CHAPTER 4

DECISION BASED INDUCTION RULE MINING
FOR WEB USAGE INFERENCES

4.1 INTRODUCTION

With the rapid growth of the World Wide Web, E-Commerce has been increasing to a faster extent than ever before keeping pace with the web. These electronic commerce behaviors incessantly produce a huge amount of Web log data. As more organizations rely on the Web to accomplish business, the conventional approaches and methods for market analysis require to be revisited in this context. Organizations often produce and gather great volumes of data in their day to day operations. Most of this information is generated routinely by Web servers and composed in server access logs. Other sources of user information comprise referrer logs which consists information about the referring pages for each page reference, and user listing or survey data collected through implements such as CGI scripts.

Examining such web log data assist the organizations to decide the repeated purchase made by the clients, cross marketing approaches across products with the increasing options available to the customer while selecting the product, or efficiency of promotional operations. Investigation of server access logs and user registration data also provide with useful information on how to customize the web site in order to meet the essential demands to be satisfied by the organization. Trade marketing on the World Wide Web examining user navigation patterns helps in targeting ads to precise groups of users.
Web mining is the process of employing data mining methods to automatically mine helpful information from web log data. Based on the consistency which analysis the targets, Web mining is split further into three different measures. They are Web content mining, Web structure mining and Web usage mining. Web usage mining is a method of data mining to identify and extract the web usage inferences and actions from Web log data. Web usage mining concentrates on methods that help the organization in a way to forecast about the behavior of user behavior whenever the user interacts with the web log data.

The functions of web usage mining from the point of internet are further divided into two different forms. They are

1. Identifying the user navigation patterns collected from web log data and
2. Extracting the user profile obtained from web log data

Web usage mining involves the identification of data and extraction of data from web log data has the form of user clickstream that refers to the users’ navigation through a web site.

The techniques used for web usage mining as shown in figure 4.1 are of different forms. They are expressive in nature derived from the statistical analysis using the web log data clickstream, mining association rules from clickstreams that relates in a way to derive which pages are highly correlated, clustering of web log data in order to evolve usage clusters, classification of web log data in order to generate user profiles. Some of the others include sequential pattern detection that evaluates the inter-session patterns in such a way that the purview of a set of page views is accompanies by another set of page views.
Figure 4.1 Web usage mining techniques
These page views are presented in the form of time-ordered set of episodes, or based on the dependency modeling to measure the significant dependencies between different variables available in the web domain using web log data. Some systems have previously been developed for this area: For example, WebSIFT employed clustering, statistical analysis or association rules, WUM consisting of the application of association rules with the help of extension of SQL), or WebLogMiner (that merges OLAP and KDD).

In this research work, Induction decision rule mining is presented for web usage mining to classify web documents automatically using web log data. The proposed induction based decision rule model is deployed for deriving the inferences and obtains the implicit hidden behavioral aspects in the web usage mining to investigate at the web server and client logs with the help of web log data. The decision based rule induction mining combines a fast decision rule induction algorithm and a method for converting a decision tree to a simplified rule set, however still rationally equal to the original tree.

4.2 WEB USAGE ANALYSIS WITH INDUCTION RULE MINING

The Induction Rule Mining of user clickstream has been used for Web Usage Analysis to perform e-commerce operations. The goal of web usage analysis with induction rule mining is to analyze meaningful relationships or associations about access to an on-line catalogue of an e-shop with the help of web log data. In order to capture the on-line catalogue of an e-shop more accurately, unlike conventional model of induction rule mining, the work uses induction rule mining from two different perspectives. From the first perspective, the problem of associations between frequently occurring sequence of visited pages through clickstream is analyzed that identify the places where certain shortcuts can be added to the web pages in order to speed
up the process and help to identify groups of customers with similar navigation pattern when browsing the web log data.

From the second perspective, the answer to the question like what will be the next page visited by a customer if the navigation pattern of a customer is known in priori. With these perspectives, the on-line catalogue can be reformed by presenting the method for shortcuts at the same time can be used to detect the killer pages, pages from which the user immediately leaves the catalogue. Two elements have to be addressed in order to perform induction rule mining with web log data. They are data understanding and data preprocessing.

