Sādhanā Vol. 30, Parts 2 & 3, April/June 2005, pp. 275–289. © Printed in India

Data mining in e-commerce: A survey

N R SRINIVASA RAGHAVAN

Department of Management Studies, Indian Institute of Science,
Bangalore 560 012, India
e-mail: raghavan@mgmt.iisc.ernet.in

Abstract. Data mining has matured as a field of basic and applied research in computer science in general and e-commerce in particular. In this paper, we survey some of the recent approaches and architectures where data mining has been applied in the fields of e-commerce and e-business. Our intent is not to survey the plethora of algorithms in data mining; instead, our current focus being e-commerce, we limit our discussion to data mining in the context of e-commerce. We also mention a few directions for further work in this domain, based on the survey.

Keywords. Data mining; e-commerce; web mining.

1. Introduction

E-commerce has changed the face of most business functions in competitive enterprises. Internet technologies have seamlessly automated interface processes between customers and retailers, retailers and distributors, distributors and factories, and factories and their myriad suppliers. In general, e-commerce and e-business (henceforth referred to as e-commerce) have enabled on-line transactions. Also, generating large-scale real-time data has never been easier. With data pertaining to various views of business transactions being readily available, it is only apposite to seek the services of data mining to make (business) sense out of these data sets.

Data mining (DM) has as its dominant goal, the generation of non-obvious yet useful information for decision makers from very large databases. The various mechanisms of this generation include abstractions, aggregations, summarizations, and characterizations of data (Carbone 2000). These forms, in turn, are the result of applying sophisticated modelling techniques from the diverse fields of statistics, artificial intelligence, database management and computer graphics.

The success of a DM exercise is driven to a very large extent by the following factors (Kohavi 2001).

- *Availability of data with rich descriptions*: This means that unless the relations captured in the database are of high degree, extracting hidden patterns and relationships among the various attributes will not make any practical sense.
- *Availability of a large volume of data*: This is mostly mandated for statistical significance of the *rules* to hold. Absence of say, atleast a hundred thousand transactions will most likely reduce the usefulness of the rules generated from the transactional database.

- *Reliability of the data available*: Although a given terabyte database may have hundreds of attributes per relation, the DM algorithms run on this dataset may be rendered defunct if the data itself was generated by manual and error prone means and wrong default values were set. Also, the lesser the integration with legacy applications, the better the accuracy of the dataset.
- *Ease of quantification of the return on investment (ROI) in DM*: Although the earlier three factors may be favourable, unless a strong business case can be easily made, investments in the next level DM efforts may not be possible. In other words, the utility of the DM exercise needs to be quantified vis-a-vis the domain of application.
- *Ease of interfacing with legacy systems*: It is commonplace to find large organizations run on several legacy systems that generate huge volumes of data. A DM exercise which is usually preceded by other exercises like extract, transformation and loading (ETL), data filtering etc, should not add more overheads to system integration.

It must now be noted that e-commerce data, being the result of on-line transactions, do satisfy all the above proper criteria for data mining. We observe that once the back-end databases are properly designed to capture customer buying behaviour, and provided that default data take care of missing and non-existent data, the first issue of availability of data with rich descriptions is taken care of. Similarly, the reliability of data collected is also ensured because it is possible to increase the so called *no-touch-throughput* in e-commerce transactions. Technologies like ebXML, BizTalk and RosettaNet enhance the quality of data that is generated. The ROI in DM exercises related to e-commerce can be easily quantified. For instance, mining the weblogs certainly enhances webserver architecture-related decisions. Improved webserver availability results in faster transactions, thus increasing the revenue. Observe that increasing the number of transactions directly results in improved profits. Lastly, e-commerce systems usually follow the MVC (Model-View-Controller) pattern with the business execution systems conforming to the model tier, the browser being the view tier and interfacing mechanisms like Java Servlets or Microsoft ASP forming the controller tier. Data mining in e-commerce mostly relies on the controller for generating the data to mine on. Thus integration issues also do not surface in this case. In summary, it is little surprise that e-commerce is the killer application for data mining (Kohavi 2001).

This paper is organized as follows. We first review some important results in data mining in § 2. We then present various applications of these techniques for conducting DM in e-commerce, in § 3. The data collection and software architecture issues consititute § 4. We then present some cases in § 5, followed by conclusions and suggestions for further work.

2. A review of data-mining methods

Given a truly massive amount of data, the challenge in data mining is to unearth hidden relationships among various attributes of data and between several snapshots of data over a period of time. These hidden patterns have enormous potential in predictions and personalizations in e-commerce. Data mining has been pursued as a research topic by at least three communities: the statisticians, the artificial intelligence researchers, and the database engineers. We now present a brief overview of some of the features of each of these approaches.

2.1 *Role of statistics in data mining*

Extracting causal information from data is often one of the principal goals of data mining and more generally of statistical inference. Statisticians have done aggregate data analyses on

data for decades; thus DM has actually existed from the time large scale statistical modelling has been made possible (Carbone 2000).

