Mining E-Commerce Data: The Good, the Bad, and the Ugly

Ron Kohavi
Senior Director, Data Mining
Blue Martini Software
2600 Campus Dr.
San Mateo, CA 94403
ronnyk@CS.Stanford.EDU

ABSTRACT

Organizations conducting Electronic Commerce (e-commerce) can greatly benefit from the insight that data mining of transactional and clickstream data provides. Such insight helps not only to improve the electronic channel (e.g., a web site), but it is also a learning vehicle for the bigger organization conducting business at brick-and-mortar stores. The e-commerce site serves as an early alert system for emerging patterns and a laboratory for experimentation. For successful data mining, several ingredients are needed and e-commerce provides all the right ones (the Good). Web server logs, which are commonly used as the source of data for mining e-commerce data, were designed to debug web servers, and the data they provide is insufficient, requiring the use of heuristics to reconstruct events. Moreover, many events are never logged in web server logs, limiting the source of data for mining (the Bad). Many of the problems of dealing with web server log data can be resolved by properly architecting the e-commerce sites to generate data needed for mining. Even with a good architecture, however, there are challenging problems that remain hard to solve (the Ugly). Lessons and metrics based on mining real e-commerce data are presented.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Applications – *Data Mining*. I.2.6 [**Artificial Intelligence**]: Learning.

General Terms

Measurement, Design, Experimentation, Human Factors.

Keywords

E-commerce, data mining, application server, web server, web site architecture.

1. INTRODUCTION

Organizations conducting e-commerce can greatly benefit from the insight that data mining of transactional and clickstream data provides. Such insight can help improve the site design, personalization strategies, customer loyalty, and profitability. Good user experiences will improve the average purchase size, the number of repeat customers, and the value of the brand name. Bad user experiences can hurt a well-known brand much more than the immediate revenue loss associated with the dissatisfied customers.

A web site serves multiple purposes besides supporting online transactions. A web site gives customers a place to get information about products and services. For example, IBM estimated a savings of $2 billion in costs in the year 2000 by offering support information to customers on the Web [1]. The web site also provides an early alert system for emerging patterns. Customers searching for specific products and failing to find them provide an early warning to merchandisers, who should consider adding them to the site's offering. Product cross-sells and up-sells can be developed based on viewing and buying patterns. New product introductions can be tested in a variety of ways: different ads can be quickly tested, target segments can be identified, and appropriate messages that increase conversion rates can be developed. The web is an amazing experimental laboratory.

The paper is organized as follows. Section 2 reviews the ingredients needed for successful data mining and illustrates that e-commerce provides all of them, making it a killer domain (the Good). Section 3 describes the problems with the obvious approach to mining e-commerce data, namely using web server logs (the Bad). Section 4 describes an alternative to the use of web server logs: logging clickstreams at the application sever layer. Section 5 looks at challenging open problems (the Ugly). Section 6 discusses lessons learned and metrics based on mining real e-commerce data. Section 7 provides a summary.

This paper concentrates on business-to-consumer web sites as the principal electronic channel, but many of the issues naturally extend to other electronic channels, such as wireless devices and business-to-business transactions. The term data mining is used here in the broad sense, sometimes referred to as Knowledge Discovery.

2. THE GOOD: E-COMMERCE IS THE KILLER DOMAIN FOR DATA MINING

For data mining to succeed, several desiderata should be satisfied, yet they are seldom present in real-world applications. In e-commerce, some are satisfied and some can be satisfied with proper design [2]. The desiderata are as follows.

1. Large amount of data. Having many records match each pattern ensures the statistical significance of patterns found and reduces the likelihood of overfitting.

   Clickstream data containing page view information can be collected extremely fast even for small sites. For example, a web site that sells an average of five items an hour will have over 1.4 million page views a month, assuming 2% conversion rate and eight pages per session. Yahoo! serves over 1 billion page views a day [3], implying that the log files alone require about 10GB an hour. With so much data, valid sampling strategies, such as sampling by users (based on cookies) or, less preferably, by sessions, become important for exploratory analysis.

