Implementation of Naive Bayesian Classifier and Ada-Boost Algorithm Using Maize Expert System

Naveen Kumar Korada , N Sagar Pavan Kumar, Y V N H Deekshitulu

Email-Id: knaveenkumar35@gmail.com

Abstract

Machine learning [1] is concerned with the design and development of algorithms that allow computers to evolve intelligent behaviors based on empirical data. Weak learner is a learning algorithm with accuracy less than 50%. Adaptive Boosting (Ada-Boost) is a machine learning algorithm may be used to increase accuracy for any weak learning algorithm. This can be achieved by running it on a given weak learner several times, slightly alters data and combines the hypotheses. In this paper, Ada-Boost algorithm is used to increase the accuracy of the weak learner Naïve-Bayesian classifier. The Ada-Boost algorithm iteratively works on the Naïve-Bayesian classifier with normalized weights and it classifies the given input into different classes with some attributes. Maize Expert System is developed to identify the diseases of Maize crop using Ada-Boost algorithm logic as inference mechanism. A separate user interface for the Maize expert system consisting of three different interfaces namely, End-user/farmer, Expert and Admin are presented here. End-user/farmer module may be used for identifying the diseases for the symptoms entered by the farmer. Expert module may be used for adding rules and questions to data set by a domain expert. Admin module may be used for maintenance of the system.

Keywords

Expert Systems, Machine Learning, Ada-Boost, Naïve Bayesian Classifier, Maize, JSP and MYSQL

I. Introduction

A. Machine Learning

Machine learning[2, 3, 4 and 6], a branch of artificial intelligence, is a scientific discipline concerned with the design and development of algorithms that allow computers to evolve behaviors based on empirical data, such as from sensor data or databases. The key issue in the development of Expert Systems is the knowledge acquisition for building its knowledge base. One simple technique for acquiring the knowledge is direct injection method in which the knowledge is collected from the domain experts by conducting programmed interviews and entering it in an appropriate place manually. But it is difficult process and time consuming. Instead, machine learning algorithms are used by making the systems learn from their past experiences. The goal of machine learning is to program computers to use training data or past experience to solve a given problem. Effective algorithms have been invented for certain types of learning tasks. Many practical computer programs have been developed to exhibit useful types of learning and significant commercial applications have begun to appear. Machine learning refers

DOI : 10.5121/ijist.2012.2305

to the changes in systems that perform tasks associated with artificial intelligence (AI). Such tasks involve recognition, diagnosis, planning, robot control, prediction, etc. Some of the machines learning algorithms are Genetic Algorithm [16], ID3 [17], ABC algorithm [18], Artificial Neural Networks [19] and C4.5 Algorithm [20] etc.

B. Expert Systems

An expert system [5] is a computer system that emulates the decision-making ability of a human expert, i.e. it acts in all respects as a human expert. Expert systems have emerged from early work in problem solving, mainly because of the importance of domain-specific knowledge. The expert knowledge must be obtained from specialists or other sources of expertise, such as texts, journal articles, and data bases. Expert system receives facts from the user and provides expertise in return. The user interacts with the system through a user interface, constructed by using menus, natural language or any other style of interaction. The rules collected from the domain experts are encoded in the form of Knowledge base. The inference engine may infer conclusions from the knowledge base and the facts supplied by the user. Expert systems may or may not have learning components. A series of Expert advisory systems [12], [13], [15] were developed in the field of agriculture and implemented in Indiakisan.net [14].

