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. 11, November 2014, pg.201 – 206

RESEARCH ARTICLE

An Endowed Takagi-Sugeno-type Fuzzy Model for Classification Problems

S.Thenmozhi[1], G.Misal[2]

PG Scholar[1], Asst. Professor[2]

Department of Computer Science & Engineering[1, 2]

V.S.B. Engineering College, Karur, India[1, 2]

thenmozhi2991@gmail.com [1], missal.india@gmail.com [2]

Abstract— Takagi-Sugeno-type fuzzy classification model is improved by proposing a new incremental learning approach. Margin selective gradient descent learning and incremental support vector machine helps the proposed fuzzy model to learn with high generalization ability. Training samples are provided incrementally one after another instead of in a single batch. This improved fuzzy model is obtained from an empty set and clustering algorithm is used for finding initial fuzzy sets and number of rules in the antecedent part. The generalization ability of the fuzzy model can be improved by proposing an online incremental linear support vector machine to tune the rule consequent parameters. Online training problem can be solved by incremental support vector machine, where only one new training sample is provided at a time. Margin selective gradient descent algorithm is used to learn antecedent parameter and avoids overtraining.

Keywords— Batch support vector machine, high generalization ability, new incremental learning approach, one-pass clustering algorithm, online training samples

I. INTRODUCTION

A model is designed by data-driven machine learning approaches using training samples. Two different modes of learning scenario are batch mode and incremental mode. In a batch learning mode, memory is not sufficient for handling all training data at a time because all training data are collected in advance and trained in a batch. In an incremental learning mode, training data are segmented and learning can done in an online fashion sequentially. Unlike batch mode, an incremental mode processes single training sample or a chunk of training samples at a time. Memory consumption is reduced for training data storage by this learning approach and is efficient for the applications where data are provided one-by-one. Fuzzy classifiers are learned to minimize the training errors.

II. PREVIOUS WORK

An incremental support vector machine is classified into two categories such as online incremental support vector machine[1]-[7] and chunk incremental support vector machine[8]-[12]. In online ISVMs, only one training sample is trained at a time and its solution is constructed recursively. In chunk ISVMs, one subset of training data are used to train support vector machine and its solution is found with the help of traditional batch SVMs. Therefore, learning efficiency is improved using chunk ISVMs by removing some new training samples based on their locations with respect to its classification hyperplane. One main disadvantage in chunk ISVMs is its unsuitableness for the new arrival of training sample. SVMs have number of support vector leads to a large support vector machine model irrespective of its learning approaches.

© 2014, IJCSMC All Rights Reserved

III. PROPOSED WORK

In the proposed system generalization ability is improved by designing the fuzzy classification model with the help of support vector machine. One common approach is support vectors are considered as fuzzy rules and both are equal in a batch SVM. Fuzzy classifiers may degenerate the performance by removing the irrelevant fuzzy rule from the original rule sets. However, clustering algorithm may helps to provide a small rule set where the rules are equal to number of clusters[13]-[16]. Antecedent parameters are not tuned by chunk ISVM and batch SVM-trained FC. To avoid these problems, online incremental SVM and margin selective gradient descent algorithm are proposed for tuning consequent and antecedent parameters. Online incremental SVM obtains high generalization ability and optimal consequent parameters. It is also used to tune samples one at a time to avoid huge data access. Antecedent parameter of a fuzzy classifier is tuned by margin selective gradient descent algorithm. This type of tuning may be done from two support vector located hyperplane using only training samples. The slack variable values are minimized by performing the operation of moving the mapped training samples in the consequent space to proper region.

IV. IMPLEMENTATION

A. SYSTEM DESIGN

The above discussed fuzzy model is represented as a diagram given below in the Fig.1.

