Towards Automated Classifier Combination for Pattern Recognition

Alper Baykut, Aytül Erçil

Sabanci University, Turkey
{alper.baykut@arcelik.com}, {aytulercil@sabanciuniv.edu}

Abstract. This study covers weighted combination methodologies for multiple classifiers to improve classification accuracy. The classifiers are extended to produce class probability estimates besides their class label assignments to be able to combine them more efficiently. The leave-one-out training method is used and the results are combined using proposed weighted combination algorithms. The weights of the classifiers for the weighted classifier combination are determined based on the performance of the classifiers on the training phase. The classifiers and combination algorithms are evaluated using classical and proposed performance measures. It is found that the integration of the proposed reliability measure, improves the performance of classification. A sensitivity analysis shows that the proposed polynomial weight assignment applied with probability based combination is robust to choose classifiers for the classifier set and indicates a typical one to three percent consistent improvement compared to a single best classifier of the same set.

1 Introduction

The ultimate goal of designing pattern recognition systems is to achieve the best possible classification performance for the task at hand. It has been observed that different classifier designs potentially offer complementary information about the patterns to be classified, which could be harnessed to improve the performance of the selected classifier. A large number of combination methods have been proposed in the literature [1][3][7][8]. A typical combination method consists of a set of individual classifiers and a combiner, which combines the results of the individual classifiers to make the final classification. In this paper, we aim to build a robust classifier combination system given a classifier set. For this purpose, current trends in classifier combination are studied and various classifier combination schemes have been devised. To study classifier combination techniques, 10 classical classifiers are gathered to form a classifier set.[2]

It is very difficult to make sense of the multitude of empirical comparisons for classifier performances that have been made. There are no agreed objective criteria by which to judge algorithms. The situation is made more difficult because rapid advances are being made in all fields of pattern recognition. Any comparative study that does not include the majority of the algorithms is clearly not aiming to be complete. Also, any comparative study that looks at limited number of data sets

cannot give reliable indicators of performance. In this study, an extensive comparative study is realized among a wealth of classifiers applied on different data sets. Their performances are evaluated using a variety of performance measures. This comparative study builds the base of the study on combining classifiers to improve the classification performance.

The set of classifiers that are used in this study are fixed, i.e. they are not optimized especially for the application at hand. Different combination schemes are developed hoping to reach reasonable classification accuracy, which is independent of the characteristic of the application. Section 2 summarizes the different classifiers used in the study, section 3 reviews the performance measures that are proposed and used throughout the study, section 4 outlines the different weighted combination algorithms that are proposed, section 5 gives the experimental results.

2. Classifier Set used in the study

This study focuses on the weighted combination of classification algorithms, in which the weights are optimized based on the performance in the training data set. To have a classifier combination, following classifiers with the given parameters are designed.

KMClus:	K-means clustering (max iteration=10; max error=0.5).
SOM:	Self organizing map clustering (max iteration=1000; learning rate=1).
FANN:	Fuzzy neural network classifier (fuzzification level=3; fuzzification type=0; hidden layer units=25; learning rate=0.001; max iteration=1000; min error=0.02).
ANN:	Artificial neural network classifier (hidden layer units=25; learning rate=0.001; max iteration=1000; min error=0.02).
KMClas:	K-means classifier.
Parzen:	Parzen classifier (alfa=1).
KNN:	K-nearest neighbour classifier (k=3).
PQD:	Piecewise quadratic distance classifier.
PLD:	Piecewise linear distance classifier.
SVM:	Support vector machine using radial basis kernel with (p=1).

The design parameters of classifiers are chosen as typical values used in the literature or by experience. The classifiers are not specifically tuned for the data set at hand even though they may reach a better performance with another parameter set, since the goal is to design an automated classifier combination based on any classifier in the classifier set. The aim of this study is to combine the classifiers' results in a robust way to achieve almost the performance of the best classifier in the classifier set or better.