C. Adaptive Boosting (Ada-Boost) Algorithm

Ada-Boost [8, 11], short for Adaptive Boosting, is a machine learning algorithm, formulated by Yoav Freund and Robert Schapire [7]. It is a meta-algorithm, and can be used in conjunction with many other learning algorithms to improve their performance. Generally learning algorithms are either strong classifiers or weak classifiers. Strong classification algorithms use the techniques such as ANN, SVM etc. Weak classification algorithms use the techniques such as Decision trees, Bayesian Networks, Random forests etc. Ada-Boost is adaptive because the instances misclassified by previous classifier are reorganized into the subsequent classifier. Ada-Boost is sensitive to noisy data and outliers. The boosting algorithm begins by assigning equal weight to all instances in the training data. It then calls the learning algorithm to form a classifier for this data, and reweighs each instance according to the classifier's output. The weight of correctly classified instances is decreased, and that of misclassified ones is increased. This produces a set of easy instances with low weight, and a set of hard ones with high weight. In the next iteration, a classifier is built for the reweighed data, which consequently focuses on classifying the hard instances correctly. Then the instances weights are increased or decreased according to the output of this new classifier. Here there are two possibilities: The first one is harder instances may become even harder and easier instances may become even easier. The second possibility is the harder instances may become easier and easier instances may become harder. After all weights have been updated, they are renormalized so that their sum remains the same as it was before. After all iterations, the final hypothesis value is calculated.

The pseudo code for Ada-Boost algorithm is given as below

- **Input**: a set S, of 'm' labeled examples: S= ((xi,yi), i=(1,2,…,m)), with labels in Y.
- **Learn** (a learning algorithm)
- **A constant** L.

[1] Initialize for all i: wj(i)=1/m // initialize the weights

[2] for j=1 to L do
[3] for all i: // compute normalized weights

$$p_j(i) = \frac{w(i)}{\sum_i^m w(i)}$$

[4] hj:=Naïve-Bayesian(S,pj) // call weak Learn with normalized weights
[5] Calculate the error of hj

$$\varepsilon_j = \sum_i p_j(i)[h_j(x_i \neq y_i)]$$

[6] if $\varepsilon_j > \frac{1}{2}$ then
[7] L=j-1
[8] go to 12
[9]

$$\beta_j = \frac{\varepsilon_j}{1-\varepsilon_j}$$

[10] for all i: // compute new weights

$$w_{j+1}(i) = w_j(i)\beta_j^{1-[h_j(x_i-y_i)]}$$

[11] end for
12] Output:

$$h_{final}(x) = \arg\max_{y \in Y} \sum_{j=1}^{L} \left(\log \frac{1}{\beta_h}\right)[h_j(x=y)]$$

D. Naïve Bayesian Classifier (Weak learner)

Naïve Bayes Classifier is a simple probabilistic A Naive Bayes Classifier [9] is a simple probabilistic classifier based on applying Bayes' theorem with strong (naive) independence assumptions. A more descriptive term for the underlying probability model would be "independent feature model". Depending on the precise nature of the probability model, Naive Bayes classifiers can be trained very efficiently in a supervised learning setting. In spite of their naive design and apparently over-simplified assumptions, Naïve Bayes classifiers have worked quite well in many complex real-world situations. A comprehensive comparison with other classification methods showed that Bayes classification is outperformed by more current approaches, such as boosted trees or random forests [10].

An advantage of the Naive Bayes classifier is that it only requires a small amount of training data to estimate the parameters necessary for classification.

E. KNOWLEDGE BASE

Expert system knowledge base contains a formal representation of the information provided by the domain experts. The information is collected from the domain experts by conducting programmed interviews. The Knowledge Base of the Maize Expert System contains the diseases and the symptoms for corresponding diseases and the cure for those diseases is encoded in the form of rules. The symptoms and diseases occurred in maize crop are represented in tabular format as below.