2. Many attributes (wide records). Entities that are mined should have many attributes.

   If the records in the data consist of only a few attributes, then simple techniques, such as bar graphs, scatter plots, and spreadsheet tables suffice for understanding the data. When there are several dozens or hundreds of attributes, automated techniques are needed to sift through the data and identify the important factors. With proper design of a site, extensive data can be collected, and a large number of attributes can be made available. For example, in the KDD Cup 2000 [4], over five hundred attributes were available per record.

3. Clean data. Noisy and corrupt data can hide patterns and make predictions harder. Manual data entry and integration with legacy systems can introduce inconsistencies and anomalies. Direct electronic collection at the source provides superior quality and highly reliable data.

   With the proper architecture, events at a web site, at a call center, or at a kiosk can be automatically logged. Clean data is, of course, a relative term. Human-entered information, such as registration forms on the web, can still contain errors but on-line validation can help against many unintentional errors that are harder to catch in paper forms. Contrast this with a clerk entering information about a person from a paper application three days after the applicant left. In one case, a bank found that 5% of their customers were born on the same date (day, month, year). It turned out that the data entry system mandated filling in the birth date, so for the paper forms that did not contain the date, it was easiest to advance to the next field by typing 111111, which became Nov 11, 1911.

4. Actionable domain. Interesting insight may be fascinating and results are often discussed, but action is rarely taken in practice because legacy systems and inflexible domains make it extremely hard and expensive to apply the new knowledge and improve processes. For example, Berry and Linoff [5] describe fantastic data mining case studies from real-life experiences, yet there is little discussion about the actions taken after the knowledge was discovered. The discoveries made in the data warehouse rarely affected the operational systems.

   In e-commerce, many discoveries can be made actionable by changing web sites, including the site layout, design, cross-sells, up-sells, and personalization. Targeted e-mail campaigns are relatively easy to execute and automate. The operational system and the analysis system can be tied into one closed-loop system more easily than in other domains.

5. Measurable Return On Investment (ROI). Evaluating changes and tracking their effect in brick-and-mortar stores is hard, expensive, and takes a long time. Paco Underhill, in his book "Why We Buy: The Science of Shopping" [6], describes how human trackers log people's behavior at stores, and the arduous effort his team makes analyzing over 14,000 hours of store videotapes every year.

   The web changes everything! Clickstreams and events can be electronically collected and the effects of changes and discoveries can be quickly translated into incremental dollars at the web site. The web allows for controlled experiments (e.g., use of control groups), multiple experiments, and an immediate evaluation of their effect on the objective (e.g., improved conversion rates). E-mail campaigns can be tracked through every stage of the conversion process: opening the e-mail, clicking through, browsing products, adding products to the shopping cart, initiating checkout, and purchasing. Reports can be generated daily or even in real time. That said, the electronic world is new and the "science" of shopping and interacting on the web is still in the alchemy days.

While technically e-commerce is a great domain for data mining, and, conversely, data mining is extremely important for e-commerce sites, business processes and social issues are still developing. Companies have to deal with channel conflicts, consistent messages across multiple touchpoints with customers, and the cost of building and maintaining electronic sites.

We conclude this section with a very important observation:

The Web is an Experimental Laboratory.

An electronic site provides a company with unparalleled access to a "laboratory" for conducting experiments, studying customer behavior, and learning about trends quickly.

It is easy to change images, test cross-sells, test messages, and quickly measure how customers react. Are customers interested in reading more content (e.g., magazine sections of a site)? Are customers reacting better to promotion A or to promotion B? Does product P sell well with Q or R? Why is everyone searching for Pokémon when you have not yet heard about it?

While not all results from the lab carry over to the physical world, many discoveries do generalize to other channels. The value of a web site is therefore not just the immediate revenue but also its value as a research center and an experimental lab!

