01 Jan 2000

Association Rules for Web Data Mining in WHOWEDA

Sanjay Kumar Madria
Missouri University of Science and Technology, madrias@mst.edu

C. Raymond

M. Mohania

Sourav S. Bhowmick

Recommended Citation

S. K. Madria et al., "Association Rules for Web Data Mining in WHOWEDA," *Proceedings of the International Conference on Digital Libraries: Research and Practice, 2000 Kyoto*, Institute of Electrical and Electronics Engineers (IEEE), Jan 2000.
The definitive version is available at https://doi.org/10.1109/DLRP.2000.942179

Association Rules for Web Data Mining in WHOWEDA

Sanjay Kumar Madria
Department of Computer Science, University of Missouri-Rolla, Rolla, MO
Chris Raymond
Department of Computer Science Purdue University, West Lafayette, IN
Sourav Bhowmick
School of Computer Engineering, Nanyang Technological University
Singapore
Mukesh Mohania
Computer Science Department, Western Michigan University Kalamazoo, MI

{madrias@umr.edu, craymond@cs.purdue.edu, assourav@ntu.edu.sg, mohania@cs.wmich.edu}

Abstract

In this paper, we discuss association rules which can be discovered from web data. The association rules are discussed within the scope of our WHOWEDA (warehouse of web data) project. WHOWEDA is supported by a web data model and a set of algebraic operators. Web data model allows uniform and integrated view of web data gathered using a user's query graph. A user's query graph describes the query by example (what user perceives as the query) and web coupling query gathers instances of such a query graph from the web and store them in the form of subgraphs (called web tuples) in a web table. We discuss association rules within this domain. An association rule defines an association between the nodes and links attributes of web tuples within a web table. There are two different classes of association rules that can be developed from data in a web table. Node-to-node associations are those rules that relate the content (defined by metadata attributes) between two or more nodes within a web tuple. Link associations are rules that show the connectivity of different URLs. Distinguishing the two types of associations provides a view of the structure of the web data. The goal of performing web association mining on web data is to better organize searching patterns through hyperlinked documents.

1. Introduction

The advent of the World Wide Web has caused a dramatic increase in the usage of the Internet. Information on the WWW is important not only to individual users, but also to the business organizations especially when the critical decision-making is concerned. Most users obtain WWW information using a combination of search engines and browsers. However, these two types of retrieval mechanisms do not necessarily address all of a user's information needs [1]. This is particularly true in the case of business organizations that currently lack suitable tools to systematically harness strategic information from the web and analyze these data to discover useful knowledge to support decision making.

The resulting growth in on-line information combined with the almost unstructured web data necessitates the development of powerful yet computationally efficient web data mining tools. Web data mining [2, 13] can be defined as the discovery and analysis of useful information from the WWW data. Web involves three types of data; contents on the web pages, the web log data regarding the users who browsed the web pages and the web structure data. Thus, the WWW data mining includes focus on three issues; *web structure mining* [3,4,12], *web content mining* [6] and *web usage mining* [2, 8, 9]. Web structure mining involves mining the web document's structures and links. Web content mining describes the automatic search of information resources available on-line. Web usage mining includes the data from server access logs, user registration or profiles, user sessions or transactions etc.

0-7695-1022-1/01 $10.00 © 2001 IEEE

In our discussion here, we focus on the web association research issues [14] with respect to the web warehousing project called WHOWEDA (*Ware*h*ouse of *W*eb *Data*) [16,17,18,19,20]. The key objective of WHOWEDA is to design and implement a web warehouse that materializes and manages useful information from the web to support strategic decision making. We are building a web warehouse [20] using the database approach of managing a web warehouse containing strategic information coupled from the web that may also inter-operate with conventional data warehouses. One of the important areas of our work involves the development of techniques for mining useful information from the web. We would be integrating WHOWEDA with intelligent tools for information retrieval and extend the data mining techniques to provide a higher level of data organization for unstructured data available on the web.