In this study, the classifiers are modified to produce class probability estimates besides their class labels for all classes. For that purpose, their distance measures or belief values are normalized in the training set as probabilities and applied on the test set. For some classifiers such as fuzzy neural network classifier, artificial neural network classifier and support vector machine, their belief values are converted to posterior probabilities using a normalized mapping. For K-means classifier, Parzen, K-nearest neighbour, piecewise quadratic and piecewise linear distance classifiers their implicit distance measures are explicitly calculated and converted to posterior probabilities for each class. One of the main contributions of this study is that all the

classifiers and clustering based classifier algorithms in the classifier set are modified to produce posterior probabilities for their class assignments for all classes.

For clustering, standard k-means clustering and self-organizing map algorithms are applied on the data in a usual way. After the final step of clustering a one-to-one assignment algorithm is applied on the final clusters of the leave-one-out technique to convert them into a classification problem by solving an optimization problem to assign clusters to class (1). The resulting mapping criteria of clusters to classes are then applied on the test sample.

$$\begin{aligned} &\text{minimize} && \sum_{c=1}^{C}\sum_{n=1}^{N} z_{nc}x_{nc} \\ &\text{subject to} && \sum_{n=1}^{N} x_{nc} = 1 \;\; c = 1\ldots C \;\; (\text{C is the number of classes}) \\ & && \sum_{c=1}^{C} x_{nc} = 1 \;\; n = 1\ldots N \;\; (\text{N is the number of samples}) \\ &\text{where} && x_{nc} = \begin{cases} 1 \text{ if sample n is assigned to class c} \\ 0 \text{ otherwise} \end{cases} \\ & && z_{nc} = \begin{cases} 1 \text{ if sample n is wrongly assigned to class c} \\ 0 \text{ otherwise} \end{cases} \end{aligned} \tag{1}$$

3. Performance Measures for evaluating classifier combination methods

To compare different classification algorithms and combination methods, performance measures should be defined. There are different performance measures which evaluate different performances of classifiers: generalization performance, learning performance, correct and wrong classification performance, real time performance, etc. Performance measures are given in equations (2). The proposed "reliability" is the probability of correct classification for that class. Classification accuracy based on classes (CAC) is the ratio of correct classifications to the sample size. Generally, this is the only performance measure used in the literature. Classification accuracy based on probabilities (CAP) is based on distances of the posterior probabilities p'_i of the classification result and the true classification probability p_i. The proposed overall classification performance (OP) is a measure, which combines the products of performance (P_i) and reliability (R_i) with the counts of samples (N_i) for corresponding class c as a weight. CR_i is the number of correct assignments for class i, WR_{ij} is the number of wrong assignments of class i to class j and UN_i is the number of unclassified samples of class i.

$$P_i = \frac{CR_i}{N_i},\; R_i = \frac{CR_i}{CR_i + \sum_{\substack{j=1 \\ j\neq i}}^{C} WR_{ji}},\; CAC = \frac{\sum_{i=1}^{C} CR_i}{\sum_{i=1}^{C} N_i},\; CAP = 1 - \frac{\sum_{i=1}^{N} |\, p_i' - p_i \,|}{\sum_{i=1}^{C} N_i},\; OP = \frac{\sum_{i=1}^{C} P_i \times R_i \times N_i}{\sum_{i=1}^{C} N_i} \tag{2}$$

4. Weighted Combination Algorithms

This study focuses on combining the results of several different classifiers in a way that provides a coherent inference, which performs a reasonable classification performance for the data set at hand. Therefore four different weighted combination algorithms with three different weight assignment are applied on the data sets. Combination algorithm based on class labels uses the classifiers' class label assignments to combine [1]. For each class, the weights of the classifier is added to the decision value of the classifier combination, if the classifier has decided on that class. The classifier combination decides the assigned class based on the maximum of these decision values.

$$\text{Assign } x_n \rightarrow w_c \text{ if } \sum_{k=1}^{K} W_k \Delta_{cnk} = \max_{i=1..C} \sum_{k=1}^{K} W_k \Delta_{ink} \tag{3}$$

If the weights are equal, this combination algorithm is the classical majority vote. The classifiers are forced to produce binary valued function Δ_{cnk} using the posterior probabilities as: $\Delta_{cnk}=1$ if $P(w_c \mid x_{nk}) = \max_{i=1..C} P(w_i \mid x_{nk})$ and 0 otherwise.

