Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 4, April 2014, pg.1328 – 1335

RESEARCH ARTICLE

Development of Mushroom Expert System Based on SVM Classifier and Naive Bayes Classifier

Prof. M.S. Prasad Babu[1], Professor, Dept. of CS & SE, Andhra University, Visakhapatnam, A.P, India

Email-Id: msprasadbabu@yahoo.co.in

Rajani Thommandru[2], M.Tech, Dept. of CS & SE, Andhra University, Visakhapatnam,A.P, India

Email-Id: rajani.261@gmail.com

K.Swapna[3], **Research Scholar in CS & SE,** Andhra University, Visakhapatnam,A.P, India

Email-Id: pentaswapna@yahoo.co.in

E.Nilima[4], Assistant Professor in Dept. of CST, GITAM University, Visakhapatnam, A.P, India

Email-id: eadha.neelima@gmail.com

ABSTRACT:

Machine learning is the ability of a machine to improve its own performance through the use of a software that employs artificial intelligence techniques. In practice, this involves creating programs that optimize a performance criterion through the analysis of data. Support vector machine (SVM) is the machine learning algorithm used for creating the classification model from training data. In this paper, Support Vector Machine and Naïve Bayes algorithms are used for classification of mushrooms. Mushroom Expert System is developed for classification of mushrooms and to predict the class of mushrooms on submission of characteristics of the mushrooms. Programmed interviews are conducted with domain experts to build the knowledge base for the mushroom expert system. Performances of both algorithms are evaluated on mushroom data in fold cross-validation. This system is a web based application for online users and developed with JSP as front end and MySQL as backend.

Keywords: Expert System, Machine learning, Support vector machines, Naïve Bayes algorithm, JSP and MySQL

I. INTRODUCTION

1. Machine Learning:

Machine Learning [4] is the study of how computer programs can improve their performance automatically through experience. In practice, this involves creating programs that optimize a performance criterion through the analysis of data. Several applications such as Face recognition, Speech recognition and Optimize robot behavior have been developed to exhibit useful types learning. The important issue in development of expert system is knowledge acquisition from the experts to build the knowledge base. One technique to the knowledge acquisition is direct injection method in which knowledge is collected form domain experts by conducting interviews. But it is time consuming process. Instead, machine learning algorithms are used to learn from its past experience. Some of the machines learning algorithms are Genetic Algorithm [11], Decision Tree Algorithm [12], Optimization Algorithm [13], Adaptive Boosting Algorithm [14], Bagging Algorithm [15] and Particle Swarm Optimization Algorithm, Bayesian Classifier Algorithm.

2. Expert System:

The expert system [3] is a knowledge system which takes computer as a tool and makes use of the expertise and knowledge consequence to comprehend and solve the problem. It imitates the macroscopically inferential activity of expert, and uses computer to inference realm knowledge which is conformed to the model described. Expert system is consists of four parts, Knowledge base, inference engine, knowledge acquisition and explaining interface with the knowledge base and inference engine as its kernel. Increasing the reliability of the systems and lowering the time consumption of the farmers in going to Mycologists for the suggestions, the proposed system provides a good interface for getting the relevant information suggested by the experts. With machine learning, however we turn all that on its head and get computers to program themselves thus leads to user satisfaction.

3. Support Vector Machines (SVM):

SVM [2, 5] is a classification, technique that is originated as an implementation of Vapnik's (1995) structural risk minimization principle. SVM are based on mapping input space to a high-dimensional feature space where linear separation is easier than input space. SVM have been used successfully for the solution of many problems. Consider a training set T= $\{X_i, Y_i\}^N_{i=1,}$ where X_i is a real-valued n dimensional input vector (i.e. $X_i \in R^n$) and y Є {+1,-1} is a label that determines the class of X_i. The SVM employed for two class problems are based on hyper planes to separate the data. The hyper plane is determined by an orthogonal vector w and a bias b, which defines the points that satisfy $W^t.X + b = 0$. By finding a hyper plane that maximizes the margin of separation q, it is intuitively expected that the classifier will have better generalization ability. The hyper plane with the largest margin on the training set can be completely determined by the nearest points to the hyper plane. Two such points are x_1 and x_2 and they are called support vectors (SV). Therefore, in its simplest form, SVM learn linear decision rules as

F(x) = sign (W^t. X +b) so that (W, b) are determined to classify correctly the training examples and to maximize q. The margin q can be calculated as

$$q = \frac{2}{||W||}$$

So, Minimize $\frac{1}{2}||W||^2$ Subject to: $y_i \ (W^t.X_i + b) \geq 1$ to get the maximum marginal classifier so we introduced Lagrange Function for the SVM quadratic problem with linear constraints as follows

$$L(w,b,\alpha) = \frac{1}{2}||W||^2 - \sum \alpha_i \ (y_i((W.X_i\ ) + b) - 1)$$

