Combination Methodologies of Multi-agent Hyper Surface Classifiers: Design and Implementation Issues

Qing He [1,2], Xiu-Rong Zhao [1,2], Ping Luo [1,2], and Zhong-Zhi Shi [1,2]

[1] The Key Laboratory of Intelligent Information Processing,
Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100080, China
[2] Graduate University of Chinese Academy of Sciences,
Beijing 100039, China
{heq,zhaoxr,luop,shizz}@ics.ict.ac.cn

Abstract. This paper describes a new framework using intelligent agents for pattern recognition. Based on Jordan Curve Theorem, a universal classification method called Hyper Surface Classifier (HSC) has been studied since 2002. We propose multi-agents based technology to realize the combination of Hyper Surface Classifiers. Agents can imitate human beings' group decision to solve problems. We use two types of agents: the classifier training agent and the classifier combining agent. Each classifier training agent is responsible to read a vertical slice of the samples and train the local classifier, while the classifier combining agent is designed to combine the classification results of all the classifier training agents. The key of our method is that the sub-datasets for the classifier training agents are obtained by dividing the features rather than by dividing the sample set in distribution environment. Experimental results show that this method has a preferable performance on high dimensional datasets.

1 Introduction

The combination of multiple classifiers can be considered as a generic pattern recognition problem in which the input consists of the results of the individual classifiers, and the output is the combined decision. For this purpose, many developed classification techniques can be applied; in fact, classification techniques such as neural networks and polynomial classifiers have served to combine the results of multiple classifiers. In this area, the early work can be found in [24, 25]. This approach almost immediately produced promising results. In this domain technological developments has increased and grown tremendously [7]-[19], partly as a result of the coincident advances in the technology itself. These include the production of very fast and low cost computers that have made many complex algorithms practicable, among which are many pattern recognition algorithms. In paper [20], L. Xu, A. Krzyzak, and C. Y. Suen propose to conduct a more systematical investigation into the problem of multi-classifier combination. Their idea consists of two parts. The first part, being closely dependent on the specific applications, includes the problems of "How many classifiers are chosen for a specific application problem? What kind of classifiers should be used? And for each classifier what types of features should be chosen?" as well as other problems that

V. Gorodetsky et al. (Eds.): AIS-ADM 2007, LNAI 4476, pp. 100–113, 2007.

relate to the construction of those individual and complementary classifiers. In paper [15], J. Kittler presents a theoretical framework for combining soft decision outputs of multiple experts employing mixed (some shared and some distinct) representations of patterns to be classified. In paper [22], Ching Y. Suen and Louisa Lam examine the main combination methods that have been developed for different levels of classifier outputs - abstract level, ranked list of classes, and measurements. In paper [23], Shaodan Lin etc. suggests a technology of information fusion using multiple agents, each of which uses a quite different classification algorithm such as decision tree algorithm, simple Naïve Bayes algorithm and the newly emerging classification algorithm based on atomic association rules. In paper [22] Ahmed Al-Ani and Mohamed Deriche present a classifier combination technique based on the Dempster-Shafer theory of evidence, which is a powerful method for combining measures of evidence from different classifiers.

In this paper, we propose multi-agents based technology to realize the combination of Hyper Surface Classifiers (HSC). Agents can imitate human beings' group decision to solve problems in the same way as group consultations of doctors. In [1] [2] a new classification method based on hyper surface (HSC) is put forward. In this method, a model of hyper surface is obtained during the training process and then it is directly used to classify large database according to whether the wind number is odd or even based on Jordan Curve Theorem. Experiments show that HSC can efficiently and accurately classify large-sized data sets in two-dimensional space and three-dimensional space. Though HSC can classify higher dimensional data according to Jordan Curve Theorem in theory, it is not as easy to realize HSC in higher dimensional space as in three-dimensional space. However, what we really need is an algorithm that can deal with data not only of massive size but also of high dimensionality. Thus in [3] a simple and effective kind of dimension reduction method without losing any essential information is proposed, which is a dimension changing method in nature. In paper [29], based on the idea of ensemble, another solution to the problem of HSC on high dimensional data sets is proposed and proven to have a preferable performance by experiments. However multi agent technique has not been utilized in paper [29]. Recently, Distributed Data Mining (DDM) [30][31] has attracted lots of attention among the data mining community. DDM refers to the mining of inherently distributed datasets, aiming to generate global patterns from the union set of locally distributed data. However, the security issue among different local datasets and the huge communication cost in data migration prevent moving all the datasets to a public site. Thus, the algorithms of DDM often adopt a computing paradigm of local processing and global synthesizing, which means that the mining process takes place at a local level and then at a global level where local data mining results are combined to gain global findings. Considering the problem of distributed computing environment and that agents can imitate human beings' group decision to solve problems just as group consultations of doctors, in this paper we extend on our work in [29]. The multi-agent techniques are introduced to realize the combination of Hyper Surface Classifiers. A combination of classifiers is a set of classifiers, whose individual classification decisions are combined in some way, typically by a weighted or equal voting, to classify new examples. Generally speaking, there're two ways of combination, horizontal combination and vertical combination. Here, vertical combination is adopted. The combination of HSC is constructed by dimension dividing rather than dimension reduction for high dimensional data. We use two types of agents: the classifier training agent and the classifier combining agent. Each classifier training agent is responsible to read a vertical slice of the samples and train the local classifier, while the classifier combining agent is designed to combine the classification results of all the classifier training agents.

