Appeared in ICDM'01: The 2001 IEEE International Conference on Data Mining
Available at http://www.kohavi.com

Integrating E-Commerce and Data Mining: Architecture and Challenges

Suhail Ansari, Ron Kohavi, Llew Mason, and Zijian Zheng

Blue Martini Software
2600 Campus Drive
San Mateo, CA, 94403, USA

{suhail,ronnyk,lmason,zijian}@bluemartini.com

Abstract

We show that the e-commerce domain can provide all the right ingredients for successful data mining. We describe an integrated architecture for supporting this integration. The architecture can dramatically reduce the pre-processing, cleaning, and data understanding effort often documented to take 80% of the time in knowledge discovery projects. We emphasize the need for data collection at the application server layer (not the web server) in order to support logging of data and metadata that is essential to the discovery process. We describe the data transformation bridges required from the transaction processing systems and customer event streams (e.g., clickstreams) to the data warehouse. We detail the mining workbench, which needs to provide multiple views of the data through reporting, data mining algorithms, visualization, and OLAP. We conclude with a set of challenges.

Note: A long version of this paper is also available [1].

1. Introduction

In *Measuring Web Success* [2], the authors claim that "Leaders will use metrics to fuel personalization" and that "firms need web intelligence, not log analysis."

Data mining tools aid the discovery of patterns in data[1] and E-commerce is the killer-domain for data mining. It is ideal because many of the ingredients required for successful data mining are easily satisfied: data records are plentiful, electronic collection provides reliable data, insight can easily be turned into action, and return on investment can be measured. To really take advantage of this domain, however, data mining must be integrated into the e-commerce systems with the appropriate data transformation bridges from the transaction processing system to the data warehouse and vice-versa. Such integration can dramatically reduce the data preparation time, known to take about 80% of the time to complete an analysis [3]. An integrated solution can also provide users with a uniform user interface and seamless access to metadata.

[1] In this paper, we use the term *data mining* to denote the wider process, sometimes called *knowledge discovery*, which includes multiple disciplines, such as preprocessing, reporting, exploratory analysis, visualization, and modeling.

2. Integrated Architecture

In this section we give a high level overview of a proposed architecture for an e-commerce system with integrated data mining. Details of the most important parts of the architecture and their advantages appear in following sections. The described system is an ideal architecture based on our experiences at Blue Martini Software. In our proposed architecture there are three main components, *Business Data Definition*, *Customer Interaction*, and *Analysis*. Connecting these components are three data transfer bridges, *Stage Data*, *Build Data Warehouse*, and *Deploy Results*. The relationship between the components and the data transfer bridges is illustrated in Figure 1. Next we describe each component in the architecture and then the bridges that connect these components.

In the *Business Data Definition* component, the e-commerce business user defines the data and metadata associated with their business. This data includes merchandising information (e.g., products, assortments, and price lists), content information (e.g., web page templates, email templates for campaigns, articles, images, and multimedia) and business rules (e.g., personalized content rules, promotion rules, and rules for cross-sells and up-sells). From a data mining perspective the key to the *Business Data Definition* component is the ability to define a rich set of attributes (metadata) for any type of data. For example, products can have attributes like size, color, and targeted age group, and can be arranged in a hierarchy representing categories like men's and women's, and subcategories like shoes and shirts. As another example, web page templates can have attributes indicating whether they show products, search results, or are used as part of the checkout process. Having a diverse set of available attributes is not only essential for data mining, but also for personalizing the customer experience.

The *Customer Interaction* component provides the interface between customers and the e-commerce business. Although we use the example of a web site throughout this paper, the term customer interaction applies more generally to any sort of interaction with customers. This interaction could take place through a web site (e.g., a marketing site or a web store), email marketing campaigns, cus-

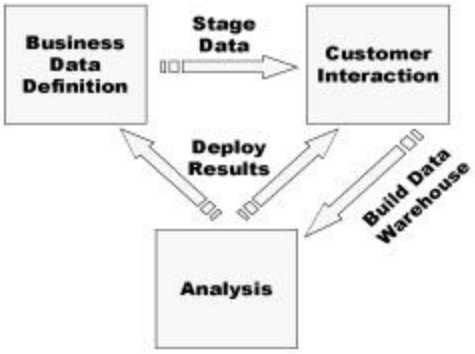

Figure 1. Proposed high-level system architecture.