Where, Lagrange multiplier, $\alpha_i \geq 0$

For L to be maximized, only training examples with $y_i.((X_i\ .W) + b) - 1 = 0$ (support vectors) will have $\alpha_i \neq 0$. As practical problems are not likely to be linearly separable, the linear SVM has been extended to a nonlinear version by mapping the training data to an expanded feature space using a nonlinear transformation $\Phi(x)$.Then, the maximum margin classifier of the data in the new space can be determined. With this procedure, the data that are non-separable in the original space may become separable in the expanded feature space. Since the training algorithm only depend on data through dot products. We can use a "kernel function" K such that

$$k(x_i, x_j) = \Phi(x_i).\Phi(x_j)$$

The most commonly used function for the dot product is the RBF kernel.

$k(a,b) = e^{-\gamma||a-b||^2}$ However, depending on the type of nonlinear mapping, the training points may not happen to be linearly separable, even in the expanded feature space. In this case, it will be impossible to find a linear classifier. Therefore, a new cost function is introduced,

$$V(W,\epsilon) = \frac{1}{2}W^t W + C\sum_{i=1}^{N}\epsilon_i$$

N non-negative slack variables ϵ_i are introduced to allow for training errors. C is a preselected positive penalty factor.

4. Naïve Bayes algorithm:

Bayesian Classifiers are statistical classifiers based on Bayes theorem A naive Bayes classifier works on principle of independence assumptions that is the presence (or absence) of a particular feature of a class is unrelated to the presence (or absence) of any other feature. It classifies the data based on the probability estimation function given the class prior probabilities. For each possible class label, multiply together the conditional probability of each feature, given the class label. This means, for us to implement the classifier, all we need to do, is compute these individual conditional probabilities for each label, for each feature, p(Fi | Cj), and multiply them together with the prior probability for that label p(Cj). The label for which we get the largest product, is the label returned by the classifier. In order to compute these individual conditional probabilities the Maximum Likelihood Estimation method is used.

II. KNOWLEDGE BASE

Knowledge base contains the data acquired from the domain experts. The information is collected from the experts by programmed interviews. The knowledge base of Mushroom Expert System contains the characteristics and class of mushrooms. The characteristics of the mushrooms and their class are represented in tabular format as shown**.**

S. No	Characteristics	Class of mushroom
1	Cap-shape is convex , cap-surface is smooth, cap-color is yellow, bruises are true, odor is almond, gill-attachment is free , gill-spacing is close, gill-size is broad, gill-color is black, stalk-shape is enlarging, stalk-root is club, stalk-surface-above-ring is smooth, stalk-surface-below-ring is smooth, stalk-color-above ring is white, stalk-color-below-ring is white, veil-type is partial, veil-color is white, ring-number is one, ring-type is pendant , spore-print-color is brown, population is numerous and habitat is grasses	Edible
2	Cap-shape is convex , cap-surface is smooth, cap-color is brown, bruises are true, odor is pungent, gill-attachment is free , gill-spacing is close, gill-size is narrow, gill-color is black, stalk-shape is enlarging, stalk-root is equal, stalk-surface-above-ring is smooth, stalk-surface-below-ring is smooth, stalk-color-above ring is white, stalk-color-below-ring is white, veil-type is partial, veil-color is white, ring-number is one, ring-type is pendant , spore-print-color is black, population is scattered and habitat is urban	Poisonous
3	Cap-shape is flat , cap-surface is scaly, cap-color is yellow, bruises are false, odor is fishy, gill-attachment is free , gill-spacing is close, gill-size is broad, gill-color is gray, stalk-shape is enlarging, stalk-root is bulbous, stalk-surface-above-ring is silky, stalk-surface-below-ring is silky, stalk-color-above ring is pink, stalk-color-below-ring is pink, veil-type is partial, veil-color is white, ring-number is one, ring-type is large, spore-print-color is chocolate, population is solitary and habitat is grasses	Poisonous

The information collected from the experts is stored in the database table as rules .The rules are in the form as shown below.
Rule 1: Cap-shape is x , cap-surface is s, cap-color is y, bruises are t, odor is a, gill-attachment is f , gill-spacing is c, gill-size is b, gill-color is k, stalk-shape is e, stalk-root is c, stalk-surface-above-ring is s, stalk-surface-below-ring is s, stalk-color-above ring is w, stalk-color-below-ring is w, veil-type is p, veil-color is w, ring-number is o, ring-type is p , spore-print-color is b, population is n and habitat is g.
Then **Edible**