S. No	Stage of the Crop	Part Effected	Symptoms	Disease	Cure
1.	Seedling	Leaves	Spots are small and pale green later becoming bleached	Phaeosphaeria Leaf Spot.	Use resistant varieties such as Comp. A 9, EH 43861, etc.
2.	Seedling	Leaves	Very small round scattered spots in the youngest leaves which increases with plant growth	Corn streak Virus.	Use systemic fungicide such as metalaxyl MX L 35 or Apron XL 35 ES 3 WS, Apron 35 WP
3.	Flowering	Leaves	Leaves of infected plants tend to be narrower and more erect.	Sorghum Downy Mildew	Spray mancozeb 2.5g copper Oxychloride 3g/liter
4.	Flowering	Leaves	Small powdery pustules present over both surfaces of the leaves.	Maize Fine Stripe Virus.	Spray carbendazim 1.5g and use metalaxyl MXL 35
5.	Early Whorl	Leaves	Lesions begin as small regular elongated necrotic spots and grow parallel to the veins.	Gray Leaf Spot.	Spray Zineb/Meneb @ 2.5-4.0 g/liter of water.
6.	Late Whorl	Leaves	Lesions with oval narrow necrotic and parallel to the veins.	Phyllosticta Leaf Spot.	Spraying of insecticides endosulfan 35EC
7.	Seedling	Root	White thin lesions along leaf surface and green tissue in plants	Flea Beetles and Flea Rootworms.	Spray Dithane M-45 @ 2-2.5 gm/liter
8.	Mid Whorl	Root	Bushy appearance due to proliferation of tillers which become chlorotic and reddish and lodging.	Bilb Bug Worms.	Seed treatment with peat based formulation
9.	Flowering	Root	Irregular section of epidermis and Perforated Leaves.	Seed and Seedling Blight	Spray of Chelamin450 @ 2.5 to 4.0 g/liter of water
10.	Late Whorl	Stem	The affected area just above the soil line is brown water-soaked soft and collapsed	Pre Flowering Stalk Rot or Pythium Stalk Rot	Spray of Sheethmar

I. PROPOSED ADA-BOOST ALGORITHM:

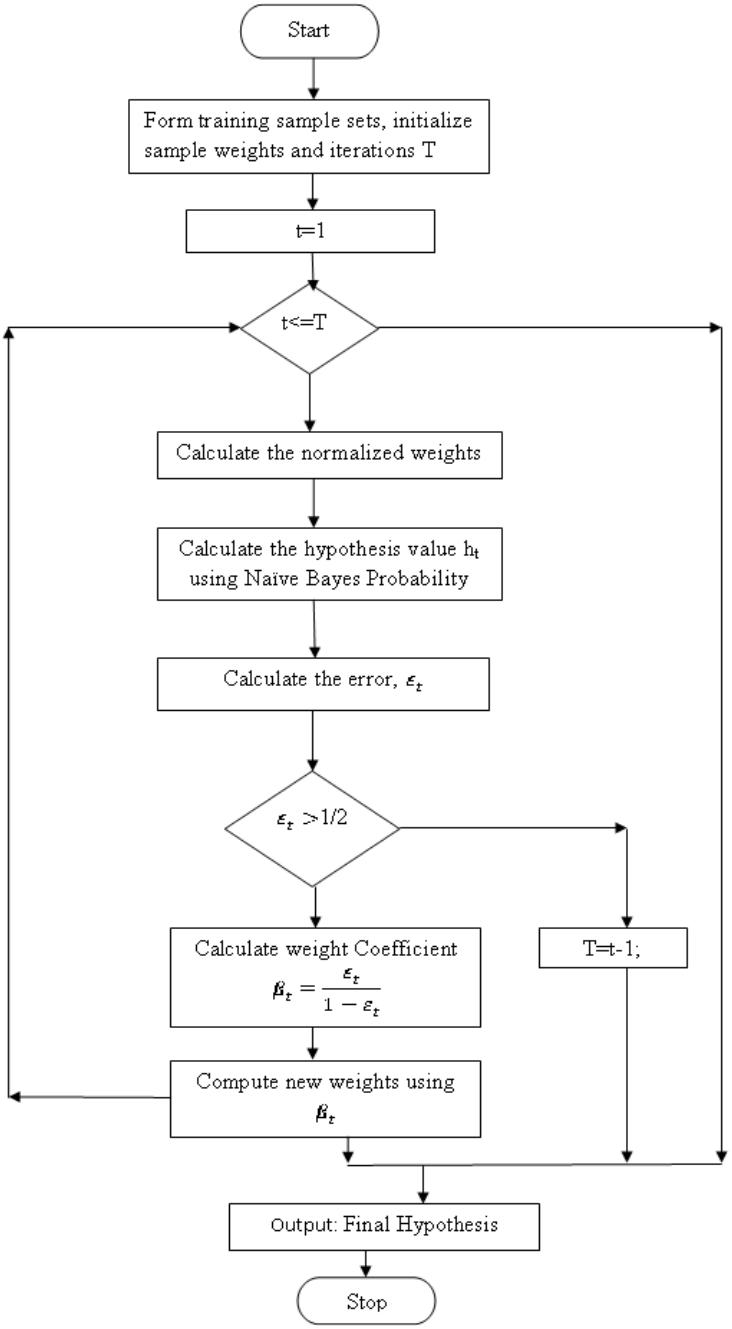
The Proposed Ada-Boost Algorithm uses the Naïve-Bayes classifier as weak learner and it uses the training data and the weights are initialized based on the number of classifiers i.e. the weights of the each class is equal to the fraction of the total number of classifiers. Select 'T', the number of rounds the algorithm has to run iteratively by adjusting the weights. In each round the weak learner is called based on the given input and the weights for each classifier and it generates a new hypothesis 'hj' in each hypothesis and the weight and the error is calculated based on the obtained hypothesis and based on the error value obtained the new weights are calculated by using the formula given below