The rest of this paper is organized as follows: In section 2, we give an outline of hyper surface classifier. Then in Section 3, after discussing the problems with HSC on high dimensional data sets, we give the combination technique used in HSC. Our experimental results are presented in Section 4, followed by our conclusions in Section 5.

2 Overview of the Classification Method Based on Hyper Surface

HSC is a universal classification method based on Jordan Curve Theorem in topology. The main differences between the well-known SVM algorithm and HSC are that HSC can directly solve the nonlinear classification problem in the original space without having to map the data to a higher dimensional space, and thus without the use of kernel function.

Jordan Curve Theorem. Let X be a closed set in n-dimensional space R^n. If X is homeomorphic to a sphere in $n-1$ dimensional space, then its complement $R^n \setminus X$ has two components, both connected, and one of them is called inside, the other called outside.

Classification Theorem. For any given point $x \in R^n \setminus X$, x is in the inside of X $\Leftrightarrow$ the wind number i.e. intersecting number between any radial from x and X is odd, and x is in the outside of X $\Leftrightarrow$ the intersecting number between any radial from x and X is even.

How to construct the separating hyper surface is an important problem. An approach has been given in [1]. Based on Jordan Curve Theorem, we have put forward the following classification method HSC [1].

Step1. Let the given samples distribute in the inside of a rectangle region.

Step2. Transform the region into a unit region.

Step3. Divide equally the region into some smaller units. If some units contain samples from two or more classes then divide them into a series of smaller units repeatedly until each unit covers at most samples from the same class.

Step4. Label each region according to the inside sample's class. Then the frontier vectors and the class vector form a string for each region.

Step5. Combine the adjacent regions of the same class and obtain a separating hyper surface then save it as a string.

Step6. Input a new sample and calculate the intersecting number of the sample about separating hyper surface. Drawing a radial from the sample can do this. Then the class of the sample is decided according to whether the intersecting number between the radial and the separating hyper surface is even or odd.

HSC has the following properties.

1) *High Efficient and Accuracy*

The classification algorithm based on hyper surface is a polynomial algorithm if the same class samples are distributed in finite connected components. Experiments show that HSC can efficiently and accurately classify density large dataset in two-dimensional or three-dimensional space for multi-classification. For large three-dimensional data up to 10^7, the speed of HSC is still very fast [2]. The reason is that the time of saving and extracting hyper surface is very short and the need for storage is very little. Another reason is that the decision process is very easy by using Jordan Curve Theorem.

2) *Ability of Generalization*

The experiment of training on small scale samples and testing on density large scale shows that HSC has strong ability of generalization[2]. According to statistic learning theory[4][5], the higher VC dimension is, the larger confidence domain is. So the difference between real risk and experimental risk possibly increase. This is the reason of excessive learning problem. So machine learning process is not only minimizing the experimental risk, but also reducing the dimension of VC. But the strategy is not useful in HSC. Because the hyper surface made by linear segmentation function. The function set has infinite VC dimension because the set can separate any more h samples that distribute in anyway. This shows that the conclusion about the bound of generalization given by Vapnik is loose when the VC dimension is too big.

3) *Robustness*

Though the data noise can not be completely clear, it can be controlled in a local region. If a noised sample locates in the inside of the hyper surface, the hyper surface will change into complex hyper surface. In this case the classification theorem is still efficient, the noise may make mistake in classification, but the influence has been controlled in a local small unit.