Rule 2: Cap-shape is x , cap-surface is s, cap-color is b, bruises are t, odor is p, gill-attachment is f , gill-spacing is c, gill-size is n, gill-color is k, stalk-shape is e, stalk-root is e, stalk-surface-above-ring is s, stalk-surface-below-ring is s, stalk-color-above ring is w, stalk-color-below-ring is w, veil-type is p, veil-color is w, ring-number is o, ring-type is p , spore-print-color is k, population is s and habitat is u
Then **Poisonous**

Rule 3: Cap-shape is f , cap-surface is y, cap-color is y, bruises are f, odor is f, gill-attachment is f , gill-spacing is c, gill-size is b, gill-color is y, stalk-shape is e, stalk-root is b, stalk-surface-above-ring is k, stalk-surface-below-ring is k, stalk-color-above ring is p, stalk-color-below-ring is p, veil-type is p, veil-color is w, ring-number is o, ring-type is l, spore-print-color is h, population is y and habitat is g.
Then **Poisonous**

The Table depicts the part of data that have collected from the Domain experts for the classification. The attributes or columns of the database table are the different characteristics of the mushrooms and their values are of the type nominal (characters).

III. Sequential Minimal Optimization (SMO)

SMO [7] solves the Lagrange Function by decomposing it into sub problems and solving the smallest possible optimization problem, involving two Lagrange multipliers, at each step.

- Select two α parameters: α_i and α_j
- Repeat while passes<maxpasses
  For i = 1……m
  Select for from i=1……m
  Select α_i at random
  If αi does not fulfill KKT conditions
  Then select α j from remaining (m-1) α's
  End for.
  Compute α = (αi +αj)/2
  End while.
- Select the threshold b such that both i^{th} and j^{th} examples satisfy KKT conditions
- The KKT conditions are

$\alpha = 0 \quad \Rightarrow \quad y(w^T + b) > 1$
$\alpha = C \quad \Rightarrow \quad y(w^T x + b) < 1$
$0 < \alpha < C \quad \Rightarrow \quad y(w^T x + b) = 1$

Pseudo code for the SMO algorithm:

Input:
C: regularization parameter
tol: numerical tolerance
max passes: max # of times to iterate over α's without changing
$(x^{(1)}, y^{(1)}), \ldots, (x^{(m)}, y^{(m)})$: training data
Output:
$\alpha \in R^m$: Lagrange multipliers for solution
$b \in R$: threshold for solution
Initialize $\alpha_i = 0$, i, b = 0.
Initialize passes = 0.
- while (passes < max passes)
  num_changed_alphas = 0.
- For i = 1, . . .m,
- Calculate $E_i = f(x^{(i)}) - y^{(i)}$

- if (($y^{(i)}E_i < -tol$ && $\alpha_i < C$) || ($y^{(i)}E_i > tol$ && $\alpha_i > 0$))
  Select $j \neq i$ randomly.
  Calculate $E_j = f(x^{(j)}) - y^{(j)}$
  Save old α's: $\alpha_i^{(old)} = \alpha_i$ and $\alpha_j^{(old)} = \alpha_j$ Compute L and H {if $y^i \neq y^j$ L=max 0, α_j- α_i),

H=min(C, C+ α_j - α_i)

Otherwise L=max (0, α_j +α_i -C), H=min(C, α_j + α_i)

- if (L == H)
  Continue to next i.
- Compute ŋ
- if (ŋ >= 0)
  Continue to next i.
- Compute and clip new value for α_j
- if ($|\alpha_j - \alpha^{(old)}| < 10^{-5}$)
  Continue to next i.
- Determine value for α_i
- Compute b1 and b2
- Compute b
- num_changed_alphas = num_changed_alphas+1
- end if
- end for
- if (num_ changed_ alphas == 0)
  Passes = Passes + 1
- Else
  Passes = 0

➢ end while

The flow diagram of the proposed SMO algorithm used in development of Mushroom Expert System is shown in Fig: 2. on giving the training data as input to the algorithm and calculates the value of α and b fro the classification of data.