The reliability of the classifier for the assigned class, as proposed in (2), is also integrated to the combination by multiplying the reliability value with the weighted class label. This idea is intuitive, if we handle the classifiers as experts in the decision theory. That is, each classifier has different reliabilities on deciding on different classes. This reliability value increases the influence on the final decision, if the classifier reliability is high for deciding this class, and decreases the influence, if it is unreliable.

$$\text{Assign } x_n \rightarrow w_c \text{ if } \sum_{k=1}^{K} W_k R_{kc} \Delta_{cnk} = \max_{i=1..C} \sum_{k=1}^{K} W_k R_{ki} \Delta_{ink} \tag{4}$$

Combination algorithm based probabilities uses the posterior probabilities of classifiers to carry out the combination. As in the case of combination based class labels on they are weighted with the classifier's weights.

$$\text{Assign } x_n \rightarrow w_c \text{ if } \sum_{k=1}^{K} W_k P(w_c \mid x_{nk}) = \max_{i=1..C} \sum_{k=1}^{K} W_k P(w_i \mid x_{nk}) \tag{5}$$

The reliability of the classifier for the assigned class is also integrated to the combination by multiplying the reliability value with the assignment probability.

$$\text{Assign } x_n \rightarrow w_c \text{ if } \sum_{k=1}^{K} W_k R_{kc} P(w_c \mid x_{nk}) = \max_{i=1..C} \sum_{k=1}^{K} W_k R_{ki} P(w_i \mid x_{nk}) \tag{6}$$

Different algorithms are used to calculate the weights of the classifiers for weighted combination. The easiest way is to combine the classifiers using equal weights, well known as simple majority. This performs a good result if the classifiers in the set are independent and unbiased [1]. The weight of classifier m_k is: $W_k=1/K$.

A better way for assigning weights to classifiers is, to assign their performance values in the training phase. In this study, it is proposed that the defined overall performance values, "OP"s, found in the leave-one-out training phase are assigned as

weights. These results are also integrated with the reliability of the classifiers. The weight of a classifier m_k is its weighted OP value.

Another proposal is to assign weights using a linear fit on the posterior probabilities of leave-one-out results. Better and reliable performance is reached with the weight assignment where the true class probabilities of the training data set and the posterior probabilities found by the classifiers are used to find a linear regression parameters for them. Least square fit parameters for the training data set is used as weights of the classifiers in the combination. The constant term which may be in the equation is not used since shifting will not affect the relative values for classification. The weights are the solution of the following equation (7).

$$\begin{bmatrix} P(w_1 \mid x_1) \\ \vdots \\ P(w_1 \mid x_N) \\ P(w_2 \mid x_1) \\ \vdots \\ P(w_C \mid x_N) \end{bmatrix} = \begin{bmatrix} W_1 \\ \vdots \\ W_k \\ \vdots \\ W_K \end{bmatrix} \begin{bmatrix} P(w_1 \mid x_1, m_1) & .. & P(w_1 \mid x_1, m_2) & .. & P(w_1 \mid x_1, m_K) \\ \vdots & & \vdots & & \vdots \\ P(w_1 \mid x_N, m_1) & .. & P(w_1 \mid x_N, m_2) & .. & P(w_1 \mid x_N, m_K) \\ P(w_2 \mid x_1, m_1) & .. & P(w_2 \mid x_1, m_2) & ... & P(w_2 \mid x_1, m_K) \\ \vdots & & \vdots & & \vdots \\ P(w_C \mid x_N, m_1) & .. & P(w_C \mid x_N, m_2) & .. & P(w_C \mid x_N, m_K) \end{bmatrix} \quad (7)$$

The extension of this polynomial weight assignment proposal is to integrate the reliability of the classifier for the assigned class (8).

