Interactive Path Analysis of Web Site Traffic

Pavel Berkhin
Accrue Software, Inc.
48634 Milmont Drive
Fremont, CA 94538

Jonathan D. Becher
Accrue Software, Inc.
48634 Milmont Drive
Fremont, CA 94538

Dee Jay Randall
Accrue Software, Inc.
48634 Milmont Drive
Fremont, CA 94538

ABSTRACT

The goal of ***Path Analysis*** is to understand visitors' navigation of a Web site. The fundamental analysis component is a path. A path is a finite sequence of elements, typically representing URLs or groups of URLs. A full path is an abstraction of a visit or a session, which can contain attributes described below. Subpaths represent interesting subsequences of the full paths.

Path Analysis provides user-configurable extraction, filtering, preprocessing, noise reduction, descriptive statistics and detailed analysis of three basic specific objects: *elements, (sub)paths*, and *couples* of elements. In each case, lists of frequent objects —subject to particular filtering and sorting — are available. We call the corresponding interactive tools *Element, Path,* and *Couple Analyzers*.

We also allow in-depth exploration of individual elements, paths, and couples: *Element Explorer* investigates composition and convergence of traffic through an element and allows conditioning based on the number of preceding/succeeding steps. *Path Explorer* visualizes in and out flows of a path and attrition rate along the path. *Couple Explorer* presents distinct paths connecting couple elements, along with measures of their association and some additional statistics.

1. INTRODUCTION

Path Analysis provides a comprehensive analysis of visitors' navigation of a Web site. We use the term analysis instead of the word reporting to stress the interactive and detailed nature of the techniques we present. Identification of Web site traffic patterns is important for designing a more efficient and user-friendly site, discovering potentially misleading, duplicate, or overlapping content, and understanding the effectiveness of referring links. Most importantly, it enables a broad variety of business applications.

A ***full path*** is a finite sequence of elements **(a b . . . c)** typically representing URLs or groups of URLs. The concept of a full path is an abstraction of a visit or a session. A full path can more generally contain additional business, timing, or visitor-related attributes. Any contiguous subsequence of a full path is a ***subpath***, or simply a ***path***. We also use the term ***couple*** to describe an ordered pair <**a,b**> of elements that occur in some path. As we will demonstrate, there is a certain duality between the concepts of a path and a couple. *Elements*, *paths* and *couples* are the three fundamental objects of path analysis.

Our path analysis implementation is built on top of an enterprise class e-business analysis platform called Accrue G2 [**1**]. Accrue G2 provides a rich infrastructure for Web traffic analysis. It collects visitor activity through a proprietary network collector (packet sniffing of http traffic), Web server plug-ins, or Web server log files. Accrue G2 shields our implementation from many technical problems related to acquisition, transformation, and preprocessing of the Web site data. For example, our implementation does not need to be concerned with sessionizing (see [**4**]) or spider/robot identification.

We operate in two main phases: In the ***pre-processing phase***, we extract data from the G2 repository, perform sampling, filtering, and noise reduction, compute general descriptive statistics, and construct compressed data structures for supporting later interactive analysis. A critical characteristic of this phase is scalability, as the raw volume of data operated on can be very large.

The efficient data structures constructed during the pre-processing phase enable the ***interactive phase***. The interactive phase provides the user with a broad variety of tools to investigate elements, paths, and couples. These tools include detailed analysis and exploration. The analysis tools provide filtered and sorted lists of these objects. The exploration tools provide valuable insights into the structure of a specific object and its relationship to other objects. Altogether, there are six major interactive tools:

Element Analyzer	***Element Explorer***
Path Analyzer	***Path Explorer***
Couple Analyzer	***Couple Explorer***

All the tools provide a number of additional measures, including frequency, coverage, and associations. The interactive phase is easy to use and provides rich analysis. A critical characteristic of the interactive phase is that it is fast. Software implementation

of interactive phase uses XML for the client-server communication. We plan to make this API public for developers of third party tools.

2. PRE-PROCESSING

The pre-processing phase consists of two major steps: data extraction and data preparation.

The ***Data extraction step*** selects Path Analysis data from Accrue G2 or another appropriate repository and passes it through the following logical stages:

Data Repository —>E—>S—>M—>P—>F—> Input Path Set.