$$w_{j+1}(i) = w_j(i)\beta_j^{1-[h_j(x_i - y_i)]} \qquad \text{Where } \beta j \text{ is error coefficient.}$$

The weak learner is called by using the new weights. The process is repeated until the error value greater than ½ or the number of iterations completes. And finally, the hypothesis value is calculated by using the given formula.

$$h_{final}(x) = \underset{y \in Y}{\arg\max} \sum_{j=1}^{L} \left(\log \frac{1}{\beta_h}\right) [h_j(x = y)]$$

The flow diagram of the proposed Ada-Boost algorithm used in development of this Expert Advisory System is shown in **Figure. 1. Flow chart of the Ada-Boost Algorithm**



A. Simple Example

The working of the proposed system is explained by considering the 10 symptoms as input. It is explained as follows

- Encode Solution: Just use 10 bits (1 or 0).
- Generate input.

S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
1	0	1	0	0	0	1	0	1	0

- Initialize the weights w_i based on the classifiers. Consider there are 5 classifiers, w_i= 1/5.
- Select the value for '**T**', the number of iterations.
- In each and every, iterations the hypothesis value 'h_j' is to be calculated.
- The error value is calculated by adding the probabilities value of the remaining diseases with their corresponding weights.
- Based on the error value the algorithm is repeated repeatedly for 'T' times by adjusting the weights.

1. Hypothesis Values Using Naïve Bayesian Classifier

The probability densities for each disease is calculated using the Naïve Bayesian classifier as follows

P (Disease1/s1 ...s10) = P (Disease1)*P (s1/Disease1)*P (s2/Disease2)….P (s10/Disease)

By using the above equation for the given input string

P(Corn Streak Virus/1,0,1,0,0,0,1,0,1,0)= 0.15002
P(Sorghum Down Mildew/1,0,1,0,0,0,1,0,1,0)=0.0

P(Postfloweringstalk rot/1,0,1,0,0,0,1,0,1,0)=0.01

P(Phaesophearia Leaf spot/1,0,1,0,0,0,1,0,1,0)=0.02922

P(Alternaria Leaf Spot/1,0,1,0,0,0,1,0,1,0)=0.022

2. Hypothesis Values Using Ada-Boost Algorithm

The probability densities for each disease is calculated using the Ada-Boost algorithm is as follows

P(Disease1/s1,..s10)= P(Disease1)*P(s1/Disease1)*P(s2/Disease2)….P(s10/Disease)

By using the above equation for the given input string

P(Corn Streak Virus/1,0,1,0,0,0,1,0,1,0)= 0.20000002

P(Sorghum Down Mildew/1,0,1,0,0,0,1,0,1,0)=0.0

P(Postfloweringstalk rot/1,0,1,0,0,0,1,0,1,0)=0.029626261

P(Phaesophearia Leaf spot/1,0,1,0,0,0,1,0,1,0)=0.12922

P(Alternaria Leaf Spot/1,0,1,0,0,0,1,0,1,0)=0.122

After 'n' iterations, the final probability value for disease Corn Streak Virus is greater than all the remaining diseases hence the final hypothesis classifies the given data to the class with name "Corn Streak Virus".