Statisticians consider the causal relationship between the dependent variables and independent variables as proposed by the user (usually the domain expert), and endeavour to capture the degree and nature of dependence between the variables. Modelling methods include simple linear regression, multiple regression, and nonlinear regression. Such models are often parameter driven and are arrived at after solving attendant optimization models. For a more detailed overview of regression methods, the reader is referred to Neter *et al* (1996) and Gujarati (2002).

The regression methods may be consider analogous to the association rules in data mining. In the latter case, rule-mining algorithms propose the correlation of itemsets in a database, across various attributes of the transactions. For instance, rules could be of the form *if a customer visits Page A.html, 90% of the times she will also visit Page B.html*. We assume here that the database (here, the web logs) has transactions recorded on a per-customer basis. Each record in the database indicates whether the customer visited a page during her entire session. Such rules can and need to be validated using the well-known statistical regression methods. Also, in some cases, the number of association rules may be very large. To draw meaningful rules that have real business value, it may be worthwhile to select the statistically most significant set of rules from the large pool of rules generated by a rule-mining algorithm.

We note that methods such as principal components analysis and factor analysis could be used to unearth hidden classes or clusters. Time series modelling on the other hand, is more relevant in sequential mining. This is used to unearth correlations and patterns in temporally ordered data. For a more detailed overview of time series methods, the reader may refer to Box *et al* (1994).

Model validations based on hypothesis testing start with an initial hypothesis of (usually) a linear relation between variable sets X and Y, and after conducting tests, the data are either shown to prove or disprove the hypothesis. Data mining involves designing a search architecture requiring evaluation of hypotheses at the stages of the search, evaluation of the search output, and appropriate use of the results. Although statistics may have little to offer in understanding search architectures, it has indeed a great deal to offer in evaluation of hypotheses in the above stages (Glymour *et al* 1996). While the statistical literature has a wealth of technical procedures and results to offer data mining, one has to take note of the following while using statistics to validate the rules generated using data mining.

- Prove that the estimation and search procedures used in data mining are consistent under conditions reasonably assumed to apply in applications
- Use and reveal uncertainty and not hide it; some data-mining approaches ignore the causal relations due to lack of sufficient data. Such caveats can be unearthed using statistical methods.
- Calibrate the errors of search to take advantages of model averaging. This is relevant where predicting the future is important, as in data mining applied to forecasting a time series. Model averaging is beneficial where several models may be relevant to build a forecast.

2.2 *The role of AI in data mining*

Artificial intelligence, on the other hand, has provided a number of useful methods for DM. Machine learning is a set of methods that enable a computer to learn relations from the given data sets. With minimal or no hypothesis from the user, learning algorithms do come up with

meaningful relations and also explain them well. Some of the most popular learning systems include the neural networks and support vector machines. We briefly present the relevant issues below.

Neural networks are predominantly used to learn linear and nonlinear relationships between variables of interest. The architecture, in general, consists of a perceptron with input and output nodes with weighted edges connecting the two nodes. A neural network with two layers is thus a bi-partite acyclic graph. The perceptron, which is the learning machine, is 'trained' in order to arrive at an optimal 'weight vector'. The output is then expressed as a (weighted) linear combination of the inputs. Learning consists of solving an underlying optimization model which is solved using gradient descent based methods.

It is worth noting here that the corresponding statistical methods available for estimating nonlinear relationships are based on the Maximum Likelihood Estimate problem. This problem is rather unwieldy since it requires the solution of highly nonlinear optimization problems; this results in tedious computations involved in solving algebraic equations. It is here that neural networks outperform their statistical counterparts, by resorting to the supervised learning methods based on gradient descent to solve such estimation problems. In other words, instead of explicitly solving equations to arrive at the maximum likelihood weights, neural networks 'learn' these weights via gradient descent based search methods.

To learn more complex relationships including multi-variate nonlinear ones, it is not uncommon to have more layers than two. Such layers are called the hidden layers. The empiricism associated with neural networks is due to the non-availability of methods that would help fix the rate of convergence and the optimal number of layers. In the above learning process, if the outputs are boolean, the problem is essentially a supervised learning mechanism to classify data sets. In such cases, often-times, a sigmoidal function (a nonlinear transformation) is applied to obtain the relevant output.

Apart from learning relationships as above, neural networks are also useful in clustering data sets. The most popular method available to cluster data sets is the K-means algorithm. Given an M-dimensional data set, the idea is to try and locate the minimal number of centroids around which the data set clusters itself. Thus the onus is to define an appropriate distance vector that helps partition the data sets into as minimally overlapping sub-sets as possible. In general, Euclidean distance metrics (including the L_2, L_1, and the L_∞ norms) are proposed for 'optimally' partitioning a given data set. The optimization is again based on minimizing the sum of squares of an appropriate error term, using classical gradient based methods.

The advantages of neural networks over the conventional statistical analysis methods are as follows (Park 2000). First, neural networks are good at modelling nonlinear relationships and interaction while conventional statistical analysis in most cases assumes linear relationship between independent variables and dependent variables. Neural networks build their own models with the help of learning process whether the relationships among variables are linear or not. Secondly, neural networks perform well with missing or incomplete data. A single missing value in regression analysis leads to removal of the entire observation or removal of the associated variable from all observations in the data set being analysed. However, neural networks update weights between input, output, and intermediate nodes, so that even incomplete data can contribute to learning and produce desired output results. Finally, neural networks do not require scale adjustment or statistical assumptions, such as normality or independent error terms. For a more detailed and comprehensive overview of neural computation and the underlying theories, the interested reader is referred to Hertz *et al* (1994) and Haykin (1998).

