} = \#trans (P_i, P_j) / \max\{PT_k \mid k = [1..m]\}$$

Where $trans (P_i, P_j)$ is the set of transactions that contain P_i and P_j .

DAV transforms the spring stiffness to the distance in a 3D sphere after the graph has relaxed and converged to a state of local minimal energy.

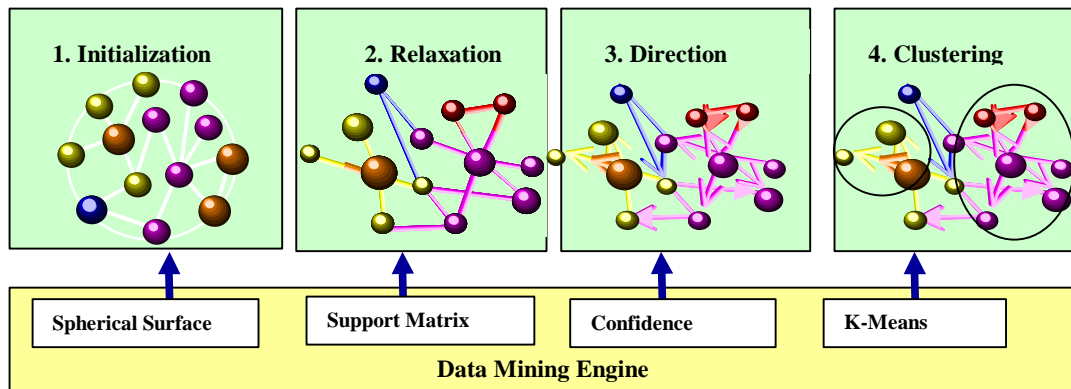


Figure 2: A Directed Association Visual Mining Component Architecture

3.4 Direction of association

DAV joins the antecedent of a rule with the consequence using a directed edge (arrow) to represent the direction of the association. The confidence levels are given in a matrix D . It is obtained by dividing the support of the item set by the support of the antecedent of the rule.

$$D = \begin{bmatrix} d_{11} & d_{12} & \dots & d_{1n} \\ \dots & & & \dots \\ d_{2i} & d_{2i} & d_{2i} & \dots \\ \dots & & & \dots \\ d_{ni} & \dots & & d_{nn} \end{bmatrix}$$

$$d(P_i, P_j) = \#trans (P_i, P_j) / \#trans (P_i)$$

d_{ij} = association confidence direction and level

$P_i \rightarrow P_j$

To identify rules with sufficient predictive power, DAV allows users to specify a minimum confidence level (threshold). The asymmetry of D between $P_i \rightarrow P_j$ and $P_j \rightarrow P_i$ is defined as the following:

- (1) Draw an arrow from P_i to P_j
If confidence level ($P_i \Rightarrow P_j$) exceeds threshold but confidence level ($P_j \Rightarrow P_i$) does not.
- (2) Draw a double arrow between P_i and P_j
If both confidence levels ($P_i \Rightarrow P_j$) and ($P_j \Rightarrow P_i$) exceed the thresholds

Figure 3 only draws the items above a minimum confidence value. The others are hidden. The user can easily follow the edges and directions to discover implications between items. For example, the user is able to find all antecedents that have “paper” as consequence. This visualization may help plan what the store should do to promote the sales of “paper”.

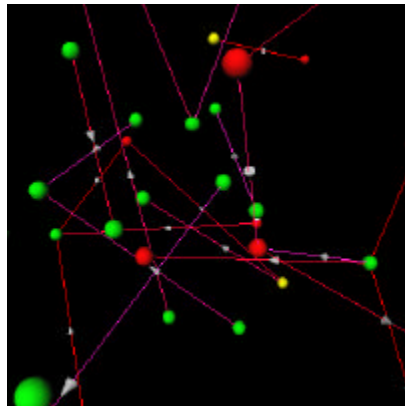


Figure 3. Directed Items with Minimum Confidence Level

3.5 Clustering

In addition to visualizing associations, DAV automatically clusters related items into groups using the “k-means” algorithm [16]. DAV wraps ellipsoids around each cluster. The number of items in a cluster and their positions control the shape of an ellipsoid.

