http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 8 (2): 241-245, 2009
ISSN 1812-5638

Selecting and Combining Classifiers Simultaneously with Particle Swarm Optimization

Li Ying Yang, Jun Ying Zhang and Wen Jun Wang
School of Computer Science and Technology, Xidian University, Xi’an, 710071, China

Abstract: A weighted combination model of multiple classifier systems based on Particle Swarm Optimization was reviewed, which took sum rule and majority vote as special cases. It was observed that the rejection of weak classifier in the combination model can improve classification performance. Inspired by this observation, we presented a problem that how to choose the useful classifiers in a given ensemble, especially in the reviewed model. In this study, a combination algorithm was proposed, which implemented classifiers’ selection and combination simultaneously with particle swarm optimization. We describe the implementation details, including particles encoding and fitness evaluation. Nine data sets from UCI Machine Learning Repository were used in the experiment to justify the validity of the method. Experimental results show that the propose model obtained the best performance on 5 out of 9 data sets, and averagely outperforms the reviewed model, majority voting, max rule, min rule, mean rule, median rule and product rule. The results were analysed from the point of the characteristic of data set.

Key words: Ensemble learning, multiple classifiers system, particle swarm optimization

INTRODUCTION

In order to achieve the best possible classification performance for a pattern recognition task at hand, we need to design many algorithms and then perform an evaluation-selection process. That is, evaluate a set of different algorithms against a representative validation set and select the best one. This is the traditional approach to supervised learning problem. It is now recognized that the key to recognition problems does not lie wholly in any particular solution. No single model exists for all pattern recognition problems and no single technique is applicable to all problems. Furthermore, the sets of patterns misclassified by the different algorithms would not necessarily overlap, which suggested that different algorithms potentially offered complementary information (Ghosh, 2002). This led to the emergence of classifiers combination. Combining multiple classifiers is a learning method where a collection of a finite number of classifiers is trained for the same classification task. It came alive in the 90’s of last century. Over the past years, this method has been considered as a more practical and effective solution for many recognition problems than using one individual classifier (Suen *et al.*, 1990). Research in this domain has increased and grown tremendously, partly as a result of the coincident advances in the technology itself. These technological developments include the production of very fast and low cost computers that have made many complex pattern recognition algorithms practicable (Suen and Lam, 2000). Classifiers combination gains better performance at the cost of computation. Research on classifiers combination follows two parallel lines of study. One is decision optimization and the other coverage optimization (Ho, 2000). Assuming a given, fixed set of carefully designed and highly specialized classifiers, decision optimization attempts to find an optimal combination of their decisions. Assuming a fixed decision combination function, coverage optimization generates a set of mutually complementary, generic classifiers that can be combined to achieve optimal accuracy. We focused on decision optimization in this study.

Majority vote is the simplest combination method and has been a much-studied subject among mathematicians and social scientists. In majority vote, each individual has the same importance. A natural extension to majority vote is to assign weight to different individual. Thus weighted combination algorithm was obtained. Since under most circumstances, there is difference between individuals, weighted combination algorithm provides a more appropriate solution. The key to weighted combination algorithm is the weights. A weighted combination model based on particle swarm optimization (PSO-WCM) is proposed in earlier study work (Yang and Qin, 2005). It is observed that the rejection of weak classifier in PSO-WCM can improve classification performance. This inspired us with a problem that how to choose the useful classifiers in PSO-

Corresponding Author: Li Ying Yang, P.O. Box 161, No. 2 South Taibai Road, Xi’an City, Shaanxi Province, 710071, People’s Republic of China

WCM. In this study, a combination model which selects and combines classifiers simultaneously with particle swarm optimization algorithm is proposed. To the best of our knowledge, former evolution-based research on classifiers combination concentrated only on classifier selection or classifier combination rather than both selection and combination (Ruta and Gabrys, 2001).

CLASSIFIERS COMBINATION ALGORITHM BASED ON PSO

Particle swarm optimization: Inspired by simulating social behavior (such as bird flocking), Everhart and Kennedy introduced Particle Swarm Optimization (PSO), which is a population-based evolutionary computation technique (Kennedy and Eberhart, 1995; Eberhart and Kennedy, 1995). In PSO, candidate solutions are denoted by particles. Each particle is a point in the search space and has two attribute values: fitness determined by the problem and velocity to decide the flying. Particles adjust their flying toward a promising area according to their own experience and the social information in the swarm. Thus they will at last reach the destination through continuous adjustment in the iteration. Given a D-dimension search space, m particles constitute the swarm. The i-th particle is denoted by $x_i = (x_{il}, x_{i2},..,x_{iD})$, i = 1, 2,...,m. Taking x_i into the objective function, the fitness for the i-th particle can be work out, which could tell the quality of current particle, i.e., the current solution. The current velocity and the best previous solution for the i-th particle are represented by $v_i = (v_{il}, v_{i2},..,v_{iD})$ and $p_i = (p_{il}, p_{i2},..,p_{iD})$. The best solution achieved by the whole swarm so far is denoted by $p_g = (p_{gl}, p_{g2},..,p_{gD})$. In Everhart and Kennedy's original version, particles are manipulated according to the following equations:

$$v_{id} = v_{id} + c_1r_1(p_{id}-x_{id})+c_2r_2(p_{gd}-x_{id}) \quad (1)$$

$$x_{id} = x_{id}+v_{id} \quad (2)$$

where i = 1,....,m; d = 1,....,D; c_1 and c_2 are two positive constants called cognitive learning rate and social learning rate respectively; r_1 and r_2 are random numbers in the range [0,1]. The velocity v_{id} is limited in $[-v_{max}, v_{max}]$ with v_{max} a constant determined by specific problem.

The original version of PSO lacks velocity control mechanism, so it has a poor ability to search at a fine grain (Angeline, 1998). Many researchers devoted to overcoming this disadvantage. Shi and Eberhart introduced a time decreasing inertia factor to Eq. 1 (Shi and Eberhart, 1998):

$$v_{id} = wv_{id} + c_1r_1(p_{id}-x_{id})+c_2r_2(p_{gd}-x_{id}) \quad (3)$$

where, w is inertia factor which balances the global wide-range exploitation and the local nearby exploration abilities of the swarm. Clerc introduced a constriction factor a into Eq. 2 to constrain and control velocities magnitude (Clerc, 1999):

$$x_{id} = x_{id} + av_{id} \quad (4)$$

The above Eq. 3 and 4 are called classical PSO, which is much efficient and precise than the original one by adaptively adjusting global variables.

Weighted combination model: Weighted Combination Model (WCM) is an extension of simple majority vote. Consider a pattern recognition problem with M classes $(C_1, C_2,..., C_M)$ and K classifiers $(R_1, R_2,...,R_K)$. For a given sample x, (i = 1,...,K) outputs $E_i = (e_i(1),...,e_i(M))$, where $e_i(j)(j = 1,...,M)$ denotes the probability that *x* is from class j according to R_i. The weight vector for classifier ensemble is represented as $\varphi = (\varphi_1,...,\varphi_K)$ with:

$$\sum_{k=1}^{K} \varphi_k = 1$$

Let $E = E(E_{R1},...,E_{RK})$. The sample *x* is classified into the class with maximum posteriori probability and the decision rule is:

$$x \to C_j, \text{ If } \sum_{i=1}^{R} \varphi_i e_i(j) = \max_{k=1}^{M}(\sum_{i=1}^{R} \varphi_i e_i(k)) \quad (5)$$

In Eq. 5, if:

$$\varphi_i = \frac{1}{K}$$

then majority vote is obtained when classifiers output at abstract level, and sum rule is obtained when classifiers output at measurement level. If there is only one 1 in the weight vector and the other elements are all 0, the combination model is equal to the individual classifier whose weight is 1.

There are two methods for acquiring the weights in WCM. One set fixed weights to each classifier according to experience or something else. The other obtains weights by training. Training methods gain better performance at the cost of computation. It has two steps: Firstly, training individual classifiers on training set; Secondly, determining the weights based on validation set. In the second step, traditional approach set the weights in directly proportional to classifiers' accuracy on validation set (Baykut and Ercil, 2003). In the earlier study, we proposed a combination algorithm which determined the weights based on PSO, that is, PSO-

WCM (Yang and Qin, 2005). Optimal weights are achieved by searching in K-dimension space. A solution is a particle in PSO and coded into one K-dimension vector $\varphi = (\varphi_1,...,\varphi_K)$. Fitness function is computed as combination model's error rate on validation set using the weights. Hence the task is converted into an optimization problem for minimum.

CLASSIFIERS SELECTION AND COMBINATION ALGORITHM BASED ON PSO

In Zhou's work (Zhou and Tang, 2002), the relationship between the ensemble and its component neural networks is analyzed from the context of both regression and classification, which reveals that it may be better to ensemble many instead of all of the neural networks at hand. Our earlier study also provided similar results. But we just recombined the remained classifiers after rejecting weak classifiers artificially. PSO algorithm was proposed for simultaneous feature extraction and feature selection in (Chen and Qin, 2006). In this study, we proposed to select and combine classifiers simultaneously with PSO, that is, PSO-SWCM (PSO Selection and Weighted Combination Model).