E is the physical Extraction of paths (potentially with pre-computed frequencies). Some configuration parameters are available.

S is Sampling, a standard data mining technique to reduce the size of the data. It can also be used to research stability and confidence intervals of computed characteristics.

M is Mapping or encoding of elements (URLs or groups of URLs) into integers. One special feature during the mapping is called ***factorization***. Factorization allows the user to examine paths at different levels of granularity; consequently the user ignores irrelevant details by zooming out to a broader view. The simplest factorization implementation is based on the *content hierarchy*.

P is optional Preprocessing of path data to incorporate useful transformations that are most relevant for a particular application. Examples include *elimination of consecutive repetitions* (**(a,a,b,c,c,c) —> (a,b,c)**), or *trimming* of elements located far from pages corresponding to business events.

F is Filtering in which conditions are specified to select a particular subset of all the input full paths. The conditions include: date and time, starting and ending elements or groups of elements, referrers or groups of referrers, visitor groups, and particular business events attributable to a whole path (e.g., registration, purchase, session length, etc).

The ***Data preparation step*** results in data compression and noise reduction. Two types of noise reduction controls are available: frequency and coverage. First, the user can specify a minimum frequency, so that paths with lower frequencies are discarded. Second, the user can specify a coverage limit. Low frequency paths are discarded (reducing the coverage) until the coverage limit is reached. The second mechanism alleviates a potential imbalance in the first approach due to the fact that shorter paths have a tendency to be more frequent. Table 1 presents an actual example of 1164619 paths collected over one day. The columns contain the compression coefficient and the coverage for this data set without any noise reduction (MF=1), and with noise reduction corresponding to minimal frequency MF=2, 3:

Min Frequency (MF)	1	2	3
Compression coefficient	4.2	66.9	117.4
Coverage ratio	100.0%	75.2%	72.7%

Table 1

The compressed path data is packaged into special data structures to enable fast querying during the interactive phase. These structures are mainly fixed length path sets sorted by certain suffixes, prefixes, and other task specific orderings. Task specific orderings allow logarithmically quick searches that relate to the query of a particular tool. An example of such a query is a request for all in-flows to a particular focus path (see section 4 below). Care is taken to maintain and sort references to the underlying list elements rather than the list elements themselves. Exploiting references allows significant savings in memory usage. Due to a lack of space, we plan to describe related data structures in a separate publication.

At this point a broad variety of descriptive statistics are computed, which provide a snapshot of the Web site traffic. These statistics include:

- Number of extracted, retained, discarded, and unique elements and full paths, along with corresponding compression coefficients and coverage ratios;

- Length related statistics for full paths (example: conditional average length of paths longer than a given length), and the distributions of extracted and retained full paths by length;

- Attrition, compression and coverage by length for full retained paths and distribution by length of fixed length (sub)paths;

- Statistics for paths of special nature, such as constant paths (e.g., **(a a … a)**), circular-2 paths (e.g., **(a b a b …)**), and cyclic paths (e.g., **(a … a)**).

3. ANALYSIS AND EXPLORATION OF ELEMENTS

Element Analyzer presents elements sorted by their overall frequency, their session entry or exit frequency, or their occurrence as paths of length one. In particular, this information provides the simplest means to find confident changes or trends in traffic data from one run to another.

Element Explorer deals with issues of composition and convergence of traffic to and from a particular element. The simplest question to ask is what are the immediate (1-step) predecessors and successors of an element **a**. The question can be generalized. An element **x** is a **k**-step predecessor of **a** if there is a path **(x z_1 ... z_{k-1} a)**. The analogous definition holds for a successor **y**. If the number of such paths (with different **z**) is **N(k,x,a)** and frequency of **a** is **N(a)**, then the *conditional probability,* **N(k,x,a)/N(a),** shows the fraction of **a**-traffic that arrives from **x** in **k** steps. An array of these probabilities for different **x** describes a ***composition*** of **a**-th predecessors. Altogether, three questions can be asked. To demonstrate the symmetry between predecessors and successors, we formulate all three questions for successors:

- What are the chances that **y** is a **k**-step successor of **a**?
- What are the chances of reaching **y** from **a** in **k** steps for the first time?
- What are the chances of reaching **y** from **a** in no more then **k** steps?