tomer service (via telephony or email), wireless application, or a bricks-and-mortar point of sale system. For effective analysis of all of these data sources, a data collector needs to be an integrated part of the *Customer Interaction* component. To provide maximum utility, the data collector should not only log sale transactions, but it should also log other types of customer interactions, such as web page views for a web site, opening of emails sent out as part of a campaign, etc. Further details of the data collection architecture for the specific case of a web site are described in Section 3. To illustrate the utility of this integrated data collection let us consider the example of an e-commerce company measuring the effectiveness of its web banner advertisements on other sites geared at attracting customers to its own site. A similar analysis can be applied when measuring the effectiveness of advertising or different personalization on its own site.

The cost of a web banner advertisement is typically based on the number of "clickthroughs." That is, there is a fee paid for each visitor who clicks on the banner advertisement. Many e-commerce companies measure the *effectiveness* of their web banner advertisements using the same metric, the number of clickthroughs, and thus fail to take into account the *sales generated* by each referred visitor. If the goal is to sell more products then the site needs to attract buyers rather than browsers. A Forrester Research report [2] stated "*Using hits and page views to judge site success is like evaluating a musical performance by its volume.*" We have seen the ratio of generated sales to clickthroughs vary by as much as a factor of 20 across a company's web banner advertisements. One advertisement generated five times as much in sales as another advertisement, even though clickthroughs from the former advertisement were one quarter of the clickstreams from the latter. The ability to measure this sort of relationship requires conflation of multiple data sources.

The *Analysis* component provides an integrated environment for decision support utilizing data transformations, reporting, data mining algorithms, visualization, and OLAP tools. The richness of the available metadata gives the *Analysis* component significant advantages over horizontal decision support tools, in both power and ease-of-use. For instance, the system automatically knows the type of each attribute, including whether a discrete attribute's values are ordered, whether the range of a continuous attribute is bounded, and textual descriptions. For a web site, the system knows that each customer has web sessions and that each web session includes page views and orders. This makes it a simple matter to compute aggregate statistics for combinations of customers, sessions, page views, and orders automatically. We examine the integrated analysis component in more detail in Section 4.

The *Stage Data* bridge connects the *Business Data Definition* component to the *Customer Interaction* component. This bridge transfers (or *stages*) the data and metadata into the *Customer Interaction* component. Having a staging process has several advantages, including the ability to test changes before having them implemented in production, allowing for changes in the data formats and replication between the two components for efficiency, and enabling e-commerce businesses to have zero down-time.

The *Build Data Warehouse* bridge links the *Customer Interaction* component with the *Analysis* component. This bridge transfers the data collected within the *Customer Interaction* component to the *Analysis* component and builds a data warehouse for analysis purposes. The *Build Data Warehouse* bridge also transfers all of the business data defined within the *Business Data Definition* component (which was transferred to the *Customer Interaction* component using the *Stage Data* bridge). The data collector in the *Customer Interaction* component is usually implemented within an On-Line Transaction Processing (OLTP) system, typically designed using entity relation modeling techniques. OLTP systems are geared towards efficient handling of a large number of small updates and short queries. This is critical for running an e-commerce business, but is not appropriate for analysis [4, 5], which usually requires full scans of several very large tables and a star schema design which business users can understand. For data mining, we need to build a data warehouse using dimensional modeling techniques. Both the data warehouse design and the data transfer from the OLTP system to the data warehouse system are very complex and time-consuming tasks. Making the construction of the data warehouse an integral part of the architecture significantly reduces the complexity of these tasks. In addition to typical ETL (Extract, Transform and Load) functionality, the bridge supports import and integration of data from both external systems and syndicated data providers (e.g., Acxiom). Since the schema in the OLTP system is controlled by the architecture, we can automatically convert the OLTP schema to a multi-dimensional star schema that is optimized for analysis.

The last bridge, *Deploy Results*, is the key to closing the loop and making analytical results actionable. It provides the ability to transfer models (e.g., association rules or collaborative filtering), scores or predicted values (e.g., classification or regression results), and new attributes constructed using data transformations (e.g., lifetime value) back into the *Business Data Definition* and *Customer Interaction* components for use in business rules for personalization.

