Effectiveness Evaluation of Rule Based Classifiers for the Classification of Iris Data Set

C. Lakshmi Devasena, T. Sumathi, V.V. Gomathi and M. Hemalatha

Abstract--- In machine learning, classification refers to a step by step procedure for designating a given piece of input data into any one of the given categories. There are many classification problem occurs and need to be solved. Different types are classification algorithms like tree-based, rule-based, etc are widely used. This work studies the effectiveness of Rule-Based classifiers for classification by taking a sample data set from UCI machine learning repository using the open source machine learning tool. A comparison of different rule-based classifiers used in Data Mining and a practical guideline for selecting the most suited algorithm for a classification is presented and some empirical criteria for describing and evaluating the classifiers are given.

Keywords--- IRIS, Fuzzy Clustering, DTNB Classifier, RIDOR Classifier, Conjunctive Rule Classifier

I.   INTRODUCTION

IN machine learning, classification refers to an algorithmic process for designating a given input data into one among the different categories given. An example would be a given program can be assigned into "private" or "public" classes. An algorithm that implements classification is known as a classifier. The input data can be termed as an instance and the categories are known as classes. The characteristics of the instance can be described by a vector of features. These features can be nominal, ordinal, integer-valued or real-valued. Many data mining algorithms work only in terms of categorical data and require that real-valued or integer-valued data be converted into groups.

Classification is a supervised procedure that learns to classify new instances based on the knowledge learnt from a previously classified training set of instances. The equivalent unsupervised procedure is clustering. It groups the input data into classes based on inherent similarity measure. Clustering and Classification are examples of general problems of pattern recognition, which assigns some sort of output value to a given input value. In machine learning, classification systems induced from empirical data (examples) are first of all rated by their predictive accuracy. In practice, however, the interpretability or transparency of a classifier is often important as well. In this connection, rule-based classifiers enjoy great popularity, because rules can easily be understood by human beings

II.   LITERATURE REVIEW

In [1], the comparison of the performance analysis of Fuzzy C mean (FCM) clustering algorithm with Hard C Mean (HCM) algorithm on Iris flower data set is done and concluded Fuzzy clustering are suitable for handling the issues related to understandability of patterns, incomplete/noisy data, mixed media information and human interaction, and can provide approximate solutions faster. In [6], the issues of determining an appropriate number of clusters and of visualizing the strength of the clusters are addressed using the Iris Data Set.

A.   Data Set Used

IRIS flower data set classification problem is one of the novel multivariate dataset created by Sir Ronald Aylmer Fisher [3] in 1936. IRIS dataset consists of 150 instances from Iris setosa, Iris virginica and Iris versicolor. Length and width of sepal and petals is measured from each sample of three selected species of Iris flower. Four features were derived from each sample; they are the Sepal Length, Sepal Width, Petal Length and Petal Width with centimeter (cm) as their units. Based on the combination of the four features, the classification of the plant is made [4].

Methodology Applied

Different rule based Classifiers are used in this work to evaluate the effectiveness of those classifiers in a classification problem. The Classifiers applied are:

Conjunctive Rule Classifier

It is a decision-making rule in which the intending buyer assigns least values for a number of factors and discards any result which does not meet the bare minimum value on all of the factors i.e. a superior performance on one factor cannot recompense for deficit on another. Conjunctive rule uses the AND logical relation to correlate stimulus attributes.

Conjunctive rule is a simple well interpretable 2-class classifier [5]:

$$r_y(x) = \bigwedge_{j \in J} \left[f_j(x) \lesseqgtr_j \theta_j \right]$$

where fj(x) – features,

$J \subseteq \{1, \ldots, n\}$ - subset of features, not very big, usually $|J| \leq 7$,

C. Lakshmi Devasena, Department of Software Systems, Karpagam University, Coimbatore, E-mail: devaradhe2007@gmail.com
T. Sumathi, Department of Software Systems, Karpagam University, Coimbatore, E-mail: tsumathijk@gmail.com
V.V. Gomathi, Department of Software Systems, Karpagam University, Coimbatore.
M. Hemalatha, Department of Software Systems, Karpagam University, Coimbatore, E-mail: hema.bioinf@gmail.com

θj – thresholds,

⪋j – one of the signs ≤ or ≥,

y – the class of the rule.

If ry (x) = 1 then the rule r classifies x to the class y. All objects x such that ry (x) = 0 are not classified by ry.

