ISSN:0975-9646

B. Veera sekhar Reddy et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (6) , 2014, 6985-6988

A Novel Approach for Transductive Decision Tree

B. Veera sekhar Reddy[1]
PG scholar,Dept. of CSE
Madanapalle Institute of Technology and Science
Madanapalle,India

Y.C.A.Padmanabha Reddy[2]
Asst.Professor, Dept. of CSE
Madanapalle Institute of Technology and Science
Madanapalle,India

S.Mohommad.ghouse[3]
Asst.Professor, Dept. of CSE
Madanapalle Institute of Technology and Science
Madanapalle,India

***Abstract*—A decision tree algorithm(ID3) may give good result when we process label data as input. In ID3 we give more data as input then it will give efficient result. With this method the complexity of ID3 is high. To reduce complexity we need to process less data as input and get efficient result. So for that we apply transduction to ID3. Here we use Naive Bayesian classifier to select the input data. By changing this method for selecting the input data can give better results than traditional ID3 when we compare accuracy as parameter**

Keywords—ID3; Naïve bayes; Transduction;

I. INTRODUCTION

Now a day's learning algorithms face a major problem i.e. lack of sufficient labeled data. For that we oftenly need a learning algorithm to train the machine with the small amount of labeled data. In real world the unlabeled data is available in large volume. But availability of labeled data is very small. So we can give labels for those unlabeled data by considering the small amount of labeled data. In this paper we focus on the decision tree algorithm. It is also known as ID3. For ID3 we give more training data as input then it gives the efficient results. Whenever the input data is high then the complexity is also high. So to reduce the complexity we need to process the small amount of training data as input to get better results. This can be achieved by applying transduction to ID3. Here mainly we focus on the selection of training data which is processed as input to ID3. We use naïve Bayesian algorithm to select the training data from the data set.

II INRODUCTION TO AREA OF WORK

Machine Learning can be defined as to improve the efficiency of the system or machine by learning or training with some complex, critical data sets. Machine Learning can be classified as Supervised Learning, Unsupervised Learning and semi supervised learning.

Supervised Learning: Supervised Learning is also known as Classification. In this we already have the Predefined training data to classify the test data. Here training data is nothing but the data which was already classified or the data with labels. Training data is nothing but known data. Test data is the data which will go to be classified by the training data. Test data is also known as Unknown data or unlabeled data. In Classification the process should be done in two steps. In first step we build a model or classifier to classify the test data based on the training data. In second step the classifier classify the test data.

Unsupervised Learning: Unsupervised learning is also known as clustering. In this we cannot use any train data. We use some statistical methods to cluster the unknown data. Here we cluster the data which is similar. But the attributes in one cluster may differ from the attributes of another cluster.

Semi-supervised Learning: Semi supervised learning is another type of machine learning. Semi supervised learning is half way between supervised and unsupervised learning. The main reason of evolving semi supervised learning is to overcome the drawbacks of both supervised and unsupervised learning. In supervised learning we need more train data to classify the test data. The designing of train data is cost effective and time consuming. In unsupervised learning we cannot cluster the unknown data accurately. To overcome the above problems the semi supervised learning is evolved here. By considering small amount of train data can label the unknown (or) test data.Semi supervised learning is two types. Semi supervised classification and Semi supervised clustering.

Semi-Supervised Classification: semi supervised classification is a special case of classification. Generally in classification we use more training data to classify the test data. But in semi supervised classification we use less train data to classify the large amount of test data. By using this semi supervised classification we reduce the usage of the training data. Presently more unknown or unlabeled data available in the market but the labeled data is not much available in the market. Because to design of the training data is cost effective and time consuming.

Semi Supervised Clustering: Semi supervised clustering is a special case of clustering. Generally in clustering we use unknown data for clustering. But in semi supervised clustering we use both labeled data and unlabeled data is used as pair wise constraints to cluster the unlabeled data.

Learning-paradigms:

1. Transductive Learning:

Transductive learning is one of the learning paradigms. Whatever the teacher told the methods in class room that methods only gave homework to students. Transductive Learning cannot handle the unseen the data.