4.2.1 Data Understanding

The web log data file identifies two different type of information from the navigation pattern of the customer. They are the page visited by the customer, page type and page content. The page type using web log data identifies and extracts the information related to the format of e-shop whereas the page content refers to the content offered on the page is identified. With the proper understanding of the web log data, two forms of analyses can be made namely analysis of product preferences and analysis of shopping behavior.

4.2.2 Data Preprocessing

The process involved in data preprocessing using web log data initially identifies the navigation pattern followed by the users with the help of the session ID. Then a file containing the visited pages appearing in sequences with a specific navigation pattern for each user is created with the inclusion of the start page and the end page.
4.2.3 Induction Rule Mining

Inductive learning methods, as one type of machine learning technique, are used to infer rules of classification by analyzing examples from a domain. The knowledge gained by techniques related to learning has different forms of representation that comprises of parameters in the form of algebraic expressions, decision tree models, representations of grammar, rule generation, formal logic-based expressions, graphs, and networks. As a means of classification, decision-tree induction techniques construct decision trees to discriminate among classes of objects. In contrast to neural networks, decision-tree induction techniques represent rules that are readily expressed using the English language. As a result, humans easily understand and map to the sets of rules. Indeed, an inducted decision tree is a set of nested if-then statements as illustrated in the Figure 4.2.

![Decision Tree Diagram]

**Figure 4.2 Induction rule mining**

Decision-tree induction tools allow users to create decision trees and produce decision rules using the web log data for both continuous (frequent visitors of a page) and discrete target variables (non-frequent visitors of a page). In an inducted decision tree, each intermediate node represents a condition (last navigation followed by the customer), and each
leaf is assigned a class or a vector of class probabilities. Instances also referred to as cases are typically in the form of attribute-value vectors that provides a measure to obtain the numerical or nominal values of a fixed collection of properties. In other words, they arrange the set of properties of a case in a pre-assigned order; that is, \((x_1, x_2, \ldots)\) as the measurement vector \(x\) corresponding to the case. A classifier or classification rule is a systematic way of predicting what class a case is in for a given set of measurements.

Suppose that the cases can fall into \(j\) classes, that is, \(C = \{c_1, c_2, \ldots, c_j\}\). Then, a classification rule is a function \(d(x)\), so that for every \(x\), \(d(x)\) is equal to one of the classes, \(c_1\) and \(c_2, \ldots, c_j\). At the start of the decision-tree construction, it is necessary to have a web log data set consisting of pre-classified instances. The term pre-classified are also known as the target variable also called as the dependent variable has a known class. That is, a learning sample consists of data \((x_1, j_1) \ldots (x_N, j_N)\) on \(N\) cases where \(x_1 \in X\) (all possible measurement space) and \(j_n \in C\) (all possible classes), \(n = 1, \ldots, N\). The learning sample can be denoted by \(L\); that is, \(L = \{(x_1, j_1), \ldots, (x_N, j_N)\}\). The goal is to build a tree that will make it possible to assign a class to the target variable of a new instance based on the values of the other fields or independent variables.

For instance, in assessing merit worthiness, whether to provide or to reject a student application for study loan is a target variable, and is determined based on given specific personal information consisting of the independent variables, parent’s occupation, parent’s income and so on. The predictability of a constructed decision tree is tested using subsequent cases, usually called the web log data set, whose correct classification has been observed. Among the algorithms for building decision trees are CART (Classification and Regression Trees), ID3, CHAID, C4.5, IC (Interval Classifier), SLIQ (Supervised Learning in Quest), and QUEST.
CART is an algorithm that analyzes both the discrete navigation pattern and continuous navigation pattern. It adopts a binary recursive splitting method. The discrete or continuous navigation pattern is obtained by partitioning the web log data in an iterative manner into discrete subgroups, based on one of the independent variables, until splitting is no longer feasible. The final result is a set of clickstream containing the observations where each clickstream results into one classification.

