CHAPTER 5

INTEGRATED CLUSTERING AND RULE INDUCTION MINING (ICRIM) FRAMEWORK FOR WEB USAGE KNOWLEDGE DISCOVERY

5.1 INTRODUCTION

Web usage mining applies data mining techniques to determine navigation patterns from the web log data, in order to recognize and provide the requirements of Web-based applications. This chapter presents a new framework, Integration of Clustering and Rule Induction Mining (ICRIM) that assess the performance of web usage knowledge discovery system using web log data. ICRIM framework incorporates the clustering model and Induction based decision rule model and applied to the web log data.

ICRIM framework uses web log data for users’ sessions performed in two step process as shown in Figure 5.1.

- Clustering
- Rule Induction Mining

Initially, primary cluster centers are selected based on statistical mode based computation. This helps to allow the iterative algorithm to meet a better local minimum. The proposed method is experienced with the web log data obtained from Internet Traffic Archive and demonstrates that refined original starting points and post processing modification of clusters definitely
direct to better solutions. The technique is scalable together with a scalable clustering algorithm to deal with the large-scale clustering problems in web data mining.

**Figure 5.1 Process in ICRIM framework**

The clustering method that is used in this work is the EM Clustering model which is properly described in chapter 3. Before clustering, the step of preprocessing is required to prepare the data. This needs, initially, discovering the components from each dataset ranging from the user Id to the pages visit to timestamp involved during a session. These are again entered into the cluster and then handling component activity measures for each reading dataset. In order to derive the distance of components in a cluster from each other, the pre-clustering is executed once based on the condition-mean measures. Once clusters are formed up and based on the clustering outcome, ICRIM produces induction rules using decision tree model.

The second step describes a technique for remodeling decision trees to small sets of induction rules, a general formalism for expressing knowledge
in Web Usage Knowledge Discovery systems. The technique makes use of the web log dataset from which the decision tree was produced based on the clickstream followed by the visitor, initially to simplify and evaluate the reliability of individual rules mined from the tree, and consequently to refine the collection of rules as a whole. The final set of induction rules is typically both simpler than the decision tree from which it was attained, and more perfect when classifying hidden cases. The detailed description of ICRIM framework is explained in the following sections.

5.2 FRAMEWORK OF ICRIM FOR WEB USAGE MINING

Web Usage Mining analyses the navigation pattern followed by the visitors of web sites in order to get an improved understanding of the users’ interests and the possibility of the page being visited by the other users based on the hit ratio generated. This information is especially valuable for E-Business sites in order to achieve improved customer satisfaction.

The proposed ICRIM Framework (Figure 5.2) is based on a web usage knowledge discovery system to provide scalability and allow for flexible multi-dimensional analysis and data mining. Moreover, for flexible multi-dimensional analysis they are coupled with different types of business warehouses, e.g. for building efficient customer relationship management. A prototype implementation is developed for web log data using commercially available tools.

The proposed web usage knowledge discovery system helps users to quickly find the information they want or find interesting. At the same time, the web usage knowledge framework discovery system also allows website owners for optimization of the website, maximize the level of web user satisfaction and optimize the costs of content management. An overview of the proposed web usage knowledge discovery system is shown below.
ICRIM framework has dynamically determined inductive rules based on clustering model and decision based inductive rule model.

With the increasing surge in the web which has also been a relentless generator of data appears in different format. It ranges from Web content data that stands as the basis of certain web documents, to the daily traces left by visitors of the website as they surf through a Website, also known as Web usage data. This data often yields interesting knowledge or patterns such as Web user access trends or profiles. The web usage mining is used to accomplish different objectives, including support for customer relationship management, and the main factor of personalizing the user’s experience on a Website.

Figure 5.2 ICRIM frameworks for web usage mining
In this work, ICRIM framework is used to extract usage patterns from Web log data. In ICRIM framework, clustering model segments user sessions into clusters or profiles that can later form the basis for personalization. ICRIM is based on using Induction rule mining as the basis for modeling.

**5.2.1 Discovering Web Data Clusters using ICRIM**

The proposal work presents the concentric factors involved in the designing of web usage mining and its different ranges of practical applications. Further a novel framework called Integration of Clustering and Rule Induction Mining is presented.