Ex: Take home exam.

2. Inductive Learning:

Inductive Learning is the one of learning paradigms. Whatever the teacher told the methods in class room that

methods not gave for class exam to the students. Inductive Learning Can handle the unseen data.
Ex: Take class exam.

3.Deductive Learning:

Deductive Learning is the one of the learning paradigms'. It will works on already proved formulas and it can give exact result. Ex: y=mx+c.

III DEFINITIONS

Class labels: Class labels can be used for Identification purpose. Normally it consist of Positive and Negative labels.

Training set: Training set is also known as Labeled set. The Size of Training set is L, and It is a Collection of Both Positive and Negative Labels.

Test set: Test set is also known as Unlabeled set. The Size of Test set is U, and It is also a Collection of Both Positive and Negative Labels.

Classification Accuracy: This is Calculation Accuracy over the unlabeled set. The Classification Accuracy on given Unlabeled set (test set) is the Percentage of Unlabeled set patterns or Tuples that are Correctly Classified by the Classifier.

IV HOLD OUT METHOD AND RANDOM SUB SAMPLING

Hold out method is used to select the training data from data set. By using this we can divide the two by third part of the data as training set and one by third part of the dataset as test set. The training set can build the model. By using this model we can estimate the accuracy of the test set. Random sub sampling is the method of repeating the hold out method. The final accuracy is taken by the average of each hold out iteration accuracies.

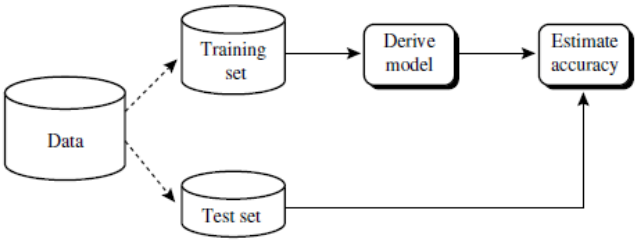


Fig.1: model for estimating accuracy using hold out

V PREPROCESS OF TRAINING DATA

The steps taken to preprocess the training data are

1. Select the training and test data sets by applying holdout method to the dataset. Training data as two by third of the data set and remaining as the test set.
2. Apply naive Bayesian algorithm to the training data to preprocess it.
3. Design the decision tree by considering the preprocessed training data as input.

VII ALGORITHM FOR CONSTRUCTING DECISION TREE USING ID3

i. D be the dataset of consisting the selected tuples that are selected by naïve Bayesian algorithm.

VI NAIVE BAYESIAN ALGORITHM

Naïve bayes is the one of the eager learner. In this a model is build based on the training data to estimate the accuracy of the test set. The working procedure of Naïve Bayesian algorithm is as follows.

i. Let D be a training set of tuples and their associated class labels. Tuple is represented by X.

ii. Suppose that there are m classes, and assigned as $C1, C2, : : : , Cm$. consider a tuple X, the classifier predicts that tuple X belongs to the class having the highest posterior probability, conditioned on X. i.e. the classifier predicts that tuple X belongs to the class Ci if and only if
$P(Ci/\boldsymbol{X}) > P(Cj/\boldsymbol{X})$ for $1 <= j <= m; j != i$.

Then we maximize $P(Ci/\boldsymbol{X})$, by using bayes theorem
$P(Ci/\boldsymbol{X}) = (P(\boldsymbol{X}/Ci)P(Ci))/P(\boldsymbol{X})$.

iii. Here P(X) is constant for all classes. So, to maximize $P(Ci/\boldsymbol{X})$ we need to maximize the $P(\boldsymbol{X}/Ci)P(Ci)$. The class prior probabilities are estimated by $P(Ci)=|Ci,D|/D$, where $|Ci,D|$ is the number of tuples of class Ci in D.

iv. If the given datasets have many attributes, it would be very complex to calculate $(P(\boldsymbol{X}/Ci)$. to reduce the complexity in evaluating $(P(\boldsymbol{X}/Ci)$ class conditional independence is made.