This rule consists of antecedents "AND"ed together and the consequent (class value) for the classification/ regression. In this case, the consequent is the distribution of the available classes (or mean for a numeric value) in the dataset. If the test instance is not covered by this rule, then it's predicted using the default class distributions/value of the data not covered by the rule in the training data. This learner selects an antecedent by computing the Information Gain of each antecedent and prunes the generated rule using Reduced Error Pruning (REP) or simple pre-pruning based on the number of antecedents.

B. Decision Table Classifier

Two variants of decision table classifiers are available. The first classifier, called DTMaj (Decision Table Majority) returns the majority of the training set if the decision table cell matching the new instance is empty, i.e., it does not contain any training instances. The second classifier, called DTLoc (Decision Table Local), is a new variant that searches for a decision table entry with fewer matching attributes (larger cells) if the matching cell is empty. This variant therefore returns an answer from the local neighborhood.

C. DTNB Classifier

This is for building and using a decision table/naive bayes hybrid classifier [6]. At each point in the search, the algorithm evaluates the merit of dividing the attributes into two disjoint subsets: one for the decision table, the other for naive Bayes. A forward selection search is used, where at each step, selected attributes are modeled by naive Bayes and the remainder by the decision table, and all attributes are modeled by the decision table initially. At each step, the algorithm also considers dropping an attribute entirely from the model.

D. JRIP Classifier

This implements a propositional rule learner, Repeated Incremental Pruning to Produce Error Reduction (RIPPER), which is proposed by William W. JRip is an inference and rules-based learner (RIPPER) that tries to come up with propositional rules which can be used to classify elements.

E. NNGE Classifier

Non-Nested Generalized Exemplars (NNGE) is an algorithm introduced by Brent, 1995. It performs generalization by merging exemplars, forming hyper rectangles in attribute space that represent conjunctive rules with internal disjunction. The algorithm forms a generalization each time a new example is added to the database, by joining it to its nearest neighbor of the same class.

The algorithm learns incrementally by first classifying, then generalizing each new example. When classifying an instance, one or more hyper rectangles may be found that the new instance is a member of, but which are of wrong class. The algorithm prunes these so that the new example is no longer a member. Once classified, the new instance is generalized by merging it with the nearest exemplar of the same class, which may be a single instance or a hyper rectangle.

F. OneR Classifier

The OneR algorithm creates a single rule for each attribute of training data and then picks up the rule with the least error rate [7]. To generate a rule for an attribute, the most recurrent class for each attribute value must be established. The most recurrent class is the class that appears most frequently for that attribute value. A rule is a set of attribute values bound to their most recurrent class with which the attribute based on.

Pseudo-code for OneR algorithm is:

For each attribute *A*,

> For each value *VA* of the attribute, make a rule as follows:
> Add up how often each class appears
> Locate the most frequent class Cf
> Generate a rule when *A=VA*; class attribute value = *Cf*
> End For-Each
> Compute the error rate of all rules
> End For-Each
> Select the rule with the smallest error rate

The number of training data instances which does not agree with the binding of attribute value in the rule produces the error rate. OneR chooses the rule with the least error rate. If two or more rules have same error rate then the rule is selected at random.

G. PART Classifier

This is a class for generating a PART decision list. It uses separate-and-conquer approach and builds a partial C4.5 decision tree in each iteration and makes the "best" leaf into a rule.

H. RIDOR Classifier

RIpple-DOwn Rule learner first generates the default rule. The exceptions are generated for the default rule with the lowest (weighted) error rate. Then it generates the "best" exceptions for each exception. Thus it carries out a tree-like expansion of exceptions and its leaf has only default rule without exceptions.

Five inner classes are defined in this class.

Ridor_node class, which implements one node in the Ridor tree. It's basically built up of a default class and its exception rules.

RidorRule class, which implements a single exception rule using REP.

The rest of the three classes are only used in RidorRule namely Antd, NumericAntd and NominalAntd.

The abstract class Antd class has two subclasses, NumericAntd and NominalAntd, to implement the corresponding abstract functions. These two subclasses implement the functions related to an antecedent with a nominal attribute and a numeric attribute respectively.

I. ZeroR Classifier

ZeroR is a learner used to test the results of the other learners. ZeroR chooses the most common category all the time. ZeroR learners are used to compare the results of the other learners to determine if they are useful or not, especially in the presence of one large dominating category.

J. Comparison Criteria

The comparison of the results is made on the basis of the following criteria:

- *Classification Accuracy*

All classification result could have an error rate and from time to time it will either fail to classify correctly, or classify wrongly. So accuracy can be calculated as follows:

Accuracy = (Instances Correctly Classified / Total Number of Instances)*100 %   _____(1)

- *Mean Absolute Error*

Mean absolute error, MAE, is the average of the difference between predicted and actual value in all test cases; it is the average prediction error. The formula for calculating MAE is given in equation shown below:

$$MAE = (|a1 - c1| + |a2 - c2| + \ldots + |an - cn|) / n$$

$$\text{_____ (2)}$$

Assuming that the actual output is a, expected output is c.