The need to abstract information in an automated manner from large quantities of data suggests the use of machine-learning principles. Machine learning seeks to acquire knowledge about a specific domain from available web log data in an automated manner. Business-to-consumer electronic commerce is a newly emerging application area of machine learning. The e-commerce stores easily accumulate the customer or sales web log data as they have built-in databases for customers, sales, and inventory. By using tracking or monitoring techniques in a Web environment, e-commerce stores accumulate customer-behavior data. As the e-commerce stores have many web log data sets, there is potential for applying machine-learning techniques.

Knowledge acquired from learning techniques are in a way valuable to understand customers’ online behavior from e-commerce stores and to gain competitive strength. The knowledge is utilized from an operational level to a strategic level from e-commerce store management. These rules are used to describe the navigational pattern of groups of users. For example, Internet storefronts uses the acquired knowledge about on-line customer behavior for user-interface design, personalized advertisements, cross-sales, intelligent customer services, customer-relationship management, customer targeting, store differentiation, and so on.
4.3 DECISION TREE MODEL ON MINED INDUCTION RULES

The significant part of the system is that the decision tree model is written as an equivalent set of human interpretable rules. The value of converting decision trees into simplified logically equivalent symbolic rule sets, instead of employing a web usage knowledge discovery system based solely on decision trees.

The web usage knowledge discovery system is based mainly on generating decision trees from web log data which is not helpful for certain types of categories for which there are no training data. However, it is not hard for an intelligent person to write logical rules to cover such a situation. Creating a decision tree by hand for the purpose of generating web inferences would be much harder, particularly for a person who is not mathematically sophisticated. It is perceived that the handwritten rule sets are incorporated using the rule sets generated by the machine, either temporarily or permanently depending on the collection of additional training data.

The rule set can be easily modified by the human user in an easy manner than modifying a decision tree. There are a number of scenarios which envisage the need for such modifications. For instance, there may be arise a difference between the training data and the actual or real data that requires smaller modification of an automatically created system. In this scenario, the modification may be achieved by way of editing a rule file. The second scenario includes the desire of the user to change an existing system in order to increase the performance level that may have minimized due to the environmental changes.

The fact that a rule set is logically equivalent to a corresponding decision tree for a particular web usage inferences problem guarantees that any mathematical analysis of the overall performance of the decision tree (as
opposed to the performance of individual rules) with respect to generate web data carries over to the rule set. This case is not true if the application of a rule set only approximated a decision tree, whereas the rule set were derived from the decision tree with the help of heuristics.

The most direct measure to transform the decision tree into an equivalent set of rules in a way understandable to the user is to initiate a set with one rule for each leaf. The set of rules are then initialized in such a way that the formation is based on the logical conjunction of the tests to obtain an unique path from the root of the tree to the leaf. In C4.5, the initial rule set derived from a decision tree is modified based on the way changes in the rule set affect the classification of training data. This can be a computational burden, but the burden is lessened by the use of plausible heuristics. However, one should note that the resulting rule set of C4.5 need not be logically equivalent to the rule set initially derived from the tree.

The approach to deriving rule sets from decision trees differs markedly from that of C4.5. Fast algorithm produces a simplified rule set that is logically equivalent to the rule set initially extracted from a decision tree. As long as the new rule set is logically equivalent to the original rule set, it does not need experimental validation to compare the new set’s overall performance with that of the original set. Instead, algorithm “picks the low-hanging fruit” by

1. Carrying out, within rules, all elementary logical simplifications related to greater than and less than, e.g., converting.

2. Removing tests that are logically superfluous in the context of the entire rule set, when those tests are readily identifiable from the structure of our decision trees.
The former simplification is standard, but the latter is not. It changes the meaning of rules associated with a particular leaf of a decision tree, while preserving the overall meaning of the rule set.

Here is the precise algorithm that implements the second rule simplification. It works for a decision tree for a two-class problem (i.e., each leaf node is labeled by a class X or its complement Y), where in the aim is to produce rules for membership in the class X.