$$\begin{bmatrix} P(w_1 \mid x_1) \\ \vdots \\ P(w_1 \mid x_N) \\ P(w_2 \mid x_1) \\ \vdots \\ P(w_C \mid x_N) \end{bmatrix} = \begin{bmatrix} W_1 \\ \vdots \\ W_k \\ \vdots \\ W_K \end{bmatrix} \begin{bmatrix} R(w_1 \mid m_1)P(w_1 \mid x_1, m_1) & .. & R(w_1 \mid m_K)P(w_1 \mid x_1, m_K) \\ \vdots & & \vdots \\ R(w_1 \mid m_1)P(w_1 \mid x_N, m_1) & .. & R(w_1 \mid m_K)P(w_1 \mid x_N, m_K) \\ R(w_2 \mid m_1)P(w_2 \mid x_1, m_1) & .. & R(w_2 \mid m_K)P(w_2 \mid x_1, m_K) \\ \vdots & & \vdots \\ R(w_C \mid m_1)P(w_C \mid x_N, m_1) & .. & R(w_C \mid m_K)P(w_C \mid x_N, m_K) \end{bmatrix} \quad (8)$$

5. Experimental Results

5.1 Classifier Results

Popular data sets from the literature are used for evaluating the different combination schemes proposed and comparing them with existing techniques. The data sets used are summarized in [2]. Classifiers' individual performances are given in Tables 1 [2]. In these tables the best performances for the data sets are marked as bold face. The number of the marked items in the last column, titled "Best" indicates the number of times the classifier has outperformed the others for 14 data sets. The standart deviations are given in last rows. In Table 1 the results show that k-nearest neighbour classifier has the best result seven times out of 14 different data sets. All classifiers show different performance on different data set, for example the k-nearest neighbour classifier, which is the best classifier based on Table 1, has at least 10 percent lower performance compared with other classifiers in some data sets like DIB, D10, WQD, BEM and HRD.

Table 1. Classification accuracy of classifiers based on class labels

CAC	BIO	DIB	D10	GID	IMX	SMR	2SD	WQD	80X	ZMM	BEM	BEV	HRD	IFD	Best
KMClus	71.1	65.9	69.0	50.5	81.8	48.1	42.3	51.7	28.9	34.4	59.0	48.5	73.0	90.0	0
SOM	84.0	64.8	75.5	45.3	72.4	54.3	37.1	84.3	71.1	52.1	45.5	87.0	**84.0**	89.3	1
FANN	70.6	74.7	70.0	36.0	68.8	74.5	36.6	93.3	28.9	8.3	34.0	86.0	81.5	71.3	0
ANN	**87.6**	**76.8**	**78.0**	50.5	86.5	77.9	45.9	96.6	82.2	66.7	53.0	85.5	81.5	82.7	3
KMClas	75.3	46.1	76.0	33.2	88.5	65.4	47.4	62.9	**93.3**	69.8	48.5	74.5	82.5	91.3	1
Parzen	58.8	62.2	21.0	28.5	15.6	24.5	57.7	32.6	35.6	63.5	47.5	82.0	59.5	76.0	0
kNN	84.5	67.6	69.0	**73.4**	**94.3**	**82.2**	**76.3**	76.4	**93.3**	**88.5**	70.0	**92.0**	74.5	95.3	7
PQD	13.9	72.8	72.5	1.9	**94.3**	75.0	43.3	**99.4**	88.9	26.0	**79.5**	65.0	82.0	92.0	3
PLD	84.5	75.7	77.0	57.5	90.6	72.6	47.4	98.3	88.9	85.4	56.5	84.0	81.5	**97.3**	1
SVM	6.2	7.6	70.5	37.4	**94.3**	71.2	26.8	25.8	91.1	36.5	73.5	86.0	80.5	94.7	1
STD	2.4	1.5	2.9	3.0	1.7	2.7	3.1	0.6	3.7	3.3	2.9	1.9	2.6	1.3	

For the sake of this study, all classifiers are like experts with different backgrounds, who try to conclude the class of the sample at hand via decision combination with an acceptable reliability. As stated before, the classifiers are not especially tuned for the training set. The results show that k-nearest neighbour classifier outperforms the others for this classifier set with the current setting of parameters for individual classifiers. Another point is that the k-means clustering and Parzen classifiers are the worst ones. The results also validate that the data sets have different characteristics, since their performance are different for different classifiers. Based on the classifier set with current parameter settings, the data sets DIB, D10, GID, SMR and 2SD are relatively hard to classify.