3. Data Collection

This section describes the data collection component of the proposed architecture. This component logs customers' transactions (e.g., purchases and returns) and event streams (e.g., clickstreams). While the data collection component is a part of every customer touch point (e.g., web site, customer service applications, and wireless applications), in this section we will describe in detail the data collection at the web site. Most of the concepts and techniques mentioned in this section could be easily extended to other customer touch points.

3.1. Clickstream Logging

Most e-commerce architectures rely on web server logs or packet sniffers as a source for clickstream data. While both these systems have the advantage of being non-intrusive, allowing them to "bolt on" to any e-commerce application, they fall short in logging high-level events and lack the capability to exploit metadata available in the application [6]. For each page that is requested from the web server, there are a huge number of requests for images and other content on the page. Since all of these are recorded in the web server logs, most of the data in the logs relates to requests for image files that are mostly useless for analysis and are commonly filtered out. Because of the stateless nature of HTTP, each request in a web log appears independent of other requests, so it becomes extremely difficult to identify users and user sessions from this data [7, 8, 9, 10]. Since the web logs only contain the name of the page that was requested, these page names have to be mapped to the content, products, etc., on the page. This problem is further compounded by the introduction of dynamic content where the same page can be used to display different content for each user. In this case, details of the content displayed on a web page may not even be captured in the web log. The mechanism used to send request data to the server also affects the information in the web logs. If the browser sends a request using the "POST" method, then the input parameters for this request are not recorded in the web log.

Packet sniffers try to collect similar data by looking at data "on the wire." These techniques still have problems identifying users (e.g., same visitor logging in from two different machines) and sessions. Also, given the myriad ways in which web sites are designed it is extremely difficult to extract logical business information by looking at data streaming across a wire. To further complicate things, packet sniffers can't see the data in areas of the site that are encoded for secure transmission and thus have difficulty working with sites (or areas of a site) that use SSL (Secure Socket Layer). Such areas of a site are the most crucial for analysis including checkout and forms containing personal data. In many financial sites including banks, the entire site is secure thus making packet sniffers that monitor the encrypted data blind and essentially useless, so the sniffers must be given access to data prior to encryption, which complicates their integration.

Collecting data at the application server layer can effectively solve all these problems. Since the application server serves the content (e.g., images, products and articles), it has detailed knowledge of the content being served. This is true even when the content is dynamically generated or encoded for transmission using SSL. Application servers use cookies (or URL encoding in the absence of cookies) to keep track of a user's session, so "sessionizing" the clickstream is trivial. Since the application server also keeps track of the user, using login mechanisms or cookies, associating the clickstream with a particular visitor is simple. The application server can also be designed to keep track of information absent in web server logs including pages that were aborted (user pressed the "stop" button while the page was being downloaded), local time of the user, speed of the user's connection and if the user had turned their cookies off. This method of collecting clickstream data has significant advantages over both web logs and packet sniffers.

3.2. Business Event Logging

The clickstream data collected from the application server is rich and interesting; however, significant insight can be gained by looking at subsets of requests as one logical event or episode [7, 11]. We call these aggregations of requests *business events*. Business events can also be used to describe significant user actions like sending an email or searching [2]. Since the application server has to maintain the context of a user's session and related data, the application server is the logical choice for logging these business events. Business events can be used to track things like the contents of abandoned shopping carts, which are extremely difficult to track using only clickstream data. Business events also enable marketers to look beyond page hit-rates to *micro-conversion rates* [12]. A micro-conversion rate is defined for each step of the purchasing process as the fraction of products that are successfully carried through to the next step of the purchasing process. Two examples of these are the fraction of product views that resulted in the product being added to the shopping cart and the fraction of products in the shopping

cart that successfully passed through each phase of the checkout process. Thus the integrated approach proposed in this architecture gives marketers the ability to look directly at product views, content views, and product sales, a capability far more powerful than just page views and clickthroughs.

Some interesting business events that help with the analyses given above include: adding items to, or removing items from the shopping cart, initiating checkout, finishing checkout, search, and registration. The search keywords and the number of results for each of these searches that can be logged with the search events give marketers significant insight into the interests of their visitors and the effectiveness of the search mechanism.