**Figure 5.3 Framework of ICRIM**

ICRIM framework analyzes the navigation pattern followed by the visitor using the clickstream and optimally segregates similar user interests.
The proposed approach is compared with hierarchical patterns (to discover patterns) and several function approximation techniques.

The ICRIM framework optimizes fuzzy clustering algorithm using an evolutionary algorithm (Figure 5.3). The raw data obtained from the log files are cleaned and pre-processing is applied followed by the application of fuzzy C means algorithm in order to identify the number of clusters. The developed clusters of data are fed to an EM model to analyze the interest of the particular page based on the navigation pattern followed by the visitors using clickstream.

The if-then rule structures are learned using an iterative learning procedure by applying evolutionary algorithm and the rule parameters are fine tuned using a back propagation algorithm. The optimization of clustering algorithm using back propagation method processes at a faster time scale in an environment which is decided based on the method being inferred and according to the problem environment.

Usually a number of cluster centers are randomly initialized and with the introduction of the decision rule induction algorithm an iterative approach is provided at the same time. The application of iterative approach using decision rule induction algorithm approximate the minimum objective function starting from a given position and leads to any of its local minima. No guarantee ensures that the decision rule induction algorithm converges to an optimum solution. The performance is very sensitive and highly depends on the initialization of cluster centers. With the initialization of cluster centers, an evolutionary algorithm is then applied to determine the optimal number of clusters and their cluster centers. The algorithm is initialized in such a way that the initial values are constrained to be within the space defined by the vectors to be clustered. In the ICRIM approach, the decision
rule induction algorithm is optimized jointly with the EM algorithm into a single global search.

ICRIM framework is used for clustering in the context of mixture models. This method estimates missing parameters of probabilistic models. The ICRIM framework is an optimization approach, which had given the initial approximation of the cluster parameters, iteratively performs two steps, i.e., the expectation step which computes the values expected for the cluster probabilities, and the maximization step computes the distribution parameters and their likelihood in the given data. The two steps are further iterated until the parameters are optimized to obtain a fixed point or until the log-likelihood function that evaluates the quality of clustering, reaches to its maximum level. To simplify the discussion, first the EM algorithm is described briefly.

5.2.2 EM Model

The EM algorithm for clustering is used to identify the maximum likelihood parameters of pages visited by a user and the count value is determined by the hit ratio produced for the specific page for which the equations cannot be solved directly. These models involve latent variables in addition to unknown parameters including supporting evidences such as the last time visited by the user and known data observations referred from the hit ratio. This either observes the missing values between the data, or the model is formed simply with the assumption of the presence of additional unobserved data points. Finding the maximum likelihood solution requires observing the derivatives of the likelihood function in accordance with the unknown values. These unknown values consist of both the parameters and the latent variables that solve the resulting equation simultaneously. In statistical models with latent variables, this usually is not possible. But in these statistical models, the results obtained are in the form of the interlocking equations. These solutions for these parameters need the values of the latent
variables and vice-versa. But at the same time, mere substitution of one set of equations into the other only results in an unsolvable equation.

The EM algorithm proceeds from the observation that one way to solve these two sets of equations numerically is to simply pick arbitrary values for one of the two web log datasets of unknowns, use them to find the other set and, then use those other values to find a better estimate of the first set, and then keep alternating between the two until both the resulting values converge to a fixed point. It is not obvious that this will work at all, but in fact it is proven that it does, and that the value is a local maximum of the likelihood function.