Artificial intelligence based methods using neural networks are used in clustering and classification methods of data mining. They can also be used in sequential mining (Park

2000). For instance, market basket analysis which concerns itself with identifying hidden customer segments could be solved using neural networks with unsupervised learning. An online webstore may want to provide different grades of service to its users depending on the frequency of customers' visits to their websites. Identifying the basket of such customer segments could be done using clustering methods. Finally, if the webstore wants to identify the factors contributing to repeat customers, they could use the nonlinear regression expressions obtained using neural networks.

2.3 *The role of database research in data mining*

Keeping in mind that data mining approaches rely heavily on the availability of high quality data sets, the database community has invented an array of relevant methods and mechanisms that need to be used prior to any DM exercise. Extract, transform and load (ETL) applications are worthy of mention in this context. Given an enterprise system like an enterprise resource planning system (ERP), it is likely that the number of transactions that happen by the minute could run into hundreds, if not thousands. Data mining can certainly not be run on the transaction databases in their native state. It requires to be extracted at periodic intervals, transformed into a form usable for analysis, and loaded on to the servers and applications that work on the transformed data. Today, software systems exist in the form of data warehousing solutions that are often bundled with the ERP system, to perform this complex and important task.

It is to be observed that data warehouses are essentially snapshots of transactional data aggregated along various dimensions (including time, geographies, demographies, products etc.) In order to run data mining algorithms, it is common practice to use the data available in the data warehouse rather than by running real time scripts to fetch transactional data. This is for the simple reason that for practical purposes, it is sufficient to include snapshots of data taken at say, weekly or monthly basis, for analysis. Real-time data is not relevant for tactical decision making, which is where data mining is used. Data warehousing is nevertheless fraught with technological challenges. The interested reader may see Marakas (2002) and Kimball *et al* (1998).

We note that database researchers have predominantly investigated association rule-mining within the field of DM. When one has terabytes of data available, the goal of database engineers in DM is to create structures and mechanisms to efficiently read in data into memory and run algorithms like A priori (Agrawal & Srikant 1994). Such algorithms assume the so-called item sets. Consider a database where the transactions pertain to a retail store. Customers buy various products and each transaction records the products bought by the customer. Observe that such databases can grow enormously in size, especially for large retailers who have web storefronts, like amazon.com. Each item set is a record in the database, with attributes mentioning if a particular product was purchased or not. The algorithms compute, given a certain support and confidence, the rules that apply on the given item sets.

A rule is of the form $X \longrightarrow Y$, where X and Y are attribute subsets of the original set of attributes. For instance, a rule could be that, of the 80% transactions where customers bought product subset X, they also bought product subset Y, and this holds good for 50% of the total transactions. Here, support is defined as the percentage of transactions where both the attributes had a value 1 (indicating that both were bought). Confidence is defined as the percentage of records with attribute value 1 for subset Y whenever the value was 1 for subset X. In the above case, the support is 50% while the confidence is 80%.

It is to be noted that A priori-like algorithms work on a given item set only. If the underlying transactional database is dynamic, then there are methods known as incremental mining proposed by the database community. Such methods resort to minimizing the number of

passes over a given database for computing the support and confidence values in rule-mining (Woon *et al* 2002). For a more detailed presentation on the above issues in DM, the interested reader is referred to Zaki (2000) and Pudi & Haritsa (2000, 2002).

Concerning e-commerce itself, it may be noted that data warehousing may or may not be required depending on the application. For instance, if one is keen on analysing the click stream patterns from users, enhancements of the web-logging schemas could be just sufficient to provide the right kind of data for mining. On the other hand, if the on-line store has an ERP system running at the back end, then ETL applications may well be required.

With respect to rule-mining itself, an application that begs attention is that of analysing the customer's pattern of website visits so that the websites could be designed and presented better. For instance, if the customer visits two or three levels deep into a website in order to conduct a transaction (like booking a ticket on line, searching for availability of trains etc.), then rule-mining could be used to unearth such patterns and identify the desirability of keeping the most frequently visited subsets of websites/homepages together. This will surely enable the customer to locate information easily while conducting a transaction at ease.

3. e-Commerce and data mining

In this section, we survey articles that are very specific to DM implementations in e-commerce. The salient applications of DM techniques are presented first. Later in this section, architecture and data collection issues are discussed.

3.1 *DM in customer profiling*

It may be observed that customers drive the revenues of any organization. Acquiring new customers, delighting and retaining existing customers, and predicting buyer behaviour will improve the availability of products and services and hence the profits. Thus the end goal of any DM exercise in e-commerce is to improve processes that contribute to delivering value to the end customer. Consider an on-line store like http:www.dell.com where the customer can configure a PC of his/her choice, place an order for the same, track its movement, as well as pay for the product and services. With the technology behind such a web site, Dell has the opportunity to make the retail experience exceptional. At the most basic level, the information available in web log files can illuminate what prospective customers are seeking from a site. Are they purposefully shopping or just browsing? Buying something they're familiar with or something they know little about? Are they shopping from home, from work, or from a hotel dial-up? The information available in log files is often used (Auguste 2001) to determine what profiling can be dynamically processed in the background and indexed into the dynamic generation of HTML, and what performance can be expected from the servers and network to support customer service and make e-business interaction productive.