Another touch point that can make significant use of business events is campaign management. Campaign Management can use business events to track the sending of an email to a specific user for a specific campaign. Business Events can also be used to track the opening of these emails, submission of survey forms sent out in the email, and user click-through on the emails resulting in visits to the web site. These business events coupled with the users' clickstream data give a comprehensive picture of the user's behavior across multiple touch points. Similarly, a business event can be collected each time that a rule or model is used in personalization and these events, coupled with the shopping-cart/checkout events, can give an excellent estimate of the effectiveness of each type of personalization.

3.3. Effective Sampling Techniques

One of the challenges in collecting clickstream and business events is the volume of data generated. It is not uncommon for a site to have tens of millions of page requests a day. Add to this the data collected for specific events to track behavior and the effectiveness of personalization and the size of this data can easily grow into hundreds of millions of records. Collecting all this data may be infeasible both from a storage perspective and the impact this may have on the performance of the web site. One solution to this problem is to sample the data at the point of collection, and only collect a subset of the data. Straightforward percentage based sampling of requests however, has disastrous implications, as this results in the recording of incomplete sessions. Even sampling at the session level is not recommended since this may result in only a percentage of a user's sessions being recorded. This prevents the tracking and analysis of repeat visitors and their behavior among other things. Our architecture supports the sampling of this data at the cookie level.

4. Analysis

This section describes the analysis component of our architecture. We start with a discussion of data transformations, followed by analysis techniques including reporting, data mining algorithms, visualization, and OLAP. The data warehouse is the source data of analyses in our architecture. Although dimensional modeling is usually a prerequisite for analysis, our experience shows that many analyses require additional data transformations that convert the data into forms more amenable to data mining.

4.1. Data Transformations

As we mentioned earlier, the business user can define product, promotion, and assortment hierarchies in the *Business Data Definition* component. Figure 2 gives a simple example of a product hierarchy. This hierarchical information is very valuable for analysis, but few existing data mining algorithms can utilize it directly. Therefore, we need data transformations to convert this information to a format that can be used by data mining algorithms. One possible solution is to add a column indicating whether the item falls under a given node of the hierarchy. Let us use the product hierarchy shown in Figure 2 as an example. For each order line or page request containing a product SKU (Stock Keeping Unit), this transformation creates a Boolean column corresponding to each selected node in the hierarchy. It indicates whether this product SKU belongs to the product category represented by the node. Figure 3 shows the enriched row from this operation.

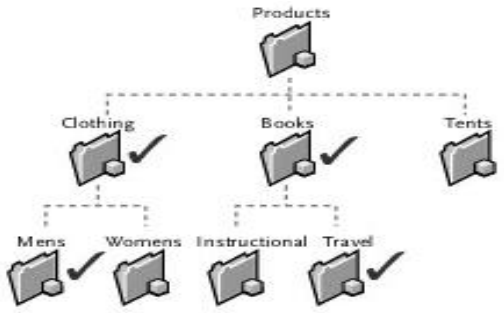

Figure 2. An example hierarchy of products.

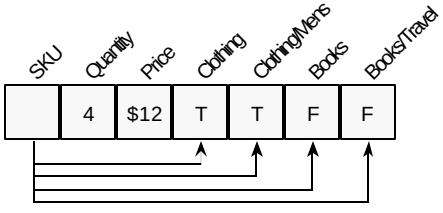

Figure 3. Data record created by the add product hierarchy transformation.

Since customers are the main concern of any e-commerce business, most data mining analyses are at the customer level. That is, each record of a data set at the final stage of an analysis is a customer signature containing all the information about the customer. However, the majority of the data in the data warehouse is at other levels such as the order header level, the order line level, and the page request level. Each customer may have multiple rows at these levels. To make this detailed information useful for analyses at the customer level, aggregation transformations are necessary [13]. Here are some examples of aggregated attributes that we have found useful:

- How much is each customer's average order amount above the mean value of the average order amount for female customers?
- What is the total amount of each customer's five most recent purchases over $30?
- What is the frequency of each customer's purchases?
- What is the recency of each customer's purchases (the number of days since the last purchase)?