3. THE BAD: WEB SERVER LOGS

Web servers can generate logs, which detail the interactions with clients, typically web browsers. Web servers generate logs in a Common Log Format (CLF) [8] or Extended Common Log Format (ECLF), which include the following fields: remote (client) host, remote logname (client identity information), username for authentication, date and time of request, request, HTTP status code, number of bytes transferred, the referring server URL, and the user agent (name and version of client). Most web servers support options to log additional fields, such as cookie and performance-related data.

Attempts to apply data mining to the web and e-commerce commonly start with web server logs as the primary or sole source of data [9][10]. The choice seems natural because many sites use web servers that support generating such logs, but it also significantly limits what data is available and creates major hurdles when additional information needs to be collected. Web server logs were designed to debug web servers, not for data mining. We now describe the problems with web server logs.

1. Web server logs do not identify sessions or users.

   HTTP is stateless, so the notion of a "session," which is crucial for mining, does not exist at the level of the web server. Weaving together the requests that form a session is currently an active research topic [10][11][20]. Common techniques rely on cookies, time, IPs, and browser user agents. Problems occur because of proxy caches, IP reassignment, browsers rejecting cookies, etc. Berendt et al. [11] recently wrote that "These [sessionization] tools are based on heuristic rules and on assumptions about the site's usage, and are therefore prone to error."

2. Web server logs need to be conflated with transactional data.

   E-commerce sites store orders in a transactional database, yet combining the order data and other transactional data with web server logs is a complex extract-transform-load (ETL) process and requires identifying the clicks relevant to each transaction. For example, one of the most common reports for e-commerce sites is the sales revenue attributable to referring sites. While the web server logs can contain the referrer URL from the HTTP field "Referer" (misspelled in the standard [13]), computing this information requires both clickstream information and order information. Other often-requested metrics are product conversion rates and checkout *micro-conversion* rates [21], which also require similar conflation. Kimball and Merz describe a *Clickstream Post-Processor* architecture for loading web server log data into a data warehouse [9]. Sane Solutions built the NetTracker product for loading clickstreams into a data warehouse, and they describe the issues in a nice whitepaper [17].

3. Web server logs lack critical events.

   Events such as "add to cart," "delete item," or "change quantity" are not available in web logs. Inferring these can be extremely hard because the log file may show that the user visited a page with five items and that the next page view was the cart. Did the user add an item or did they click on "show cart"? One of the most important metrics for e-commerce is the value of abandoned shopping carts, yet these events are not computable from web logs.

4. Web server logs do not store web *form* information.

   When a user fills out a form, such as search form, it is important to know what information was entered in order to improve the site. A recent e-metrics study conducted at Blue Martini Software shows that 11% of searches failed [12]. Knowing the keywords used can help companies add synonyms and improve their product mix.

5. Web server logs contain URLs, not the semantic information of what the URLs contain.

   URLs need to be mapped to the semantic information describing what they contain. What product is presented when a given URL is shown? Which pages are part of the checkout process or registration? Which are books versus electronics? The same page may have multiple versions in different languages, yet they are equivalent for most analysis purposes. Reverse mapping needs to be done in these cases to create a common entity name for data mining.

6. Web server logs lack information for modern sites that generate dynamic content.

   Dynamic sites are built on a small set of URL templates that are reused to present different information, making it harder to extract information from web server logs. Which product was presented to the user if all products are presented using the template product.jsp? Was something dynamic presented, such as a promotion? What about an out of stock message? Did search succeed or fail? How many results were returned after a successful search, and did the number of results overwhelm the user?

7. Web server logs are flat files on multiple file systems, possibly in different time zones.

   Large sites will have multiple web servers, each logging data into separate files, usually on different file systems. The situation is more complex if the web servers are geographically distributed in different time zones, since the combined data must be in one time zone (usually GMT). Web server logs are typically in ASCII, which is an inefficient way to store large amounts of structured data.

8. Web server logs contain redundant information

   Most entries in the web log are of no interest for mining. For example, they contain the request for every image. Because a typical web page contains multiple images and because the same page view usually contains the same images (with minor exceptions for personalization), 90% of the web server logs is commonly pruned.

9. Web server logs lack important information that can be collected using other means.

   The HTTP header [13], which is the source of information for the web logs, does not contain important information such as the user local time or their screen size.