Companies like Dell provide their customers access to details about all of the systems and configurations they have purchased so they can incorporate the information into their capacity planning and infrastructure integration. Back-end technology systems for the website include sophisticated DM tools that take care of knowledge representation of customer profiles and predictive modelling of scenarios of customer interactions. For example, once a customer has purchased a certain number of servers, they are likely to need additional routers, switches, load balancers, backup devices etc. Rule-mining based systems could be used to propose such alternatives to the customers.

3.2 *DM in recommendation systems*

Systems have also been developed to keep the customers automatically informed of important events of interest to them. The article by Jeng & Drissi (2000) discusses an intelligent framework called PENS that has the ability to not only notify customers of events, but also to predict events and event classes that are likely to be triggered by customers. The event notification system in PENS has the following components: Event manager, event channel manager, registries, and proxy manager. The event-prediction system is based on association rule-mining and clustering algorithms. The PENS system is used to actively help an e-commerce service provider to forecast the demand of product categories better. Data mining has also been applied in detecting how customers may respond to promotional offers made by a credit card e-commerce company (Zhang *et al* 2003). Techniques including fuzzy computing and interval computing are used to generate if-then-else rules.

Niu *et al* (2002) present a method to build customer profiles in e-commerce settings, based on product hierarchy for more effective personalization. They divide each customer profile into three parts: basic profile learned from customer demographic data; preference profile learned from behavioural data, and rule profile mainly referring to association rules. Based on customer profiles, the authors generate two kinds of recommendations, which are interest recommendation and association recommendation. They also propose a special data structure called profile tree for effective searching and matching.

3.3 *DM in web personalization*

Mobasher (2004) presents a comprehensive overview of the personalization process based on web usage mining. In this context, the author discusses a host of web usage mining activities required for this process, including the preprocessing and integration of data from multiple sources, and common pattern discovery techniques that are applied to the integrated usage data. The goal of this paper is to show how pattern discovery techniques such as clustering, association rule-mining, and sequential pattern discovery, performed on web usage data, can be leveraged effectively as an integrated part of a web personalization system. The author observes that the log data collected automatically by the Web and application servers represent the fine-grained navigational behaviour of visitors.

3.3a *Data to be captured by weblogs:* Depending on the goals of the analysis, e-commerce data need to be transformed and aggregated at different levels of abstraction. e-Commerce data are also further classified as usage data, content data, structure data, and user data. Usage data contain details of user sessions and pageviews. The content data in a site are the collection of objects and relationships that are conveyed to the user. For the most part, the data comprise combinations of textual material and images. The data sources used to deliver or generate data include static HTML/XML pages, images, video clips, sound files, dynamically generated page segments from scripts or other applications, and collections of records from the operational database(s). Site content data also include semantic or structural metadata embedded within the site or individual pages, such as descriptive keywords, document attributes, semantic tags, or HTTP variables. Structure data represent the designer's view of the content organization within the site. This organization is captured via the inter-page linkage structure among pages, as reflected through hyperlinks. Structure data also include the intra-page structure of the content represented in the arrangement of HTML or XML tags within a page. Structure data for a site are normally captured by an automatically generated *site map* which represents the hyperlink structure of the site. The operational database(s) for

the site may include additional user profile information. Such data may include demographic or other identifying information on registered users, user ratings on various objects such as pages, products, or movies, past purchase or visit histories of users, as well as other explicit or implicit representations of a users' interests.

Once the data types are clear, data preparation is easily achieved by processes such as data cleansing, pageview identification, user identification, session identification, the inference of missing references due to caching, and transaction (episode) identification (Mobasher 2004). The author then proposes association rules, sequential and navigational patterns, and clustering approaches for personalization of transactions as well as webpages. The above preprocessing tasks ultimately result in a set of n pageviews, $P = p1, p2, \ldots, pn$, and a set of m user transactions, $T = t1, t2, \ldots, tm$, where each $t_i \in T$ is a subset of P. Conceptually, one can view each transaction t as an l-length sequence of ordered pairs: $t = \langle (p_1^t, w(p_1^t)), (p_2^t, w(p_2^t)), \ldots, (p_l^t, w(p_l^t))\rangle$. where each $p_i^t = p_j$ for some $j \in 1, \ldots, n$, and $w(p_i^t)$ is the weight associated with pageview p_i^t in the transaction t. The weights can be determined in a number of ways, in part based on the type of analysis or the intended personalization framework. For example, in collaborative filtering applications, such weights may be determined based on user ratings of items. In most web usage mining tasks, the focus is generally on anonymous user navigational activity where the primary sources of data are server logs. This allows one to choose two types of weights for pageviews: weights can be binary, representing the existence or non-existence of a pageview in the transaction; or they can be a function of the duration of the pageview in the user's session.