- *Root Mean-Squared Error*

RMSE is frequently used measure of differences between values predicted by a model or estimator and the values actually observed from the thing being modeled or estimated. It is just the square root of the mean square error as shown in equation given below:

$$\sqrt{\frac{(a_1-c_1)^2 + (a_2-c_2)^2 + \ldots + (a_n-c_n)^2}{n}}$$

$$\text{_____ (3)}$$

The mean-squared error is one of the most commonly used measures of success for numeric prediction. This value is computed by taking the average of the squared differences between each computed value and its corresponding correct value.

The classification accuracy, mean absolute error and root mean squared error are calculated for each machine learning algorithm.

III. EXPERIMENTAL RESULTS

This work is performed using Machine learning tool to predict the effectiveness of all the rule- based classifiers. The performance of the various algorithms measured in Classification Accuracy, RMSE and MAE values as shown in Table 1 Comparison among classifiers based on the correctly classified instances are shown in Figure 1. Comparison among classifiers based on MAE and RMSE values are shown in Figure 2. The confusion matrix arrived for the classifiers are shown from Table 2 to Table 10. The overall ranking is done based on the classification accuracy, MAE and RMSE values

and it is given in Table 1. Based on the results arrived, NNGE Classifier has got the first position in ranking followed by OneR, RIDOR, JRIP, PART and other classifiers as shown in Table 1.

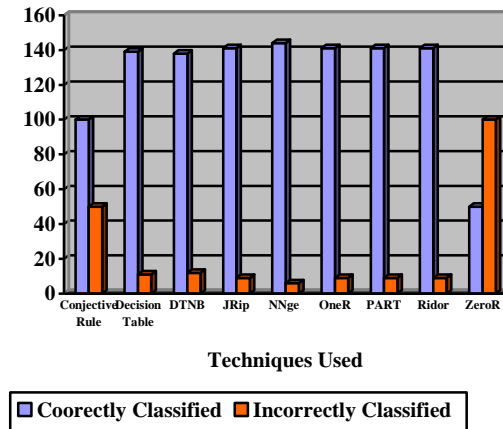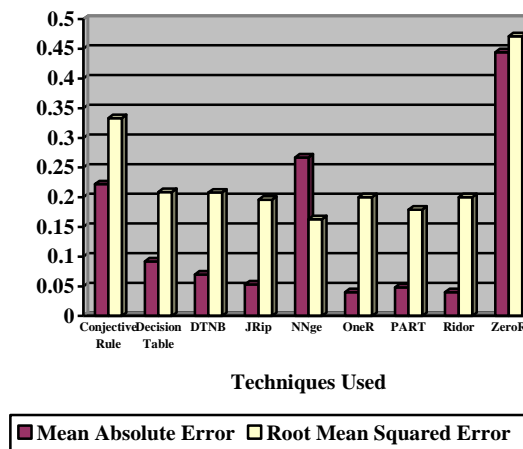Figure 1: Comparison based on Number of Instances Correctly Classified

Figure 2: Comparison based on MAE & RMSE Values.

Table 1: Overall Result of Rule Based Classifiers

Technique Used	Correctly Classified	Classification Accuracy (%)	MAE	RMSE	Time (in sec)	Accuracy Rank
Conjective Rule	100	66.67	0.222	0.333	0.02	8
Decision Table	139	92.67	0.092	0.209	0.03	6
DTNB	138	92.0	0.07	0.208	0.02	7
JRip	141	94.0	0.053	0.196	0.02	4
NNge	144	96.0	0.267	0.163	0.02	1
OneR	141	94.0	0.04	0.2	0	2
PART	141	94.0	0.048	0.179	0.03	5
Ridor	141	94.0	0.04	0.2	0.02	3
ZeroR	50	33.33	0.444	0.471	0	9

Table 2: Confusion Matrix for Conjunctive Rule Classifier

	a	b	c
a = Iris-setosa	50	0	0
b = Iris-versicolor	0	50	0
c = Iris-virginica	0	50	0

Table 3: Confusion Matrix for Decision Table Classifier

	a	b	c
a = Iris-setosa	50	0	0
b = Iris-versicolor	0	44	6
c = Iris-virginica	0	5	45

Table 4: Confusion Matrix for DTNB Classifier

	a	b	c
a = Iris-setosa	50	0	0
b = Iris-versicolor	0	44	6
c = Iris-virginica	0	6	44