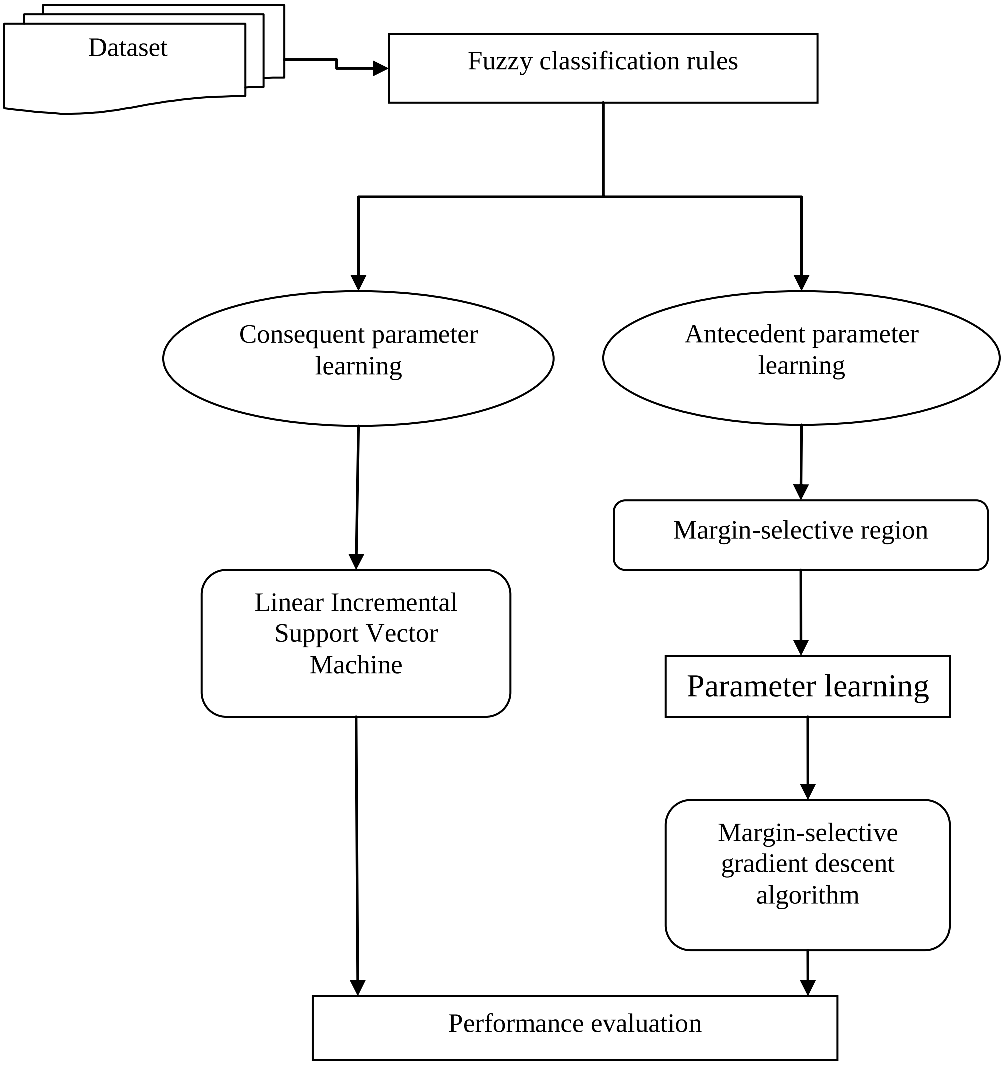

Fig.1. Fuzzy model architecture

Here input is given from the dataset and fuzzy classification rules are initialized for that given input. The data set taken for this type of fuzzy model has limited input values only. After rule initialization both parameter

© 2014, IJCSMC All Rights Reserved

learning can be done with the help of their respective learning approach algorithms. Finally, the performance of this fuzzy model can be calculated by measuring the accuracy and error rate.