5.2 Classifier Combination Results

As the first combination algorithm, defined weight assignment algorithms are used on class label based classifier combination, where the classifiers have produced only the class assignment results. The results for this combination can be summarized as in Table 3. The performance of the combination is marked as bold face, if it is better than the average of the top three best performance of the classifier set. Results are grouped for three different performance measures (OP, CAC, CAP). Results of different weight assignments are given in rows. Simple majority voting results are given in rows with "MV" title and the results next to "MV" rows, titled "xR", tabulate the integration of reliability for combination. For the weight assignment titled "OP", overall performance values are assigned as weights. The results integrated with the reliability of the classifiers are given in the rows next to it. The last rows are for weight assignment titled "Poly", where linear regression values are assigned as weights and reliabilities are integrated in the last row.

For class based classifier combination the results of the majority voting are not so promising as expected, since the classifiers in the set are not specifically chosen to be independent and unbiased. The "Best" column indicates the number of times the combination algorithm performs better than the average of the top three classifier. The results show that even this performance can be improved using the classifier reliability for the class label assignments. For the class label based combination algorithm, the integration of the reliability of the classifiers improves the performance of the combination. The results show that the performance assignment as weights is better than the other methods of combination weight assignments. If we consider the

standart deviations of the results, we can say that BIO, DIB and WQD datasets can be better classified using class label based combination method. In Table 2 the classifiers' class probability estimates are used to combine using all the weight assignment algorithms. We note that the integration of the reliability improves the performance. The results compared to the class label based combination ones show us a different behaviour, the polynomial weight assignment is better than the others. This can be explained by the discrete type of function values of the class based combination compared to continuous type of function values for polynomial weight assignment.

Table 2. Probability based classifier combination

OP Prob	BIO	DIB	D10	GID	IMX	SMR	2SD	WQD	80X	ZMM	BEM	BEV	HRD	IFD	Best
MV	**80.0**	54.5	55.3	**42.1**	89.0	**64.4**	39.1	77.1	**93.5**	65.4	48.5	**80.3**	**71.3**	91.6	6
xR	**78.0**	51.4	55.6	**52.8**	89.4	**67.6**	**56.2**	84.9	**93.5**	71.9	**58.5**	**79.7**	**71.3**	91.6	8
OP	**79.5**	54.3	55.8	**53.2**	89.9	**68.9**	**62.5**	82.6	**91.3**	**76.2**	**61.3**	**80.3**	**70.9**	91.6	9
xR	**78.5**	51.0	55.8	**54.3**	89.9	**71.0**	**66.5**	84.5	**91.3**	**78.1**	**61.0**	78.7	**70.9**	91.6	8
Poly	**81.8**	**59.5**	**61.2**	**56.7**	**90.8**	**72.0**	**98.5**	90.8	**93.5**	**81.7**	**64.7**	**87.4**	**70.1**	92.2	12
xR	**79.1**	**58.1**	**61.4**	**56.0**	**90.8**	**73.2**	**67.8**	87.6	**97.8**	**81.6**	**60.8**	**87.9**	**70.4**	92.2	12
CAC Prob	BIO	DIB	D10	GID	IMX	SMR	2SD	WQD	80X	ZMM	BEM	BEV	HRD	IFD	Best
MV	**89.2**	73.6	74.0	**63.1**	93.8	**79.3**	**62.4**	87.1	**95.6**	79.2	69.0	**89.0**	**84.0**	95.3	7
xR	**87.1**	71.5	74.0	**71.5**	93.8	**81.7**	**74.7**	91.0	**95.6**	**83.3**	**76.0**	88.0	**84.0**	95.3	8
OP	**88.7**	73.4	74.5	**71.0**	93.8	**82.2**	**78.9**	90.4	**93.3**	**86.5**	**77.5**	**89.0**	**84.0**	95.3	9
xR	**88.1**	71.4	74.5	**72.9**	93.8	**83.7**	**80.9**	91.6	**93.3**	**87.5**	**77.5**	88.0	**84.0**	95.3	8
Poly	**90.2**	**77.1**	**78.0**	**74.3**	**94.3**	**84.6**	**99.0**	94.9	**95.6**	**89.6**	**79.0**	**93.0**	**83.5**	95.3	12
xR	**88.7**	**76.2**	**78.0**	**72.9**	**94.3**	**85.1**	**70.6**	93.3	**97.8**	**89.6**	**76.5**	**93.5**	**83.5**	95.3	12
CAP Prob	BIO	DIB	D10	GID	IMX	SMR	2SD	WQD	80X	ZMM	BEM	BEV	HRD	IFD	Best
MV	63.2	57.6	60.4	77.5	74.3	59.9	51.8	67.6	68.5	82.8	54.4	67.2	67.3	74.2	0
xR	63.6	58.9	60.3	79.7	75.2	60.9	56.5	68.3	72.0	84.1	56.1	68.3	67.3	74.3	0
OP	65.8	58.2	60.8	79.3	75.5	61.8	60.2	68.4	72.2	84.4	57.4	69.8	67.5	74.6	0
xR	65.5	59.4	60.8	82.0	76.0	62.6	64.7	68.5	73.6	85.1	58.9	70.3	67.6	74.7	0
Poly	**81.0**	**67.5**	68.3	**86.6**	**96.1**	**77.7**	**92.5**	**86.4**	**95.0**	**94.6**	**67.9**	88.3	75.0	94.4	10
xR	76.6	**70.7**	69.7	84.9	**96.5**	**85.1**	**83.8**	**86.4**	89.8	**97.3**	**76.2**	**92.4**	74.4	94.7	8