II. COMPARATIVE STUDY

The hypothesis values computed by the Naïve-Bayesian Classifier and the Ada-Boost algorithm are tabulated in table 1. The hypothesis value is maximum to Corn Streak Virus.

H.AB= Hypothesis value of Ada-Boost algorithm
H.NB= Hypothesis value of Naïve-Bayesian Classifier

Increase in
Accuracy % = (H.AB – H.NB)/H.NB *100.

From the table
H.AB for Corn Streak Virus=0.20000002
H.NB for Corn Streak Virus=0.15002

Increase in accuracy for Corn Streak Virus= 33%

Table 1: Hypothesis Values of the Algorithms

Disease Name	Hypothesis Value using Naïve-Bayesian Classifier	Hypothesis Value using Ada-Boost Algorithm
Corn Streak Virus	0.15002	0.2002
Sorghum Down Mildew	0.0	0.0
Post flowering stalk rot	0.01	0.0292
Phaesophearia Leaf spot	0.0291	0.12922
Alternaria Leaf Spot	0.022	0.122

Based on the hypothesis values, the error values are calculated. A graph is drawn taking the number of iterations on X-axis and the error values of the both Naïve-Bayesian Classifier and the Ada-Boost algorithms are taken on the Y- axis (figure 2).

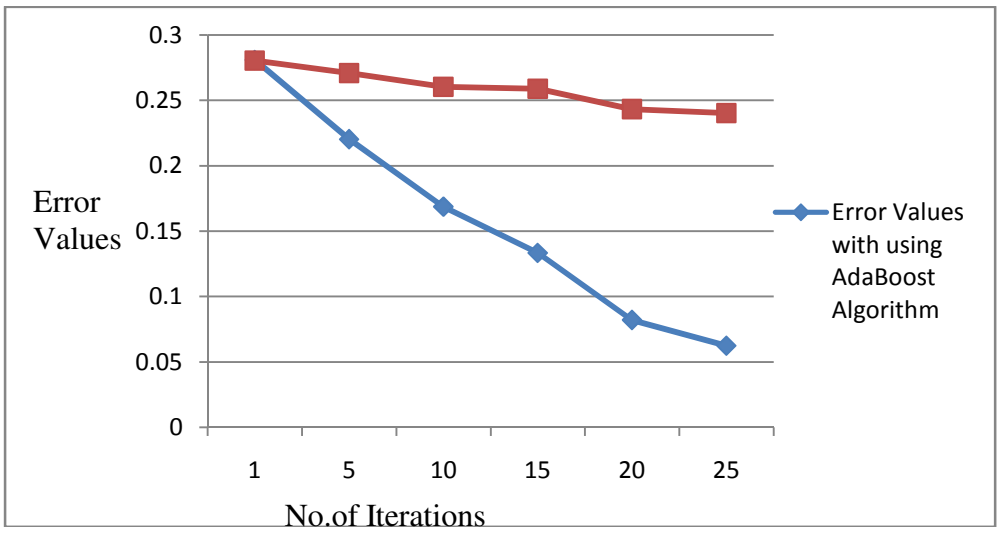


Figure 2: Graph describing the performance of Ada-Boost Algorithm

It can be observed that, as the number of iterations increases the miss-classification error values is decreased in Ada-Boost algorithm than pure Naïve-Bayesian classifier algorithm.