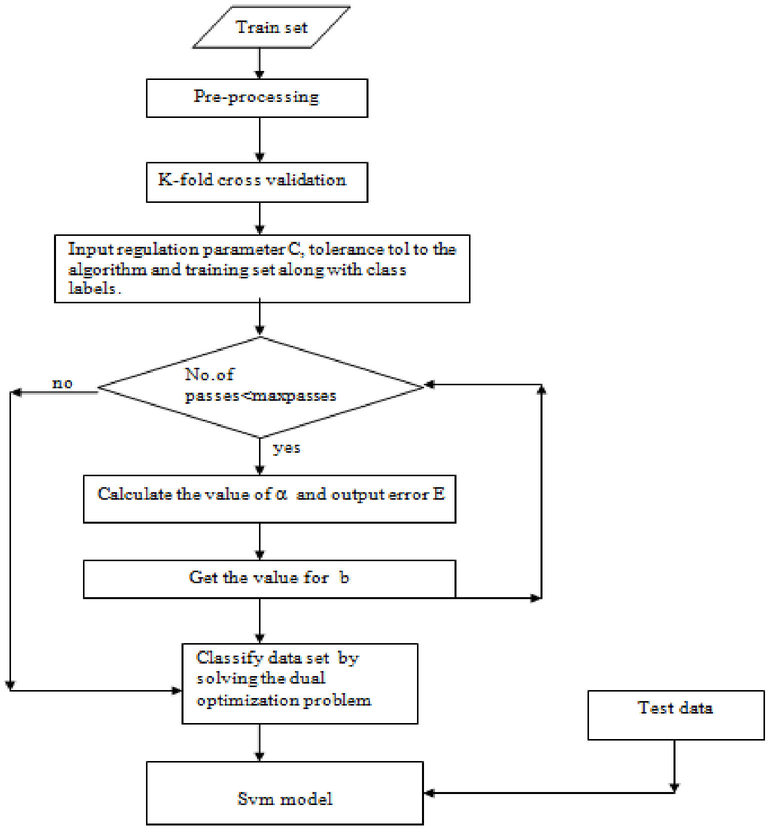

Fig2: SVM-based classification using SMO

IV. COMPARATIVE STUDY

The performance of the SVM classifier using SMO and Naïve Bayes classifier on Mushroom data set is evaluated using the accuracies obtained in each fold of the cross-fold validation. The 10-fold cross validation is performed on mushroom data meaning that the whole training dataset is divided into 10 subsets of equal length, each of which was in turn used as an independent test data set. In each subset 75 % data is used for training the classifier and 25 % data is used for testing the classifier.

No. of folds	Accuracy using Svm classifier (%)	Accuracy using Naïve Bayes classifier (%)
2	87.2	82.4
3	86.4	82.6
4	86.4	82.1
5	87.7	82.4
6	86.3	82.4
7	86.3	81.1
8	86.4	82.6
9	86.3	83.0
10	86.7	82.4

Table 1: Accuracy values

A graph is drawn between the accuracies and No. of folds that were obtained in each fold while performing the classification on mushroom data as shown below based on the values of obtained while performing the fold wise cross validation.

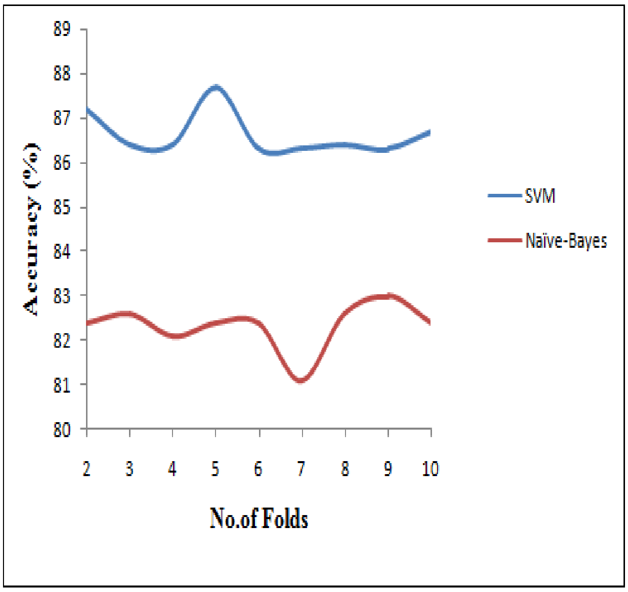

Fig3: Graph describing the performances of SVM and Naïve Bayes

Based on above graph, the SVM Classifier gives better accuracy when compared to Naïve Bayes classifier on mushroom data. So, SVM based classification is used for the prediction of unclassified or new instances.

V. MUSHROOM EXPERT SYSTEM ARCHITECTURE

The Mushroom Expert System, SVM and Naive Bayes algorithms are implemented to classify the mushroom data. Basing on the classification model the system can predict the class of mushrooms on submission of Mushroom characteristics. This system consists of SVM classification model, SMO algorithm and knowledge base which are used in the inference mechanism as shown in Fig3.