The two last questions deal with the issue of ***convergence***. Element Explorer allows the user to specify element **a**, the

number of steps **k**, the conditioning element (**a** or **x/y**) and the particular question of interest.

4. ANALYSIS AND EXPLORATION OF PATHS

Path Analyzer presents the most frequent paths of any or of fixed length, optionally filtered by starting, ending, or intermediate groups of elements, or other special criteria (e.g., non constant paths only). In a case study of three industrial sites, Table 2 shows fixed 4-long subpaths. The table cells contain total coverage (%) of all 4-subpaths by the **N** most frequent 4-subpaths.

	Education Site	Finance Portal	Computer Vendor
Total number of 4-long paths	4,081,707	8,336,165	2,526,607
N=32	15.20%	78.50%	48.90%
N=64	20.60%	81.10%	52.90%
N=128	26.40%	83.60%	57.20%
N=256	33.30%	85.20%	61.40%

Table 2

The most profound business inferences can be made from the fact that a few high frequency objects of interest cover such a significant portion of the overall data. This allows us to employ compression to dramatically decrease queries for additional information. There is an interesting generalization of this approach to overlay traffic data with certain business measures, such as the amount of sales. By assigning a particular weight to each extracted path, proportional to its business measure, we can focus the analysis on a particular business goal.

Path Explorer drills into the data related to a particular focus path. Figure 1 schematically presents path exploration:

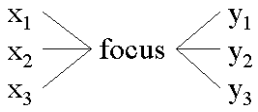

Figure 1

If the focus path is **(a b c)**, we would like to know all in-flows **x** and out-flows **y**, along with their corresponding frequencies. It is also important to be able to expand a focus path in either direction through an **x** or **y**, or to shrink the focus path by removing an element from either end. To provide a case study for this paper without divulging proprietary customer data, we present simplified output (Figure 2) of the encoded data instead of the actual graphical user interface.

```
  Freq In-Flow   Focus    Out-Flow Freq
-----------------------------------------
+ [950]    5                   1 [672] +
+ [168]    3                1884 [538] +
+ [133]  118                  49 [197] +
+ [106]   28                   3 [ 96] +
+ [ 84] 1038                6423 [ 86] +
+ [ 60]    4                  14 [ 71] +
+ [ 46] 1884                 994 [ 53] +
+ [ 38]   16                  13 [ 16] +
+ [ 31]  994   4 ->            4 [ 14] +
+ [ 27]  995     29 ->        12 [ 11] +
+ [ 20]    7       6423 ->    21 [  8] +
+ [ 17]   27         1884 ->  15 [  8] +
+ [ 14]   29                   7 [  7]
+ [  8] 6423                  11 [  7] +
  [  7] 1018                  5 [  6] +
+ [  6] 1008                 29 [  5] +
+ [  6]   12                  45 [  3] +
  [  5] 1000                 42 [  3] +
+ [  5]    1                 41 [  2]
+ [  5]   79                 129 [  2]
+ [  5] 1001                1010 [  2] +
  [  4] 1017                  63 [  2] +
```

Figure 2

In this output we explore a focus path (4 29 6423 1884). In-flows are immediate predecessors of the focus path. Out-flows are corresponding successors. The plus sign identifies a possible expanding of the path, and the numbers in the brackets represent frequencies. For example, element 5 preceded the focus as an in-flow 950 times. Expanding the focus path to the left to include the element 5, results in the output in Figure 3, corresponding to a focus path (5 4 29 6423 1884).

```
  Freq In-Flow   Focus    Out-Flow Freq
-----------------------------------------
+ [231]  995
+ [167]  994
+ [ 47]    1
+ [ 40]    3                   1 [397] +
+ [ 40]    5                1884 [290] +
+ [ 36] 1038                  49 [105] +
+ [ 33]  999                   3 [ 34] +
+ [ 22] 1000                6423 [ 30] +
+ [ 15] 1001  5 ->            994 [ 30] +
+ [ 15]    4    4 ->          14 [ 27] +
+ [ 12]   13      29 ->         5 [  5] +
+ [ 11] 1884        6423 ->    12 [  4] +
+ [  8] 1024          1884 ->  15 [  3] +
+ [  7]   48                  13 [  3]
+ [  7] 1017                  21 [  3]
+ [  6]  996                   7 [  2]
  [  6] 1089                1010 [  2] +
  [  5] 1018                  29 [  2]
+ [  4]   27
+ [  3]   11
+ [  3]  998
```