With respect to our web data mining approach, we argue that extracting information from a very small subset of all HTML web pages is also an instance of web data mining. In WHOWEDA, we focus on mining a subset of web pages returned in response to a user's query graph and stored in one or more *web tables*. We believe that due to the complexity and vastness of the web, mining information from a subset of web stored in the web tables is more feasible option. Our web warehousing approach allows us to do this effectively as we materialize only the results returned in response to a user's query graph.

Association rules have proven to be a useful measure of the relationships between data and are used to identify improved marketing strategies for retailers [10,11]. In much the same way that associations link transaction items, they can show the linked nature of web documents [6,15]. The goal of performing web association mining on web data is to better organize searching patterns through Internet documents.

An association rule defines an association between the attributes of a web tuple within a web table. There are two different classes of rules that can be developed from data in a web table. Node-to-node associations are those rules that relate the content (defined by metadata attributes in our model) between two or more nodes within a web tuple. Link associations are rules that show the *connectivity* of different URLs. Distinguishing the two types of associations provides a view of the structure of the web data.

This paper will define the terms necessary for mining the web data, give algorithms for computing the association rules and discuss algorithms with the help of examples.

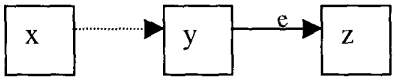
2. WHOWEDA

Our web data model [17,18, 19] in WHOWEDA consists of a hierarchy of web objects. The fundamental objects are Nodes and Links, where nodes correspond to HTML text documents and links correspond to hyper-links interconnecting the documents in the WWW. These objects consist of a set of attributes as follows: Nodes = [url, title, format, size, date, text] and link = [source-url, target-url, label, link-type]. Note that a link-type can be global link (link among web pages hosted by different web servers), local link (link among web pages hosted by same server) and interior link (link with in the same page). In our web warehouse, Web Coupling System [7] is a database system for managing and manipulating coupled information extracted from the Web. We have defined a set of coupling operators to manipulate the web tables and correlate additional useful and related information [7].

We materialize web data as web tuples stored in web tables. Web tuples, representing directed connecting graphs, are comprised of web objects (Nodes and Links). We associate with each web table a web schema that binds a set of web tuples in a web table. A web schema contains the meta-data that binds a set of web tuples to a web table in the form of connectivities and predicates defined on node and link variables. Connectivities represent structural properties of web tuples by describing possible paths between node variables. Predicates on the other hand specify the additional conditions that must be satisfied by each tuple to be included in the web table. In our warehouse, a user expresses a web query in the form of a query graph consisting of some nodes and links representing web documents and hyperlinks in those documents, respectively. Each of these nodes and links can have some keywords imposed on them to represent those web documents that contain the given keywords in the documents and/or hyperlinks. When the query graph is posted over the WWW, a set of web tuples each satisfying the query graph are harnessed from the WWW. Thus, the web schema of a table resembles the query graph used to derive the web

tuples stored in web table. Note that the results are returned as web tuples. Note that some nodes and links in the query graph may not have keywords imposed. They are called unbound nodes and links, respectively.

Example: *Consider a query to find all companies who sell color laser printers starting with the web page www.buy.com. The query above may be expressed as follows(constraints on x, y, z and on e are given below) :*



The query graph shown above is assigned as the schema of the web table generated in response to the desired query. The schema corresponding to the above query graph can be formally expressed as

$<X_n, X_l, C, P>$ where

- X_n is the set node variables; x,y,z in the example above.
- X_l is the set of link variables; - (unbound link) and e in the example
- C is set of connectivities ; $k_1 \Lambda k_2$ where k_1 = x<->y, k_2 = y<e>z
- P is a set of predicates as follows : $p_1 \Lambda p_2 \Lambda p_3 \Lambda p_4$ such that
  - p_1 (x) = [x.url EQUALS www.buy.com],
  - p_2 (e) = [e.label CONTAINS "color laser"],
  - p_3 (y) = [y.text CONTAINS "printer companies"] and
  - p_4 (z) = [z.text CONTAINS "laser printer"].