Some of these attributes are very difficult to construct using standard SQL statements, and need powerful aggregation transformations. We have found RFM (Recency, Frequency, and Monetary) attributes particularly useful for the e-commerce domain.

E-commerce data contains many date and time columns. We have found that these date and time columns convey useful information that can reveal important patterns. However, the common date and time format containing the year, month, day, hour, minute, and second is not often supported by data mining algorithms. Most patterns involving date and time cannot be directly discovered from this format. To make the discovery of patterns involving dates and times easier, we need transformations which can compute the time difference between dates (e.g., order date and ship date), and create new attributes representing day-of-week, day-of-month, week, month, quarter, year, etc. from date and time attributes.

Based on the considerations mentioned above, the architecture must support a rich set of transformations.

4.2. Reporting, Algorithms, and Visualization

In this section we discuss the importance of reporting, analytical algorithms and data visualization. Basic reporting is a bare necessity for e-commerce. Through generated reports, business users can understand how a web site is working at different levels and from different points of view. Example questions that can be answered using reporting are:

- What are the best selling products?
- What are the most frequent failed searches?
- What are the conversion rates by brand?
- What are the top referrers by sales amount?
- What are the top abandoned products?

Our experience shows that some reporting questions such as the last two mentioned above are very hard to answer without an integrated architecture that records both event streams and sales data.

Beyond basic reporting, we have found simple attribute statistics based on distributions, averages, minima and maxima for continuous attributes and top distinct values (with counts and percentages) for discrete attributes to be extremely useful. Given that typical e-commerce data contains many hundreds of attributes it is important to be able to quickly identify interesting attributes based on their distribution (e.g., attributes with a single populated value or with a uniform distribution are typically uninteresting). It is also useful to be able to examine the distributions of attributes against a given target attribute. Again, it is important to be able to reduce the number of attributes that need to be examined by identifying those attributes that are correlated with the target attribute.

Model generation using data mining algorithms is a key component of the architecture. Classification, regression, clustering, sequence analysis, association rules, and collaborative filtering all reveal patterns about customers, their purchases, page views, etc. By generating models, we can answer questions like:

- What characterizes customers that prefer certain promotions to others?
- What characterizes customers that accept cross-sells and up-sells?
- What characterizes visitors that do not buy?
- What articles generate sales for particular products?
- How do the navigation paths differ for browsers and buyers?

Models can be used for business insight, generating scores and predictions (to be later used in personalization) or can be directly deployed to the *Customer Interaction* component to form the basis of a real-time personalization or recommendation engine.

Based on our experience, in addition to automated data mining techniques, it is necessary to provide interactive model modification tools to support business insight. Models either automatically generated or created by interactive modifications can then be examined or evaluated on test data. The purpose is to let business users understand their models before deploying them. For example, we have found that for rule models, measures such as confidence, lift, and support at the individual rule level and the individual conjunct level are very useful in addition to the overall accuracy of the model.

Given that humans are very good at identifying patterns from visualized data, visualization and OLAP tools can greatly help business users to gain insight into business problems by complementing reporting tools and data mining algorithms. Our experience suggests that visualization tools are very helpful in understanding generated models and web site operational data. For example, on real client

data, we have seen that plotting a bar-chart of the number of sessions by day of the week has shown that traffic is higher on Tuesday and Wednesday.

To understand why this was so, it was interesting to look at a heat-map as shown in Figure 4. The x-axis is date (one value for every day) while the y-axis is the hour of the day. Each intersection therefore represents an hour on a given day and its color (shade of gray) is assigned