4) *Independ on The Distribusion Feature of Samples*

In fact, HSC can solve the nonlinear classification problem which the samples distribute in any shape in a finite region. The method has nothing to do with the shape of the data, even though the shape is interlock or crisscross. The common condition as other classification methods is that the samples must reflect the feature of data distribution.

3 Hyper Surface Classifiers Combination

3.1 Problems with HSC on High Dimensional Data Sets

According to Jordan Curve Theorem, HSC can deal with any data sets regardless of their dimensionality on the theoretical plane. But in practice, there exist some problems in both time and space in doing this directly. It is not as easy to realize HSC in higher dimensional space as in three-dimensional space. However, what we really need is an algorithm that can deal with data not only of massive size but also of high dimensionality. Fortunately, there exist many methods of dimensionality reduction for us to use. And another important and effective kind of dimensionality reduction method without losing any essential information is proposed in [3]. This method rearranges all of the numerals in the higher dimensional data to lower dimensional data without changing the value of all numerals, but only change their position according to some orders, and thus very suitable for HSC. In paper [29], based on the idea of ensemble, another solution to the problem of HSC on high dimensional data sets is proposed and proven to have a preferable performance by experiments. However multi agent technique has not been utilized in paper [29].

Another problem of high dimensional data is that the different dimension data for the same object usually spread in a distribution environment. DDM refers to the mining of inherently distributed datasets, aiming to generate global patterns from the union set of locally distributed data. However, the security issue among different local datasets and the huge communication cost in data migration prevent moving all the datasets to a public site. Thus, the algorithms of DDM often adopt a computing paradigm of local processing and global synthesizing, which means that the mining process takes place at a local level and then at a global level where local data mining results are combined to gain global findings.

In our work, we solve the problem based on the idea of combination and perfect results are gained. This method will be described in further detail in the next part.

3.2 The Combination of HSC

We propose multi-agents based technology to realize the combination of Hyper Surface Classifiers. Agents can imitate human beings' group decision to solve problems in the same way as group consultations of doctors. According to this, every classifier is designed to be an agent with given condition attributes. We use two types of agents: classifier training agent and classifier combining agent. Classifier training agent learns a classification model using a specified algorithm on a dataset and predicts an unlabelled instance using the learned model. Classifier combining agent realizes the information fusion of multiple classifier training agents' prediction results and gives the final class label of the unlabelled instance. When agents collectively do a classifying job, first of all, every classifier training agent must complete its learning on its own and produce an independent classifying model with given different condition attributes. Then when a classifying prediction task is performed, the classifier combining agent sends the instances to be forecast to every classifier training agent. Classifier training agents use classifying model to forecast the label of the instance, meanwhile, it evaluates the instance's properties and sends the overall evaluation results with the class label to the classifier combining agent. The classifier combining agent decides the final class label based on the results of all classifier training agents in terms of control logic such as voting or weighted voting. Thus, classifier combining agent probably surpasses the capability limits of single classifiers