Eric Schmitt et al. [7] wrote that "Using hits and page views to judge site success is like evaluating a musical performance by its volume." In the early days of a site, counting HTTP requests is a good metric to ensure sufficient performance, but as sites mature and perform higher-level analyses, more information is needed than what web server logs can provide.

4. THE ALTERNATIVE TO WEB SERVER LOGS: APPLICATION SERVER LOGGING

If web server logs have so many deficiencies, what other options are there? There are two approaches commonly used today: one is based on packet sniffing and the other is based on logging clickstreams at the application server layer [15].

Packet sniffing is a technique whereby data sent out from the web server to the client is monitored [16]. It provides a non-intrusive way to augment web server logs with additional information. The approach is certainly useful as more information is logged, but many of the disadvantages listed in Section 3 are not mitigated. Moreover, sniffing does not work if traffic is encrypted, making the additional information unavailable at the most crucial times: registration and checkout.

Application server logging does away with web server logs; instead, all the logging is done at the application layer [18][19]. We now review the problems with web server logs described in Section 3 and address how they are handled with application server logging.

1. Web server logs do not identify sessions or users.

   The application server controls sessions, user registration, login, and logout. These can be directly logged and no sessionization heuristics are needed.

2. Web server logs need to be conflated with transactional data.

   The application layer writes the order data, and if it also logs clickstream events, it is possible to generate a single comprehensive log (e.g., in a database) with consistent IDs between tables.

3. Web server logs lack critical events.

   The application layer must know about events such as "add to cart" and can log these. In addition, it can log specific interesting events, such as a browser reset (user hitting refresh or clicking on a link prior to the page download being completed).

   In addition to simple one-page events, high-level "business events" can be logged once a complete event or scenario, such as search or sending an e-mail, is completed [7][18].

4. Web server logs do not store web *form* information.

   Clearly this is something that can be done at the application server layer that parses the form. An interesting idea is to log the reposting of forms because of user errors. Such logging of field-level errors can help improve the forms.

5. Web server logs contain URLs, not the semantic information of what the URLs contain.

   At the application layer of a dynamic site, significant semantic information is available about the content of the page being displayed.

6. Web server logs lack information for modern sites that generate dynamic content.

   Clearly the URL itself becomes less important when logging information at the application server layer.

7. Web server logs are flat files on multiple file systems, possibly at different time zones.

   The application server logs can be generated directly into the database, so that transaction level integrity holds. Times can be stored in GMT, possibly with another field for the user browser's local timezone offset. Synchronization of application servers should still be done [9].

8. Web server logs contain redundant information.

   Redundancy is trivially eliminated when the application server controls logging.

9. Web server logs lack important information that can be collected using other means.

   Any information that can be collected can also be logged into the same database with the appropriate keys.

Application server logging clearly resolves all the problems previously mentioned. Blue Martini Software's Commerce application includes the described log capabilities, showing that it can work well in practice. An early version of data generated by the application server for a real site was used in the KDD Cup 2000 [4]

For existing sites, however, re-architecting the whole system for application server logging may be too expensive. A third approach, which is semi-intrusive, involves inserting calls (e.g., JavaScript) inside the web page to another system, such as WebTrendsLive [22]. Such calls can mitigate some of the issues discussed. For example, when a sale happens, information can be sent notifying another system of the sale. The main advantage of this approach is that it does not require significant re-architecture of the system while it can capture events. The main disadvantages are as follows.

1. Additional work has to take place to instrument the site with JavaScript, in effect duplicating the application server code that handles similar events. All the semantic information needs to be rewritten; as the web site evolves, keeping everything in sync is error-prone and expensive.
2. JavaScript raises browser compatibility issues and can be turned off by users.
3. JavaScript raises privacy concerns more than server-side logging.
4. The data is collected at a third system, i.e., not in weblogs or in the application database. Building a unified data warehouse with conflated sources is harder.
5. Many events at the application layer are still impossible to track (e.g., forms).