$$P(\boldsymbol{X}/Ci) = n\square\ P(xk/Ci)$$
$$k=1$$
$$= P(x1/Ci)*P(x2/Ci)*\ldots\ldots*P(xn/Ci).$$

v. In order to predict the class label of X, $P(\boldsymbol{X}/Ci)P(Ci)$ is evaluated for each class Ci . the predicted class label of tuple X is the class Ci if and only if $P(\boldsymbol{X}/Ci)P(Ci) > P(\boldsymbol{X}/Cj)P(Cj)$ for $1 <= j <= m, j! = i$

vi. In other words the predicted class label is the class Ci for which $P(\boldsymbol{X}/Ci)P(Ci)$ is maximum.

VIII EXTRACTING OF BEST FEATURES BY NAÏVE BAYESIAN ALGORITHM

The working procedure is as follows.

i. Let D be the dataset consisting of N tuples.

ii. List out all the tuples consisting with each label.

iii. Find the probability of occurrence to each tuple with each label.

iv. List out all the tuples which can give maximum probability value. i.e. the selected tuples have most probable occurrences in the dataset.

v. In the above manner we select the training data which is processed as input for ID3.

ii. Create a node N.

iii. If tuples in D are of same class C, then return N as leaf node and labeled with class C.

iv. Else apply attribute selection method to find best splitting criterion.

v. Label node N with splitting criterion.

vi. (a).if the splitting attribute is discrete valued then multi way splitting is allowed.

(b). if the splitting attribute is continuous valued then the splitting is as follows. A<=split-point and A>split-point.

(c). if the splitting attribute is discrete and binary then it has two split points.

if the outcome has no splitting criteria then assign label for it.

vii. Else repeat the above procedure until the tree does not have splitting region.

IX. Experimental Results

The five standard data sets with testing experiments on algorithms are organized . The five data sets are present in UCI Machine Learning Repository [13]. Those are listed below.

TABLE I Data-sets with no. of features, labeled and unlabeled

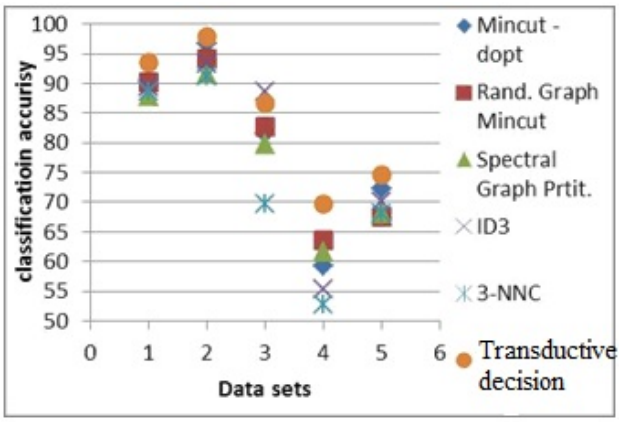
Data-set	Number Of Features	|L|	|U|	Distance Function
VOTING	16	45	390	Jaccard Coefficient
MUSH	22	20	1000	Simple Matching
IONO	34	50	300	Euclidean
BUPA	6	45	300	Euclidean
PIMA	8	50	718	Euclidean

Note: same data-sets are used in the classifiers that are used for the comparison purpose are listed as graph in cut, randomized graph min cut, 3-NNC and ID3. We select these methods for comparison because all these are similar to the method described in this paper.

TABLE II CA (%) FOR VARIOUS CLASSIFIERS

Data set	graph mincut-±opt	Rand. graph mincut	spectral graph partitioning	3-NNC	ID3	Transductive decision tree
VOTING	90.4	90.3	87.9	88.7	89.3	90.5
MUSH	96.8	94.3	91.7	91.0	93.2	96.9
INO	81.7	82.9	79.8	69.6	88.5	90.1
BUPA	59.2	63.6	61.7	52.5	55.4	62.3
PIMA	72.4	67.6	67.8	68.2	69.8	73.2

By using the table II the below graph is plotted. where the values of X axis indicates the datasets whatever present in the above table II. Y axis indicates the classification accuracy retrieved by each algorithm.