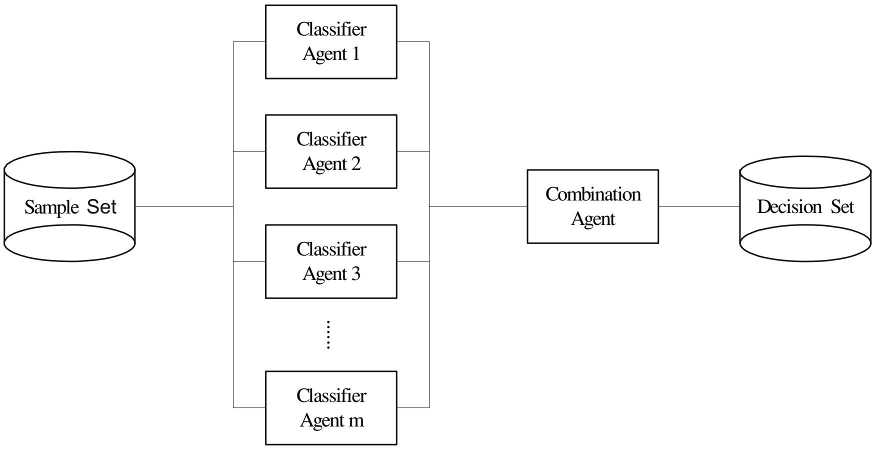

Fig. 1. The Combination of multiple classifier agents

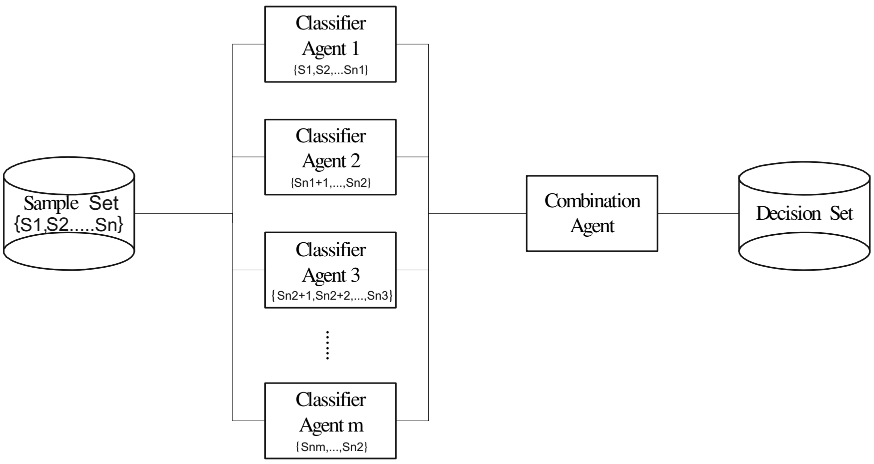

Fig. 2. The horizontal combination of multiple classifier agents

(see Fig1). Hence, the agent-based system of heterogeneous HSC classifiers can achieve a high classifying accuracy without additional fusion training of classifier group. Generally speaking, there're two ways of combination, horizontal combination (see Fig2) and vertical combination (see Fig3). The most important difference sub-datasets are obtained by dividing the features rather than by dividing the sample set, so in the case of no inconsistency, the size of each sub-dataset is equal to the original sample set, with totally occupying a little more storage space than the original sample set. Here, vertical combination is adopted.

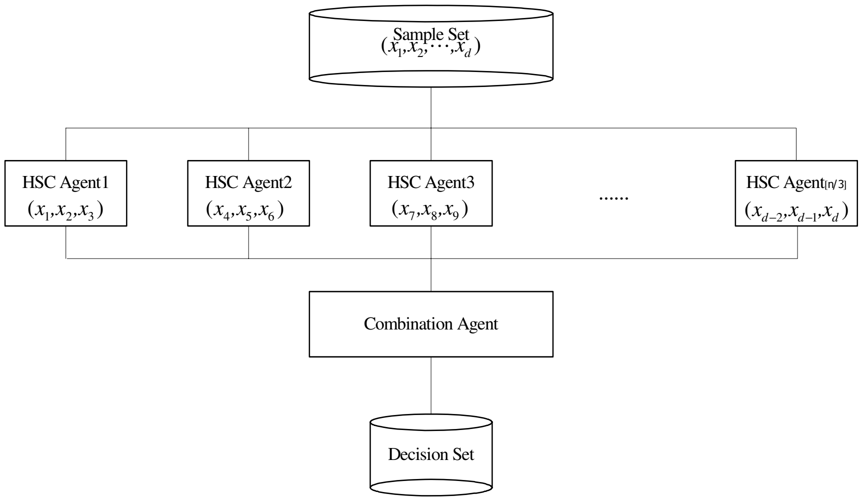

Fig. 3. The vertical combination of HSC agents

Aiming at the memory shortage problem of HSC on high dimensional data sets, we provide a solution based on the idea of combination. By attaching the same importance to each feature, firstly we group the multiple features of the data according to some rules to form some sub-datasets, then start a training process and generate a classifier for each sub-dataset, and the final determination is reached by integrating the series of classification results in some way. The following is its detailed steps.

The Agent Training Process

Step1. Get the dimension of conditional attributes d from the data set of training samples.

Step2. Divide the features into $\lceil d/3 \rceil$ subsets, where $\lceil d/3 \rceil$ is the smallest integer number that is greater than $\lceil d/3 \rceil$. The i-th subset covers the features $3i-2$, $3i-1$, $3i$, $i = 1, 2, 3, \cdots, \lceil d/3 \rceil$, and the decision attribute. If d cannot be divided exactly by 3, then one or two features in the $d-1$-th subset are added to the last subset. After this, the data set of training samples has been divided into $\lceil d/3 \rceil$ sub-datasets, each of which has the same size, with three conditional attributes and one decision attribute.

