A Systematic Approach to Classifier Selection on Combining Multiple Classifiers for Handwritten Digit Recognition

Jongryeol Kim*, Kukhwan Seo and Kyusik Chung
Dept. of Electronic Eng. Soongsil University
1-1, Sangdo 5-Dong Dongjak-Gu, Seoul, Korea. 156-743
E-mail:{,tatata,kchung }@q.soongsil.ac.kr

Abstract

Much research has been done on combining multiple classifiers for handwritten character recognition to improve the performance of the classifier. Given a fixed set of classifiers using the same or different kinds of feature set, they focus on the methodology to combine all of the classifiers. In this paper, given a variable set of classifiers, we focus on the methodology to determine which subset of classifiers to achieve the optimal combination results. In order to evaluate the dependency between classifiers, we propose a similarity measure between them which can be calculated from the errors generated by each classifier. This similarity measure allows to compare the performance of one combination case with that of the other cases relatively without doing experiments. Using five individual classifiers with different feature set such as gradient, structural, UDLRH, mesh and LSF, we perform handwritten digit recognition experiments. With three combination methods such as Major Voting, Borda Count, and LCA, we perform combination experiments for all possible cases of 3 classifiers selected among the above 5. Then, we compare their rankings in terms of the recognition rate with that in terms of the similarity measure. This comparison shows the effectiveness of the proposed method.

1 Introduction

As an alternative to improve the performance of handwritten character recognition, combining multiple classifiers has been proposed and its related research has been actively performed recently. Given a fixed set of classifiers using the same or different kinds of feature set, they combine all of the classifiers to achieve the optimal combination results. The typical combination methods include BKS method[1,2], Majority Voting method[3], Borda Count method[4], LCA(Linear Confidence Accumulation)[5], Fuzzy fusion method[6], and Neural Network method[7,8]. Note that most combination methods except BKS assume that all the classifiers to be combined are independent. Classifiers can be categorized into three types based on the information content in their output[9]. Type I classifiers provide only a single class as their final output. Type II classifiers produce a ranked list of all or subset of the classes. Type III classifiers associate each class with a score which can be used as a measure of confidence for the class. Since outputs of Type III can be reduced to that of Type I or II, they contain more information.

Given a fixed set of classifiers, most researchers focus on the methodology to combine all of the classifiers. However, there has been little research on the evaluation of the combination process or the analysis of dependency between classifiers[10,11]. In this paper, given a variable set of classifiers, we focus on the methodology to determine which subset of classifiers to achieve the optimal combination results. In order to evaluate the dependency between classifiers, we propose a similarity measure between them which can be calculated from the errors generated by each classifier. Based on this similarity, we can perform classifier selection by determining a subset of classifiers with the lowest similarity measure. Note that this is possible without doing experiments. With five classifiers, we perform experiments for handwritten digit recognition using NIST Database. With three combination methods such as Majority Voting, Borda Count, and LCA, we perform combination experiments for all possible cases of 3 classifiers selected among the above 5. Then, we compare their rankings in terms of the recognition rate with that in terms of the similarity measure.

This paper is organized as follows. Section 2 describes how to calculate similarity measure, and Section 3 shows the experiments and their results. Section 4 describes the conclusion.

2 Similarity Measure

Assuming that K classifier candidates with their output and error characteristics are given in advance and a subset of them, L(L<K), are used in classifier combination, we deal with classifier selection problem. In order to evaluate the effectiveness of a classifier combination, we propose a similarity measure between or among combined classifiers based on the output errors generated by individual classifier. Similarity measure here indicates how much the errors generated by individual classifier are correlated each other. This is a relative value which tells about the extent of possibility of error generation at the final output when the classifiers are combined. We may say that the

*To the memory of Prof. Jongryeol Kim who was very special to many people at Cheonan National Junior College and EE Dept. of Soongsil University

0-8186-7898-4/97 $10.00 © 1997 IEEE

probability of error generation at the final output of classifier combination increases in proportion to the size of similarity measure among combined classifiers.