X Conclusion

The same as on top of indicated methodology will give better result and reduces the time and space complexities for decision tree. Because whenever the no of inputs are reduced then automatically complexity reduces.

REFERENCES

[1] O. Chapelle, B. Scholkopf, and A. Zein, *Semi- Supervised Learning*. Cambridge, Massachusetts: The MIT Press, 2006.

[2] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. S. lkopf, "Learning with local and global consistency," in *Advances in Neural Information Processing Systems*, S. Thrun, L. Saul, and B. S. lkopf, Eds., vol. 16. Cambridge, MA: The MIT Press, 2004, pp. 321–328.

[3] Vapnik, *Statistical Learning Theory*. John Wiley & Sons: A Wileyinterscience Publication, New York, 1998.

[4] V. Vapnik, *Estimation of Dependences Based on Empirical Data*, 2nd ed. New York: Springer Series in Statistics, Springer-Verlag, 2006.

[5] K. Bennett, "Combining support vector and mathematical programming methods for classification," in *Advances in kernel methods – support vector learning*, B. Scholkopf et al., Ed. MIT-Press, 1999.

[6] T. Joachims, "Transductive inference for text classification using support vector machines," in *Sixteenth International Conference on Machine Learning*. Bled Slovenia: Morgan Kaufmann, 1999, pp. 200–209.

[7] A. Blum and S. Chawla, "Learning from labeled and unlabeled data using graph mincut," in *Eighteenth International Conference on Machine Learning*. Morgan Kaufmann, 2001, pp. 19–26.

[8] A. Blum, J. Lafferty, M. Rwebangira, and R. Reddy,"Semi-supervised learning using randomized mincuts," in *International Conference on Machine Learning*. Morgan Kaufmann, 2004.

[9] P.S. Bradley and Usama M. Fayyad: Refining initial points for k-means clustering. In *Proceedings Fifteenth International Conference on Machine Learning*, pages 91-99, San Francisco, CA, 1998, Morgan Kaufmann.

[10] X. Zhu, Z. Gharahmani, and J. Lafferty, "Semi-supervised learning using Gaussian fields and harmonic functions," in *20th International Conference on Machine Learning*, 2003, pp. 912–919.

[11] A. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: A review," *ACM Computing Surveys*, vol. 31, no. 3, pp. 264– 323, 1999.

[12] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*, 1st ed. Cambridge University Press, 2000.

[13] P.M.Murphy, *UCI Repository of Machine Learning Databases [http://www.ics.uci.edu/mlearn/MLRepository.html]*, Department of Information and Computer Science, University of California, Irvine, CA, 2000.

[14] R. O. Duda, P. E.Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. John Wiley & Sons: A Wiley-interscience Publication, 2000.

[15] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*,

Academic Press, 2001.
[16] B. V. Dasarathy, "Data mining tasks and methods: Classification Nearest-neighbor approaches," in *Hand*
[17] *book of data mining and knowledge discovery*. New York: Oxford University Press, 2002, pp. 288–298.
[18] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*. Cambridge, MA, U.S.A: The MIT Press, 1990.
[19] R. Motwani and P. Raghavan, *Randomized Algorithms*. Cambridge UK: Cambridge university Press, 1995.
[20] http://en.wikipedia.org/wiki/Machine_learning Machine Learning.
[21] A book titled "*Enhancement to Selective Incremental Approach for Transductive Nearest Neighbour Classification*", ISBN 978-3-656-342502, published by GRIN Verlag GnbH, Norderstedt Germany.
[22] S.Md Ghouse, Y.C.A Padmanabha Reddy, E. Madhusudhana Reddy, (2012), "An Enhancement for Tranaductive Nearest Neighbor Classification" International journal of Advanced Computing (ISSN: 2233-2433), 2012, Volume 35, Special Issue 2, PP 361-366.