Step3. For each sub-dataset, eliminate the inconsistency that may have been caused in Step2. For each sample in this sub-dataset, if there exist some other samples in which the values of conditional attributes are all the same with those of this sample but only the value of decision attribute is different, then we say inconsistency occurs. In that case, we just have to delete these samples from this sub-dataset.

Step4. For each sub-dataset after Step3, start a HSC training process independently and save the training result as the model. And till now, we have got $\lceil d/3 \rceil$ HSC classifiers, which we define as classifiers combination.

The Agent Classification Process

Step1. Get the dimension of conditional attributes d from the data set of testing samples.

Step2. Divide the features into $\lceil d/3 \rceil$ subsets, and the i-th subset covers the features $3i-2$, $3i-1$, $3i$, $i=1,2,3,\cdots,\lceil d/3 \rceil$. If d cannot be divided exactly by 3, then one or two features in the $d-1$-th subset are added to the last subset. After this, the data set of testing samples has been divided into $\lceil d/3 \rceil$ sub-datasets, each of which has three conditional attributes.

Step3. For each sub-dataset after Step3, start the corresponding HSC classifier and save the result of classification. So now we have got $\lceil d/3 \rceil$ classification results for each testing sample in the original data set.

Step4. The final decision for each testing sample in the original data set is reached by voting. Here we adopt the *plurality* voting scheme in which the collective decision is the classification result reached by more classifiers than any other. Notice that we attach the same importance to all the classifiers, so in the case of the same number of votes for two or more classification results, we can randomly choose one of them.

HSC Combination classifies high dimensional data sets by trying to analyze multiple slices of the training and testing samples. Furthermore, the combination can be far less fallible than any single HSC classifier.

3.3 Voting Methods

Among the combination methods for HSC Agents, majority vote is the simplest to implement, since it requires no prior training, and it has been used as early as 1974 [26]. The use of this method is especially appropriate in situations where other quantifiable forms of output cannot be easily obtained from individual classifiers agent, or where the use of other accurate combination methods may be too complex. Obvious examples of the former are some structural classifiers. This combination method has also been found to be highly effective. From this process of simple majority vote in which the decision of each classifier carries equal weight, various refinements can be made. This can be done by assigning different weights to each classifier agent to optimize the performance of the combined classifiers agent on the training set. For the first refinement, weights can be generated by a genetic algorithm and assigned to the vote of each classifier to determine the optimal values for an objective function. This function can incorporate conditions on the recognition and error rates. We adopted plurality voting method. The combined classifier agent decides for the testing sample belonging to class C_j if the number of classifiers that support it is considerably bigger than the number of classifiers that support any other class.

3.4 Development Toolkit and System Implementation

MAS environments for Combination HyperSurface Classifiers System (CHCS) is very important. The reason for adopting MAS technology is that it is in with the following characteristics and requirements. MAS is a natural distributed computing environment, allowing agents running on different computing hosts simultaneously to achieve high availability. Moreover, MAS middleware helps to form a unified computing interface and makes the distributed applications more easily built. MAS is an open system In an open system, the structure of the system itself is capable of dynamically changing. Its components may not be known in advance, could change over time, and may be highly heterogeneous in that they are implemented by different people, at different times, using different software tools and techniques. In training and testing complex process, the output of an early job is the input of the following jobs. Thus, distributed execution of this process involves complex process control and management. To address this problem, a number of agents, specializing in solving constituent sub-problems in autonomous ways, need to be developed. They are then coordinated through standardized Agent Communication Language, which is well supported in a MAS environment by some communication protocols.

We adopted MAGE [27] as toolkit and MAS environment. MAGE is a middleware that facilitates the development of MA systems. Each instance of the MAGE runtime environment for each computing host is called a container, as it can contain several agents. Each container on that host presents the corresponding computing resource. Several containers can connect with each other through networks to form a natural distributed computing environment. From the view of programmers, it seems that all the containers are running on the same computer. We encapsulate each algorithm into the behavior of different agents. Such an agent can be regarded as an algorithm entity that can run on any host. To initiate an agent on a host means to perform the behavior of that agent, and thus utilize the corresponding computing resource. In paper [28], we implement this system in an established multi-agent system environment, in which the reuse of existing data mining algorithms is achieved by encapsulating them into agents.