Given a statistical model consisting of a set $X$ of observed data, a set of unobserved latent data or missing values $Z$, and a vector of unknown parameters $\theta$, along with a likelihood function is in Equation 5.1

$$L(\theta; X, Z) = p(X, Z | \theta) \quad (5.1)$$

The maximum likelihood estimate (MLE) of the unknown parameters is determined by the marginal likelihood of the observed data in Equation 5.2

$$L(\theta; X) = p(X | \theta) = \sum_{Z} p(X, Z | \theta) \quad (5.2)$$

However, this quantity is often intractable

The EM algorithm seeks to find the MLE of the marginal likelihood by iteratively applying the following two steps:
Expectation step (E step): The expected value of the log likelihood function is calculated with respect to the conditional distribution of \( Z \) given \( X \) under the current estimate of the parameters \( \theta^{(t)} \) in Equation 5.3:

\[
Q(\theta | \theta^t) = \mathbb{E}_{z|X,\theta^t(t)}[L(\theta; X, Z)]
\]  

(5.3)

Maximization step (M step): The parameter that maximizes this quantity is found in Equation 5.4:

\[
\theta^{(t+1)} = \arg\max_{\theta} Q(\theta | \theta^t)
\]  

(5.4)

It is noted that in typical models to which EM is applied:

1. The observed data points \( X \) are either discrete (user ID for a specific session following certain kind of navigation pattern) or continuous (timestamp obtained from the web log dataset). There may in fact be a vector of observations associated with each data point.

2. The missing values (aka latent variables) \( Z \) are either in the form of discrete, measured from a fixed number of values with one latent variable per observed data point.

3. The parameters are continuous, and are of two kinds: The parameters are also in the form of continuous that are either associated with all the data points and parameters that are associated with a specific value of a latent variable

However, it is possible to apply EM to other sorts of models.
The motivation is as follows. If the value of the navigation patterns NP is known, usually the value of the latent variables and clickstream CS is found by maximizing the log likelihood over all possible values of CS, either simply by iterating over CS or through an algorithm such as the Viterbi algorithm for hidden Markov models. Conversely, if the value of the latent variables CS is known, then an estimate of the parameters NP is found fairly grouping the observed data points according to the value of the associated latent variable and averaging the values, or some function of the values, of the points in each group. This suggests an iterative algorithm, in the case where both NP and CS are unknown:

1. First, initialize the parameters NP to some random values.
2. Compute the best value for CS given these parameter values.
3. Then, use the just-computed values of CS to compute a better estimate for the parameters NP. Parameters associated with a particular value of CS will use only those data points whose associated latent variable has that value.
4. Iterate steps 2 and 3 until convergence.

The algorithm as just described above follows a local minimum of the cost function and that is referred to as the hard EM. The k-means algorithm is an example of this class of algorithms.

The resulting algorithm is commonly called as the soft EM that is normally associated and used with EM. The number of iterations used to evaluate these weighted average are referred to as the soft counts. The probabilities computed for $Z$ are posterior probabilities and are what is computed in the E step. The soft counts used to compute new parameter values are what are computed in the M step.
5.3 DECISION INFERENCES FROM ICRIM ON WEB USAGE DATA

Web usage mining is attained initially by providing visitors traffic information based on Web log dataset. Web log dataset were used first by the webmasters and system administrators for the reasons of how much traffic they are receiving, how many requests fail, and what kind of errors are being produced, etc. However, Web log dataset record and trace the visitors’ navigation pattern. Web log dataset is one way to gather Web traffic data.

DECISION RULES

Rule1: If (VisitedPages <20 && Count [IPAddr] <5) = “Not Preffered ClickStream”

Rule2: If (VisitedPage <30 && Count [IPAddr] >5) = “Either Preffered or Not Preffered ClickStream”

Rule3: If (VisitedPage >30 && Count [IPAddr] <5) = “Either Preferred or Not Preffered ClickStream”

Rule4: If (VisitedPage >30 && Count [IPAddr] >5) = “Preffered ClickStream”

Once the web traffic data is received from the web log dataset, it is aggregated with other relational databases, over which the data mining models are implemented. Through the data mining technique, Induction based decision rule model, visitors’ navigational behavior and clickstream patterns are identified and interpreted. 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 class of the examples. The instances are classified by sorting them down the tree from the root node to leaf node. To make the discussion simple, the Induction based decision rule algorithm that derives the inferences of the user is given below.

```
Input: Training samples, Set of candidate attributes, Attribute-list A_i where i = 1, 2, 3, 4 consisting of rules R1, R2, R3 and R4, d_i be set of samples in samples for which test-attribute = a_i;
Output: Decision rules and Create a node P;
Step 1: If (training samples belong to same cluster P) then
  Return P as a leaf node labeled with the cluster L
End If
Step 2: If A_i (attribute list is empty) then
  Return P as a leaf node labeled with the most common cluster in samples (majority voting) in A_i
  Select test attribute, among attribute-list A_i with highest information gain ratio
  Label node P with test-attribute
End If
Step 3: For each known value t_i of test-attribute
  Grow a branch from node P for the condition test-attribute= t_i
Step 4: If (d_i is empty) then
  Attach a leaf labeled with the most common clusters in samples
Else
  Attach the node returned by using decision rule
Update rules into knowledgebase
End if
Step 4.6 End for
Step 4.6 End for
```