Figure 3

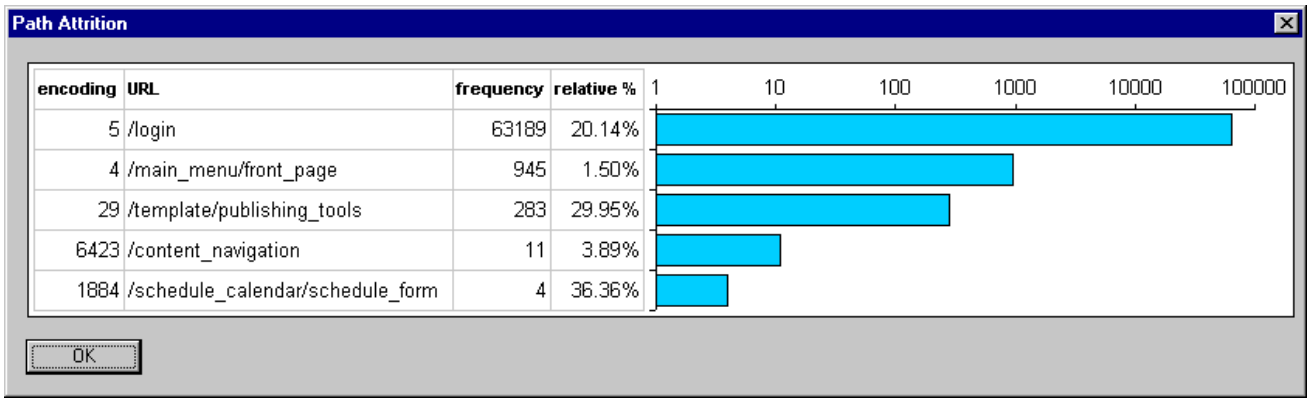

Figure 4

Another view of Path Explorer shows levels of attrition along the paths (logarithmic scale is used), as illustrated by Figure 4. For example, 29.95% of traffic through a path (5 4) passed to the element 29.

5. ANALYSIS AND EXPLORATION OF COUPLES

Couple Analyzer presents sets of filtered *couples* sorted by certain criterion. Recall that a couple is an ordered pair <**a,b**> of elements which has occurred in some (sub)path. Frequent couples are fingerprints of Web site traffic, equally as interesting as frequent paths.

However, the fact that the number of steps taken to connect **a** and **b** can vary adds extra complexity to the concept of a couple. In addition to an overall frequency, **N(a,b)**, a couple has a particular frequency, **N(k,a,b)**, for each number of connecting steps **k**, and $\mathbf{N(a,b) = \Sigma_k N(k,a,b)}$. We display the average number of steps (AvgS in Figure 5) connecting a couple. A couple with a high average number of steps is a precious commodity, since it represents a delayed influence. In the opposite extreme case, a couple with an average step number equal to one represents a physical link. Filtering out couples with low relative **1**-frequency **N(1,a,b)/N(a,b)** presents a customer with a valuable insight to redesign the site by introducing useful links. Due to the intricacies of modern Web site design (for example proxy servers), a couple can have a non-trivial 1-frequency without being a link. The details are beyond the scope of this paper.

Other important information about a couple relates to different measures of common frequency of couple elements (see Confid and Simil in Figure 5). Conditional probability **P(b|a) = N(a,b)/N(a)** is one such measure. It is known as confidence in the context of *association rules* [**2**]. We also use a symmetric measure based on the concept of similarity, **s(a,b) = N(a,b)/(N(a)+N(b)-N(a,b))**. Both measures are scaled to (0,1]. In the extreme case when **a** and **b** only happen together, **P(b|a)=s(a,b)=1**. Filtering for couples with high similarity and frequency (*support*) provides an important list of intimately related URLs.