3.5 Architecture of the CHSC System

We use two types of agents: the classifier training agent and the classifier combining agent. Each classifier training agent is responsible to train the local classifier, while the classifier combining agent is designed to combine the classification results of all the classifier training agents. The whole system is organized into two layers as shown in Figure 3: the HSC Agent function layer and the Combination Agent management and control layer. HSC Agent function layer provides software resources of the HSC algorithm in the form of classifier training agents. When a host does not have the software copy of a classifier training agent, it sends a request to the agent provider of software resources, and then the provider answers it with the mobile agent equipped with the required software copy. Thus it increases the computing capability of the host. Classifier training agents perform various operations in all phases of the data mining process, including sampling, preprocessing, training, classifier combining agent perform model combination and evaluation.

4 Experiments

As we can predict, the main problem with HSC Combination is inconsistency in sub-datasets. In that case, we have to delete some samples to keep consistent. As a result, the actual number used for training or testing decreases, it will affect our final results. Take the data set of Pima-diabetes from the UCI machine learning repository for example, the Table1 below illustrates this point.

Table 1. Classification results of HSC Combination on the data set of Pima

Rate / Agent	Actual No. of Samples for Training	Recall	Accuracy
HSC Agent 1	566	100%	90.00%
HSC Agent 2	542	100%	85.05%
HSC Agent 3	542	100%	94.85%
HSC Agents Combination	568	88.73%	87.50%

Note: Pima has 8 dimensions and 768 samples, 568 of which are used for training and the rest for testing.

Notice that even though the recall rate and accuracy of each sub-classifier are just high, the overall rates are much less than those of other data sets. This is mainly because that in this data set there exists more inconsistency after dividing the data set into sub-datasets, and as a result the actual number of samples participated in training or testing process decreases. But when computing the recall rate and accuracy, we need to divide the results by the number of overall samples, which are bigger than the number of actual samples in sub-dataset.

The accuracy is also related with another factor. During the voting process, it is quite posssible that two or more classification results have the same number of votes. In that case, we can randomly choose one of them as the final result, but which one we choose actually affects the accuracy very much in practice.

However, HSC Combination performs perfectly well on data sets that have little inconsistency after dividing them into sub-datasets. Now we show some results on several high dimensional data sets taken from the UCI machine learning repository in the following tables.

Table 2. Classification results of HSC Combination on high dimensional data sets

Data Sets	Dimensions	Number of Training Samples	Number of Testing Samples	Number of Sub-Classifiers	Recall	Accuracy
Iris	4	100	50	2	100%	98.00%
Wine	13	128	50	5	100%	100%
Wdbc	30	369	200	10	100%	100%
Sonar	60	158	50	20	100%	100%

From this table, we can see that HSC Combination works fairly well on these high dimensional data sets, and the recall rate and accuracy are very good. Specially, for the data set of iris, there are only two classifiers, the accuracy is lower than other high dimensional data sets. And it is obvious that the higher the dimension is, the higher accuracy is. Now,we give the classification results on each data set in details.

Table 3. Classification results of HSC Combination on the data set of Iris

	Recall	Accuracy
HSC Agent 1	100%	98.00%
HSC Agent 2	100%	92.00%
HSC Agents Combination	100%	98.00%

Table 4. Classification results of HSC Combination on the data set of Wine

	Recall	Accuracy
HSC Agent 1	100%	88.00%
HSC Agent 2	100%	84.00%
HSC Agent 3	100%	96.00%
HSC Agent 4	100%	94.00%
HSC Agent 5	100%	94.00%
HSC Agents Combination	100%	100.00%

Table 5. Classification results of HSC Combination on the data set of Wdbc

	Recall	Accuracy
HSC Agent 1	100%	99.00%
HSC Agent 2	100%	98.00%
HSC Agent 3	100%	97.00%
HSC Agent 4	100%	95.00%
HSC Agent 5	100%	94.00%
HSC Agent 6	100%	86.50%
HSC Agent 7	100%	97.50%
HSC Agent 8	100%	99.50%
HSC Agent 9	100%	89.00%
HSC Agent 10	100%	96.50%
HSC Agents Combination	100%	100%

For the data set of Wdbc, ten HSC Agents have been obtained, two of which are presented in the following Fig.4 and Fig. 5.

From experimental results and analysis above, we can see that HSC Combination performs very well on high dimensional data sets and is perfectly suitable for the data sets in which samples are different in each slice.