Table 5: Confusion Matrix for JRIP Classifier

	a	b	c
a = Iris-setosa	50	0	0
b = Iris-versicolor	0	45	5
c = Iris-virginica	0	4	46

Table 6: Confusion Matrix for NNGE Classifier

	a	b	c
a = Iris-setosa	50	0	0
b = Iris-versicolor	0	47	3
c = Iris-virginica	0	3	47

Table 7: Confusion Matrix for OneR Classifier

	a	b	c
a = Iris-setosa	50	0	0
b = Iris-versicolor	0	44	6
c = Iris-virginica	0	3	47

Table 8: Confusion Matrix for PART Classifier

	a	b	c
a = Iris-setosa	49	1	0
b = Iris-versicolor	0	47	3
c = Iris-virginica	0	5	45

Table 9: Confusion Matrix for RIDOR Classifier

	a	b	c
a = Iris-setosa	50	0	0
b = Iris-versicolor	0	46	4
c = Iris-virginica	0	5	45

Table 10: Confusion Matrix for ZeroR Classifier

	a	b	c
a = Iris-setosa	50	0	0
b = Iris-versicolor	50	0	0
c = Iris-virginica	50	0	0

IV. CONCLUSION

In this study, Rule based classifiers are experimented to estimate classification accuracy of that classifier in a classification problem. The experiment was done using an open source Machine Learning Tool. The performances of the classifiers were measured and results are compared using the Iris Data set. Among nine classifiers (Conjunctive Rule Classifier, Decision Table Classifier, DTNB Classifier, OneR Classifier, JRIP Classifier, NNGE Classifier, PART Classifier, RIDOR Classifier and ZeroR Classifier) NNGE Classifier performs well in the classification problem. OneR classifier, RIDOR Classifier and JRIP classifier are coming in the next category to classify the data.

REFERENCES

[1] Pawan Kumar and Deepika Sirohi. (2010). "Comparative Analysis of FCM and HCM Algorithm on Iris Data Set", International Journal of Computer Applications (0975 – 8887), Volume 5, No.2, August 2010, pp 33 – 37.
[2] David Benson-Putnins, Margaret monfardin, Meagan E. Magnoni, and Daniel Martin. "Spectral Clustering and Visualization: A Novel Clustering of Fisher's Iris Data Set".
[3] Fisher, R.A. (1936) "The use of multiple measurements in taxonomic problems" Annual Eugenics, 7, pp.179 – 188.
[4] Patrick S. Hoey. "Statistical Analysis of the Iris Flower Dataset".
[5] Konstantin Vorontsov and Andrey Ivahnenko. (2011). "Tight Combinatorial Generalization Bounds for Threshold Conjunction Rules"

4th International Conference on Pattern Recognition and Machine Intelligence (PReMI'11), June 27 – July 1, 2011.

[6] Shengli Sheng, Charles X. Ling. (2005). "Hybrid Cost-sensitive Decision Tree, Knowledge Discovery in Databases", PKDD 2005, Proceedings of 9th European Conference on Principles and Practice of Knowledge Discovery in Databases. Lecture Notes in Computer Science 3721 Springer 2005, ISBN 3-540-29244-6.

[7] Gaya Buddhinath and Damien Derry. "A Simple Enhancement to One Rule Classification".

C. Lakshmi Devasena is presently doing PhD in Karpagam University, Coimbatore, Tamil Nadu, and India. She has completed M.Phil. (computer science) in the year 2008 from Bharathidasan University and MCA degree in 1997 from RVS College of Engineering and Technology, Dindigul (Madurai Kamaraj University) and B.Sc (Mathematics) in 1994 from Gandhigram Rural Institute. She has Eight years of teaching experience and Two years of industrial experience. Her area of research interest is Image processing and Data mining. She has published 10 papers in International Journals and Presented 15 papers in National and international conferences. At Present, she is working as Lecturer in Karpagam University, Coimbatore.

M. Hemalatha completed MCA MPhil., PhD in Computer Science and currently working as an Assistant Professor and Head, Department of software systems in Karpagam University. Ten years of Experience in teaching and published more than fifty papers in International Journals and also presented seventy papers in various National conferences and one international conferences Area of research is Data mining, Software Engineering, bioinformatics, Neural Network. She is also reviewer in several National and International journals.

T. Sumathi is presently doing PhD in Karpagam University, Coimbatore, Tamilnadu, India and has completed M.Phil (computer science) in the year 2006 and MCA degree in 1999 and B. Sc(computer science) in 1996. Major research area is Image processing and title for the research is image annotation. At Present, she is working as Lecturer in Karpagam University, Coimbatore.