B. SYSTEM MODULES

1). Fuzzy Classification Rules: The fuzzy model consists of first order Takagi-Sugeno-type fuzzy IF-THEN rules. These rules are in the following form:

Rule i: If x_1 is A_{i1} and ... and x_n is A_{in}

$$\text{Then} \quad \hat{y}' = a_{i0} + \sum_{j=1}^{n} a_{ij} \tag{1}$$

where $x_1,...,x_n$ are inputs, A_{ij} is denoted as a fuzzy set and a_{ij} is represented as a real number. The following Gaussian membership function is applied for the fuzzy model as:

$$M_{ij}(x_j)=\exp\{-(x_j-m_{ij})^2/2\sigma^2_{ij}\} \tag{2}$$

here m_{ij} and σ_{ij} are center and width of the fuzzy set. σ_{ij} in different fuzzy sets are useful for the improvement of classification performance. The rule firing length is determined by

$$\mu_i(\vec{x})=\prod_{j=1}^{n} M_{ij}(x_j)=\exp\{-\sum_{j=1}^{n}[(x_j-m_{ij})^2/2\sigma^2_{ij}]\} \tag{3}$$

where $\vec{x}=[x_1,...,x_n]$. If there are r rules the output calculation can be done by adding simple weight sum operations and is shown as:

$$y' = \sum_{i=1}^{r} \mu_i \left(\sum_{j=0}^{n} a_{ij}\, x_j\right) + b = \sum_{i=1}^{r}\sum_{j=0}^{n} a_{ij}\, \mu_i x_j + b \tag{4}$$

The following can be added to the original T-S-type fuzzy model by the addition of the bias term b as:

Rule i+1 : IF *x1* is A*r*+1*n and ... and xn is* A*r*+*1n*

$$\text{Then} \quad \hat{y}' = \sum_{j=1}^{n} a_{r+1j}\, x_j + b$$

where $M_{r+11}(x_1)=...= M_{r+1n}(x_n)\equiv 1$ and $a_{r+11}=...=a_{r+1n}=0$.

The one-pass clustering algorithm helps to generate this fuzzy model from an empty rule set. The basic concept of using clustering algorithm is to consider the clusters as a fuzzy rule for rule generation. The rule firing length $\mu_i(\vec{x})$ is used for generating rule as online and for each datum; the cluster I with maximum firing strength is obtained as:

$$I=\arg\max_{1\le i\le r} \mu_i(\vec{x}) \tag{5}$$

If $\mu_I< \mu_{th}$, then a new cluster will be generated, that is $r\leftarrow r+1$. If $\mu_I> \mu_{th}$, then no new cluster will be generated. A new Gaussian fuzzy set A_{rj} is generated in each variable x_j before the rule generation caused by input datum $\vec{x}(t)$ and the following equation is:

$$m_{rj} =x_j(t),\ \sigma_{rj}=\sigma_c, \quad j=1,...,n \tag{6}$$

where σ_c is denoted as a pre-defined constant as 0.5 and a smaller number of rules are generated by the larger σ_c values.

2). Online Linear Incremental Support Vector Machine for Consequent Parameter Learning: In this online linear incremental support vector machine parameter learning, all training data and their labels are assumed to be given by a set $D^{(N)} = \{(\vec{x}_1,y_1),(\vec{x}_2 y_2),...,(\vec{x}_N,y_N)\}$, where $\vec{x}_i \epsilon \mathbb{R}^n$ and $y_i \in (+1,-1\}$. The following softly constrained optimization problem can be solved by the batch SVM learning approach with the help of finding separate function $f(\vec{x}) = \vec{w}^T\vec{x} + b$ as:

$$\text{Min}_{\vec{w},\xi_i,b} \frac{1}{2} w^T\vec{w} + C\sum_{i=1}^{N}\xi_i$$

$$\text{Subject to } y_i(\vec{w}^T\vec{x}_i + b \ge 1 - \xi_i, \xi_i \ge 0, \quad i = 1 \dots N \tag{7}$$

where $0.5\, w^T\vec{w} + C\sum_{i=1}^{N}\xi_i$ is the cost function called soft margin, C is denoted as user-defined positive parameter, ξ_i is a slack variable, and $\sum \xi_i$ is denoted as an upper bound on number of training errors. This optimization problem has its solution is in [1] as:

$$\vec{w} = \sum_{i=1}^{N} \alpha_i y_i \vec{x}_i \tag{8}$$

here α_i is a Lagrange multiplier and the separating function is:

$$f(\vec{x}) = \vec{w}^T\vec{x} + b = \sum_{i=1}^{N} y_i\alpha_i < \vec{x},\vec{x}_i > +b$$

$$\sum_{i\in SV}^{N} y_i\alpha_i < \vec{x},\vec{x}_i > +b \tag{9}$$

The necessary conditions from Karush-Kuhn-Tucker (KKT) are:

$$\times \begin{cases} \ge 0, & if\ \alpha_i = 0 \\ = 0 & if\ 0 < \alpha_i < C \\ \le 0 & if\ \alpha_i = C \end{cases} \tag{10}$$

Based on this above condition from (10), $D^{(N)}$ is divided into three categories as: for ($0 < \alpha_i < C$), the set S is represented as unbounded SVs. For ($\alpha_i = C$), the set E is represented as bounded SVs. Finally for ($\alpha_i = 0$),

© 2014, IJCSMC All Rights Reserved

the set *O* represents the non-SVs. In an online incremental SVM, one training sample is trained at a time and the existing optimal solution helps to build a new solution. When a new training sample $(\vec{x}_{N+1}, y_{N+1})$ is available, the old solution will be updated and assume that the samples in $D^{(N)}$ never migrate across sets *S*, *O*, and *E*.
The fuzzy model using this online linear ISVM tune the free antecedent parameter a_{ij} in the consequent part. The output in (4) is as follows:

$$y' = \vec{a}T\vec{\emptyset} + b \quad (11)$$

where

$$\vec{a} = [a_{10} \dots a_{1n} \dots \dots a_{r0} \dots a_{rn}]T \epsilon \mathbb{R}^{r(n+1)\times 1} \quad (12)$$

and

$$\vec{\emptyset} = [\mu_1 x_0 \dots \mu_1 x_n \dots \dots \mu_r x_0 \dots \mu_r x_n]T \epsilon \mathbb{R}^{r(n+1)\times 1} \quad (13)$$

The online linear incremental SVM evaluates only the consequent parameter vector $\vec{a}$ instead of changing the structure of T-S-type fuzzy model[18]-[20]. From (13), the input datum $\vec{x}_k = [x_{k0}, \dots, x_{kn}]$ in online linear ISVM-trained T-S-type fuzzy model is converted to the feature vector $\vec{\emptyset}_k$, where

$$\vec{\emptyset}_k = [\mu_1 x_{k0} \dots \mu_1 x_{kn} \dots \dots \mu_r x_{kn}]^T \in R^{r(n+1)\times 1} \quad (14)$$

The feature space has training data pairs which are represented as follows:

$$D^{(N)} = \{(\vec{\emptyset}_1, y_1), (\vec{\emptyset}_2, y_2) \dots (\vec{\emptyset}_N, y_N)\} \quad (15)$$

The optimal linear separating function in the feature space from (9) is

$$y'(\vec{\emptyset}) = \sum_{k=1}^{N} y_k \alpha_k < \vec{\emptyset}, \vec{\emptyset}_k > + b \quad (16)$$

The parameters a_{ij} are solved accordingly and the solution is given as follows:

$$a_{ij} = \sum_{k=1}^{N} y_k \alpha_k \mu_i x_{kj} = \sum_{k \in SV} y_k \alpha_k \mu_i x_{kj} \quad (17)$$

The new consequent parameters $a_{ij}^{(N+1)}$ may be obtained by the substitution of new coefficient $\alpha_i^{(N+1)}$ as follows:

$$a_{ij}^{(N+1)} = \sum_{k=1}^{N+1} y_k \alpha_k^{(N+1)} \mu_i x_{kj} \quad (18)$$

The parameter a_{ij} gets updated using $a_{ij}^{(N+1)}$ in the recursive form , and the update equation is as follows:

$$a_{ij}^{(N+1)} = \sum_{k=1}^{N+1} y_k (\alpha_k + \Delta\alpha_k) \mu_i x_{kj} \quad (19)$$

The new bias value $b^{(N+1)}$ is updated. Minimization of the soft margin helps to reduce the bound on the generalization error in the feature space and helps to improve the fuzzy model generalization ability. The recursive update makes it feasible to train the fuzzy model with one sample at a time to minimize the soft margin.