To summarize, web server logs have many deficiencies, which can be overcome if the site is architected properly and logging is done at the application server layer [18]. Existing sites can instrument the web pages or use packet sniffers, but will have to merge multiple data streams and the data will still lack important information.

5. THE UGLY: OPEN ISSUES

While the main problems described in the previous section (the Bad) have possible architectural solutions, the problems described below are not as pretty and are considered open research issues.

1. Crawler/bot/spider/robot identification.

   Bots and crawlers can dramatically change clickstream patterns at a web site. For example, Keynote (www.keynote.com) provides 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. Internet Explorer 5.0 supports automatic synchronization of web pages when a user logs in, when the computer is idle, or on a specified schedule; it also supports offline browsing, which loads pages to a specified depth from a given page. Finally, shopping bots, such as mySimon (www.mysimon.com), regularly scan commerce sites for product prices. These options create additional clickstreams and patterns. Identifying such bots to filter their clickstreams is a non-trivial task, especially for bots that pretend to be real users.

2. 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. As many have observed, about 80% of the time to complete an analysis is spent in data transformations [23]. By automating the initial transfer and building a warehouse, the 80% can hopefully be reduced, but business users still find it too hard to answer questions with today's data mining tools without help from technical data miners.

3. Taking action and operationalizing the findings.

   When users discover nuggets, taking action is not always easy. The insight is usually in terms of transformed data that may not be available at the operational side and may be too expensive to compute. For example, if the insight depends on the number of times a user visited a set of web pages in the past, it is too expensive to compute this value at a live site, so it may be approximated or updated through a batch process.

4. Scalability of data mining algorithms.

   With so much 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 non-linearly; and (ii) generated models are too complicated for humans to comprehend.

6. LESSONS AND STATISTICS

This section is a biased set of lessons learned from the authors' experiences mining customer data at Blue Martini Software. The statistics and metrics are all based on sites built on the Blue Martini architecture (mostly business to consumer) and may not generalize to other types of sites. Your mileage may vary.

1. Spend the time to identify crawlers and bots. Many analyses have been made with significant crawler traffic skewing results. We have observed that up to 70% of traffic can come from crawlers and bots, with an average of 25%-30%. The statistics below are after crawler removal.
2. Buyers and Browsers have very different browsing patterns. The average visitor spends 5 minutes on the site and views 8-10 pages, while a buyer spends an average of 30 minutes at the site, viewing 50 pages. These numbers are consistent across multiple sites.
3. Half the sessions are shorter than a minute with about a third of sessions never going past the home page. Spend a lot of time making the home page great!
4. Only a small percentage of visitors use search (6%), but those that search buy more. 11% of searches failed. Make sure to track the failed searches and improve the search and the products mix available at the site.
5. About a third (15%-50%) of the shopping carts are abandoned (high variations across sites).
6. Never provide defaults in forms if you want unbiased answers. We found that people often accept the defaults even when they are wrong. It is better to make the answer optional and set to "Please choose."
7. Synchronize clocks on all machines. On unsynchronized systems we have seen users who ordered before their first visit to the site!
8. "Nobody" reads the privacy policy (less than 0.5%)

7. SUMMARY

E-commerce sites can generate great data for mining, containing all the right ingredients (the Good). However, the naïve approach of using web logs is insufficient for many business questions and additional data must be collected and conflated (the Bad, if you did not design the site properly). There are several challenges that make data mining hard (the Ugly), and we hope to see them addressed by the community. Finally, several lessons and statistics were presented based on our experience with mining e-commerce data at Blue Martini Software.

8. ACKNOWLEDGMENTS

I would like to thank the data mining team at Blue Martini Software for their ongoing discussions about these topics. I would like to thank Eric Bauer, Jon Becher, Rob Cooley, Rajesh Parekh, and Zijian Zheng for their comments on this paper.

9. REFERENCES

[1] Peter Burrows. The Era of Efficiency. Business Week, pages 92-99, June 18 2001.