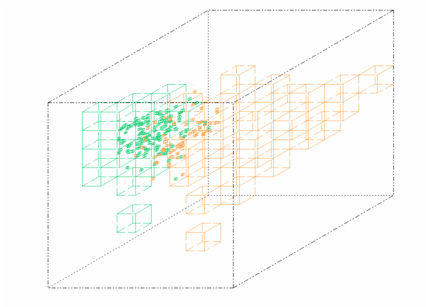

Fig. 4. HSC Agent 1 Model of the data set of Wdbc

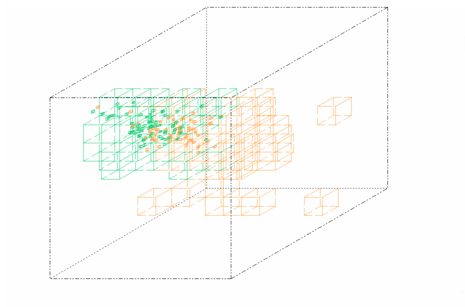

Fig. 5. HSC Agent 10 Model of the data set of Wdbc

5 Conclusions

For high dimensional data that the different dimension data for the same object usually spread in a distribution environment, a new framework using intelligent agents for pattern recognition is proposed. Considering that agents can imitate human beings' group decision to solve problems in the same way as group consultations of doctors, we propose multi-agents based technology to realize the combination of HSC Agents for high dimensional data sets. Two types of agents are introduced: classifier training agent and classifier combining HSC Agent. Each classifier training agent is designed to be an agent with given conditional attributes instead of all conditional attributes, so it is responsible to read a vertical slice of the samples and train the local classifier. The classifier combining agent is designed to combine the classification results of all the classifier training agents by way of voting. The most important difference between HSC combination and the traditional combination is that the sub-datasets are obtained by dividing the features rather than by dividing the sample set, so in the case of no inconsistency, the size of each sub-dataset is equal to the original sample set, with totally occupying a little more storage space than the original sample set. Experiments show that this method has a preferable performance on high dimensional data sets, especially on those in which samples are different in each slice. The time complexity of HSC Combination is $o((nd+n^2d)/3)$, where n denotes the size of the sample set; d denotes dimensional number.

Acknowledgements

This work is supported by the National Science Foundation of China (No. 60435010, 90604017, 60675010), the 863 Project (No.2006AA01Z128), National Basic Research Priorities Programme (No. 2003CB317004) and the Nature Science Foundation of Beijing (No. 4052025).

References

1. He, Q., Shi, Z. Z., Ren, L. A.: The Classification Method Based on Hyper Surface. In: Proc. 2002 IEEE International Joint Conference on Neural Networks, Las Vegas (2002) 1499–1503
2. He, Q., Shi, Z. Z., Ren, L. A., Lee, E. S.: A Novel Classification Method Based on Hyper Surface. Int. J. of Mathematical and Computer Modeling 38 (2003) 395–407
3. He, Q., Zhao, X. R. and Shi, Z. Z.: Classification based on dimension transposition for high dimension data. Soft Computing, 11, 4 (2006) 329–334
4. Vapnik, V. N., Golowich, S., Smola, A.: Support Vector Method for Function Approximation, Regression Estimation, and Signal Processing. In: M. Mozer, M. Jordan, and T. Petsche (eds.): Neural Information Processing Systems, Vol. 9. MIT Press, Cambridge, MA (1997)
5. Vapnik V. N.: The Nature of Statistical Learning Theory. Springer-Verlag, New York, (1995)
6. Fulton, W.: Algebraic topology. Springer-Verlag, New York (1995)
7. Alkoot, F. M., Kittler, J.: Experimental Evaluation of Expert Fusion Strategies. Pattern Recognition Letters 20 (1999) 1352–1369
8. Cordella, L. P., Foggia, P., Sansone, C., Tortorella, F. and Veneto M.: Reliability Parameters to Improve Combination Strategies in Multi-Expert Systems. Pattern Analysis and Applications 2 (1999) 205–214
9. Ho, T. K., Hull, J. J. and Srihari, S. N.: Decision Combination in Multiple Classifier Systems. IEEE Trans. PAMI 16 (1994) 66–75
10. Huang, Y. S. and Suen, C. Y.: A Method of Combination of Multiple Experts for the Recognition of Unconstrained Handwritten Numerals. IEEE Trans. PAMI 17 (1995) 90-94
11. Ji, C. and Ma, S.: Combinations of Weak Classifiers. IEEE Trans. Neural Networks 8 (1997) 32–42
12. Kang, H. J., Kim, K. and Kim, J. H.: Optimal Approximation of Discrete Probability Distribution with kth-order Dependency and Its Application to Combining Multiple Classifiers. Pattern Recognition Letters 18 (1997) 515–523
13. Kang, H.J. and Lee, S.W.: Combining Classifiers Based on Minimization of a Bayes Error Rate. In: Proceedings of the 5th International Conference on Document Analysis and Recognition, Fort Collins, Colorado, USA (1999) 124–129
14. Kittler, J.: Combining Classifiers: A Theoretical Framework. Pattern Analysis and Applications 1 (1998) 18–27
15. Kittler, J.: On Combining Classifiers. IEEE Trans. PAMI 20 (1998) 226–239
16. Lam, L. and Suen, C. Y.: Application of Majority Voting to Pattern Recognition: An Analysis of Its Behavior and Performance. IEEE Trans. Systems, Man, and Cybernetics 27 (1997) 553–568
17. Rahman, A.F.R. and Fairhurst, M.C.: An Evaluation of Multi-expert Configurations for the Recognition of Handwritten Numerals. Pattern Recognition 31 (1998) 1255–1273
18. Rahman, F.R. and Fairhurst, M. C.: Serial Combination of Multiple Experts: A Unified Evaluation. Pattern Analysis and Applications 2 (1999) 292–311
19. Wang, D., Keller, J. M., Carson, C.A., McAdoo-Edwards, K.K., and Bailey C. W.: Use of Fuzzy-Logic-Inspired Features to Improve Bacterial Recognition through Classifiers Fusion. IEEE Trans. SMC 28B (1998) 583–591
20. Xu, L., Krzyzak, A., and Suen, C. Y.: Methods of Combining Multiple Classifiers and Their Application to Handwritten Numeral Recognition. IEEE Trans. Systems, Man, and Cybernetics 22 (1992) 418–435