III. MAIZE EXPERT SYSTEM ARCHITECTURE

The Proposed architecture of the Maize Expert System consists of Rule Based Expert System, Ada-Boost algorithm and Knowledge Base which are used in the inference mechanism. It is represented in the figure 3

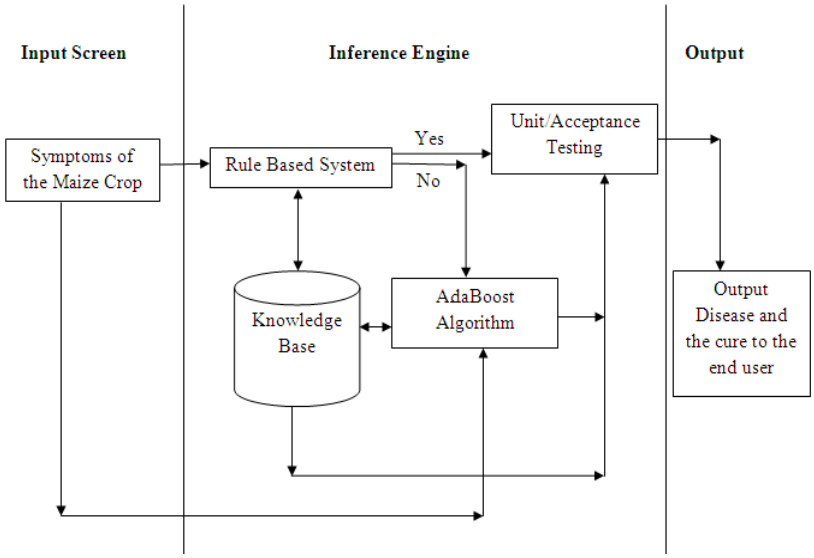


Figure 3: Proposed architecture of Expert System

The User Interface of the Maize expert system consisting of three different interfaces namely, End-user/farmer, Expert and Admin, is presented here. End-user/farmer module may be used for identifying the diseases for the symptoms entered by the farmer. Expert module may be used for adding rules and questions to data set by a domain expert. Admin module may be used for maintenance of the system.

IV. RESULTS

Figure 4 Screen for selecting symptoms in Maize Expert System

Description: In this screen shot, the user can submit the observed symptoms to the maize advisory system through online by selecting the appropriate radio buttons for the processing of the symptoms observed.

Figure 5 Displaying advices to the farmer

Description: In this screen shot, the algorithm takes the input given by the user and classifies the input into the **Corn Streak Virus** class and generating the following advices.

Effected With: **Corn Streak Virus**

Cure is: **Spraying of insecticides endosulfan 35EC @ 600-750 ml/n5g.**

V. CONCLUSIONS

According to the results, the performance of the Naïve- Bayesian classifier (weak learner) is improved by 33.33% with the help of Ada-Boost algorithm and it generates accurate results by reducing the miss-classification error values by increasing the iterations. Using this algorithm as inference mechanism a Maize Expert Advisory System is developed using Java Server Pages (JSP) and MYSQL database as backend for classifying the given symptoms given by the farmers into the corresponding disease class and suggest advices to the improvement of the crop.

VI. REFERENCES

[1] Dietterich, T. G., 2000, "An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization," *Machine Learning,* pp *139-158.*

[2] Reviews of "Machine Learning by Ryszard S. Michalski, Jaime G. Carbonell, and Tom M. Mitchell", Tiago Publishing Company, 1983, ISBN 0-935382-05-4.

[3] Corinna Cortes and Vladimir Vapnik. "Support-vector networks", Machine Learning, 20(3):273–297, 1995.

[4] C.A.Coello, Gary B.Lamont and David A.Van Veldhuizen, "Evolutionary Algorithms for Solving Multi-Objective Problems", *2nd Edition, Springer,* 2007.

[5] Dr. B D C N Prasad, P E S N Krishna Prasad and Y Sagar, "A Comparative Study of Machine Learning Algorithms as Expert Systems in Medical Diagnosis (Asthma)" *–CCSIT 2011, Part I, CCIS 131,* pp 570– 576.

[6] Rennie J, Shih L, Teevan J, and Karger D, "Tackling The Poor Assumptions of Naive Bayes Classifiers," In Proceedings *of the Twentieth International Conference on Machine Learning (ICML).* 2003.

[7] Yoav Freund and Robert E. Schapire, "Experiments with a new boosting algorithm", In *Machine Learning: Proceedings of the Thirteenth International Conference*, pages 148–156, 1996.

