MACHINE LEARNING APPROACH FOR FEATURE CLASSIFICATION USING SUPERVISED LEARNING ALGORITHMS

Meenakshi Garg[1], Kiran Joshi[2]

[1]Department of Computer Application, Sardar Patel Institute of Technology, Mumbai, India
[2]Department of Information Technology, Sardar Patel Institute of Technology, Mumbai, India
Email: [1]meenakshi1404@gmail.com

ABSTRACT

Due to the large volumes of data as well as the complex and dynamic properties data, data mining based techniques have been applied to datasets. With recent advances in computer technology large amounts of data could be collected and stored. Machine Learning techniques can help the integration of computer-based systems in any environment providing opportunities to facilitate and enhance the work of various industry professionals. It ultimately improves the efficiency and quality of data and information.

This paper presents the implementation of four supervised learning algorithms C4.5 Decision tree Classifier (J48), Instance Based Learning (IBK), Naive Bayes (NB) and Decision Stump in WEKA environment. The classification models were trained using various UCI datasets. The trained models were then used for classification & association which will help in decision making process. The Prediction Accuracy of the Classifiers was evaluated using 10-fold Cross Validation and the results have been compared to obtain the accuracy.

Keywords: *Machine Learning, C4.5, NB, J48, IBK, Decision Stump, WEKA*

I. INTRODUCTION

Due to rapid progress of information technology the amount of information stored in databases is rapidly increasing. These huge databases contain a wealth of data and constitute a potential goldmine of valuable business information. In the ever-changing environment, the form and structure of databases in every sector is changing abruptly [1].

Finding the valuable information hidden in those databases and identifying appropriate models is a difficult task. Using Data mining techniques it is possible to identify useful patterns and associations in the huge database which will support Decision making.

II. PROBLEM STATEMENT

The major problem in most of the sectors data is that data is not in standard format. Also data set is huge & does not contain any pattern. Data is not discrete in any Sector. It becomes very difficult to classify and discretize the continuous data and draw conclusion for decision making from continuous data.

There are several tools available which use different algorithms to solve this problem. We have used four Supervised learning algorithms on different domains. We investigated various issues involved with huge data sets.

III. MACHINE LEARNING METHODS

Machine learning methods have been successfully applied for solving classification problems in different applications. In Machine learning, algorithms we try to automatically filter the knowledge from example datasets. This knowledge can be used to make predictions about original data in the future and to provide insight into the nature of the target concept [6] [7]. The example data typically consists of a number of input patterns or examples to be learned. Each example is described by a vector of measurements or features along with a label which denotes the category or class the example belongs to. Machine learning systems typically attempt to discover regularities and relationships between features and classes in learning or training phase. A second phase called Classification uses the model induced during learning to place new examples into appropriate classes [7].

For analyzing the data set classification of data the four machine learning algorithms J48, IBK (Instance base learner) Naïve Bayes Classifier, Decision stump are used [8]. J48 algorithm is an implementation of the C4.5 technique to induce decision trees for

classification. A decision-tree model is built by analyzing the training data and the model is used to classify the trained data. J48 generates decision trees. The node of the J48 decision trees evaluates the existence and the significance of every individual feature.

A. J48 Algorithm

J48 (enhanced version of C4.5) is based on the ID3 algorithm developed by Ross Quinlan, with additional features to address problems that ID3 was unable to deal. In practice, C4.5 uses one successful method for finding high accuracy hypotheses, based on pruning the rules issued from the tree constructed during the learning phase. However, the principal disadvantage of C4.5 rule sets is the amount of CPU time and memory they require. Given a set S of cases, J48 first grows an initial tree using the divide-and-conquer algorithm as follows:

- If all the cases in S belong to the same class or S is small, the tree is leaf labeled with the most frequent class in S.
- Otherwise, choose a test based on a single attribute with two or more outcomes. Make this test as the root of the tree with one branch for each outcome of the test partition S into corresponding subsets S_1, S_2 …, according to the outcome for each case, and apply the same procedure recursively to each subset [4].