The query will return all web tuples satisfying the web schema given above. Thus, it is likely that many instances of the query graph shown above will be returned as web tuples.

3. Web Data Collection and Filtering

The Internet is too unstructured to be mined in a direct fashion. Therefore, there are some important steps (see Figure 1) before web data mining algorithms can be developed and applied. The first and most important step is the collection of web data to mine; domain selection. The next step is data cleaning. However, in the vast domain of WWW, these seem to be the difficult steps.

Domain Selection - In our web warehouse approach, this step is implemented with the help of *web coupling* query. Web coupling query gathers relevant information from the web that relates to some topic or piece of information that a user desires and frame his query as an example. Thus, it is based on a user's query graph. Web coupling query is used as a starting point for the generation of association rules between documents in the web warehouse. Web coupling query gathers instances of the query graph as directed connected graphs called *web tuples*.and organizes them into *web tables*, whose schema is a directed *query graph*. The nodes in the graphs represent pages on the WWW, and the links represent hyperlinks in the pages. Within the scope of web tables and web tuples, we discover association rules.

Data Cleaning -. To this end, we have defined a collection of web algebraic operators [5], which can manipulate and filter the web data returned in response to the web coupling query. Some of the operators for data cleaning are *Web select* and *Web project*. For more details, refer to [5].

Web Select – The select operation on a web table extract web tuples from a web table satisfying certain constraints. However, since the schema of web tables is more complex than that of relational tables, selection constraints have to be expressed as predicates on node and link variables. The web select operation augments the schema of web tables by incorporating new conditions into the schema. However, unlike the relational counterpart, the selection criteria of a web operation are incorporated into the schema of the resultant web table. Thus, it makes it different from select of relational table.

Given a web table *W*, a web select operation finds tuples from *W* satisfying some selection criteria. Formally, the selection criteria can be represented (similar to a web schema) by a 4-tuple < *Xsn, Xsl, Cs, Ps* > where *Xsn* and *Xsl* are node and link variables respectively, and *Cs* and *Ps* represents the additional selection conditions imposed on the *W*. Node and link variables used in the selection criteria may be found in the schema of the input web table *W*.

Web Project – A web project operator is used to isolate the data of interest, allowing subsequent queries to run over a smaller, perhaps more structured web data. The web project operation on a web table extract portions of a web tuple

satisfying certain conditions. However, since the schema of web tables is more complex than that of relational tables, projection conditions have to be expressed as node and link variables and/or connectivities between the node variables. The web project operation reduces the number of node and link variables in the original schema and the constraints over these variables.

Given a web table W with schema M = as $<X_n, X_l, C, P>$, a web projection on W computes a new web table W' with schema M' = $<X'_n, X'_l, C', P'>$. The components of M' depends on the project conditions. A user may explicitly specify any one of the conditions or any combination of the three conditions discussed below to initiate a web project operation.

- Set of node variables: To project a set of node variables from the web table.
- Start-node variable and end-node variable: To project all the instances of node variables between two node variables.
- Node variable and depth of links: To restrict the set of nodes to be projected within a limited number of links starting from specified node variable.

4. Definitions

In our web warehouse approach, a query graph Q captures instances $\{I_1, I_2, \ldots. I_n\}$ from the web. Each of these instances represents a subgraph represented as (N, E) where N is the set of nodes (web pages) and E is the set of edges (links). In web association mining, we are interested in generating relationship among the node contents and connectivities.

Associations are based on the user specified support and confidence. In the case of node-to-node associations, node support is the percentage of tuples in a particular web table that contain the associated data within a given connectivity. When dealing with link associations, link support is the percentage of links in a web table that relate the source and destination URLs.

Confidence is the other measure of association between data. It determines the probability of finding one item based on another. Node confidence is the percentage of nodes, containing a distinct subject matter, that conform to the relationship of the data. Link confidence is the percentage of times that a URL links to the other URL in the relationship. It is important to make the distinction between the types of confidence and support, as the algorithms presented later depend on the type of association to be generated.