4 Applications

DAV is a java-based client-server model. It is built on a VisMine [3] platform. VisMine uses a web browser with the Java activator that allows the user to mine large volumes of transaction data. The Web interfaces are based on standard HTML and the

use of Java applets. The client can run on a notebook. The user at the client side visually mines the knowledge results. The server is integrated with the data warehouse and mining engines.

We have applied DAV to several data mining visualization applications. In this paper, we illustrate its use in market basket analysis and customer profiling applications.

4.1 A Market basket analysis

One of the common problems electronic store managers want to solve is how to use e-customer purchase history for cross selling and up selling. They want to understand which products are purchased together and when to make real-time recommendations. Using our “directed association” system, we prototyped a market basket analysis visualization application to discover product affinities and relationships from transaction data.

An e-commerce manager can navigate a DAV-generated product sales graph and answer questions on which product groups are frequently bought together, how strong the correlation is, and in which direction. For instance, from the previous example where 85% of the people who buy a printer also buy paper, this visualization may help determine what products should be sold together with printers. Also, it helps to find what products may be impacted if the store discontinues selling printers.

Figure 4 illustrates a series of market basket analysis graphs to visualize data taken from one of the Hewlett-Packard sample shopping web site. The vertices (balls) represent products. The distances represent the support of items bought together.

Figure 4 (A) illustrates the initial layout of the graph generated from a web log. In this sample dataset, there are 182 different products (represented as balls), 250,000 transactions, and 1,383 edges. The color of the ball is used to show how often the product appears in the transaction database over a period of time. The most tightly related product is in the center and all others are evenly distributed around.

Figure 4 (B) shows the graph after it has been relaxed with 212 iterations and reached the local minima. The relaxation is based on the support/product affinities. The highly related products are self-organized into individual groups. The user can select an area to zoom in for further analysis shown in Figure 4(C). Figure 4(D) represents the graph after zoom-in. Figure 4(E) represents the graph after automatic clustering. Each highly related group is wrapped with an ellipsoidal surface for visibility. DAV allows users to interact with the graph. When the user clicks on a cluster, such as the large cluster at the upper right side in Figure 4(E), the detail information (i.e. associated product names, prices...) are displayed in a separate window as the user clicks on a selected cluster. For rapid discovery of patterns, DAV is able to monitor multiple simultaneous views of associated products.

4.2 Customer profiling

We applied this technology to analyze customer profiles. As illustrated in Figure 5, we use balls to represent customers making transactions on the web. DAV places customers with similar purchasing behaviors (i.e. product type, \$ amount, geographical

location, and income) near to each other. The store manager can rapidly discover patterns and issue coupons to the right customers for promotions.

5 Conclusion

Information visualization of e-commerce applications is an emerging technology. It needs new techniques to visualize large volumes of massive transaction data. At Hewlett-Packard Laboratories, we have integrated a mass-spring system into a visual mining platform. We have used the system to visually mine over a dataset containing 500,000 transactions covering 600 different products for market basket analysis. DAV provides a useful, fast, and interactive way for e-commerce managers to easily navigate through large-volume purchasing data to find product affinities for cross selling and up selling. Further research is continuing on scalability issues.

Acknowledgements

Thanks to Sharon Beach from HP Research Laboratories for her encouragement; to Prof. Daniel A. Keim from Halle University, to Graham Pollock from Agilent Laboratories for suggestions and comments; to Patrick Barthelemy -- "Template Graphics Software" for technical supports.