There are usually many tests that could be chosen in this last step. J48 uses two heuristic criteria to rank possible tests: information gain, which minimizes the total entropy of the subsets $\{S_i\}$ (but is heavily biased towards tests with numerous outcomes).

B. Naive Bayes Classifier

The Naive Bayes Classifier (Probabilistic Learner) technique is based on Bayesian theorem and is used when the dimensionality of the inputs is high. Naïve Bayes classifiers assume that the variable value on a given class is independent of the values of other variable. The Naive-Bayes inducer computes conditional probabilities of the classes given the instance and picks the class with the highest posterior. Depending on the precise nature of the probability model, Naive Bayes classifiers can be trained very efficiently in a supervised learning mode.

Assume that each instance is represented as a vector $X = (a1, a2, \ldots, an)$, where $a1, a2, \ldots,$ an are measures of attributes $A1, A2, \ldots, An$. We suppose there are m classes $C1, C2, \ldots, Cm$, and then given a new unknown instance X, by Bayesian theorem, the posterior probability of X that belongs to $Ci\ (1 \le i \le m)$ is:

$$P(C_i \mid X) = \frac{P(X \mid C_1)\, P(C_1)}{\sum_{k=1}^{M} P(C_K)\, P(X \mid C_K)} \quad (1)$$

The Equation 1, describes the Bayesian theorem [2] [3].

C. IBK

IBK is an implementation of the k-nearest-neighbors classifier. Each case is considered as a point in multi-dimensional space and classification is done based on the nearest neighbors. The value of 'k' for nearest neighbors can vary. This determines how many cases are to be considered as neighbors to decide how to classify an unknown instance.

For example, for the 'iris' data, IBK would consider the 4 dimensional space for the four input variables. A new instance would be classified as belonging to the class of its closest neighbor using Euclidean distance measurement. If 5 is used as the value of 'k', then 5 closest neighbors are considered. The class of the new instance is considered to be the class of the majority of the instances. If 5 is used as the value of k and 3 of the closest neighbors are of type 'Iris-setosa', then the class of the test instance would be assigned as 'Iris-setosa'.

The time taken to classify a test instance with nearest-neighbor classifier increases linearly with the number of training instances that are kept in the classifier. It has a large storage requirement. Its performance degrades quickly with increasing noise levels.

It also performs badly when different attributes affect the outcome to different extents. One parameter that can affect the performance of the IBK algorithm is the number of nearest neighbors to be used. By default it uses just one nearest neighbor.

D. Decision stump

A decision stump is a machine learning model consisting of a one-level decision tree [5]. That is, it is a decision tree with one internal node (the root) which is immediately connected to the terminal nodes. A decision stump makes a prediction based on the value of just a single input feature.

Depending on the type of the input feature, several variations are possible. For nominal features, one may build a stump which contains a leaf for each possible feature value. Such classifiers have sometimes been also called "1-rules" stump with the two leaves, one of which corresponds to some chosen category, and the other leaf to all the other categories. For binary features these two schemes are identical. A missing value may be treated as a yet another category. For continuous feature, some threshold feature value is selected, and the stump contains two leaves for values below and above the threshold.

Decision stumps are often used as components (called "weak learners" or "base learners") in machine learning ensemble techniques such as bagging and boosting.

IV. IMPLEMENTATION

The data analysis was carried out using WEKA software environment for machine learning. The Weka, Open Source, Portable, GUI-based workbench is a collection of state-of-the-art machine learning algorithms and data pre processing tools.

In this section we compare the results of four supervised learning algorithms like J48, IBk, Naïve Bayes Classifier, Decision stump. We have taken value of $k=1$ in IBk algorithm. The simulations were conducted four times with using four different datasets. All datasets were pre-processed. Missing values are replaced numeric or binary attributes are converted to nominal type using various preprocessing methods.