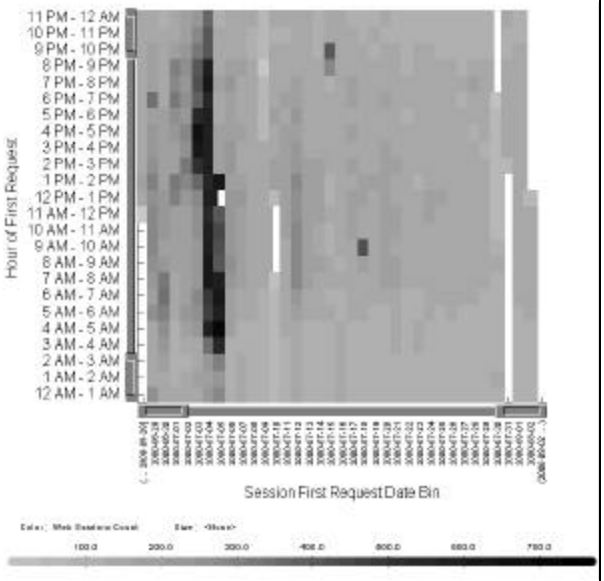

Figure 4. A heat-map showing the date, hour of day, and session count in gray.

based on the number of sessions that started at that hour. The heat-map quickly reveals that the reason for the spike on Tuesday is the July 4th holiday, when traffic increased dramatically. It is also instructive to look at the white spaces in the figure, on July 10th and July 30th-July 31st. These indicate site downtime and prior to that downtime there is relatively little traffic at the site (lighter grays).

4.3. Closed Loop Interaction Management

One of the primary advantages of this integrated architecture is closing the loop on customer interactions, analysis and personalization. This is arguably the most difficult part of the knowledge discovery process to implement in a non-integrated system. However, the shared metadata across all three components means that results can be directly reflected in the data that defines the e-commerce company's business.

The architecture supports several personalization engines that run within the *Customer Interaction* component. Rules defined by marketing users can be deployed for offering promotions to visitors, or displaying specific products or content to a certain type of visitor. These rules can be based on several different types of data, including customer attributes, customer shopping and browsing history, and the current shopping basket. Data mining models (such as those based on association rules or collaborative filtering) can be deployed from the *Analysis* component to provide real-time dynamic recommendations for products, assortments or even content. The architecture can also use control groups so that personalization is only activated for a fraction of the target visitors. This enables analysts to directly look at sales or results for visitors when the rules were and were not activated. As an example of closing the loop, customers can be scored on their propensity to accept a cross-sell and the site can be personalized based on these scores. Clickstream data, business events and other transactional data can then be analyzed to measure the effectiveness of these cross sells. These results can then be used to further improve the model and new scores can then be used in personalizing the site. Managing email campaigns is another good example of closing the loop. A recent Forrester report says that email personalization and list quality are key factors in determining the effectiveness of email campaigns [14]. As an example, consider two email campaigns sent out with two different promotional offers in them. The *Customer Interaction* component enables business users to track the success of these campaigns at multiple levels like the number of users that opened each email, clicked through on each email, and actually used the promotion to make a purchase. Using the analysis component, users can delve into the details of customers that responded to each kind of promotional offer and further tune the segmentation model (lists) for different types of promotions and campaigns. Similarly, the content used in the emails can also be targeted to a specific, more granular, customer segment. In this way the integrated architecture allows business users to personalize interactions, track user responses, measure success, get insight into the effect of the personalization, calculate ROI and further personalize the interactions, effectively closing the loop.

5. Challenges

In this section we describe several challenging problems based on our experiences in mining e-commerce data. The complexity and granularity of these problems differ, but each represents a real-life area where we believe improvements can be made. Except for the first two challenges, the problems deal with data mining algorithmic challenges.

Make Data Mining Results Comprehensible

Business users, from merchandisers who make decisions about the assortments of products to creative designers who design web sites to marketers who decide where to spend advertising dollars, need to understand the results of data mining. Few data mining models, however, are

easy to understand. Classification rules are the easiest, followed by classification trees. A visualization for the Naïve-Bayes classifier [15] was also easy for business users to understand in the second author's past experience.

The challenge is to define new types of models and ways of presenting them to business users.

Make Data Mining Process Accessible

The ability to answer a question given by a business user usually requires some data transformations and technical understanding of the tools. Our experience is that even commercial report designers and OLAP tools are too hard for most business users. Two common solutions are (i) provide templates (e.g., reporting templates, OLAP cubes, and recommended transformations for mining) for common questions, something that works well in well-defined vertical markets, and (ii) provide the expertise through consulting or a services organization. The challenge is to find ways to empower business users so that they will be able to serve themselves.