The following describes the details of similarity measure. Let M be the number of pattern class, and let C_i, i $\in$ A={1, ... , M} be each class. And let e_k, k=1,...,K indicate a classifier where K is the total number of classifiers. Given a database X of input patterns, an input pattern x belongs to X, x$\in$ X. Let $e_k(x) = \{m_k^i(x) || \forall i (1 \leq i \leq M)\}$ indicate an output vector of measurement values of classifier k for an input pattern x, where m_k^i(x) indicates a measurement value of class i in e_k(x). The higher value of m_k^i(x) represents the higher possibility of the input pattern x to belong to class i. As the type and the scale of a classifier output are different depending on the classifier used, the output of each classifier need be normalized using the following transformation function T.

$$T : m_k^i -> t_k^i$$
$$where \;\; t_k^i = \frac{(m_k^i)^2}{\sum_{i=1}^{M}(m_k^i)^2} t_k^i \qquad (1)$$

After this normalization process is applied, all the outputs of the classifiers are transformed into the normalized values as follows:, $\sum_{i=1}^{M} t_k^i = 1$, $0 \leq t_k^i \leq 1.0$, $\forall k (1 \leq k \leq K)$, $\forall (\leq i \leq M)$. If we express the normalized output of the true (target) class of classifier k for input pattern x by t_k^{true} and $t_k^{true} = max_{1 \leq j \leq M} t_k^j$, then the error ε_k(x) for input pattern x can be defined as:

$$\varepsilon_k(x) = t_k^{max} - t_k^{true} \qquad (2)$$

If we assume that L classifiers among K classifiers are selected for classifier combination, the number of the possible set of L classifiers from K classifier candidates is given by Z = $\begin{pmatrix} K \\ L \end{pmatrix}$. The classifier selection here is to choose one of Z cases which brings out the best performance.

Let S_{Lh} denote the similarity measure for the h-th one among Z cases where $1 \leq h \leq Z$. The calculation of S_{Lh} is done as follows: Firstly, we divide all the combination output results into several disjoint cases, according to the numbers of classifiers (among L classifiers) generating the incorrect outputs for a given input x, and then sum the similarity measure for individual case with different weight. Secondly, we approximate the similarity measure (among L classifiers) for individual case by summing the similarity measure between a pair of classifiers for each possible pair of classifiers. Thirdly, we define the similarity measure between a pair of classifiers in terms of error correlation between them.

Then, we can express S_{Lh} as follows:

$$S_{Lh}(e_k(x)) = \sum_{j=0}^{L} \alpha_j S_{Lhj}(e_k(x))$$

$$= \sum_{j=0}^{L} \alpha_j (\beta_j \sum_{m<n, m,n=1,2,3....L} \sigma_{mn})$$

$$\sigma_{mn} = \frac{\sum_{x \in Test} \varepsilon_m(x) \varepsilon_n(x)}{N} \qquad (3)$$

where S_{Lhj}, α_j, β_j, and N indicate the similarity measure for the case where j classifiers among z generate incorrect outputs, the contribution factor of the case j to S_{Lh}, the normalization factor of the case j, and the total number of test patterns, respectively.

For the case of j=0, $\sigma_{mn} = 0$ because $\varepsilon_m(x)$ and $\varepsilon_n(x)$ are zero. Therefore, $S_{Lh0}(e_k(x)) = 0$. For the case of j=1, $\sigma_{mn} = 0$ because either one of $\varepsilon_m(x)$ and $\varepsilon_n(x)$ is zero. Therefore, $S_{Lh1}(e_k(x)) = 0$. Actually there is a possibility that the combination results for this case lead to an incorrect final output. However, its possibility approaches to zero according as the number of classifiers in combination becomes larger. Therefore, we set it to be zero. Thus, S_{Lh} can be reexpressed as follows:

$$S_{Lh}(e_k(x)) = \sum_{j=2}^{L} \alpha_j S_{Lhj}(e_k(x)) \qquad (4)$$

In this paper, assuming that L is 3, we further explain how to determine α_j, β_j as below. Note the probability of each case to make the combination result incorrect. The probability in the case of $S_{3h2}(e_k(x))$ can be said to approach 2/3 since it belongs to the case where two classifiers give error outputs and the other classifier give a correct output. Also, the probability in the case of $S_{3h3}(e_k(x))$ can be said to approach 1 since it belongs to the case where all the three classifiers give error outputs. Therefore, we can set α_2 and α_3 to be 2/3 and 1, respectively. β_2 and β_3 can be approximately calculated as follows: In calculating , $S_{3h2}(e_k(x))$, $\varepsilon_m(x) \cdot \varepsilon_n(x)$ is calculated one time for an input x whereas in calculating $S_{3h3}(e_k(x))$, it is calculated three times for an input x. Therefore, $S_{3h3}(e_k(x))$ need be divided by three from the viewpoint of normalization and we set β_2 and β_3 to be 1 and 1/3, respectively.