We have used four UCI datasets Adult, Zoo, Congress Voting and Balance [9]. We used 15 attributes and 2500 instances from adult dataset, 18 attributes and 101 instances from Zoo Dataset, 17 attributes and 437 instances from Congress Voting Dataset and 5 attributes and 625 instances from Balance Dataset. Below table 1 shows he kappa statistics for different classifiers using different algorithms on four different datasets.

Table 1. Kappa Statistics for Different Classifiers Criteria

	Classifiers	J48	IBK	NB	DS
Kappa Statistic	Adult	0.51	0.39	0.510	0.41
	Zoo	0.94	0.94	0.90	0.44
	Congress Voting	0.90	0.84	0.793	0.90
	Balance	0.32	0.70	0.839	0.14
Mean Absolute Error (MAE)	Adult	0.22	0.22	0.179	0.28
	Zoo	0.13	0.01	0.020	0.13
	Congress Voting	0.08	0.07	0.097	0.07
	Balance	0.30	0.23	0.218	0.35
Root Mean Squared Error (RMSE)	Adult	0.35	0.47	0.373	0.38
	Zoo	0.25	0.09	0.103	0.25
	Congress Voting	0.20	0.24	0.294	0.20
	Balance	0.42	0.32	0.284	0.43

We have preprocessed all datasets by removing missing values using unsupervised filters. We have converted UCI data set into WEKA compatible (.arff) file format. The Prediction Accuracy of the Classifiers was evaluated using 10-fold Cross Validation and the results are shown in Table 2.

Table 2. Percentage of Correctly Classified Instances

	UCI DATA SET	NB	J48	IBK	DS*
Correctly Classified Instances percentage	Adult	83.233	82.553	77.19	75.11
	Zoo	93.069	92.079	96.03	60.39
	Congress Voting	90.046	96.75	92.36	95.65
	Balance	91.36	63.2	83.84	54.08
Incorrectly Classified Instances percentage	Adult	16.766	17.447	22.80	24.89
	Zoo	6.9307	7.9208	3.960	39.60
	Congress Voting	9.9537	3.2407	7.638	4.347
	Balance	8.64	36.8	16.16	45.92
Time taken to build the model (in Seconds)	Adult	0.02	0.11	0.06	0.03
	Zoo	0.04	0.03	0.07	0.02
	Congress Voting	0.05	0.05	0.05	0.03
	Balance	0.03	0.03	0.06	0.05

*DS - Decision Stump

Table 2 shows Percentage of correctly classified instances, incorrectly classified instances along with the time taken to build different classification models. As shown in Table 2 that NB and J48 have similar classification accuracy for most of the domains.

V. RESULTS

In Figures 1 percentage of correctly classified instances of different UCI datasets using different supervised learning algorithm is given. The *X*-axis shows the names of different data sets used. The *Y*-axis shows the names of different classification algorithms used where as *Z*-AXIS shows Percentage of correctly classified instances.

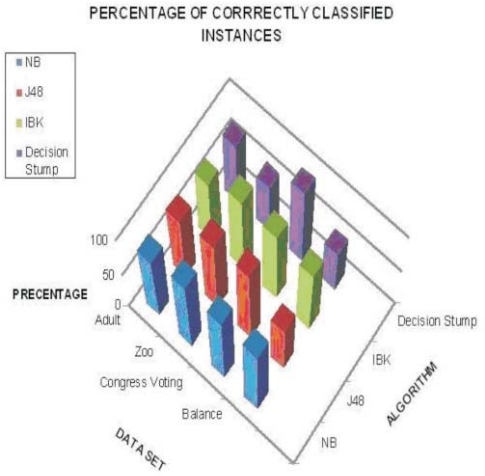

Fig. 1. Graph of percentage Correctly classified Instances vs classification algorithms

The below figure 2 shows graph kappa statistics of four supervised learning algorithms with respect to different data sets. The Figures 2 shows kappa statistics of different UCI datasets using four Supervised learning algorithm (NB, J48, IBK, Decision Stump). The *X*-axis shows the names of different data sets used. The *Y*-axis shows the names of different classification algorithms used where as *Z*-AXIS shows kappa statistics.