Or each leaf labeled X, a rule for membership is created in class X by forming a conjunction of conditions obtained by traversing the path from the root to X, but use those conditions only in conformance with the following: For each node N on a path from the root to a leaf labeled X, the condition attached to the parent of N is to be part of the conjunction only if the sibling criterion holds for N. The node N is not the root, and the sibling node of N is not a leaf node labeled by X. The resulting set of rules is then logically equivalent to the original tree. It should be noted that the sibling criterion, and its use here, is novel. For example, the decision tree shown in Figure 4.3 can be considered in which each feature count is assumed as integer-valued.

If one applies the algorithm to the decision tree in Figure 4.3, one sees that for each leaf labeled X, there is only one condition on the path from the root to the leaf that is not to be omitted from the conjunction, and so one immediately obtains the equivalent rule set.
Figure 4.3 Decision based induction rule mining on Web data

Web usage mining using decision based induction rule mining with web log dataset is obtained by providing the traffic information of the visitors on the basis of Web log data log files. Web log data files were used first by the webmasters and system administrators for the reasons of how much traffic is received, how many requests fail, how many visitors followed the same set of navigation pattern and what kind of errors are being produced”, etc. However, Web log data files also record and follow the visitors’ navigation pattern to derive the on-line behaviors. Web log data is one way to gather Web traffic data.
After the web traffic is received from the web log data files, the data related to the navigation pattern followed by the visitors are joined with other relational databases, over which the data mining models are implemented. Through the data mining technique, Induction based decision rule model, visitors’ navigation patterns using web log data are identified and interpreted.

4.3.1 Decision Rules

Decision rules are used in classification and prediction. It is simple and yet a powerful way of knowledge representation. The models generated by decision rules are represented in the form of tree structure. A leaf node indicates the navigation pattern followed by a visitor during a specific timestamp. The instances are classified by sorting them down the tree from the root node to leaf node. This work used Induction based decision rule algorithm which is described in the forthcoming section (4.3.2) followed by information gain with the decision rules described below.

Decision Rules

R1: \[ \text{If} \ (\text{VisitedPages} < 20 \ \&\& \ \text{Count} \ [\text{IPAddress}] < 5) = \text{“Not Preffered ClickStream”} \]

R2: \[ \text{If} \ (\text{VisitedPage} < 30 \ \&\& \ \text{Count} \ [\text{IPAddress}] > 5) = \text{“Either Preffered or Not Preffered ClickStream”} \]

R3: \[ \text{If} \ (\text{VisitedPage} > 30 \ \&\& \ \text{Count} \ [\text{IPAddress}] < 5) = \text{“Either Preferred or Not Preffered ClickStream”} \]

R4: \[ \text{If} \ (\text{VisitedPage} > 30 \ \&\& \ \text{Count} \ [\text{IPAddress}] > 5) = \text{“Preffered ClickStream”} \]
The information gain measure Induction based decision rule algorithm is used to select the test attribute at each node in the tree using web log data. Such a measure is referred to as an attribute selection measure of the goodness of split based on the session id obtained through the web log data. The attribute with the highest number of hit ratio is selected as the test attribute for the current navigation pattern. As a result, the attribute with highest number of hit ratio being selected minimizes the information needed for sample classification in the resultant partitions. This information-based approach reduces the perceived number of tests needed for object classification and results in a simple tree.

Let \( D \) be a set of web log dataset samples with their corresponding labels. If there are \( k \) classes and the training set contains \( d_i \) samples of class \( C \) and \( s \) is the total number of samples in the web log dataset. Expected information needed to classify a given samples is calculated by Equation 4.1

\[
C(D_1, D_2, \ldots, D_k) = \sum_{i=1}^{k} \frac{D_i}{D} \log_2 \left( \frac{D_i}{D} \right)
\]  

(4.1)

4.3.2 Induction Based Decision Rule Algorithm

<table>
<thead>
<tr>
<th><strong>Input:</strong></th>
<th>Training samples, represented by Discrete attributes; Set of candidate attributes, Attribute-list</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Output:</strong></td>
<td>Decision rules</td>
</tr>
<tr>
<td></td>
<td><strong>Create</strong> a node P;</td>
</tr>
</tbody>
</table>