3). Margin Selective Gradient Descent Algorithm for Antecedent Parameter Learning: From the equation (6), m_{ij} and σ_{ij} are determined. The generalization performance reduction and overtraining are caused by minimizing output error in all training samples with the help of gradient descent algorithm in neural fuzzy classifiers. MGDA is mainly used to avoid overtraining and also for increasing the separation margin width through learning training samples in the margin-selective region. The slack variable ξ_k after online ISVM is:

$$\xi_k = \begin{cases} 0, & y_k y_k' \geq 1 \\ 1 - y_k y_k', & otherwise \end{cases} \quad (20)$$

The main objective of MGDA is to minimize ξ_k in the feature space using the following function as:

$$e_k = \frac{1}{2}\xi_k^2 = \frac{1}{2}(y_k' - y_k)^2, \quad y_k y_k' < 1 \quad (21)$$

The MGDA is not applied to the samples in classified region and the misclassified samples in MS region. Therefore, MGDA is mainly applied to the training samples which are located in the margin-selective region by satisfying the condition as $-1-\beta \leq y_k y_k' < 1$. The parameters m_{ij} and σ_{ij} for an input sample $\vec{x}_k$ are updated as follows:

$$m_{ij}(k+1) = m_{ij}(k) - \frac{\partial e_k}{\partial m_{ij}}, \quad i=1,\dots,r, \quad j=1,\dots,n \quad (22)$$

where

$$\frac{\partial e_k}{\partial m_{ij}} = (y_k' - y_k)(\textstyle\sum_{j=0}^{n} a_{ij} x_{kj}) \mu_i \frac{2(x_{kj} - m_{ij})}{\sigma_{ij}^2} \quad (23)$$

and

$$\sigma_{ij}(k+1) = \sigma_{ij}(k) = \frac{\partial e_k}{\partial \sigma_{ij}}, \quad i=1,\dots,r, \quad j=1,\dots,n \quad (24)$$

© 2014, IJCSMC All Rights Reserved

where

$$\frac{\partial e_k}{\partial \sigma_{ij}} = \left(y_k^{'} - y_k\right)\left(\sum_{j=0}^{n} a_{ij} x_{kj}\right)\mu_i \frac{2(x_{kj}-m_{ij})^2}{\sigma_{ij}^3} \quad (25)$$

For an offline dataset, the linear online ISVM and MGDA are incrementally applied to the consequent and antecedent parameter learning for the whole dataset, because all training samples are available in advance. For an online training dataset, the linear online ISVM and MGDA parameter learning are applied to only one training sample at a time, because online training samples deals one sample at a time without any repeated training procedures.

V. Conclusions

An improved fuzzy model is proposed for the classification problems in T-S-type fuzzy model. To achieve a small model clusters are used for determining the rule number in instead of using support vectors. Generalization ability of this fuzzy model is improved with the ISVM for parameter tuning and also helps to reduce a soft margin. In addition, online classification problems are also handled by online ISVM. Batch training algorithm causes problems such as preload of the large amount of training data and this can be avoided by using online ISVM. Antecedent parameter learning by margin selective gradient descent algorithm performs an operation of moving the selected mapped training samples in the consequent space to a proper location.

References

[1] C. P. Diehl and G. Cauwenberghs, “SVM incremental learning, adaptation and optimization,” in *Proc. Int. Joint Conf. Neural Netw.*, 2003, pp. 2685–2690.

[2] A. Shilton, M. Palaniswami, D. Ralph, and A. C. Tsoi, “Incremental training of support vector machines,” *IEEE Trans. Neural Netw.*, vol. 16, no. 1, pp. 114–131, Jan. 2005.

[3] P. Laskov, C.Gehl, S. Kr¨uger, andK. R.M¨uller, “Incremental support vector learning: Analysis, implementation and applications,” *Mach. Learn. Res.*, vol. 7, pp. 1909–1936, 2006.