[2] Ron Kohavi and Foster Provost. Applications of data mining to electronic commerce. Data Mining and Knowledge Discovery, 5(1/2), 2001. http://robotics.Stanford.EDU/~ronnyk/ecommerce-dm

[3] Yahoo! Reports First Quarter 2001 Financial Results, Press Release April 11, 2001. http://biz.yahoo.com/bw/010411/0403.html

[4] Ron Kohavi, Carla Brodley, Brian Frasca, Llew Mason, and Zijian Zheng. KDD-Cup 2000 organizers' report: Peeling the onion. SIGKDD Explorations, 2(2):86-98, 2000. http://www.ecn.purdue.edu/KDDCUP

[5] Michael J. A. Berry and Gordon S. Linoff. Mastering Data Mining. John Wiley & Sons, Inc, 2000.

[6] Paco Underhill. Why We Buy: The Science of Shopping. Touchstone Books, Rockefeller Center, 1230 Avenue of the Americas, New York, NY 10020, 2000.

[7] Eric Schmitt, Harley Manning, Yolanda Paul, and Joyce Tong. Measuring web success. Forrester Report, November 1999.

[8] The Common Logfile Format. http://www.w3.org/Daemon/User/Config/Logging.html#common-logfile-format

[9] Ralph Kimball and Richard Merz. The Data Webhouse Toolkit: Building the Web-Enabled Data Warehouse. John Wiley & Sons, 2000.

[10] Robert Cooley, Bamshad Mobasher, and Jaideep Srivastava. Data preparation for mining world wide web browsing patterns. Knowledge and Information Systems, 1(1), 1999.

[11] Bettina Berendt, Bamshad Mobasher, Myra Spiliopoulou, and Jim Wiltshire. Measuring the accuracy of sessionizers for web usage analysis. In Workshop on Web Mining at the First SIAM International Conference on Data Mining, pages 7-14, April 2001.

[12] EMetrics Study, Blue Martini Software, 2001. http://developer.bluemartini.com/developer/articles/index.jsp

[13] J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee. Hypertext transfer protocol - http/1.1. RFC 2616. http://www.w3.org/Protocols/rfc2616/rfc2616.html

[14] Robert Cooley. Drinking from the firehose: Converting raw web traffic and e-commerce data streams for data mining and marketing analysis. In Web Data Mining Conference, San Francisco, CA, 2000. http://www.webusagemining.com/sys-tmpl/webdataminingworkshop

[15] Jon Becher and Ron Kohavi. E-commerce and clickstream mining tutorial. First SIAM International Conference on Data Mining, 2001. http://robotics.Stanford.EDU/~ronnyk/miningTutorialSlides.pdf

[16] Accrue Software Inc. Web Mining whitepaper: Driving business decisions in web time, March 2000. http://www.accrue.com/forms/webmining.html

[17] Sane Solutions. Analyzing web site traffic, 2000. http://www.sane.com/products/NetTracker/whitepaper.pdf

[18] Suhail Ansari, Ron Kohavi, Llew Mason, and Zijian Zheng. Integrating e-commerce and data mining: Architecture and challenges. In WEBKDD'2000 workshop: Web Mining for E-Commerce---Challenges and Opportunities. http://robotics.Stanford.EDU/~ronnyk/WEBKDD2000/index.html

[19] Robert W Cooley. Web Usage Mining: Discovery and Application of Usage Patterns from Web Data. Doctorate, University of Minnesota, 2000.

[20] Lara D. Catledge and James E. Pitkow. Characterizing browsing strategies in the World-Wide Web. Computer Networks and ISDN Systems, 27(6): 1065-1073, 1995.

[21] Stephen Gomory, Robert Hoch, Juhnyoung Lee, Mark Podlaseck, and Edith Schonberg. Analysis and visualization of metrics for on-line merchandizing. In WEBKDD'99 workshop on Web Usage Analysis and User Profiling, 1999. http://www.wiwi.hu-berlin.de/ myra/WEBKDD99

[22] WebTrends Live. http://www.webtrendslive.com/

[23] 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. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, pages 89-95. AAAI Press, 1996.