3 Experiments and Results

3.1 Experimental Environment and Classifiers

In[12], we compare the performance of features, by experiment, which can be used for Handwritten English Alphabet, Digit and Korean alphabet character recognition. From the performance comparison, we select five feature types where four of them show the highest discrimination power to the three different set of Alphabets and Digits whereas the last one does not, but is the same kind of feature type as the others: Gradient, Structure, UDLRH(Up Down Left Right Hole), Mesh, and LSF(Large Stroke Feature) features. Note that these features are a part of GSC feature set developed at CEDAR[13].

Table 1: Number of Neurons for Each Layer in NN

Experiment		N_{in}	N_{hi}	N_{out}
Digit	C_G	192	96	10
	C_S	192	96	10
	C_U	80	40	10
	C_M	64	32	10
	C_L	32	16	10

Table 2: Classification Results for Single Classifier

Classifiers	Train set		Test Set	
	Recog. rate	sub. rate	Recog. rate	Sub. rate
C_G	99.83	0.17	97.56	2.44
C_S	99.75	0.25	96.14	3.86
C_U	99.77	0.23	98.13	1.87
C_M	99.15	0.85	94.86	5.14
C_L	88.11	11.89	86.12	13.88

In these experiments, we use five kinds of neural network classifiers with different feature set: Classifier_Gradient(C_G), Classifier_Structure(C_S), Classifier_UDLRH(C_U), Classifier_Mesh(C_M), and Classifier_LSF(C_L). We perform experiments in IBM-PC Pentium-100 using a neural network simulator called NWORKS[14]. The classifier used in our experiment is a multi-layered perceptron with one hidden layer trained with backpropagation algorithm and learning rate 0.5. The training and test data for Digit recognition are taken from NIST database[15]. The number of patterns used in our experiments for training and test are 10766 and 10909, respectively. Note that the test set is used to calculate similarity measure and evaluate the classifier combination. Table 1 shows the number of neurons for each layer in the neural network classifier.

3.2 Classifier Combination Methods

In these experiments, we adopted three kinds of combination methods: Majority Voting for type 1, Borda Count for type 2, and LCA for type 3. As each classifier is implemented by a neural network, it generates an output vector of real values ranging from 0 to 1. In case of Majority Voting, the output of the neural network classifier is transformed into 1 or 0 depending on whether it is the highest output. In case of Borda count, the output of the neural network classifier is transformed into a decreasing rank order.

$$t_k^i = 1 - \frac{rank - 1}{max.rank}.$$

In case of LCA, the output of the neural network classifier is used without any transformation. In these experiments, we do not normalize outputs of the classifiers because each classifier used is of the same kind classifier.

Table 3: Combination Results with LCA

Combined Classifiers	LCA			
	Rec.	Sub.	Rel.	S_{3h}
$C_G + C_S + C_U$	98.61(1)	1.39	98.61(1)	0.71(1)
$C_G + C_S + C_M$	98.07(6)	1.93	98.07(6)	1.04(6)
$C_G + C_S + C_L$	97.83(7)	2.17	97.83(7)	1.15(7)
$C_G + C_U + C_M$	98.58(2)	1.42	98.58(2)	0.74(2)
$C_G + C_U + C_L$	98.49(4)	1.51	98.49(4)	0.83(3)
$C_G + C_M + C_L$	97.48(9)	2.52	97.48(9)	1.36(9)
$C_S + C_U + C_M$	98.50(3)	1.50	98.50(3)	0.83(3)
$C_S + C_U + C_L$	98.29(5)	1.71	98.29(5)	0.97(5)
$C_S + C_M + C_L$	97.30(10)	2.70	97.30(10)	1.50(10)
$C_U + C_M + C_L$	97.76(8)	2.24	97.76(8)	1.31(8)