[8] A. J. M. Abu Afza, Dewan Md. Farid, and Chowdhury Mofizur Rahman, "A Hybrid Classifier using Boosting, Clustering, and Naïve Bayesian Classifier", *World of Computer Science and Information Technology Journal (WCSIT) ISSN: 2221-0741 Vol. 1, No. 3,*105-109, 2011.

[9] Rish Irina, "An empirical study of the naive Bayes classifier", *IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence.*

[10] Caruana.R and Niculescu-Mizil, "An empirical comparison of supervised learning algorithms", In Proceedings *of the 23rd international conference on Machine learning*. 2006.

[11] Atsushi Takemura, Akinobu Shimizu, and Kazuhiko Hamamoto, "Discrimination of Breast Tumors in Ultrasonic Images Using an Ensemble Classifier Based on the AdaBoost Algorithm With Feature Selection", *IEEE Transactions on Medical Imaging, vol. 29, no. 3,* 2010.,

[12] Prof. M.S. Prasad Babu, N.V. Ramana Murty, S.V.N.L.Narayana, "A Web Based Tomato Crop Expert Information System Based on Artificial Intelligence and Machine learning algorithms", *International Journal of Computer Science and Information Technologies, Vol. 1 (1), (ISSN: 0975-9646).,* 2010, pp6-15.

[13] Prof. M.S. Prasad Babu, Mrs. J. Anitha, K. Hari Krishna, "A Web Based Sweet Orange Crop Expert System using Rule Based System and Artificial Bee Colony Optimization Algorithm" , *International Journal of Engineering Science and Technology ,vol.2(6)*,2010.

[14] www.indiakisan.net

[15] Prof. M.S. Prasad Babu, N.Thirupathi Rao, "Implementation of Parallel Optimized ABC Algorithm with SMA Technique for Garlic Expert Advisory System", *International Journal of Computer Science, Engineering and Technology (IJCSET), Volume 1, Issue 3,* October 2010.

[16] Prof. M.S. Prasad Babu, M. Sudara Chinna, "Implementation of Rank Based Genetic Algorithm for Cotton Advisory System", *International Journal of Computer Science and Technology, vol 9(6*), 2010.

[17] Andrew Colin, "Building Decision Trees with the ID3 Algorithm", *Dr. Dobbs Journal June* 1996.

[18] Prof. M.S. Prasad Babu, Mrs. J. Anitha, K. Hari Krishna, "A Web Based Sweet Orange Crop Expert System using Rule Based System and Artificial Bee Colony Optimization Algorithm", *International Journal of Engineering Science and Technology ,vol.2(6)*,2010.

[19] Krogh, A., and Vedelsby, J., "Neural Networks Ensembles, Active Learners". *In Tesauro, G; Touretzky, D., and Leen, T., eds., NIPS-7, pp.231-238.Cambridge, MA: MIT press.* 1995

[20] J. R. Quinlan "Improved use of continuous attributes in C 4.5". *Journal of Artificial Intelligence Research, 4:77-90,* 1996.

Authors

NAVEEN KUMAR KORADA received the Bachelor's in Computer Science Engineering from JNTU and Master's Degree in Computer Science Technology with Artificial Intelligence and Robotics (CST with AI & R) from Andhra University, Visakhapatnam, and Andhra Pradesh, India. He is currently Software Engineer in TATA BSS. He has two published papers in International Journals. His area of interest includes Networks and Robotics.

N SAGAR PAVAN KUMAR received the Bachelor's in Computer Science Engineering from JNTU and Master's Degree in Software Engineering (SE) from GITAM University, Visakhapatnam, and Andhra Pradesh, India. He is currently Assistant Professor in the Department of CSE, Vignan College of Engineering (Women), Visakhapatnam.

Y V N H DEEKSHITULU received the Bachelor's in Computer Science Engineering from JNTU and Master's Degree in Computer Science Technology with Artificial Intelligence and Robotics (CST with AI & R) from Andhra University, Visakhapatnam, and Andhra Pradesh, India. He is currently Software Engineer in TATA BSS.