References

- [1] Rakesh Agrawal, Tomasz Imielinski, Arun Seamil, "Mining Association Rules Between Sets of Items in Large Databases", Sigmod 5/93, Washington.
- [2] T.C.Sprenger, M.H. Gross, "Ivory – An Object Oriented Framework for Physics-Based Information Visualization in Java", IEEE InfoVis98, North Carolina.
- [3] Ming Hao, Umesh Dayal, Meichun Hsu, etc. "A Java- based Visual Mining Infrastructure and Applications", IEEE InfoVis99, CA.
- [4] Pak Chung Wong, Paul Whitney, Jim Thomas, "Visualizing Association Rules for Text Mining", IEEE InfoVis99, CA.
- [5] Giuseppe Di Battista, Peter Eades, "Graph Drawing Algorithms for the Visualization of Graph", Prentice Hall, 1999.
- [6] Mihael Ankerst, Stefan Berchtold, Daniel A Keim, "Similarity Clustering of Dimensions for an Visualization of Multidimensional Data", IEEE InfoVis99, CA.
- [7] "Quest": IBM Data Mining Technologies.
- [8] "MineSet": SGI MineSet 3.0 Enterprise Edition.
- [9] T.C.Sprenger, M.H. Gross, A. Eggenberger, M.Kaufmann:"A Framework for Physically-based Information Visualization". Eight Euro Graphics-Workshop on Visualization in Scientific Computing, France, 1997.
- [10] M.H. Gross, T.C. Spenger, J.Finger: "Visualizing Information on a Sphere", IEEE VisInfo97.
- [11] LikeMinds: LikeMinds Partner Program.
- [12] Advizor: Visual Insights data visualization.
- [13] A.S.Glassner: Principles of Digital Image Synthesis, Morgan Kaufmann Publishers, San Francisco, 1995.
- [14] R.J.Hendley, N.S.Drew, A.M.Wood & R.Beale, "Case Study Narcissus: Visualizing Information", IEEE InfoVis95.

- [15] Stephen G. Eick, Joseph L. Steffen, Eric E. Sumner, Jr.: "SeeSoft-A Tool for Visualizing Line Oriented Software Statistics", IEEE Transactions on Software Engineering, 1992
- [16] J. MacQueen "Some Methods for Classification and Analysis of Multivariate Observations", The 5th Berkeley symposium on mathematical statistics and probability, Berkeley, CA. 1967.
- [17] G.G. Robertson , J.D. Mackinlay, S. K. Card, " Cone Tree: Animated 3D visualizations of hierarchical information. SIGCHI'91, 1991.
- [18] John Lamping and Ramana Rao, "Laying out and Visualizing Large Trees Using a Hyperbolic Space". ACM /UIST'94, 1994
- [19] Tamara Munster, "Exploring Large Graphs in 3D Hyperbolic Space" IEEE Computer Graphics. Vol. 18, Number 4. 1998
- [20] Matthias Kreuseler, Norma Lopez, Heidrun Schumann, "A Scalable Framework for Information Visualization" InforVis 2000, 2000, Utah.
- [21] T.C.Sprenger, R. Brunella, M.H. Gross, "H-BLOB: A Hierarchical Visual Clustering Method Using Implicit Surfaces", IEEE/VIS2000.