Support Multiple Granularity Levels

Data collected in a typical web site contains records at different levels of granularity. Page views are the lowest level with attributes such as product viewed and duration. Sessions include attributes such as browser used, initiation time, referring site, and cookie information. Each session includes multiple page views. Finally, customer attributes include name, address, and demographic attributes. Each customer may be involved in multiple sessions.

Mining at the page view level by joining all the session and customer attributes violates the basic assumption that records are independently and identically distributed. If we are trying to build a model to predict who visits page X, and Joe happens to visit it very often, then we might get a rule that if the visitor's first name is Joe, they will likely visit page X. The rule will have multiple records (visits) to support it, but it clearly will not generalize beyond the specific Joe. This problem is shared by mining problems in the telecommunication domain [16]. The challenge is to design algorithms that can support multiple granularity levels correctly.

Utilize Hierarchies

Products are commonly organized in hierarchies. A product hierarchy is usually three to eight levels deep. A customer purchases individual products, but generalizations are likely to be found at higher levels (e.g., families or categories). Some algorithms have been designed to support tree-structured attributes [17], but they do not scale to large product hierarchies. The challenge is to support such hierarchies within the data mining algorithms.

Scale Better: Handle Large Amounts of Data

Yahoo! has over 1.2 billion page views per day [18]. The challenge is to find useful techniques (other than sampling) that will scale to this volume of data. Are there aggregations that should be performed on the fly as data is collected?

Support and Model External Events

External events, such as marketing campaigns (e.g., promotions and media ads), and site redesign change patterns in the data. The challenge is to be able to model such events, which create new patterns that spike and decay over time.

Detect Robots and Crawlers

Robots and crawlers can dramatically change clickstream patterns at a web site. For example, Keynote (www.keynote.com) provides site performance measurements. The Keynote robot can generate a request multiple times a minute, 24 hours a day, 7 days a week, skewing the statistics about the number of sessions, page hits, and exit pages (last page at each session). Search engines conduct breadth first scans of the site, generating many requests in short duration. Identifying such robots to filter their clickstreams is a non-trivial task, especially for robots that pretend to be real users.

Support Slowly Changing Dimensions

Visitors' demographics change: people get married, their children grow, their salaries change, etc. With these changes, their needs, which are being modeled, change. Product attributes change: new choices (e.g., colors) may be available, packaging material or design change, and even quality may improve or degrade. These attributes that change over time are often referred to as "slowly changing dimensions" [4]. The challenge is to keep track of these changes and provide support for such changes in the analyses.

Handle date/time and cyclical attributes

Significant numbers of date and time attributes are collected from clickstreams and order data. Few algorithms handle these properly. Some algorithms consider them to be unique streams, rendering them useless. Others look at the date as a continuous attribute, which is useful but makes it hard to capture specific intervals since splits typically occur on a single threshold. A combination of a date and time attribute is more useful than date alone, but rarely supported. One transformation commonly done to dates is to convert them to other attributes, such as day of week. However, cyclical attributes, such as day of week and hour of the day, need to be recognized as a special type because it does not make much sense to look at day-of-week greater than Tuesday. It would be much better to capture a consecutive range of days, such as Saturday to Sunday.

Today, the alternative is typically to specifically construct such attributes as Weekend or Morning (from hour).

6. Summary

We proposed an architecture that successfully integrates data mining with an e-commerce system. The proposed architecture consists of three main components: *Business Data Definition*, *Customer Interaction*, and *Analysis*, which are connected using data transfer bridges. This integration effectively solves several major problems associated with horizontal data mining tools including the enormous effort required in pre-processing of the data before it can be used for mining, and making the results of mining actionable. The tight integration between the three components of the architecture allows for automated construction of a data warehouse within the *Analysis* component. The shared metadata across the three components further simplifies this construction, and, coupled with the rich set of mining algorithms and analysis tools (like visualization, reporting and OLAP) also increases the efficiency of the knowledge discovery process. The tight integration and shared metadata also make it easy to deploy results, effectively closing the loop. Finally we presented several challenging problems that need to be addressed for further enhancement of this architecture.

Acknowledgments

We would like to thank other members of the data mining and visualization teams at Blue Martini Software, and Cindy Hall. We wish to thank our clients for sharing their data with us and helping us refine our architecture and improve Blue Martini's products.