3.3 Experimental Results

Table 2 shows the classification results for each single classifier, respectively, where experiments were performed without using rejection. Table 3, 4, and 5 show the results of classifier combination using LCA, Majority Voting, and Borda Count, respectively. In those tables, the first column indicates which three classifiers are combined. The second(Rec.), the third(Sub.), and the fourth(Rej.) columns indicate the correct recognition rate, the substitution rate, and the rejection rate if available, respectively. The fifth(Rel.) and the last(S_{3h}) columns indicate the reliability of the combination, and similarity measure, respectively where the reliability is calculated by Rec./(Rec. + Sub.). Note that the last column contains 100 X S_{3h}. The numbers in the parenthesis of the Rec. and S_{3h} columns indicate the rankings of the values in each column.

The experimental results show the existence of a strong relationship among the three rankings of recognition rate(Rec.), reliability(Rel.), and Similarity Measure(S_{3h}) columns. This relationship among the rankings is shown to be stronger in the following order of combination methods: LCA, Majority Voting and Borda Count. Except a case of 3th and 4th, the rankings among the three columns in Table 3 are matched exactly. This good result may come from that the outputs of neural network classifiers are used to Type III classifier combination without information loss. In Table 4 or 5, there are some cases of unmatching among the rankings. This seems due to errors deviating from our assumption on α_2 and α_3. In Table 4, it may also be due to the characteristics of Major Voting method which produces more rejection results. In Table 5, it may also be due to the loss of information during the transformation of an output value of a neural network classifier (to ranking orders) where the output of a winner becomes very closer to 1 whereas the outputs of the other classes become very closer to 0. These experimental results show that we can estimate the performance of classifier combination based on similarity measure.

4 Conclusion

In this paper, we propose a systematic approach to classifier combination based on similarity measure which can be calculated from the errors of each single classifier. In order to evaluate the effectiveness of our proposed method, we perform handwritten digit

Table 4: Combination Results with Majority Voting

Combined Classifiers	Majority Voting				
	Rec.	Sub.	Rej	Rel.	S_{3h}
$C_G + C_S + C_U$	98.19(1)	1.39	0.42	99.60(1)	1.37(1)
$C_G + C_S + C_M$	97.50(5)	1.73	0.77	98.25(7)	1.98(6)
$C_G + C_S + C_L$	97.23(7)	1.75	1.02	98.23(8)	2.11(7)
$C_G + C_U + C_M$	98.01(2)	1.17	0.82	98.82(2)	1.50(2)
$C_G + C_U + C_L$	97.87(3)	1.26	0.87	98.73(4)	1.56(3)
$C_G + C_M + C_L$	96.53(9)	1.81	1.66	98.16(9)	2.61(9)
$C_S + C_U + C_M$	97.82(4)	1.18	1.00	98.81(3)	1.70(4)
$C_S + C_U + C_L$	97.43(6)	1.35	1.22	98.64(5)	1.87(5)
$C_S + C_M + C_L$	96.14(10)	1.92	1.94	98.05(10)	2.91(10)
$C_U + C_M + C_L$	96.65(8)	1.62	1.73	98.35(6)	2.48(8)

Table 5: Combination Results with Borda Count

Combined Classifiers	Borda Count				
	Rec.	Sub.	Rej	Rel.	S_{3h}
$C_G + C_S + C_U$	98.62(1)	1.23	0.15	98.77(1)	0.41(1)
$C_G + C_S + C_M$	97.84(4)	1.81	0.35	98.19(4)	0.72(6)
$C_G + C_S + C_L$	97.05(7)	2.45	0.50	97.54(7)	0.75(7)
$C_G + C_U + C_M$	98.32(2)	1.38	0.30	98.62(2)	0.55(3)
$C_G + C_U + C_L$	97.51(5)	2.10	0.39	97.89(5)	0.54(2)
$C_G + C_M + C_L$	96.80(9)	2.55	0.65	97.44(9)	1.23(8)
$C_S + C_U + C_M$	98.25(3)	1.45	0.30	98.55(3)	0.65(4)
$C_S + C_U + C_L$	97.39(6)	2.13	0.48	97.85(6)	0.67(5)
$C_S + C_M + C_L$	96.80(10)	2.55	0.65	97.44(10)	1.39(10)
$C_U + C_M + C_L$	96.98(8)	2.45	0.57	97.54(8)	1.25(9)