Based on CAC, the proposed polynomial weight assignment for weighted classifier combination outperforms the other combination methods. Comparing single classifier and combination results, except for the data sets WQD, BEM and IFD, the combination performance is even higher than the best classifier's performance. For example for the DIB data set, the best possible class performance is 76.8 percent, which is increased to 77.1 percent. Considering the standart deviations of the single classifier results we can even state that the BIO, DIB, SMR, 2SD, and 80X data sets are classified better than the single best classifiers.

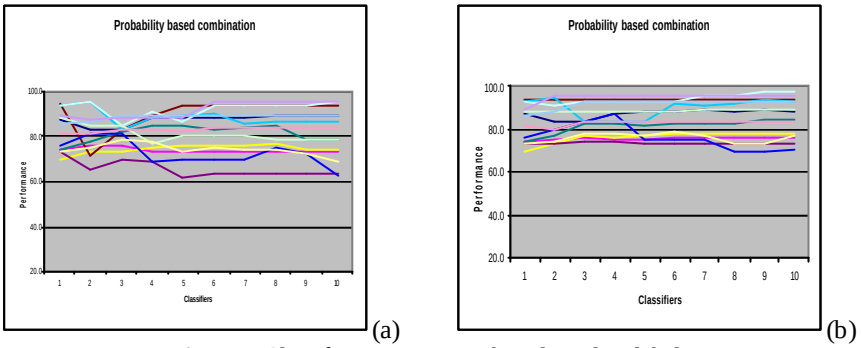
The analysis of removing the worst and best classifiers improves the classification performance for some data sets. For the worst classifier removal, improvements are better than the case of removing the best classifier. A detailed study [2] shows that the behaviour is different depending on the characteristics of the data sets. In fact the removal of worst classifier improves the performance as expected, but the result for the best classifier was an unexpected result. A closer look at the k-nearest neighbour classifier based on the sum of squared errors of posterior probabilities is given in Table 3. The bold face values indicate the best sum of squared errors among the classifiers on the same data set on that column. The last column titled "Best" indicates the number of times the classifier has the best sum of squared error of posterior probability among the classifier collection. The results show that the k-nearest neighbour classifier is one of the best ones, but not in all cases.