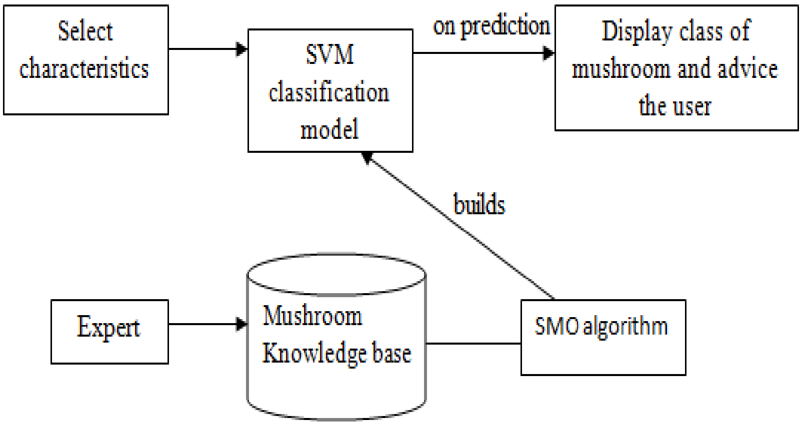

Fig4: Mushroom Expert System Architecture

VI. RESULTS

By applying the proposed algorithm SMO the results are below

Fig 5: screenshot for selection of characteristics of Mushrooms

Description: In this screenshot, user can submit the characteristics of the mushrooms by selecting options in the dropdown list.

Fig 6: screenshot that displays the class of mushrooms as result

Description: In this screen shot, the algorithm classifies the given input as Poisonous mushrooms and advices user on eating them.

VII.CONCLUSION

This paper presents a new application domain for the use of SVM. In this, SVM used for the classification of training data and predicting the class of new instance based on the classification model built by SVM on training process .Experiments have conducted on the mushrooms data shows that SVM gives better accuracy when compared to Naïve Bayer's algorithm. The performance of the SVM classifier is evaluated using the accuracy. The results obtained shows that SVM is an efficient tool for this application domain.

REFERENCES

[1] Missouri Dept. of Conservation. "Poisonous Mushrooms."
http://mdc.mo.gov/nathis/mushrooms/mushroom/poisonous.htm

[2] Fast training of SVM's using Sequential Minimal Optimization by Sudip Khanna & Ajit Datar.

[3] Mustafa Acikka , and Mehmet Fatih Akay " Support vector machines for predicting the admission decision of a candidate to the School of Physical Education and Sports at Cukurova University" Expert Systems with Applications Volume 36, Issue 3, Part 2, April 2009, Pages 7228-7233.

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

[5] A.J. Smola, P.L. Bartlett, B. Schölkopf, and D. Schuurmans "Advances in Large Margin Classifiers" MIT Press, 2000.

[6] Ray LaBarge. ECE 539. 11/11/2008. "Distinguishing Poisonous from Edible Wild Mushrooms".

[7]. CS 229, Autumn 2009 "The Simplified SMO Algorithm".

[8]. Bendi Venkata Ramana and Prof. M. Surendra Prasad Babu "A Critical Study of Selected Classification Algorithms for Liver Disease Diagnosis" International Journal of Database Management Systems (IJDMS), Vol.3, No.2, May 2011.

[9]. Prof. M.S. Prasad Babu, N.V. Ramana Murty and S.V.N.L. Narayana "A Web based tomato crop expert Information system based on Artificial Intelligence ISSN: 0975-9646 AND MACHINE LEARNING ALGORITHMS" International Journal of Computer Science and Information Technologies**,** Vol. 1 (1), 2010, 6-15.

[10]. "Several Views of Support Vector Machines" by *Charlie Frogner* 9.520: Statistical Learning Theory and Applications February 17th, 2010.

[11]. ZHANG. J, Chung. H and Lo. W. L, "Clustering-Based Adaptive Crossover and Mutation Probabilities for Genetic Algorithms", IEEE Transactions on Evolutionary Computation vol.11, no.3, pp. 326–335, 2007.

[12]. Rokach, L.; Maimon, O. (2005). "Top-down induction of decision trees classifiers-a survey". *IEEE Transactions on Systems, Man, and Cybernetics, Part C* 35 (4): 476–487.doi:10.1109/TSMCC.2004.843247.

[13]Ashlock, D. (2006), *Evolutionary Computation for Modeling and Optimization*, Springer, ISBN 0-387-22196-4.

[14]. 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.

[15] Bühlmann, P., Yu, B.: Analyzing bagging. Annals of Statistics 30, 927-961 (2002).