References

[1] Suhail Ansari, Ron Kohavi, Llew Mason, and Zijian Zheng, Integrating E-Commerce and Data Mining, *Blue Martini Software Technical Report*, 2001. Available from the articles section of http://developer.bluemartini.com.

[2] Eric Schmitt, Harley Manning, Yolanda Paul, and Joyce Tong, Measuring Web Success, *Forrester Report*, November 1999.

[3] Gregory Piatetsky-Shapiro, Ron Brachman, Tom Khabaza, Willi Kloesgen, and Evangelos Simoudis, An Overview of Issues in Developing Industrial Data Mining and Knowledge Discovery Applications, *Proceeding of the second international conference on Knowledge Discovery and Data Mining*, 1996.

[4] Ralph Kimball, *The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses*, John Wiley & Sons, 1996.

[5] Ralph Kimball, Laura Reeves, Margy Ross, Warren Thornthwaite, *The Data Warehouse Lifecycle Toolkit : Expert Methods for Designing, Developing, and Deploying Data Warehouses*, John Wiley & Sons, 1998.

[6] Ron Kohavi, Mining e-commerce data: The good, the bad, and the ugly (invited industrial track talk). In Foster Provost and Ramakrishnan Srikant, editors, *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, August 2001. http://robotics.Stanford.EDU/users/ronnyk/goodBadUglyKDDItrack.pdf

[7] Robert Cooley, Bamshad Mobashar, and Jaideep Shrivastava, Data Preparation for Mining World Wide Web Browsing Patterns, *Knowledge and Information Systems*, 1, 1999.

[8] Bettina Berendt, Bamshad Mobasher, Myra Spiliopoulou, and Jim Wiltshire, Measuring the Accuracy of Sessionizers for Web Usage Analysis, *Workshop on Web Mining at the First SIAM International Conference on Data Mining*, 2001.

[9] J. Pitkow, In search of reliable usage data on the WWW, *Sixth International World Wide Web Conference*, 1997.

[10] Shahana Sen, Balaji Padmanabhan, Alexander Tuzhilin, Norman H. White, and Roger Stein, The identification and satisfaction of consumer analysis-driven information needs of marketers on the WWW, *European Journal of Marketing*, Vol. 32 No. 7/8 1998.

[11] Osmar R. Zaiane, Man Xin, and Jiawei Han, Discovering Web Access Patterns and Trends by Applying OLAP and Data Mining Technology on Web Logs, *Proceedings of Advances in Digital Libraries Conference (ADL'98)*, Santa Barbara, CA, 1998.

[12] Stephen Gomory, Robert Hoch, Juhnyoung Lee, Mark Podlaseck, Edith Schonberg, Analysis and Visualization of Metrics for Online Merchandizing, *Proceedings of WEBKDD'99*, Springer 1999.

[13] B. Mobasher, H. Dai, T. Luo, M. Nakagawa, Y. Sun, and J. Wiltshire, Discovery of Aggregate Usage Profiles for Web Personalization, *Proceedings of KDD'2000 Workshop on Web Mining for E-Commerce - Challenges and Opportunities (WEBKDD'2000)*, p. 1-11, 2000.

[14] Jim Nail, Chris Charron, Tim Grimsditch, and Susan Shindler, The Email Marketing Dialogue, *Forrester Report*, January 2000.

[15] Barry Becker, Ron Kohavi, and Dan Sommerfield. Visualizing the Simple Bayesian Classifier, Chapter 18, pages 237-249, *Information Visualization in Data Mining and Knowledge Discovery*, Morgan Kaufmann, Publishers, San Francisco, 2001.

[16] Saharon Rosset, Uzi Murad, Einat Neumann, Yizhak Idan, and Gadi Pinkas, Discovery of Fraud Rules for Telecommunications: Challenges and Solutions, *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1999.

[17] Hussein Almuallim, Yasuhiro Akiba, and Shigeo Kaneda, On Handling Tree-Structured Attributes, *Proceedings of the Twelfth International Conference on Machine Learning*, p.12-20, 1995.

[18] *Yahoo! Inc. Second Quarter 2001 Financial Results*, July 11, 2001.