3.3b *An illustration:* For example, consider a site with 6 pageviews A, B, C, D, E, and F. Assuming that the pageview weights associated with a user transaction are determined by the number of seconds spent on them, a typical transaction vector may look like: (11, 0, 22, 5, 127, 0). In this case, the vector indicates that the user spent 11 seconds on page A, 22 seconds on page C, 5 seconds on page D, and 127 seconds on page E. The vector also indicates that the user did not visit pages B and F during this transaction. Given this representation, the set of all m user transactions can be conceptually viewed as an $m \times n$ transaction-pageview matrix which we shall denote by TP. This transaction-pageview matrix can then be used to perform various data mining tasks. For example, similarity computations can be performed among the transaction vectors (rows) for clustering and k-NN neighborhood formation tasks, or an association rule discovery algorithm, such as Apriori, can be applied (with pageviews as items) to find frequent itemsets of pageviews (Mobasher 2004).

3.4 *DM and multimedia e-commerce*

Applications in virtual multimedia catalogs are highly interactive, as in e-malls selling multimedia content based products. It is difficult in such situations to estimate resource demands required for presentation of catalog contents. Hollfelder *et al* (2000) propose a method to predict presentation resource demands in interactive multimedia catalogs. The prediction is based on the results of mining the virtual mall action log file that contains information about previous user interests and browsing and buying behaviour.

3.5 *DM and buyer behaviour in e-commerce*

For a successful e-commerce site, reducing user-perceived latency is the second most important quality after good site-navigation quality. The most successful approach towards reducing user-perceived latency has been the extraction of path traversal patterns from past users

access history to predict future user traversal behaviour and to prefetch the required resources. However, this approach is suited for only non-e-commerce sites where there is no purchase behaviour. Vallamkondu & Gruenwald (2003) describe an approach to predict user behaviour in e-commerce sites. The core of their approach involves extracting knowledge from integrated data of purchase and path traversal patterns of past users (obtainable from webserver logs) to predict the purchase and traversal behaviour of future users.

Web sites are often used to establish a company's image, to promote and sell goods and to provide customer support. The success of a web site affects and reflects directly the success of the company in the electronic market. Spiliopoulou & Pohle (2000) propose a methodology to improve the *success* of web sites, based on the exploitation of navigation-pattern discovery. In particular, the authors present a theory, in which success is modelled on the basis of the navigation behaviour of the site's users. They then exploit web usage miner (WUM), a navigation pattern discovery miner, to study how the success of a site is reflected in the users' behaviour. With WUM the authors measure the success of a site's components and obtain concrete indications of how the site should be improved.

In the context of web mining, clustering could be used to cluster similar click-streams to determine learning behaviours in the case of e-learning, or general site access behaviours in e-commerce. Most of the algorithms presented in the literature to deal with clustering web sessions treat sessions as sets of visited pages within a time period and do not consider the sequence of the click-stream visitation. This has a significant consequence when comparing similarities between web sessions. Wang & Zaiane (2002) propose an algorithm based on sequence alignment to measure similarities between web sessions where sessions are chronologically ordered sequences of page accesses.

4. Data collection and software architecture

4.1 *Enabling data collection in e-commerce*

It may be observed that there are various ways of procuring data relevant to e-commerce DM. Web server log files, web server plug-ins (instrumentation), TCP/IP packet sniffing, application server instrumentation are the primary means of collecting data. Other sources include transactions that the user performs, marketing programs (banner advertisements, e-mails etc), demographic (obtainable from site registrations and subscriptions), call centres and ERP systems.

It is quite common to expend about 80% of any DM effort in e-commerce in data filtering. This is largely in part to the heavy reliance on the web logs that are generated by the HTTP protocol. This protocol being stateless, it becomes very difficult to cull out customer buying behaviour-related information along with the product details. Ansari *et al* (2001) describe an architecture for supporting the integration of DM and e-commerce. The architecture is found to dramatically reduce the preprocessing, cleaning, and data understanding effort. They emphasize the need for data collection at the application server layer and not the web server, in order to support tagging of data and metadata that is essential to the discovery process. They also describe the data transformation bridges required from the transaction processing systems and customer event streams (e.g. click streams) to the data warehouse.

4.2 *Analysing web transactions*

Once the data are collected via any of the above mentioned mechanisms, data analysis could follow suit. This could be done along session level attributes, customer attributes, product

attributes and abstract attributes. Session level analysis could highlight the number of page views per session, unique pages per session, time spent per session, average time per page, fast vs. slow connection etc. Additionally, this could throw light on whether users went through registration, if so, when, did the users look at the privacy statement; did they use search facilities, etc. The user level analysis could reveal whether the user is an initial or repeat or recent visitor/purchaser; whether the users are readers, browsers, heavy spenders, original referrers etc. (Kohavi 2001).

The view of web transactions as sequences of pageviews allows one to employ a number of useful and well-studied models which can be used to discover or analyse user navigation patterns. One such approach (Sarukkai 2000) is to model the navigational activity in the website as a Markov chain. In the context of web transactions, Markov chains can be used to model transition probabilities between pageviews. In web-usage analysis, they have been proposed as the underlying modelling machinery for web prefetching applications or to minimize system latencies.