VI. CONCLUSION

In this paper we evaluated accuracy of four different classification algorithms (J48, NB, DS, IBk). According to the simulation results NB and J48

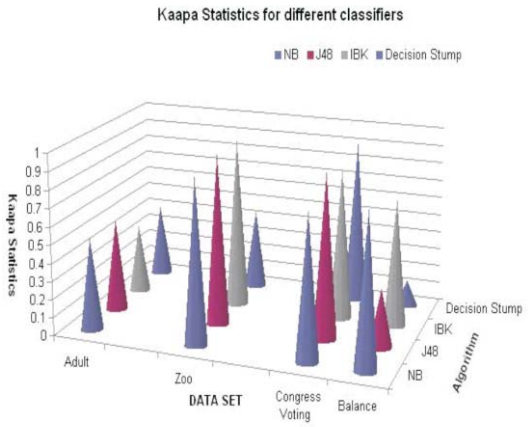

Fig. 2. Graph of kappa statistics of four supervised learning algorithms

classifiers performs the best on most of the domains. Furthermore, in few cases IBK gives similar performance as compared with NB. As per kappa statistics of four supervised learning algorithms NB performs better than other algorithms for most of the domains.

VII. REFERENCES

[1] Yu Yan; Haiying Xie Research on the Application of Data Mining Technology in Insurance Informatization. HIS '09. Ninth International Conference on Hybrid Intelligent Systems, 2009 Volume: 3 Publication Year: 2009, Page(s): 202 – 205 Digital Object Identifier: 10.1109/HIS.2009.255

[2] Wang Ding; Songnian Yu; Qianfeng Wang; Jiaqi Yu; Qiang Guo; 2008 "A Novel Naive Bayesian Text Classifier" in 2008 International Symposiums on Information Processing (ISIP) pp:78-82 DOI: 10.1109/ISIP.2008.54

[3] N. Friedman, D. Geiger and M. Goldszmidt, 1997 "Bayesian Network Classifiers," Machine Learning, 29(2-3): pp. 131-163.

[4] Xindong Wu, Vipin Kumar, J. Ross Quinlan, Joydeep Ghosh, Qiang Yang, Hiroshi Motoda, Geoffrey J. McLachlan, Angus F. M. Ng, Bing Liu, Philip S. Yu, Zhi-Hua Zhou, Michael Steinbach, David J. Hand, Dan Steinberg: 2008 Top 10 algorithms in data mining. Knowl. Inf. Syst. 14(1): 1-37 DOI: 10.1007/s10115-007-0114-2

[5] Kosiantis S.B., 2009, S.B. Dept. of Math., Univ. of Patras, Patras, Greece Local Random Subspace Method for Constructing Multiple Decision Stumps in *International Conference on Information and Financial*

Engineering, 2009. ICIFE 2009. Issue Date : 17-20 On page(s): 125 – 129, Print ISBN: 978-0-7695-3606-4 INSPEC Accession Number: 10804695 Digital Object Identifier : 10.1109/ICIFE.2009.22

[6] Pradeep Singh 2009 "Comparing the Effectiveness of Machine Learning Algorithms for Defect Prediction", International Journal of Information Technology and Knowledge Management, 2, No.2, pp. 481-483.

[7] MeeraGandhi G. 2010 "Machine Learning Approach for Attack Prediction and Classification using Supervised Learning Algorithms" International Journal of Computer Science & Communication Vol. 1, No. 2, pp. 247-250.

[8] Ian H.Witten, Eibe Frank 2005, "Data Mining – Practical Machine Learning Tools and Techniques," 2nd Edition, Elsevier.

[9] UC Irvine Machine Learning Repository http://archive.ics.uci.edu/ml/