To provide the reader with some taste of the introduced concepts, we show a simplified version of Couple Analyzer output in Figure 5. For example, the first couple happens 47.9% in one step, 22.9% in two steps, and so on with ellipsis representing remaining infrequent occurrences. Two filters are employed in this example to exclude cycles and to set minimum average step to two. In particular, a couple <1038, 1017> is an instance of a stable user pattern that happens most frequently in two steps (68%).

Couple Explorer drills into the data related to a particular couple <**a,b**>. Figure 6 schematically presents couple exploration:

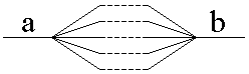

Figure 6

There is a duality between Figures 1 and 6. Couple Explorer tracks all the paths connecting **a** and **b**, sorted by frequency (or by length and frequency). For special couples (say, **b** is a checkout or registration page) the most frequent connecting paths are important from a business perspective. For usability purposes it is important to filter such paths by different criteria. The following filters are implemented:

- filtering by proper paths (connecting path is *proper* if it

```
No. AvgS  Simil Confid   Freq  <Couple>    %-age: 1Step   2    3    4    5    6    7
-------------------------------------------------------------------------------------
 1. 2.08  0.100  0.132  34033  <  994, 1001>        47.9 22.9 13.9  7.8  4.3  2.1  ...
 2. 2.38  0.080  0.106  27231  <  994, 1000>        32.0 30.5 18.5 10.1  5.3  2.5  1.1  ...
 3. 2.07  0.055  0.067  17115  <  994,  995>        43.8 27.0 15.9  7.8  3.4  1.5  ...
 4. 2.27  0.045  0.220  10301  < 1008,  994>        10.2 64.8 16.2  6.0  1.8  ...
 5. 3.13  0.038  0.160   9059  < 1038,  994>        13.0  6.7 49.2 20.1  7.4  2.6  ...
 6. 2.32  0.091  0.140   7930  < 1038, 1017>         4.6 68.1 20.7  4.6  1.3  ...
 7. 2.36  0.105  0.177   6031  <    4,    3>        27.1 42.1 12.6  8.7  5.6  2.9  1.0  ...
```

Figure 5

does not contain ends inside),
- filtering by including elements,
- filtering by non including elements,
- filtering by length,
- filtering by frequency.

Special vertical applications naturally arise as soon as the customer provides lists of elements related to particular business events.

6. FUTURE WORK

We identify the following directions as important for further development:

- Automatic discovery of interesting objects is essential for industrial usability. Research of how to configure interesting objects and what constitutes an interesting object is required.

- Comparison of two or more data sets in order to identify significant differences. Comparison is important for two reasons: 1) to identify changes (trends) with respect to time, and 2) to identify differences between segments of filtered data (for example, for different groups of customers, referrers, with and without advertisement, promotion, etc.).

- Predictive mining is an important component of eCRM and naturally integrates with the presented Path Analysis infrastructure. For example, consider the possibility of dynamically updating page content for a customer on a particular path.

7. RELATED WORK

General sequence analysis has a long history. Applications cover many fields, including biology [**19**], [**16**], information retrieval [**12**], [**17**], and time series [**5**]. A sequential patterns approach to retail basket data mining is proposed in [**3**] and [**18**]. Academic and experimental work in Web data analysis is exploding in popularity. This paper concerns itself with Web site log data analysis. The significance of frequent long subsequences was identified in [**15**]. Elegant measures of Web page importance are suggested in [**7**]. For sessionizing Web log data see [**4**]. There are successful attempts of general Web net classification [**11**], [**13**] and [**14**]. Industrial Web mining is discussed in [**10**]. Exploratory data analysis of a Web site with emphasis on human interaction is researched in [**8**]. Innovative approaches to visualization of Web data are proposed in [**6**] and [**9**].

Nearly every vendor of Web analysis products provides some sort of path analysis solution; as such, a complete list is impractical. Most of these solutions focus on reporting most frequent paths and provide little, if any, interactivity.

8. CONCLUSIONS

This paper presents a comprehensive approach to the analysis of Web site traffic. It identifies important underlying objects and a range of user interfaces for their exploration. The special data structures used allowed fast interactive analysis of site elements, paths, and couples, and in-depth exploration of these objects. An infrastructure is laid out for the flexible development of business applications, taking advantage of batch mode and interactive tools to explore site navigation activity.