The induction based decision rule algorithm is briefed in algorithm 2 which consists of four parts. The first part, checks for the training samples.
If it belongs to the same cluster then the node P is returned as the leaf node with the cluster L. The second part identifies if the attribute list is empty and if so P is returned as a leaf node followed by the selection of test attribute with the highest information gain ratio. Next for each known value of test attribute $t_i$, a branch is grown from the node P. Finally, if $d_i$ is empty, then using the decision rule the node is attached and rules are updated into knowledge base.

5.4 EXPERIMENTAL EVALUATION OF ICRIM

This section depicts the experimentations, designed to evaluate the result quality of Integration of Clustering and Rule Induction Mining (ICRIM) Framework. The experimental evaluation was conducted using WebLog Dataset chosen from http://www.race.u-tokyo.ac.jp/~uchida/blogdata/. The data was in the original arff format used by Weka tool. ICRIM framework is used for User Modeling in Web Usage Mining System. The performance is executed on Weka tool, data mining software for estimation part of the system.

In this research, the statistical/text data produced by the log file analyzer were utilized. Choosing helpful data is a significant task in the data pre-processing task. After various preface examination, the statistical data encompassing of domain byte requests, hourly page requests and daily page requests were chosen as center of the cluster models for discovering Web users' usage patterns. It is also significant to eliminate the unrelated and piercing data in order to construct a defined model.

The further input index number was also incorporated to differentiate the time series of the data. The most freshly retrieved data was indexed at the top whereas the least recently retrieved data was located at the bottom. Moreover the inputs or quantity of requests and quantity of pages (bytes) and index number, the cluster data offered by the clustering algorithm
was also employed as a further input variable. The data were re-indexed depending on the cluster information. The task is to forecast the Web traffic quantity on an hourly and every day basis. The results of these experiments are illustrated in the next subsection.

**Results and Discussion on ICRIM**

In this investigation, data conversion is performed using two steps before employing induction based decision rule algorithm. There are around 800 URLs WebLog Dataset. Assigning each URL address in the session to sequential numeric values is the earliest step. It is not possible to give 800 attributes to Weka so for minimizing the number of attributes; each eight sequence of attributes is assigned to one attribute based on EM algorithm. The performance of ICRIM is evaluated in terms of decisive rules mined, mean absolute error and Root mean square error. Table 5.1 represents the sample values of Number of rules mined using ICRIM Framework on WebLog Dataset.

**Table 5.1 Number of rules mined using ICRIM on WebLog dataset**

<table>
<thead>
<tr>
<th>Number of Users (Pages Visited)</th>
<th>Number of Rules Mined</th>
<th>Induction Based Decision Rule Method</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ICRIM</td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>38</td>
<td>29</td>
</tr>
<tr>
<td>400</td>
<td>45</td>
<td>32</td>
</tr>
<tr>
<td>600</td>
<td>52</td>
<td>38</td>
</tr>
<tr>
<td>800</td>
<td>58</td>
<td>42</td>
</tr>
<tr>
<td>1000</td>
<td>63</td>
<td>45</td>
</tr>
<tr>
<td>1200</td>
<td>67</td>
<td>52</td>
</tr>
<tr>
<td>1400</td>
<td>69</td>
<td>55</td>
</tr>
<tr>
<td>1600</td>
<td>76</td>
<td>59</td>
</tr>
<tr>
<td>1800</td>
<td>83</td>
<td>63</td>
</tr>
</tbody>
</table>
Figure 5.4 illustrates the number of rules mined using ICRIM with Weblog dataset. As the number of pages visited by the user increases, the rules mined using ICRIM also gets increased. Compared to the Induction based decision rule method the rules mined using ICRIM is higher and the variance achieved is 20-26% higher than rules mined using induction based decision rule method. This is because ICRIM mines the rule by integrating both the cluster and rule induction method for the pages visited resulting in higher rules being mined which in a way help the organization to increase the volume of sale by applying ICRIM.