We formally define :

Web Association: A relationship between nodes of specific content within a given connectivity, which has sufficient web support and web confidence. In terms of web tuples this is defined $\exists$ x, y, z such that $\{x \in N \wedge y \in N \wedge z \in V \wedge x<z>y \wedge$ web support $\geq$ min. support $\wedge$ web confidence $\geq$ min.-confidence$\}$.

Web Support: Support for a Web Association determines the prevalence of the association within the table. Web Support is determined by the number of instances of a connectivity that relates nodes of a particular content.

Web Confidence: The confidence in a web association is the probability of finding a web association in a web table. It is calculated by dividing the support for a relationship of the type x <z> y by the prevalence of the node x in the table.

For example, assume that for a given query Q, we have a web table of 100 tuples and there exist 65 nodes that corresponded to pages about software, and there are 50 nodes that correspond to Internet utilities. Further assume that 27 of the software nodes are linked through a specific link label to one of the utility pages. Of the 65% of pages containing information about software, 41.5% contain links to pages with Internet utilities. In this case, the web support is 27, which is the count of the instances of the given relationship. The web confidence is the 41.5% probability of finding a link to an Internet utility from a software page. Thus, the concepts of support and confidence of associations can be adapted to the web tuples stored in the warehouse.

5. Association Rules

The process of mining web tables for association rules in a web warehouse answer two questions. The first is the question of how the data in the web table is related. For example, retailers can get an idea of how the content on their site is related to of other retailers that sell accessories. This would be an example of a node relationship, where the title or content of one page is consistently paired with the title or content of another page. These associations are made using meta tag attributes such as title, text etc. The limited capacity to immediately recognize content complicates the generation of node rules. The second question to be answered

is how interlinked are certain documents within a table. The confidence of this type of association is the measure of the luminosity of the document. It helps site designers and site users to get an idea of what is available from a given page. It is especially useful for comparing types of services offered by various sites. The purpose of mining web tables is not to reveal content of the data, but structure and patterns within the data.

Algorithms that have been developed for the mining of association rules in a relational environment work in two separate phases [10]. The first phase is counting. All instances of an item or set of items are counted, and those sets that meet the user specified support are kept. These are called frequent data sets, or frequent item sets. In the case of web data, there will be two frequent item sets generated. The frequent node set and the frequent link set. They conform to the node support and link support parameters respectively The itemset for nodes corresponds to the node attributes with same (predicate, value) pair such as Title CONTAINS "DBMS" where Title is a node attribute and CONTAINS is a predicate with value DBMS. Note that here itemset may contain items with similar predicates and equal values for multiple attributes such as (title, date). The itemset for *link* consists similar values for attribute pair such as source URL and destination or target URL. In case of links, link type can also be differentiated; a source URL can be linked to destination URL through a local, global or interior link. This is an example of an itemset containing items from multiple attributes for link association. The frequent sets are stored in a tabular format, with pairings of attributes, and the corresponding support counts. In phase two, rules are generated from the frequent sets by determining confidence as a function of support. The rule generation for node and link confidence will be calculated by the same method. The key to mining web data is the creation of the frequent sets, since the data is not in a normalized format. Algorithms to generate these sets from web tables are given in next section.

5. 1 Creating Frequent Sets

Frequent link sets serve as the starting point for association rule generation. According to our web data model, a link is comprised of a source and destination URL, a label, and a link type. Only the source and destination URLs will be used to distinguish the links. A count of each distinct URL and a count of each pairing of URLs are stored along with the link attributes. The resulting set of links is a relational table with a schema as shown in Figure 1.