[4] G. Cauwenberghs and T. Poggio, “Incremental and decremental support vector machine learning,” in *Advances in Neural Information Processing Systems*, T. K. Leen, T. G. Dietterich, and V. Tresp, Eds. Cambridge, MA, USA: MIT Press, 2000, vol. 13, pp. 409–415.

[5] M. Karasuyama and I. Takeuchi, “Multiple incremental decremental learning of support vector machines,” *IEEE Trans. Neural Netw.*, vol. 21, no. 7, pp. 1048–1059, Jul. 2010.

[6] M. Karasuyama and I. Takeuchi, “Multiple incremental decremental learning of support vector machines,” *IEEE Trans. Neural Netw.*, vol. 21, no. 7, pp. 1048–1059, Jul. 2010.

[7] S. Ertekin, L. Bottou, and C. L. Giles, “Nonconvex online support vector machines,” *IEEE Trans. Pattern Anal.Mach. Intell.*, vol. 33, no. 2, pp. 368–381, Feb. 2011.

[8] R. Roscher, B. Waske, and W. Forstner, “Incremental import vector machines for classifying hyper spectral data,” *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 9, pp. 3463–3473, Sep. 2012.

[9] P. Mitra, C. A. Murthy, and S. K. Pal, “Data condensation in large databases by incremental learning with support vector machines,” in *Proc. Int. Conf. Pattern Recog.*, 2000, pp. 708–711.

[10] C. Domeniconi and D. Gunopulos, “Incremental support vector machine construction,” in *Proc. Int. Conf. Data Mining*, 2001, pp. 589–592.

[11] S. Ruping, “Incremental learning with support vector machines,” in *Prof. Int. Conf. Data Mining*, 2001, pp. 641–642.

[12] K. Ikeda and T. Yamasaki, “Incremental support vector machines and their geometrical analyses,” *Neuro computing*, vol. 70, pp. 2528–2533, 2007.

[13] S. Cheng and F. Y. Shih, “An improved incremental training algorithm for support vector machines using active query,” *Pattern Recog.*, vol. 40, pp. 964–971, 2007.

[14] C. F. Juang, S. H. Shiu, and S. J. Shiu, “Fuzzy system learned through fuzzy clustering and support vector machine for human skin color segmentation,” *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 37, no. 6, pp. 1077–1087, Nov. 2007.

[15] C. F. Juang, S. H. Chiu, and S. W. Chang, “A self-organizing TS-type fuzzy network with support vector learning and its application to classification problems,” *IEEE Trans. Fuzzy Syst.*, vol. 15, no. 5, pp. 998–1008, Oct. 2007.

[16] C. F. Juang and G. C. Chen, “A TS fuzzy system learned through a support vector machine in principal component space for real-time object detection,” *IEEE Trans. Ind. Electron.*, vol. 59, no. 8, pp. 3309–3320, Aug. 2012.

© 2014, IJCSMC All Rights Reserved

[17] G. C. Chen and C. F. Juang, "Object detection using color entropies and a fuzzy classifier," *IEEE Computer. Intell. Mag.*, vol. 8, no. 1, pp. 33–45, Feb. 2013.
[18] Y. Chen and J. Z. Wang, "Support vector learning for fuzzy rule-based classification systems," *IEEE Trans. Fuzzy Syst.*, vol. 11, no. 6, pp. 716–728, Dec. 2003.
[19] C. T. Lin, C. M. Yeh, S. F. Liang, J. F. Chung, and N. Kumar, "Support vector- based fuzzy neural network for pattern classification," *IEEE Trans. Fuzzy Syst.*, vol. 14, no. 1, pp. 31–41, Feb. 2006.
[20] S. M. Zhou and J. Q. Gan, "Constructing L2-SVM-based fuzzy classifiers in high-dimensional space with automatic model selection and fuzzy rule ranking," *IEEE Trans. Fuzzy Syst.*, vol. 15, no. 3, pp. 398–409, Jun. 2007.

© 2014, IJCSMC All Rights Reserved