21. Al-Ani, A, Deriche, M: A New Technique for Combining Multiple Classifiers Using the Dempster-Shafer Theory of Evidence. Journal of Artificial Intelligence Research 17 (2002) 333–361
22. Suen, C. Y. and Lam, L: Multiple Classifier Combination Methodologies for Different Output Levels. In J. Kittler and F. Roli (Eds.): LNCS Vol.1857 (2000) 52–66
23. Lin, S. D., Han G. Q., Yuan, X.: Information Fusion of Agent Based Heterogeneous Multi-Classifiers. In Proceedings of the Fourth International Conference on Machine Learning and Cybernetics, Guangzhou (2005) 18–21
24. Gader, P. D., Hepp, D., Forrester, B., and Peurach, T.: Pipelined Systems for Recognition of Written Digits in USPS ZIP Codes. In Proc. U. S. Postal Service Advanced Technology Conf. (1990) 539–548
25. Mandler, E. and Schuermann, J.: Combining the Classification Results of Independent Classifiers Based on the Dempster/Shafer Theory of Evidence. Pattern Recognition and Artificial Intelligence (1988) 381–393
26. Spanjersberg, A. A.: Combination of Different Systems for the Recognition of Handwritten Digits. In Proc. 2nd Int. Joint Conf. on Pattern Recognition, Copenhagen (1974) 208–209
27. Shi, Z. Z., H. Zhang, Y. Cheng, Y. Jiang, Q. Sheng and Z. Zhao: MAGE: An Agent-Oriented Programming Environment. In Proceedings of the IEEE International Conference on Cognitive Informatics, New York (2004) 250–257
28. Luo, P., Lu¨, K., Huang, R., He, Q., and Shi, Z. Z.: A Heterogeneous Computing System for Data Mining workflows in Multi-Agent Environments. Expert Systems 23 (2006) 258–272
29. Zhao, X. R., He, Q. & Shi, Z. Z: HyperSurface Classifiers Ensemble for High Dimensional Data Sets. In Proc. International Symposium on Neural Networks. Lecture Notes in Computer Science, Vol. 3971. Springer-Verlag, Berlin Heidelberg New York (2006) 1299–1304
30. Fu, Y.: Distributed data mining: An overview. IEEE TCDP newsletter (2001)
31. Cannataro, M., Congiusta, A., Pugliese, A., Talia, D., Trunfio, P.: Distributed data mining on grids: Services, tools, and applications, IEEE Transactions on Systems, Man and Cybernetics 34 (6) (2004) 2451–2465