Source URL	Dest. URL	Count

Figure 1. Frequent Link Set

The algorithm shown in Figure 2 will develop such a table for processing by the rule generation procedure. The generation of the frequent node set differs substantially from that of the link associations. As mentioned previously, here nodes are associated based on metadata attributes. For nodes, the schema information within the query graph is used to exclusively limit pairings of pages. A query graph has a specified number of connectivities for the nodes. These dictate which nodes are visible from each other in a web tuple. Figure 3 shows the process of developing the frequent set of node-to-node associations. The schema of this table will also be different, as shown in Figure 4. Unlike the previous algorithm, this one is not a one-pass operation. Multiple sweeps must be made of the data due to the extra restrictions of the connectivities.

Figure 2: Algorithm to generate frequent link sets

```
Begin: CreateLinkSet()

For each Link L
{
  If {L.src, null} exists in table, increment count
  Else create entry in table for {L.src, null};

  If {L.src, L.dest} exists in table, increment count
  Else create entry in table for {L.src, L.dest};

  Increment total number of links
}

For each entry in table T
{
 If (T.entry.count/T.count) < Minimum Support Then
    Remove T.entry;
}
End CreateLinkSet();
```

Figure 3: Creating frequent node sets

```
Begin CreateFreqNodeSet();
Table T;
For each node in web table W
{
 If (node exists in T) increment count;
 Else add entry in T for {nolabel, node, null}
}

For Each Web tuple in W
{
 For Each connectivity K:
  {
  If ({K.linklabel, K.left, K.right} exists) increment
count;
  Else create entry in T for ({K.linklabel, K.left,
K.right};
  }
}

For Each Entry in T
{
 If (T.entry.count/ W.numTuples) < MinSup then
 Remove T.entry;
}
End; CreateFreqNodeSet();
```

5. 2 Deriving Rules

The next task is to generate link associations from the frequent link table described earlier. These rules will have the form "X% of links originating from www.x.com have www.y.org as their destination". In this skeleton rule, X% is the link confidence of the association. The confidence of a link association is simply a function of the support for the given URL's in the frequent link set. The algorithm to generate all acceptable associations from a frequent link set is given in Figure 5. The purpose of these rules is to give insight to the accessibility of different documents within a given search criteria. The derivation of node associations is done in a similar way, except that it is node confidence that is being computed. The basic formula for the computation confidence is exactly the same. Figure 6 describes the algorithm to derive node associations. An example of the entire process of generating link and node associations is given in the next section.

Figure 4: Schema of the frequent node set

Label	L Node	R Node	Count

Figure 5: Deriving link associations

```
Begin DeriveLinkRule():

For each entry E in table T
{
 If (E.Dest != NULL AND E.src != NULL)
  {
   E' = find(E.src, NULL);
   Conf = E.count / E'.count;
   If (Conf > MinConf) Keep Rule
  }
}
End DeriveLinkRule();
```

Figure 6: Deriving node associations

```
Begin DeriveNodeRule():

For each entry E in table T
{
  If (E.rnode != NULL)
   {
    E' = find(E.lnode, NULL);
    Conf = E.count / E'.count
    If (Conf > MinConf) Keep Rule
   }
}
End DeriveNodeRule();
```

6. Conclusions

The explosion of data on the Internet has necessitated better methods for searching not only content but also structure in web documents. WHOWEDA provides a method for searching both necessities. Web mining takes this one step further by revealing the patterns and linked properties of documents in web tuples. The success of this process is dependant on the ability to detect the subject of the content in a web page. The rise of XML shows promise of an improvement to this pitfall. Also, the algorithms presented are merely prototypes and need to be refined for speed and efficiency. We are studying classification and clustering among web tuples in WHOWEDA.

References

1. H. Vernon Leighton and J. Srivastava. Precision Among WWW Search Services (Search Engines): Alta Vista, Excite, Hotbot, Infoseek, Lycos. http://www.winona.msus.edu/is-f/library-f/webind2/webind2.htm, 1997.

2. R. Cooley, B. Mobasher and J. Srivsatava. Web Mining: Information and Pattern Discovery on the Word Wide Web. In Proceedings of the 9th IEEE International Conference on Tools with AI (ICTAI,97), Nov. 1997.