![Figure 5.4 Measure of rules mined using ICRIM](image)

Table 5.2 Mean Absolute Error using ICRIM on WebLog dataset

<table>
<thead>
<tr>
<th>Decisive Rules</th>
<th>Mean Absolute Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ICRIM</td>
</tr>
<tr>
<td>5</td>
<td>0.5254</td>
</tr>
<tr>
<td>10</td>
<td>0.5471</td>
</tr>
<tr>
<td>15</td>
<td>0.5669</td>
</tr>
<tr>
<td>20</td>
<td>0.5718</td>
</tr>
<tr>
<td>25</td>
<td>0.5971</td>
</tr>
<tr>
<td>30</td>
<td>0.6287</td>
</tr>
<tr>
<td>35</td>
<td>0.6413</td>
</tr>
</tbody>
</table>
In Table 5.2 Mean Absolute Error measures the average magnitude of the errors observed based on the navigation pattern and clickstreams of the visitors using WebLog dataset. The mean absolute error measures the accuracy for continuous variables. The Mean Absolute Error is the average over the verification sample of the absolute values of the differences between forecast and the corresponding observation. The Mean Absolute Error is a linear score which means that all the individual differences are weighted equally in the average. Figure 5.5 shows that the proposed ICRIM Framework has low Mean Absolute Error compared with existing Induction based decision rule method. The variance achieved using ICRIM is 5-10% lesser than the induction based decision rule method.

**Figure 5.5 Measure of mean absolute error**
Table 5.3 Root mean square error using ICRIM with WebLog dataset

<table>
<thead>
<tr>
<th>Discovered Rules</th>
<th>Root Mean Square Error (%)</th>
<th>Induction Based Decision Rule Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0.5784</td>
<td>0.8523</td>
</tr>
<tr>
<td>10</td>
<td>0.5894</td>
<td>0.8632</td>
</tr>
<tr>
<td>15</td>
<td>0.6004</td>
<td>0.8876</td>
</tr>
<tr>
<td>20</td>
<td>0.6154</td>
<td>0.8965</td>
</tr>
<tr>
<td>25</td>
<td>0.6198</td>
<td>0.9234</td>
</tr>
<tr>
<td>30</td>
<td>0.6269</td>
<td>0.9767</td>
</tr>
<tr>
<td>35</td>
<td>0.6321</td>
<td>0.9965</td>
</tr>
</tbody>
</table>

Figure 5.6 Measure of root mean square error

In Table 5.3 Root Mean Square Error is a quadratic scoring rule which measures the average magnitude of the error using WebLog dataset. The equation for the Root Mean Square Error is given in both of the
The root mean square error is the difference between forecast and corresponding observed values are squared and then averaged over the sample. Finally, the square root of the average is taken. Since the errors are squared before they are averaged, the Root Mean Square Error gives a relatively high weight to large errors. This means the Root Mean Square Error is most useful when large errors are particularly undesirable. From Figure 5.6, it is evident that the Root Mean Square Error is relatively low in ICRIM when compared to that of the induction based rule decision method using WebLog dataset. The variance achieved using ICRIM with WebLog dataset is 20-30% lesser than the induction based rule decision method.

Table 5.4 represents the Rule mining Accuracy (%) for the different number of attributes.