In PSO-SWCM, the main interest is in representing weights space and all possible subsets of the given classifiers ensemble. As shown in Eq. 6, each particle is encoded into a real-valued vector which includes two parts. The first part is a weight vector where the i-th one is the weight assigned to the i-th classifier and the other part is a masking vector representing whether or not the i-th classifier is selected. If the mask value for a given classifier is negative, the classifier is not considered for combination. Otherwise, if the mask value is positive, the classifier is scaled according to the associated weight and included in the combination. K is the number of classifiers in original ensemble.

$$[(\varphi_1, \varphi_2,...,\varphi_K)(f_1, f_2,...,f_K)] \qquad (6)$$

Fitness function is computed as combination model's error rate on validation set using the mask values and the weights. The task is then converted into an optimization problem for minimum. In order to implement PSO-SWCM algorithm, a K-dimension-binary-code vector $B = (B_1,...,B_K)$ is defined as following:

$$B_i = \begin{cases} 1, & \text{if } f_i > 0 \\ 0, & \text{if } f_i \le 0 \end{cases} \qquad (7)$$

Then a given sample $\boldsymbol{x}$ is classified into the class with maximum posteriori probability and the decision rule is:

$$x \to C_j, \text{If } \sum_{i=1}^{R} B_i \varphi_i e_i(j) = \max_{k=1}^{M}\left(\sum_{i=1}^{R} B_i \varphi_i e_i(k)\right) \qquad (8)$$

RESULTS AND DISCUSSION

We trained a group of classifiers and then combined them to verify the combination model proposed in this work. Five classifiers used in this paper are: LDC, Linear Discriminant Classifier; QDC, Quadratic Discriminant Classifier; KNNC, K-Nearest Neighbor Classifier with K = 3; TREEC, a decision tree classifier; BPXNC, a neural network classifier based on MATHWORK's trainbpx with 1 hidden layer and 5 neurons in this hidden layer. Six combination rules were included in our experiments for the sake of comparison: majority vote rule, max rule, min rule, mean rule, median rule and product rule (Kittler *et al*., 1998).

PSO-WCM and PSO-SWCM were applied to nine real world problems from the UCI repository: Balance, Letter, Vehicle, Glass, Waveform, Sat, Iris, Pima and Wine (Blake *et al*., 1998). For each dataset, 2/3 examples were used as training data, 1/6 validation data and 1/6 test data. In other combination rules or individual classifiers, 2/3 examples were used as training data and 1/3 test data. All experiments were repeated for 10 runs and averages were computed as the final results. Note that all subsets were kept the same class probabilities distribution as original data sets. The characteristics of these data sets are shown in Table 1.

Experiments setting: Since there are 5 classifiers, the number of weights is 5. A particle in PSO-WCM was coded into one 5-dimension vector $\varphi = (\varphi_1, \varphi_2, \varphi_3, \varphi_4, \varphi_5)$. A particle in PSO-SWCM was coded into the vector $[(\varphi_1, \varphi_2, \varphi_3, \varphi_4, \varphi_5)(f_1, f_2, f_3, f_4, f_5)]$ and the corresponding binary vector $B = (B_1, B_2, B_3, B_4, B_5)$. φ_i was initialized as random number in the range [0,1] and then normalized with the constraint

$$\sum_{k=1}^{5} \varphi_k = 1$$

f_i was initialized as random number in the range [-1,1].

Table 1: Data sets used in the study

Dataset	Samples	Inputs	Outputs
Iris	150	4	3
Balance	625	5	3
Vehicle	846	19	4
Letter	20000	16	26
Sat	4435	36	6
Pima	768	9	2
Glass	846	18	4
Waveform	5000	21	3
Wine	178	13	3

Table 2: Error rate of individual classifiers

Data	LDC	QDC	KNNC	TREEC	BPXNC
Iris	0.0250	0.0333	0.0333	0.1250	0.0417
Balance	0.1096	0.0731	0.1038	0.5385	0.0750
Vehicle	0.2256	0.1640	0.2833	0.2560	0.1668
Letter	0.3256	0.1245	0.0596	0.3488	0.9863
Sat	0.1612	0.1468	0.1124	0.1813	0.3255
Pima	0.2422	0.2547	0.2672	0.3172	0.2484
Glass	0.3644	0.6373	0.3789	0.3033	0.3946
Waveform	0.1454	0.1523	0.1902	0.3012	0.1417
Wine	0.0101	0.0253	0.3121	0.0786	0.0089