3. Sourav S. Bhowmick, S. K. Madria, W.-K. Ng, E.-P. Lim, Web Bags : Are They Useful in Web warehouse? In proceedings for 5th International Conference on Foundation of Data Organization, Japan, Nov. 1998. Also, to appear in the book Information Organization and Databases. K. Tanaka and S. Ghandeharizadeh (Eds.). Kluwer Academic Publishers, 2000

4. T. Bray, Measuring the Web. In Proceedings of the 5th Intl. WWW Conference, Paris, France, 1996.

5. Wee-Keong Ng, Ee-Peng Lim, Chee-Thong Huang, Sourav Bhowmick, Fengqiong Qin. Web Warehousing : An Algebra for Web Information. In Proceedings of the IEEE Advances in Digital Libraries Conference, Santa Barbara, U.S.A., April 1998.

6. Shian-Hua Lin, Chi-Sheng Shih, Meng Chang Chen, et al. Extracting Classification Knowledge of Internet Documents with Mining Term Associations: A Semantic Approach. In Proceedings of 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Melbourne, Australia, 1998.

7. Sourav S. Bhowmick, W.-K. Ng, E.-P. Lim. Information Coupling in Web Databases. In Proceedings of the 17th International Conference on Conceptual Modelling (ER'98), Singapore, November 16-19, 1998.

8. D. Backman and J. Rubbin, Web log analysis: Finding a Recipe for Success. http://techweb.comp.com/nc/811/811cn2.html, 1997.

9. J. Pitkow, In Search of Reliable Usage Data on the WWW. In Proceedings of the 6th International World Wide Web Conference, Santa Clara, California, April, 1997.

10. H. Mannila, Methods and Problems in Data Mining. In Proceedings of International Conference on Database theory, Delphi, Greece, January 1997.

11. R. Aggarwal, R. Srikant, Fast Algorithms for Mining Association Rules. In proceedings of the 20th VLDB Conference, Santiago, Chile, 1994.

12. Ellen Spertus, ParaSit : Mining Structural Information on the Web. In proceedings of 6th International WWW Conference , April, 1997.

13. Myra Spiliopoulou: Data Mining for the Web. PKDD 1999: 588-589

14. Sanjay Kumar Madria, Sourav S. Bhowmick, Wee Keong Ng, Ee-Peng Lim: Research Issues in Web Data Mining, DaWaK 1999: 303-312

15. Lisa Singh, Peter Scheuermann, Bin Chen: Generating Association Rules from Semi-Structured Documents Using an Extended Concept Hierarchy. CIKM 1997: 193-200

16. Sourav, S.B., Madria, S. K., W. K. Nag and Ee Peng Lim, Cost-benefit Analysis of Web Bags in a Web Warehouse: An Analytical Approach, WWW Journal, Vol. 3, No. 3, 2000.

17. Sourav S. B., Wee-Keong, Madria, S.K., Lim Ee Peng, Detecting and Representing Relevant Web Deltas using Web Join, in IEEE proceedings of International Conference on Distributed Computing System (ICDCS'2000), March 2000, Taiwan.

18. Sourav S. B., Madria, S.K., et al., Reverse Osmosis to Reduce Cognative Overheads in a Web Warehouse, appeard in the IEEE proceedings of 7th International Conference on Parallel and Distributed Systems (ICPADS'2000), 4th to 7th July, 2000, Japan.

19. Sourav S. B., Madria, S.K., et al., π-Web Join in a Web Warehouse, in IEEE Proceedings of 6th Intl. Conf. on Database Systems for Advanced Applications (DASFAA'99), April, 1999.

20.Sourav S. B., Madria, S.K., et al., Web Warehousing : Design and Issues, in Proceedings of International Workshop on Data Warehousing and Data Mining (DWDM'98) in conjunction with ER'98, 16-20th Nov. 1998, Singapore, Lecture Notes in Computer Science, Vol. 1552, Springer-verlag.