Hu & Cercone (2002) present a new approach called on-line analytical mining for web data. Their approach consists of data capture, *webhouse* construction, pattern discovery and pattern evaluation. The authors describe the challenges in each of these phases and present their approach for web usage mining. Their approach is useful in determining the most profitable customers, the difference between buyers and non-buyers, identification of website parts that attract most visits, parts of website that are session killers, parts of the site that lead to the most purchases, identifying the typical path of customers that leads to a purchase or otherwise etc. The webhouse is akin to the data warehouse.

4.3 *An architecture for DM*

In a B2B e-commerce setting, it is very likely that vendors, customers and application service providers (ASP) (usually the middlemen) have varying DM requirements. Vendors would be interested in DM tailored for market basket analysis to know customer segments. On the other hand, end customers are keen to know updates on seasonal offerings and discounts all the while. The role of the ASP is then to be the common meeting ground for vendors and customers. Krishnaswamy *et al* (2000) propose a distributed DM architecture that enables a DM to be conducted in such a naturally distributed environment. The proposed distributed data mining system is intended for the ASP to provide generic data mining services to its subscribers. In order to support the robust functioning of the system it possesses certain characteristics such as heterogeneity, costing infrastructure availability, presence of a generic optimisation engine, security and extensibility. Heterogeneity implies that the system can mine data from heterogeneous and distributed locations. The proposed system is designed to support user requirements with respect to different distributed computing paradigms (including the client-server and mobile agent based models). The costing infrastructure refers to the system having a framework for estimating the costs of different tasks. This implies that a task that requires higher computational resources and/or faster response time should cost the users more on a relative scale of costs. Further, the system should be able to optimise the distributed data mining process to provide the users with the best response time possible (given the constraints of the mining environment and the expenses the user is willing to incur). The authors have indeed designed and implemented such a framework.

Maintaining security implies that in some instances, the user might be mining highly sensitive data that should not leave the owner's site. In such cases, the authors provide the option to use the mobile-agent model where the mining algorithm and the relevant parameters are shipped to the data site and at the end of the process the mobile agent is destroyed on

the site itself. The system is extensible to provide for a wide range of mining algorithms (Krishnaswamy *et al* 2000). The authors provide a facility wherein the user can register their algorithms with the ASP for use in their specific distributed DM jobs.

5. Cases in e-commerce data mining

In this section, we first present an interesting application of DM in e-commerce. We then present some important lessons learnt by some authors while implementing DM in e-commerce.

5.1 *Distributed spatial data mining*

In various e-commerce domains involving spatial data (real estate, environmental planning, precision agriculture), participating businesses may increase their economic returns using knowledge extracted from spatial databases. However, in practice, spatial data is often inherently distributed at multiple sites. Due to security, competition and a lack of appropriate knowledge discovery algorithms, spatial information from such physically dispersed sites is often not properly exploited. Lazarevic *et al* (1999) develop a distributed spatial knowledge discovery system for precision agriculture. In the proposed system, a centralized server collects proprietary site-specific spatial data from subscribed businesses as well as relevant data from public and commercial sources and integrates knowledge in order to provide valuable management information to subscribed customers. Spatial data mining software (Koperski *et al* 1996) interfaces this database to extract interesting and novel knowledge from data. Specific objectives include a better understanding of spatial data, discovering relationships between spatial and nonspatial data, construction of spatial knowledge-bases, query optimization and data reorganization in spatial databases. Knowledge extracted from spatial data can consist of characteristic and discriminant rules, prominent structures or clusters, spatial associations and other forms.

Challenges involved in spatial data mining include multiple layers of data, missing attributes and high noise due to a low sensibility of instruments and to spatial interpolation on sparsely collected attributes. To address some of these problems, data are cleaned by removing duplicates, removing outliers and by filtering through a median filter with a specified window size (Lazarevic *et al* 1999).

The goal of precision agriculture management is to estimate and perform site-specific crop treatment in order to maximize profit and minimize environmental damage. Through a knowledge discovery (KDD) process, Lazarevic *et al* (1999) propose learning algorithms that perform data modelling using data sets from different fields in possibly different regions and years. Each dataset may contain attributes whose values are not manageable (e.g. topographic data), as well as those attributes that are manageable (e.g. nutrient concentrations).

In order to improve prediction ability when dealing with heterogeneous spatial data, an approach employed in the proposed system by Lazarevic *et al* (1999) is based on identifying spatial regions having similar characteristics using a clustering algorithm. A clustering algorithm is used for partitioning multivariate data into meaningful subgroups (clusters), so that patterns within a cluster are more similar to each other than are patterns belonging to different clusters. Local regression models are built on each of these spatial regions describing the relationship between the spatial data characteristics and the target attribute.

5.2 *DM applied to retail e-commerce*

Kohavi *et al* (2004) have attempted a practical implementation of data mining in retail e-commerce data. They share their experience in terms of lessons that they learnt. They classify the important issues in practical studies, into two categories: business-related and technology-related. We now summarize their findings on the technical issues here.