Table 3: Error rate comparison of combination algorithms

Data	VOTEC	MAXC	MINC	MED-IANC
Iris	0.0250	0.0417	0.0417	0.0250
Balance	0.0962	0.0442	0.0788	0.0904
Vehicle	0.1354	0.1686	0.2432	0.2150
Letter	0.1086	0.6285	0.9312	0.1145
Sat	0.1164	0.2110	0.3102	0.1133
Pima	0.2406	0.2500	0.2500	0.2406
Glass	0.3357	0.5313	0.5674	0.3643
Waveform	0.1454	0.1605	0.1786	0.1556
Wine	0.0064	0.0073	0.0110	0.0043

Table 4: Error rate comparison of combination algorithms

Data	MEANC	PRODC	PSO-WCM	PSO-SWCM
Iris	0.0333	0.0333	0.0250	0.0250
Balance	0.0808	0.0846	0.0420	0.0327
Vehicle	0.2144	0.2011	0.1786	0.1756
Letter	0.3906	0.9255	0.0636	0.0542
Sat	0.1334	0.2876	0.1080	0.0985
Pima	0.2281	0.2266	0.2281	0.2062
Glass	0.3643	0.6233	0.3644	0.3643
Waveform	0.1447	0.1369	0.1499	0.1360
Wine	0.0043	0.0043	0.0043	0.0043

Classical PSO was adopted in this study. Parameters were set as following: size of the swarm N=20; inertia factor *w* linearly decreases from 0.9 to 0.4; $c_1 = c_2 = 2$; constriction factor a =1; for *i*-th particle, each dimension in position vector x_i and velocity vector v_i were initialized as random number in the range [0,1] and [-1,1]; max iteration = 800.

Table 2 shows that different classifier achieved different performance for the same task. But no classifier is superior for all problems. Some classifiers' error rates are even worse than 1/2, which is the accuracy of a completely random guess. It proclaimed again the limitation of single classifier in pattern classification task.

The combination performance of 5 classifiers by majority vote, max rule, min rule, mean rule, median rule, product rule, PSO-WCM and PSO-SWCM, were given in Table 3 and 4. It is shown that PSO-SWCM outperforms all comparison combination rules and the best individual classifier on data sets balance, letter, sat, pima, waveform. These data sets have a common characteristic, that is, the sample size is large. Therefore, the optimal weights obtained on validation set are also representative on test set. The same thing is not true on smaller data sets for the obvious reason that over fitting tends to occur. Optimal weights might appear in initial process, so the succeeding optimization makes no sense. But for the dataset Glass and Vehicle, which is comparatively larger, PSO-WCM and PSO-SWCM do not obtained the best result. This is due to another characteristic of the data sets. These two data sets both have comparatively more features. In order to reach the original distribution of given data set, much more samples are needed for data set with more features.

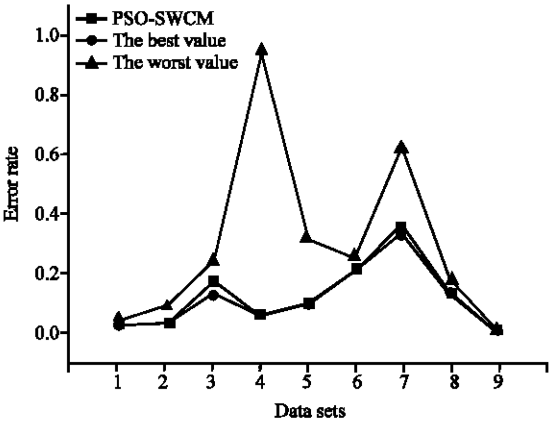

Fig. 1: The error rate comparison

It is also shown in the above tables that PSO-SWCM outperforms PSO-WCM, which reveals the superiority of selection operation incorporated with combination in multiple classifier system. This is in agreement with Zhou's ''many could be better than all".

For the sake of visualization, we drew the error rate comparison plot in Fig. 1. The proposed PSO-SWCM model was compared with the best and the worst values from the total 8 combination methods in our experiments. It was shown that the performance of PSO-SWCM model is close to the best performance.