<table>
<thead>
<tr>
<th>No. of attributes</th>
<th>Rule mining accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ICRIM</td>
</tr>
<tr>
<td>2</td>
<td>97.7</td>
</tr>
<tr>
<td>4</td>
<td>97.2</td>
</tr>
<tr>
<td>6</td>
<td>97.1</td>
</tr>
<tr>
<td>8</td>
<td>96.4</td>
</tr>
<tr>
<td>10</td>
<td>96.1</td>
</tr>
<tr>
<td>12</td>
<td>95.2</td>
</tr>
<tr>
<td>14</td>
<td>94.4</td>
</tr>
<tr>
<td>16</td>
<td>94.2</td>
</tr>
</tbody>
</table>
In general, if the set of rules is too small, some effective rules may be missed. On the other hand, if the rule set is too large, the training data set may be overfit. From the figure 5.7, one can see that the peaks of rule mining accuracy are achieved at the beginning of the curves. That is, number of attributes increases, accuracy falls down. However, according to the experimental results, Proposed ICRIM framework achieves better accuracy i.e., 7% higher than CBA and 14% higher than C4.5. Table 5.5 gives the Mean absolute error (MAE) of different methods such as ICRIM, CBA, C4.5.

On the whole, the proposed ICRIM performs better in the evaluation of web usage knowledge discovery system using the weblog dataset. From the Performance results, ICRIM gives 5 to 10 % better results in Mean Absolute Error and 20 to 30 % better results in Root Mean Square Error.
5.5 PARAMETRIC EVALUATION OF ICRIM ON WEBLOG DATASET

Data sets generally used in testing clustering comprise a hidden variable, which specifies the class with which every tuple is associated. Whereas there is no assurance that any known classification corresponds to an optimal clustering, it is however enlightening to compare clustering with pre-specified classifications of tuples.

Table 5.5 Time Taken to Generate Induction Rules

<table>
<thead>
<tr>
<th>No. of Instances</th>
<th>Time Taken to Generate Induction Rules (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ICRIM</td>
</tr>
<tr>
<td>20000</td>
<td>88</td>
</tr>
<tr>
<td>40000</td>
<td>156</td>
</tr>
<tr>
<td>60000</td>
<td>198</td>
</tr>
<tr>
<td>80000</td>
<td>255</td>
</tr>
<tr>
<td>100000</td>
<td>304</td>
</tr>
<tr>
<td>120000</td>
<td>365</td>
</tr>
<tr>
<td>140000</td>
<td>435</td>
</tr>
<tr>
<td>160000</td>
<td>499</td>
</tr>
<tr>
<td>177811</td>
<td>575</td>
</tr>
</tbody>
</table>

To perform this, we use performance measures from Information Retrieval, specifically Cluster Precision. Cluster precision is defined as the ratio of relevant retrieved items consisting of pages visited, hit ratio, navigation pattern followed by sequence of users and retrieved items in the web usage data.

To compare with the full model size (i.e., model size = no. of items) the same test was also run, considering all similarity values and the
best k was picked for prediction generation. The entire process was repeated for three different x values (training/test ratios).

Table 5.5 measures and generates values, the Time Taken to Generate Induction Rules of the three methods (ICRIM, CBA and C4.5) with respect to different instances of the web access sequences.

![Figure 5.8 Time Taken to Generate Induction Rules](image)

This experiment (as shown in Figure 5.8), have measured the Time Taken to Generate Induction Rules of the three methods (ICRIM, CBA and C4.5) with respect to different instances of the web access sequences. The experimental results in Figure 5.8 have shown that the ICRIM has better result than the CBA and C4, while the size of input database becomes larger.

### 5.6 SUMMARY

This chapter has implemented a new framework, Integration of Clustering and Rule Induction Mining (ICRIM) that evaluate the performance of web usage knowledge discovery system with the help of WebLog dataset. ICRIM integrated clustering model and Induction based decision rule model.
Web data clusters have been discovered and using the web log dataset the navigation pattern and clickstreams are used to analyze the user behavior and optimally segregated similar user interests. The proposed web usage knowledge discovery system helps the users to quickly find the information they want or find interesting. Whatever the number of clients has, ICRIM has designed to provide the result in real time and generated inferences and implicit hidden behavioral aspects in the web usage mining to investigate at the web server and client logs. On the other hand, website owners were allowed to optimize the website, increase the satisfaction level of web user interaction and save on the costs of content management. ICRIM has dynamically determined inductive rules based on clustering model and decision based inductive rule model. The performance and results of ICRIM is discussed in detail in the following chapter.