If samples are entire same cluster P, then

<table>
<thead>
<tr>
<th><strong>Return</strong></th>
<th>P as a leaf node labeled with the cluster L;</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>End If</strong></td>
<td></td>
</tr>
</tbody>
</table>
If attribute list is empty then

Return P as a leaf node labeled with the most common cluster in samples (majority voting)

Select test attribute, the attribute among attribute-list with the highest information gain ratio;

Label node P with test-attribute;

End If

For each known value $t_i$ of test-attribute

Grow a branch from node P for the condition test-attribute = $t_i$;

Let $d_i$ be set of samples in samples for which test-attribute = $a_i$;

If $d_i$ is empty

Attach a leaf labeled with the most common clusters in samples;

Else

Attach the node returned by generate decision rule

Update rules into knowledgebase

End if

End for

Information Gain

A feature $Q$ with values $\{q_1,q_2,\ldots,q_n\}$ can divide the web log data training set into $n$ subsets $\{d_1,d_2,\ldots,d_n\}$ where $d_j$ is the subset which has the value $q_j$ for feature $Q$. Furthermore let $D_j$ contain $D_{ij}$ samples of class $i$. Entropy of the feature $Q$ is Equation 4.2.

$$E(Q) = \sum_{j=1}^{n} \frac{(D_{ij} + \ldots + D_{kj})}{D \times C(D_{ij} + \ldots + D_{kj})}$$  \hspace{1cm} (4.2)
Information gain for Q is calculated as Equation 4.3.

\[
\text{Gain}(Q) = \sum_{i=1}^{n} \left| R_i \right| / |R| \times \log_2 \left( \frac{|R_i|}{|R|} \right)
\]  

(4.4)

In experiment, information gain is obtained for cluster labels by employing a binary discrimination for each cluster following the navigation pattern as mentioned in the rules. That is, for each cluster, a dataset instance is considered in-cluster, if it has the same label: outclass, if it has a different label. As a result when compared to evaluating single information gain as a general evaluation on the relevance of the feature for every cluster, the information gain is measured for each cluster. Thus, this advantage makes the feature to difference the given cluster from other clusters.

The notion of information gain established earlier tends to favor attributes that have a greater number of values. For example if an attribute A has the distinct value for each record, then Info (A, R) is 0, thus Gain (A, R) is maximal. To compensate for this, it was suggested to use the following ratio instead of gain.

SplitInfo is the information due to the split of R on the basis of the value of the navigation pattern followed by a visitor A, which is defined by Equation 4.4.

\[
\text{SplitInfo}(X) = \sum_{i=1}^{n} \left| R_i \right| / |R| \times \log_2 \left( \frac{|R_i|}{|R|} \right)
\]  

(4.4)

and Gain Ratio is then calculated by Equation 4.5

\[
\text{GainRatio}(A, R) = \frac{\text{Gain}(A, R)}{\text{SplitInfo}(A, R)}
\]  

(4.5)
The gain ratio, expresses the proportion of useful information generation split, i.e., that appears helpful for classification. If the split is nearer to the trivial value then the split information is too small and as a result this cannot be used further. To provide solution to the above problem, the criteria for gain ratio is selected in such a way that the ratio gain to splitinfo obtained is to be of higher value. This is designed in such a way that the information gain must be large, at least as great as the average gain over all tests examined.

4.4 EXPERIMENTAL EVALUATION OF DECISION BASED INDUCTION RULE MINING

This section discusses experimental performance of proposed decision based induction rule mining model. For experimental purpose, web log data WebLog Dataset is chosen from http://www.race.u-tokyo.ac.jp/~uchida/blogdata/. The WebLog dataset comprises of the time, IP address, session id, page request URL and referrer. In addition to this, the WebLog data have a generated session ID and so the identification of users was relatively easy (Each sequence of pages was treated with the same ID as one session.)

The whole dataset consists of 522 410 sessions, of which 318 523 only contain a single page visit. In the following, dataset is reduced to the 203 887 sessions that contain at least two page visits. The average length of these sessions was 16; In addition to the log data, following information are considered: table shop listed all e-shops (7 entries), table category listed general product categories (65 entries), table product contained product subcategories (157 entries) and table brand listed all sold brands (197 entries).