Table 3. Sum of squared errors on probabilities of classifier set

SSE	BIO	DIB	D10	GID	IMX	SMR	2SD	WQD	80X	ZMM	BEM	BEV	HRD	IFD	Best
KMClus	74.3	89.7	56.5	370.5	174.6	70.4	71.6	141.6	135.3	520.1	85.1	91.4	65.2	109.1	0
SOM	32.0	70.3	49.0	109.3	55.2	91.3	125.8	31.5	57.8	95.8	109.0	26.0	32.0	**21.3**	1
FANN	32.6	**33.8**	**38.0**	116.2	56.1	**34.0**	69.2	**27.3**	85.8	224.0	62.4	34.3	**29.0**	35.1	5
ANN	**31.3**	41.1	39.9	80.8	69.9	38.4	58.5	29.7	48.2	131.4	54.4	40.6	34.2	48.3	1
KMClas	95.0	95.4	53.1	380.7	165.2	65.3	71.4	143.0	92.4	528.5	83.1	85.2	52.4	109.1	0
Parzen	72.9	70.2	96.8	86.2	95.2	93.5	79.2	87.0	92.7	82.4	86.4	59.3	76.7	84.4	0
kNN	41.6	54.7	55.5	**51.9**	**33.7**	43.1	**37.0**	49.1	**33.0**	**36.0**	52.9	32.9	45.7	33.1	5
PQD	95.8	80.5	53.6	386.0	195.0	74.7	72.3	147.3	129.3	639.0	75.3	87.5	53.9	140.4	0
PLD	53.0	51.9	42.1	183.6	121.0	56.4	56.0	76.0	69.2	190.3	51.9	56.4	30.6	51.5	0
SVC	53.1	54.7	77.9	69.4	73.7	46.0	128.6	75.5	85.9	68.4	**47.6**	**21.2**	34.5	68.9	2

To have an automated classifier, we may have a collection of fuzzy neural networks, artificial neural networks, knearest neigbour and support vector machine based classifiers, and combine them. The design of this subset of classifiers is based on their sum of squared error on posterior probabilities in Table 3. The four classifiers chosen, have the lowest sum of squared errors. The results of this set was almost as good as the original classifier set with all the classifiers. The probability based combination with overall performance values assigned as weights performs better than the whole classifier set for classification accuracy based on probability.

A more detailed sensitivity analysis is done on data sets separately [2]. For each data set the classifiers are sorted based on the sum of squared errors, respectively. Beginning with the best classifier, all classifiers are incrementally added to the classifier set. Their performance changes are graphically presented in Figure 1.



(a) (b)

Figure 1. Classification accuracy based on class labels

As can be seen in Figure 1(a), the classification accuracy based on class labels for the majority vote of the classifier set, after the best classifier is in the subset, the performance first drops by adding the second or third best classifier and then it begins to improve sometimes to its initial level and sometimes above it. This is due to the fact that the best classifiers in the classifier set are not independent of each other, that is they are correlated in the misclassification of the same test samples. A complete search for the best classifier subset may also be carried out by considering all possible subsets of the original classifier set. A further research should be done on the class characteristics of the data sets, since the performance of classifiers are not just data set dependent, their performance also depend on the different classes of the data set. For some classifiers, certain classes of some data sets may be classified more reliably than the other classes of the same data set. In fact in this study, this kind of analysis is added to the combination by the proposed reliability factors.

Comparing the sensitivity of adding classifiers to the set in the figures 1(a) and (b) show that polynomial weight assignment is more robust for addition of new classifiers. Another point is that probability based classifier combination is more robust to class label based combination algorithms, as can be noticed by the smaller variability in the performance, which ranges between 70 percent and 100 percent (Figure 1(b)), compared to the large variation of the simple majortity vote case of 60-100 percent range (Figure 1(a)). Based on this analysis we can outline a guide for designing a classifier set for combination.

Assuming that at the design time of the classifier set, the time is not a critical issue, all available classifiers should be trained. If possible, they may be tuned for better performance. At training phase the time costly leave-one-out algorithm should be used. The reliabilities after this training should be recorded for testing phase. Than a sensitivity analysis should be carried out: either beginning with best classifier based on the sum of squared probability errors, all classifiers are added to the classifier set and the new performance is traced after each step till all classifiers are added, or all possible subsets of classifiers for classifier set should be considered. Either the best set or if the performance difference is not so high, all classifiers should be selected for classifier set. Finally using results of leave-one-out, the polynomial weights and the class reliabilities of classifiers should be calculated and used for the testing phase. The probability based combination algorithm is more robust as the combination algorithm.