recognition experiments. For classifiers to be used in our combination experiments, we adopt five classifiers with different feature set such Gradient, Structural, UDLRH, Mesh and LSF. Given these 5 classifiers, we repeat experiments of classifier combination with 3 classifiers by varying the subset of classifiers. For combination methods, we adopt Majority Voting for Type I, Borda Count for Type II, and LCA for Type III. The experimental results show that we can estimate the performance of classifier combination based on similarity measure. Further study need be done on the generalization of the similarity measure calculation method and the verification of its effectiveness with more classifiers and various combination methods.

References

[1] Y. S. Huang and C. Y. Suen, "A Method of Combining Multiple Experts for the Recognition of Unconstrained Handwritten Numerals", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 1, pp. 90-94, Jan., 1995.

[2] Y. S. Huang and C. Y. Suen, "An Optimal Method of Combining Multiple Experts for Handwritten Numerical Recognition", *the Third International Workshop on Frontiers in Handwriting Recognition*, pp. 11-20, Buffalo, New York, USA, 1993.

[3] L. Lam and C. Y. Suen, "Increasing Experts for Majority Vote in OCR: Theoretical Considerations and Strategies", *the 4th International Workshop on Frontier in Handwriting Recognition*, pp. 245-254, 1994.

[4] T. K. Ho, J. J. Hull, and S. N. Srihari, "Decision Combination in Multiple Classifier Systems", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 16, NO. 1, pp. 66-75, Jan., 1994.

[5] Y. S. Huang and C. Y. Suen, "Combination of Multiple Classifiers with Measurement Values", *Proc. the 2nd International Conference on Document Analysis and Recognition*, pp. 598-601, 1993.

[6] Fumiaki Yamaoka, Yi Lu, Adnan Shaout and M. Shridhar, "Fuzzy Integration of Classification Results in a Handwritten Digit Recognition System", *the 4th International Workshop on Frontier in Handwriting Recognition*, pp. 255-264, 1994.

[7] D. S. Lee and S. N. Srihari, "A Theory of Classifier Combination: The Neural Network Approach", *Proc. 4th International Conference on Document Analysis and Recognition*, pp. 42-45, 1995.

[8] Y. S. Huang, K. Liu and C. Y. Suen, "A Neural Network Approach for Multi-Classifier Recognition Systems" , *the 4th International Workshop on Frontier in Handwriting Recognition*, pp. 235-244, 1994.

[9] L. Xu, A. Krzyzak, and C. Y. Suen, "Methods of Combining Multiple Classifiers and Their Applications to Handwriting Recognition", *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. 22, NO. 3, pp418-435, 1992.

[10] G. Dimauro, S. Impedovo, G. Pirlo and S. Rizzo, "Multiple Experts: A New Methodology for the Evaluation of the Combination Processes", *the 5th International Workshop on Frontier in Handwriting Recognition*, pp. 131-136, 1995.

[11] H. J. Kang and J. H. Kim, "Combining Multiple Classifiers based on Dependency and Its application", *Journal of the Korea Information Science Society*, Vol. 22, No. 11, 1995.

[12] J.Yoon, J.Kim, and K.Chung, "Comparison of Feature Performance in Off-line Handwritten Character Recognition", in Proceedings of 1996 IEEE invited Workshop on Pattern Recognition for Multimedia Techniques, Taegu, Korea, pp.35-45, Oct. 1996.

[13] J. T. Favata, G. Srikantan and S. N. Srihari, "Handprinted Character/Digit Recognition using a Multiple Feature/Resolution Philosophy", *the 4th International Workshop on Frontier in Handwriting Recognition*, pp. 57-66, 1995.

[14] "Neural Works Professional III and Neural Works Explorer", *Neural Ware Inc.*, 1989.

[15] M. D. Garris, R. A. Wilkinson, " NIST Special Database: 3, Binary Image of Handwritten Segmented Character", Feb. 1992