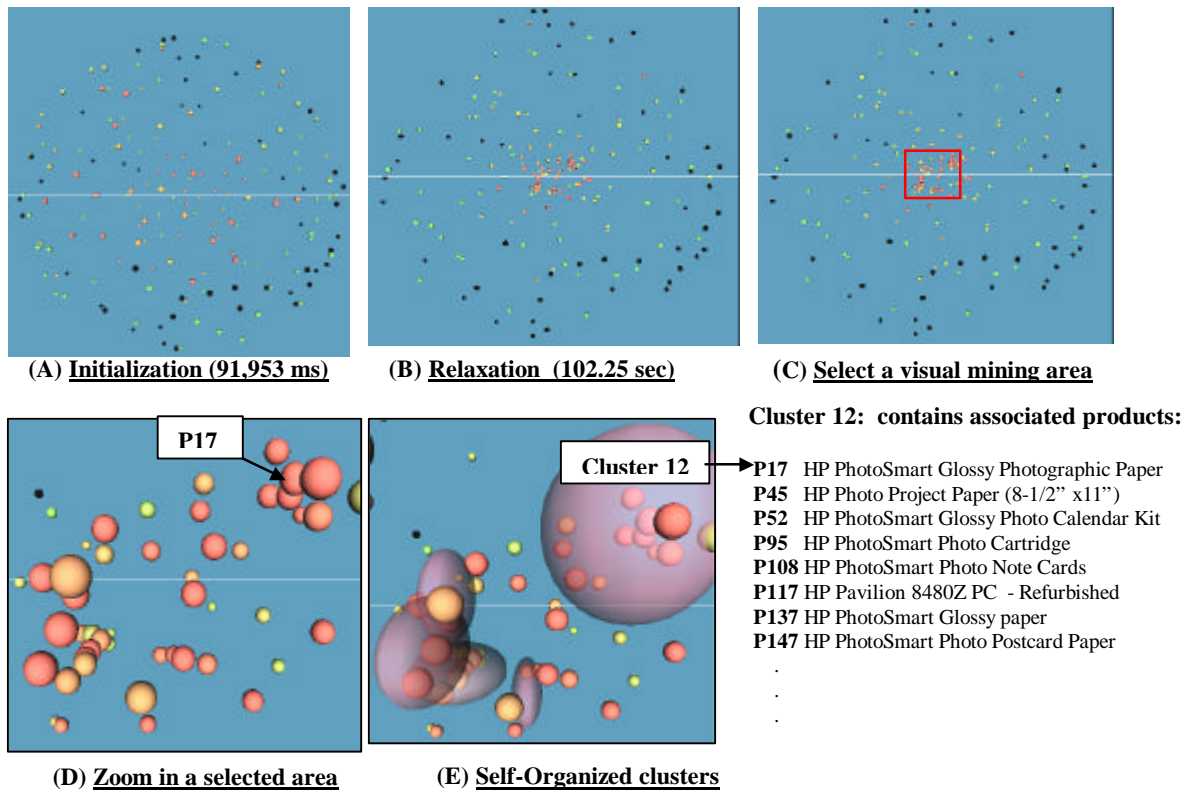


Figure 4: An Example of Market Basket Analysis (Hewlett Packard E-Store)

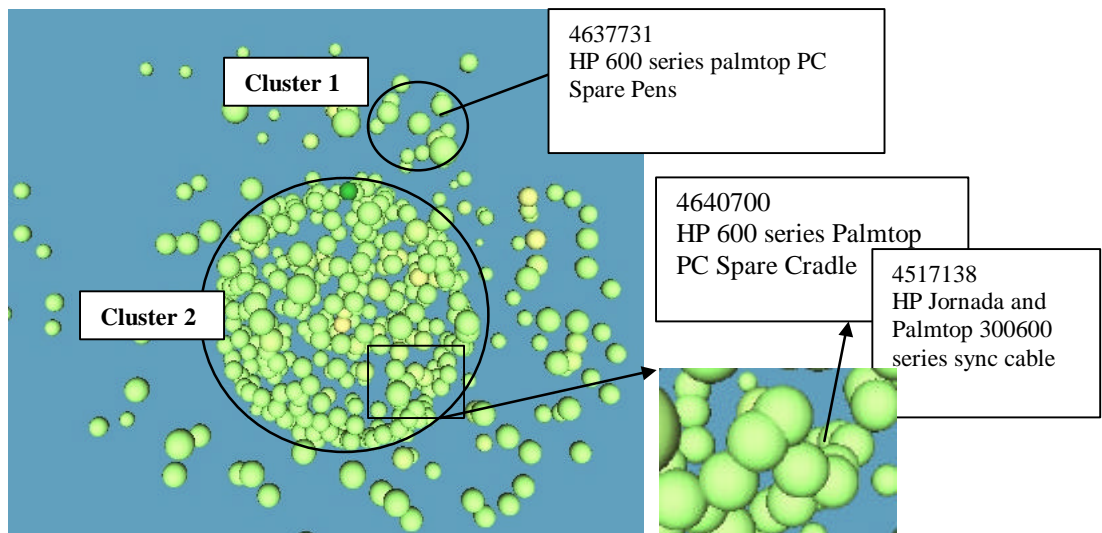


Figure 5: An Example of Customer Profiling (Purchasing Similar Products)
Grouping 478,000 customers into 12 clusters from 171,000 real-time transactions