6. SUMMARY AND CONCLUSIONS

In this study different classifier combination schemes are proposed and realized in an integrated framework. All classifiers and combination schemes are evaluated using a variety of performance measures. When combining different classifiers, the weighted combination methods are applied as the combination schemes. The two basic questions concerning the weighting methods: what to weight and how to weight, direct us to new ideas of alternatives for combination methods and for weight assignments. Class and probability based combination methods are applied on the data set and experimentally demonstrated that the proposed probability based weighted combination method is a robust way of combining classifiers. The weights of classifiers are basically based on their performances in the training phase, assuming that they will achieve almost the same performance for the test samples. Overall performance values, originated by the class performance and reliability values proportional to the class population for a specific class after the leave-one-out training phase, is proposed for weight assignment. A better way of assigning weights is proposed by using the least square fit parameters of true and estimated posterior probabilities in the leave-one-out training, and it is called polynomial weight assignment. The classical equal weight assignment is also implemented and tested in the framework to compare the effectiveness of proposed methods.

Sensitivity analysis of selecting a classifier subset to achieve best performance possible with the current classifier set is performed. The basic idea behind selection of a classifier for the classifier set is that, the individual classifier which will be added

in the set should not be strongly correlated in the misclassification of the current classifiers in the classifier set. This criterion is not easily satisfied, since even different classes of the same data set may have different characteristics. That is, the classifier's performance may vary for the different classes of the same data set. The results of different sensitivity analysis show that the probability based combination with polynomial weights is a robust way to combine classifiers. Probability based combination with polynomial weights achieves the best possible performance with the current set of classifiers at hand.

To have a complete comparative study, all the proposed combination schemes and weight assignments are applied on the data sets and their performances are evaluated using the proposed performance measures. Some of the performance measures proposed in this study include the reliabilities of the classifiers for their decisions based on their training performances. The reliability values integrate their trust for their decisions, which is used with the assigned weights, to improve the performance of correct classification and reduce the overall misclassification error. Hence, the integration of the reliability measure in the combination improves the classification performance; even the simplest combination scheme of equal weight assignment for classifiers is improved. The main observation of this study is that a typical one to three percent consistent improvement compared to a single best classifier is seen when combining classifiers using the polynomial weight assignment applied with probability based combination.

7. REFERENCES

[1] Alexandre, L., A. Campilho and M. Kamel, 2000, "Combining Unbiased and Independent Classifiers Using Weighted Average", *11th Portuguese Conference on Pattern Recognition*, pp. 495-498, Porto, Portugal.

[2] Baykut, A., 2002, "Classifier Combination Methods in Pattern Recognition", PhD. Thesis, Bogaziçi University, Istanbul, Turkey.

[3] Bauer, E. and R. Kohavi, 1999, "An Empirical Comparison of Voting Classification Algorithms: Bagging", *Boosting and Variants, Machine Learning*, Vol. 36.

[4] Christianini, N. and J. S. Taylor, 2000, *Support Vector Machines and other kernel-based learning methods*, Cambridge University Press, UK.

[5] Cortes, C. and V. Vapnik, 1995, "Support Vector Networks", *Machine Learning*, Vol. 20, pp. 273-297.

[6] Duda, R. O. and P. E. Hart, Stork, D.G. 2001, *Pattern Classification*, John Wiley&Sons.

[7] Kittler, J., 1998, "Combining Classifiers: A Theoretical Framework", *Pattern Analysis and Applications*, Vol. 1, No. 1, pp. 18-28.

[8] Kittler, J., M. Hatef, R. P. W. Duin and J. Matas, 1998, "On Combining Classifiers", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 20, No. 3, pp. 226-240.

[9] Shalkoff, R. J., 1992b, Pattern Recognition: Statistical, Structural and Neural Approaches, John Wiley&Sons.