(1) Collecting data at the right level of abstraction is very important. Web server logs were originally meant for debugging the server software. Hence they convey very little useful information on customer-related transactions. Approaches including sessionising the web logs may yield better results. A preferred alternative would be have the application server itself log the user related activities. This is certainly going to be richer in semantics compared to the state-less web logs, and is easier to maintain compared to state-ful web logs.
(2) Designing user interface forms needs to consider the DM issues in mind. For instance, disabling default values on various important attributes like Gender, Marital status, Employment status, etc., will result in richer data collected for demographical analysis. The users should be made to enter these values, since it was found by Kohavi *et al* (2004) that several users left the default values untouched.
(3) Certain important implementation parameters in retail e-commerce sites like the automatic time outs of user sessions due to perceived inactivity at the user end, need to be based not purely on DM algorithms, but on the relative importance of the users to the organization. It should not turn out that large clients are made to lose their shopping carts due to the time outs that were fixed based on a DM of the application logs.
(4) Generating logs for several million transactions is a costly exercise. It may be wise to generate appropriate logs by conducting random sampling, as is done in statistical quality control. But such a sampling may not capture rare events, and in some cases like in advertisement referral based compensations, the data capture may be mandatory. Techniques thus need to be in place that can do this sampling in an *intelligent* fashion.
(5) Auditing of data procured for mining, from data warehouses, is mandatory. This is due to the fact that the data warehouse might have collated data from several disparate systems with a high chance of data being duplicated or lost during the ETL operations.
(6) Mining data at the right level of granularity is essential. Otherwise, the results from the DM exercise may not be correct.

6. Conclusions and future work

In this paper, we have presented how web mining (in a broad sense, DM applied to e-commerce) is applicable to improving the services provided by e-commerce based enterprises. Specifically, we first discussed some popular tools and techniques used in data mining. Statistics, AI and database methods were surveyed and their relevance to DM in general was discussed. We then presented a host of applications of these tools to DM in e-commerce. Later, we also highlighted architectural and implementation issues.

We now present some ways in which web mining can be extended for further research. With the growing interest in the notion of semantic web, an increasing number of sites use structured semantics and domain ontologies as part of the site design, creation, and content delivery. The notion of *Semantic Web Mining* was introduced by Berendt *et al* (2002). The primary challenge for the next-generation of personalization systems is to effectively integrate

semantic knowledge from domain ontologies into the various parts of the process, including the data preparation, pattern discovery, and recommendation phases. Such a process must involve some or all of the following tasks and activities (Mobasher 2004).

(1) *Ontology learning, extraction, and preprocessing*: Given a page in the web site, we must be able extract domain-level structured objects as semantic entities contained within this page.
(2) *Semantic data mining*: In the pattern discovery phase, data mining algorithms must be able to deal with complex semantic objects.
(3) *Domain-level aggregation and representation*: Given a set of structured objects representing a discovered pattern, we must then be able to create an aggregated representation as a set of pseudo objects, each characterizing objects of different types occurring commonly across the user sessions.
(4) *Ontology-based recommendations*: Finally, the recommendation process must also incorporate semantic knowledge from the domain ontologies.

Some of the challenges in e-commerce DM include the following (Kohavi 2001).

- *Crawler/bot/spider/robot identification*: Bots and crawlers can dramatically change clickstream patterns at a web site. For example, some websites like (www.keynote.com) provide site performance measurements. The Keynote bot 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. Tools need to have mechanisms to automatically sieve such noisy data in order for DM algorithms to yield sensible and pragmatic proposals.
- *Data transformations*: There are two sets of transformations that need to take place: (i) data must be brought in from the operational system to build a data warehouse, and (ii) data may need to undergo transformations to answer a specific business question, a process that involves operations such as defining new columns, binning data, and aggregating it. While the first set of transformations needs to be modified infrequently (only when the site changes), the second set of transformations provides a significant challenge faced by many data mining tools today.
- *Scalability of data mining algorithms*: With a large amount of data, two scalability issues arise: (i) most data mining algorithms cannot process the amount of data gathered at web sites in reasonable time, especially because they scale nonlinearly; and (ii) generated models are too complicated for humans to comprehend.

The above challenges need to be better addressed in real world tools.

Episode mining involves mining not one-time events, but mining for a historical pattern of events. Episode-mining methods rely on extensions of rule-mining methods. Alternate approaches could be explored here. Support vector machines (Haykin 1998) have taken the centre stage of late, in learning linear and nonlinear relationships. Their applications in episode mining could be an exciting area for further work.

The author would like to thank Prof. Narahari for encouraging him to write this article. He also thanks Mr. Venkataramana for his help in bibTeX.

References

Agrawal R, Srikant R 1994 Fast algorithms for mining association rules. In *20th Int. Conf. on Very Large Databases* (New York: Morgan Kaufmann) p 487–499

Ansari S, Kohavi R, Mason L, Zheng Z 2001 Integrating e-commerce and data mining: architecture and challenges. In *Proc. 2001 IEEE Int. Conf. on Data Mining* (New York: IEEE Comput. Soc.) pp 27–34

Auguste D M 2001 Customer service in e-business. *IEEE Internet Comput.* 5(5): 90–91

Berendt B, Hotho A, Stumme G 2002 Towards semantic web mining. In *Proc. First Int. Semantic Web Conference,* Sardinia, Italy

Box G, Jenkins G, Reinsel G 1994 *Time series analysis: Forecasting and control* 3rd edn (Englewood Cliffs, NJ: Prentice Hall)