CONCLUSION

In this study, we presented a PSO based ensemble algorithm, PSO-SWCM, to address the selection and combination problem in multiple classifier systems. The approach consists of generating a number of classifiers using different classification algorithms, and using only those classifiers which is chosen by PSO, then combining them according to the weights determined by PSO. Note that the selection of classifiers and the set of weights are simultaneous. Experiments were carried out on nine real world problems from the UCI repository. We observe that PSO-SWCM works rather well than PSO-WCM that we described in our previous work and other six algorithms used in bibliographies.

REFERENCES

Angeline, P.J., 1998. Evolutionary Optimization Versus Particle Swarm Optimization: Philosophy and Performance Differences. In: Evolutionary Programming, Porto, V.W., N. Saravanan, D. Waagen and A.E. Eiben (Eds.). Vol. 7, Springer-Verlag, Berlin, ISBN: 978-3-540-64891-8, PP: 601-610.

Baykut, A. and A. Ercil, 2003. Towards Automated Classifier Combination for Pattern Recognition. In: Multiple Classifier Systems, Windeatt, T. and F. Roli (Eds.). Vol. 2709, Springer-Verlag, Berlin, ISBN: 978-3-540-40369-2, pp: 94-105.

Blake, C., E. Keogh and C.J. Merz, 1998. UCI repository of machine learning databases, 1998. www.ics.uci.edu/~mlearn/MLRepository.html.

Chen, J.Y. and Z. Qin, 2006. Dimensionality reduction for evolving RBF networks with particle swarms. Proceedings of 3rd International Symposium on Neural Networks, Chengdu, China, LNCS 3971, May 28-June 1, Springer Berlin/Heidelberg, pp: 1319-1325.

Clerc, M., 1999. The swarm and the queen: Towards a deterministic and adaptive particle swarm optimization. Proceedings of the Congress of Evolutionary Computation, July 6-9, Washington, DC., USA., pp: 1958-1962.

Eberhart, R.C. and J. Kennedy, 1995. A new optimizer using particle swarm theory. Proceedings of the 6th International Symposium on Micro Machine and Human Science, Oct. 4-6, Nagoya, Japan, pp: 39-43.

Ghosh, J., 2002. Multiclassifier systems: Back to the future. Proceedings of the 3rd International Workshop on Multiple Classifier Systems, LNCS 2364, J un. 24-26, Springer-Verlag London, UK., pp: 1-15.

Ho, T.K., 2000. Complexity of classification problems and comparative advantages of combined classifiers. Proceedings of 1st International Workshop, MCS 2000 Cagliari, Italy, LNCS 1857, Jun. 21-23, Springer Berlin/Heidelberg, pp: 97-106.

Kennedy, J. and R. Eberhart, 1995. Particle swarm optimization. Proceedings of the IEEE International Conference on Neural Networks, Nov. 27-Dec. 1, Perth, Australia, pp: 1942-1948.

Kittler, J., M. Hatef, R.P.W. Duin and J. Matas, 1998. On combining classifiers. IEEE Trans. Pattern Anal. Mach. Intell., 20: 226-239.

Ruta, D. and B. Gabrys, 2001. Application of the evolutionary algorithms for classifier selection in multiple classifier systems with majority voting. Proceedings of the 2nd International Workshop on Multiple Classifier Systems, LNCS 2096, July 2-4, Springer Berlin/Heidelberg, pp: 399-408.

Shi, Y. and R. Eberhart, 1998. A Modified particle swarm optimizer. Proceedings of the IEEE Congress on Evolutionary Computation, May 4-9, Piscataway, NJ., pp: 69-73.

Suen, C.Y., C. Nadal, T.A. Mai, R. Legault and L. Lam, 1990. Recognition of totally unconstrained handwriting numerals based on the concept of multiple experts. Frontiers in handwriting recognition. Proceedings of the International Workshop on Frontiers in Handwriting Recognition, 1990, Montreal, Canada, pp: 131-143.

Suen, C.Y. and L. Lam, 2000. Multiple classifier combination methodologies for different output levels. Proceedings of 1st International Workshop on Multiple Classifier Systems, Cagliari, Italy, LNCS 1857, Jun. 21–23, Springer Berlin/Heidelberg, pp: 52-66.

Yang, L.Y. and Z. Qin, 2005. Combining classifiers with particle swarms. Proceedings of 1st International Conference on Advances in Natural Computation, Changsha, China, LNSC 3611, Aug. 27-29, Springer Berlin/Heidelberg, pp: 756-763.

Zhou, Z.H., J. Wu and W. Tang, 2002. Ensembling neural networks: Many could be better than all. Artif. Intell., 137: 239-263.