The experimental evaluation was conducted using WebLog dataset. The data is in the original arff format used by Weka tool. The characteristics
of the dataset used are given in table 4.1. Induction based decision rule algorithm is used for User Modeling in Web Usage Mining System.

Ultimately with the navigation pattern or ClickStream followed by a specified set of visitors it can determine what might be the best site followed by the visitor. With this information obtained from the social network, the organizations benefit by deriving the decision making process and changes to be made accordingly to increase the marketing volume. Decision trees systems are incorporated in product-selection systems offered by many vendors. They are great for situations in which a visitor comes to a Web site with a particular need. But once the decision has been made, the answers to the questions contribute little to targeting or personalization of that visitor in the future.

Figure 4.4 shows the number of rules mined based on the number of clickstreams followed by the visitors using WebLog dataset from the values given in Table 4.2. Figure 4.4 depicts the number of rules mined based on the number of navigation pattern or ClickStream followed. As the number of ClickStreams increases, the number of rules being mined also gets increased. Compared with the existing EM algorithm, proposed Induction based decision rule algorithm mines the rule better. This is because based on the induction mining and number of clickstream followed by a set of visitors the decision rule is formed accordingly. At the same time, the user ID is also used to derive the navigation pattern followed by the specified set of visitors resulting in the higher number of rule generated. The variance achieved using Induction Rule Mining algorithm is 10-15% higher than the existing EM algorithm.
Table 4.1 Sample WebLog dataset used in the experiments

<table>
<thead>
<tr>
<th>User IDs</th>
<th>URIs</th>
<th>Timestamp</th>
</tr>
</thead>
<tbody>
<tr>
<td>146390</td>
<td><a href="http://sugamo.jugem.cc?eid=268">http://sugamo.jugem.cc?eid=268</a></td>
<td>2004-10-14 00:00:00</td>
</tr>
<tr>
<td>146540</td>
<td><a href="http://sugamo.jugem.jp?eid=333">http://sugamo.jugem.jp?eid=333</a></td>
<td>2004-10-13 00:00:00</td>
</tr>
<tr>
<td>146541</td>
<td><a href="http://redkyudan.jugem.jp?eid=372">http://redkyudan.jugem.jp?eid=372</a></td>
<td>2004-10-12 00:00:00</td>
</tr>
<tr>
<td>146538</td>
<td><a href="http://brownshoes.jugem.cc?eid=171">http://brownshoes.jugem.cc?eid=171</a></td>
<td>2008-10-1 00:00:00</td>
</tr>
<tr>
<td>146548</td>
<td><a href="http://ersamo.jugem.cc?eid=468">http://ersamo.jugem.cc?eid=468</a></td>
<td>2008-10-3 00:00:00</td>
</tr>
<tr>
<td>146565</td>
<td><a href="http://sapamo.jugem.cc?eid=258">http://sapamo.jugem.cc?eid=258</a></td>
<td>2008-10-30 00:00:00</td>
</tr>
<tr>
<td>146555</td>
<td><a href="http://sertamo.jugem.cc?eid=288">http://sertamo.jugem.cc?eid=288</a></td>
<td>2008-10-23 00:00:00</td>
</tr>
<tr>
<td>146572</td>
<td><a href="http://sewfgamo.jugem.cc?eid=258">http://sewfgamo.jugem.cc?eid=258</a></td>
<td>2008-10-12 00:00:00</td>
</tr>
</tbody>
</table>

Table 4.2 Number of rules mined on WebLog dataset

<table>
<thead>
<tr>
<th>Number of Click Streams</th>
<th>Number of Rules Mined</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Induction Rule Mining Algorithm</td>
</tr>
<tr>
<td>50</td>
<td>32</td>
</tr>
<tr>
<td>100</td>
<td>37</td>
</tr>
<tr>
<td>150</td>
<td>39</td>
</tr>
<tr>
<td>200</td>
<td>46</td>
</tr>
<tr>
<td>250</td>
<td>53</td>
</tr>
<tr>
<td>300</td>
<td>59</td>
</tr>
<tr>
<td>350</td>
<td>62</td>
</tr>
</tbody>
</table>
Figure 4.4 Measure of number of rules mined