Carbone P L 2000 Expanding the meaning of and applications for data mining. In *IEEE Int. Conf. on Systems, Man, and Cybernetics* (New York: IEEE) pp 1872–1873

Glymour C, Madigan D, Pregibon D, Smyth P 1996 Statistical inference and data mining. *Commun. ACM* 39(11):

Gujarati D 2002 *Basic econometrics* (New York: McGraw-Hill/Irwin)

Haykin S 1998 *Neural networks: A comprehensive foundation* 2nd edn (Englewood Cliffs, NJ: Prentice-Hall)

Hertz J, Krogh A, Palmer R G 1994 *Introduction to the theory of neural computation* (Reading, MA: Addison-Wesley)

Hollfelder S, Oria V, Ozsu M T 2000 Mining user behavior for resource prediction in interactive electronic malls. In *IEEE Int. Conf. on Multimedia and Expo* (New York: IEEE Comput. Soc.) pp 863–866

Hu X, Cercone N 2002 An olam framework for web usage mining and business intelligence reporting. In *Proc. IEEE Int. Conf. on Fuzzy Systems, FUZZ-IEEE'02* (New York: IEEE Comput. Soc.) pp 950–955

Jeng J J, Drissi Y 2000 Pens: a predictive event notification system for e-commerce environment. In *The 24th Annu. Int. Computer Software and Applications Conference, COMPSAC 2000*, pp 93–98

Kimball R, Reeves L, Ross M, Thornthwaite W 1998 *The data warehouse lifecycle toolkit: Expert methods for designing, developing, and deploying data warehouses* (New York: Wiley)

Kohavi R 2001 Mining e-commerce data: The good, the bad, and the ugly. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2001)* (New York: ACM Press) pp 8–13

Kohavi R, Mason L, Parekh R, Zheng Z 2004 Lessons and challenges from mining retail e-commerce data. *Machine Learning J.* (Special Issue on Data Mining Lessons Learned)

Koperski K, Adhikary J, Han J 1996 Spatial data mining: Progress and challenges. *J. Data Mining Knowledge Discovery*

Krishnaswamy S, Zaslavsky A, Loke S W 2000 An architecture to support distributed data mining services in e-commerce environments. In *Second Int. Workshop on Advanced Issues of E-Commerce and Web-Based Information Systems, WECWIS 2000*, pp 239–246

Lazarevic A, Xu X, Fiez T, Obradovic Z 1999 Clustering-regression-ordering steps for knowledge discovery in spatial databases. In *Proc. of IEEE/INNS Int. Conf. on Neural Neural Networks*, Washington

Marakas G M 2002 *Modern data warehousing, mining, and visualization: core concepts* (Englewood Cliffs, NJ: Prentice-Hall)

Mobasher B 2004 Web usage mining and personalization. In *Practical handbook of internet computing* (ed.) M P Singh (CRC Press)

Neter J, Kutner M, Nachtsheim C J, Wasserman W 1996 *Applied linear statistical models* (New York: McGraw-Hill/Irwin)

Niu L, Yan X W, Zhang C Q, Zhang S C 2002 Product hierarchy-based customer profiles for electronic commerce recommendation. In *Int. Conf. on Machine Learning and Cybernetics* pp 1075–1080

Park S 2000 Neural networks and customer grouping in e-commerce: a framework using fuzzy art. In *Academia/Industry Working Conference on Research Challenges*, pp 331–336

Pudi V, Haritsa J 2000 Quantifying the utility of the past in mining large databases. *Inf. Syst.* 25: 323–344

Pudi V, Haritsa J 2002 On the efficiency of association rule-mining algorithms. In *Advances in knowledge discovery and data mining* Lecture Notes in Artificial Intelligence (LNAI) 2336, (eds) M Chen, P Yu, B Liu (Berlin: Springer) pp 80–91

Sarukkai R R 2000 Link prediction and path analysis using markov chains. In *Proceedings of the 9th International World Wide Web Conference*, Amsterdam

Spiliopoulou M, Pohle C 2000 Data mining to measure and improve the success of web sites. *J. Data Mining and Knowledge Discovery*

Vallamkondu S, Gruenwald L 2003 Integrating purchase patterns and traversal patterns to predict http requests in e-commerce sites. In *IEEE Int. Conf. on e-commerce*, pp 256–263

Wang W, Zaiane O R 2002 Clustering web sessions by sequence alignment. In *13th Int. Workshop on Database and Expert Systems Applications* pp 394–398

Woon Y K, Ng W K, Lim E P 2002 Online and incremental mining of separately-grouped web access logs. In *Proc. Third Int. Conf. on Web Information Systems Engineering, WISE 2002* (New York: IEEE Comput. Soc.) p 53–62

Zaki M J 2000 Scalable algorithms for association mining. *IEEE Trans. Knowledge Data Eng.* 19: 372–390

Zhang Y Q, Shteynberg M, Prasad S K, Sunderraman R 2003 Granular fuzzy web intelligence techniques for profitable data mining. In *12th IEEE Int. Conf. on Fuzzy Systems, FUZZ '03* (New York: IEEE Comput. Soc.) pp 1462–1464