9. ACKNOWLEGMENTS

All the work described in this paper was performed while the authors were employed by Accrue Software, Inc. Many other people at Accrue contributed to this effort, including but not limited to: Phil Aaronson who provided invaluable domain expertise on the intricacies of Web data; Alan Baer who detailed some early efforts at path analysis developed at Accrue called NavGraph and BuyPath; Judson Groshong who supplied comprehensive marketing research on the state of the art in path analysis; and John D'Albis who helped with discussions of the underlying software architecture.

10. REFERENCES

[1] Accrue Software, Inc. http://www.accrue.com/Products/Accrue_G2/g2_overview.html.

[2] R.Agrawal, H.Mannila, R.Srikant, H.Toivonen, and A.I.Verkamo. Fast Discovery of Association Rules. *U.Fayyad et al (eds.). Knowledge Discovery and Data Mining*, MIT Press, 307-328, 1996.

[3] R.Agrawal and R.Srikant. Mining Sequential Patterns. *Proc. of the 14th Int'l Conference on Data Engineering*, Taipei, Taiwan, 1995.

[4] B.Berendt, B.Mobasher, M.Spiliopoulou, and J.Wiltshire. Measuring the Accuracy of Sessionizers for Web Usage Analysis, A Summary of Results, *Workshop on Web Mining, First SIAM Intl. Conf. On Data Mining*, Chicago, 2001, 7-14.

[5] D.Berndt and J.Clifford. Finding patterns in time series in advances. *U.Fayyad et al (eds.). Knowledge Discovery and Data Mining*, MIT Press, 37-59, 1996.

[6] I.Cadez, D.Heckerman, C.Meek, P.Smyth, and S.White. Visualization of Navigation Patterns on a Web Site Using Model-Based Clustering, *KDD-2000*, 280-284, Boston, MA, 2000.

[7] P.K.Chan. A non-invasive learning approach to building web user profiles, *WebKDD-99 Workshop on Web Usage Analysis and User Profiling*, 7-12, San Diego, 1999.

[8] E.H.Chi, P.Pirolli, and J.Pitkow. The Scent of a Site: A System for Analyzing and Predicting Information Scent, Usage, and Usability of a Web Site. *CHI-2000*, The Hague, The Netherlands, 161-168, 2000.

[9] E.H.Chi, J.Pitkow, J.Mackinlay, P.Pirolli, R.Gossweiler, and S.K.Card. Visualizing the Evolution of Web Ecologies. *ACM Conference on Human Factors in Software, CHI -98*, Los Angeles: 400-407, 1998.

[10] R.Cooley, B.Mobasher, and J.Srivastava. Web Mining: Information and Pattern Discovery on the World Wide Web, Department of Computer Science

University of Minnesota Minneapolis, MN 55455, USA, 1997.

[11] B.A.Huberman, P.Pirolli, J.Pitkow and R.J.Lukose. Strong Regularities in World Wide Web Surfing. *Science 280*, 95-97, 1998.

[12] P.Pirolli and S.K.Card. Information Foraging. *Psychological Review*, 106(4), 643-675, 1999.

[13] P.Pirolli and J.E.Pitkow. Distributions of Surfers' Paths Through the World Wide Web: Empirical Characterization. *World Wide Web*, 2(1-2), 29-45, 1999.

[14] J.E.Pitkow. Summary of WWW Characterizations. *Web Journal*, 2(1-2), 3-13, 1998.

[15] J.E.Pitkow and P.Pirolli. Mining longest repeated subsequences to predict World Wide Web surfing. *Second USENIX Symposium on Internet Technologies and Systems*, 1999.

[16] M.A.Roytberg. A search for common patterns in many sequences. *Computer Applications in the Biosciences*, 8(1), 57-64, 1992.

[17] G.Salton and M.McGill. Introduction to Modern Information Retrieval. McGraw-Hill Comp., NY, 1983.

[18] R.Srikant and R.Agrawal. Mining Sequential Patterns: Generalizations and Performance Improvements, *Proc. of the Fifth Int'l Conference on Extending Database Technology (EDBT)*, Avignon, France, 1996.

[19] M.S.Waterman, ed.. *Mathematical Methods for DNA Sequence Analysis*. CRS Press, 1989.