Table 4.3 Decisive rules

<table>
<thead>
<tr>
<th>Number of Pages Visited</th>
<th>Induction Rule Mining Algorithm</th>
<th>EM Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>45</td>
<td>35</td>
</tr>
<tr>
<td>10</td>
<td>52</td>
<td>43</td>
</tr>
<tr>
<td>15</td>
<td>58</td>
<td>50</td>
</tr>
<tr>
<td>20</td>
<td>63</td>
<td>55</td>
</tr>
<tr>
<td>25</td>
<td>65</td>
<td>62</td>
</tr>
<tr>
<td>30</td>
<td>68</td>
<td>64</td>
</tr>
<tr>
<td>35</td>
<td>74</td>
<td>68</td>
</tr>
<tr>
<td>40</td>
<td>78</td>
<td>70</td>
</tr>
<tr>
<td>45</td>
<td>80</td>
<td>72</td>
</tr>
</tbody>
</table>
Table 4.3 and Figure 4.5 show the number of decisive rules mined according to the number of pages visited by a visitor. The decisive rules mined are proportionate to the number of pages visited by the visitor based on the ClickStream followed by the visitor. When the number of pages visited by the visitor is increased, the decisive rule generated also gets increased. This is because according to the gain ratio criterion, higher the number of pages visited by the visitor, higher the decisive rules generated. Comparing with the existing EM algorithm, proposed Induction based decision rule algorithm performs well. The variance achieved using Induction Rule Mining algorithm is 8-15% higher than the EM algorithm.

Table 4.4 gives the comparative values of EM algorithm and Induction based decision rule algorithm based on the number of pages visited.
Table 4.4 Execution time

<table>
<thead>
<tr>
<th>Number of Pages Visited</th>
<th>Execution Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Induction Rule Mining Algorithm</td>
</tr>
<tr>
<td>5</td>
<td>45</td>
</tr>
<tr>
<td>10</td>
<td>52</td>
</tr>
<tr>
<td>15</td>
<td>60</td>
</tr>
<tr>
<td>20</td>
<td>65</td>
</tr>
<tr>
<td>25</td>
<td>70</td>
</tr>
<tr>
<td>30</td>
<td>75</td>
</tr>
<tr>
<td>35</td>
<td>77</td>
</tr>
</tbody>
</table>

Figure 4.6 Measure of execution time

Figure 4.6 gives the comparative study of EM algorithm and Induction based decision rule algorithm based on the Execution time for WebLog dataset. Execution time is measured in terms of seconds (s). Induction based decision rule algorithm takes lesser time to execute when
compared with the EM algorithm. This is because using the information gain the induction based decision rule algorithm predicts the occurrences of events using the ClickStream and with the navigational pattern followed by the visitors obtained using the user ID in lesser time when compared to the EM algorithm. The variance achieved using Induction based decision rule algorithm 5-8% lesser than the existing EM algorithm.

4.5 SUMMARY

In this chapter, the proposed web usage knowledge discovery system introduces a technique of decision rule induction to induce knowledge that can facilitate the web log dataset in the web usage mining. Induction decision rule mining is proposed in this chapter for web log data. Proposed Induction based decision rule model is exploited for generating inferences and implicit hidden behavioral aspects in the web usage mining which investigates at the web server and client logs.

The induction process is based on the decision tree model. The decision based rule induction mining combines a fast decision rule induction algorithm and a method for converting a decision tree to a simplified rule set, however still rationally equal to the original tree. It works on the assumption that the web log available data is considered for processing as data and with the help of the knowledge repository large amount of decision rules are produced. Proposal system thus limits the number of rules by inducing only rules that are relevant to the user’s need based on the ClickStream and navigational pattern followed by the visitor. Relevancy is guided by query predicates and the user ID.

The next chapter introduces the Integrated clustering and rule induction mining (ICRIM) in the documentation and presents how it is used for web and knowledge discovery.