Similarity-based Regularization for Semi-Supervised Learning for Handwritten Digit Recognition

D. Barbuzzi, G. Pirlo
Department of Computer Science
University of Bari Aldo MORO
Bari, Italy
{donato.barbuzzi, giuseppe.pirlo}@uniba.it

S. Uchida, V. Frinken
Faculty of Information Science and Electrical Engineering
Kyushu University
Fukuoka, Japan
{uchida, vfrinken}@ait.kyushu-u.ac.jp

D. Impedovo
Department of Mechanics, Mathematics and Management
Polytechnic of Bari
Bari, Italy

***Abstract*— This paper presents an experimental analysis on the use of semi-supervised learning in the handwritten digit recognition field. More specifically, two new feedback-based techniques for retraining individual classifiers in a multi-expert scenario are discussed. These new methods analyze the final decision provided by the multi-expert system so that sample classified with a confidence greater than a specific threshold is used to update the system itself. Experimental results carried out on the CEDAR (handwritten digits) database are presented. In particular, error rate, similarity index and a new correlation score among them are considered in order to evaluate the best retraining rule. For the experimental evaluation, an SVM classifier and five different combination techniques at abstract and measurement level have been used. Finally, the results show that iterating the feedback process, on different multi-expert systems built with the five combination techniques, one retraining rule is winning over the other respect to the best correlation score.**

Keywords*— Semi-Supervised Learning; Feedback-based Strategies; Handwritten Digit Recognition; Multi-Expert Intelligent System; SVMs.

I. INTRODUCTION

The basic problem in character classification is to assign a digitized character to its symbolic class. In the case of a printed text image, this is referred to as optical character recognition (OCR). In the case of handwritten text, it is loosely referred to as intelligent character recognition (ICR). The recognition of a character from a single, machine-printed font family on a paper document can be done very accurately. In contrast, difficulties arise when handwritten character are to be handled [1]. In this paper, we discuss the recognition of handwritten single digits.

In general, the recognizers need a large amount of handwritten digits and the corresponding labels for training. Creating this ground truth, however, is a costly and tedious task since it needs to be done by human annotators. On the other hand, unlabeled data can be obtained cheaply, but there are few ways to use them. Making use of both labeled and unlabeled data for classifier training is known as semi-supervised learning [2,3]. Most work on semi-supervised learning deals with the standard classification scenario, but the basic observation is that, if an ensemble of classifiers is available, there is no analysis and use of their common behavior of classification given the input to be recognized [2, 3].

Recently, for this purpose, multi-expert systems have been used for two reasons: (1) to capture the variability of the numerous handwritten digits and (2) to select the most profitable samples when new data becomes available.

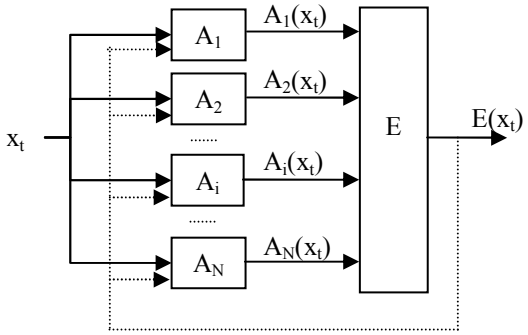

Fig. 1 Feedback in a Multi-Expert Parallel System

The main contribution of this paper is to propose two new feedback-based strategies for retraining individual classifiers in a multi-expert scenario (see Fig. 1). More specifically, it is exploited the collective behavior of a multi-expert system to label and select the most profitable samples from a set of unlabeled data. The whole system classifies a large set of unlabeled data and uses the most confidently recognized patterns to create a new training set for each expert in the system. A single expert is iteratively retrained by enlarging the original training set with this new dataset, respect to a specific retraining rule.

Two different feedback-based strategies are analyzed: the first rule corrects the misclassification of a single expert with respect to the final decision. The second rule reinforces the classifier's knowledge base but at each iteration all training

data for the individual experts are redistributed. Finally, a new correlation score is used in order to evaluate the retraining rules with respect to error rate and similarity among the individual classifiers.

As already discussed in the literature, the achievement of better performance depends on the iteration of the feedback process [4], on the combination strategy of the multi-expert system, but also on the data distribution and similarity between samples in the training set, feedback set and test set [5, 6, 7, 8].

The approach proposed in this paper has the fundamental objective of simultaneously minimizing its classification error rate as well as the experts' correlation score to avoid an overspecialization. All experiments have been carried out on the handwritten digits of the CEDAR database. An SVM classifier and five different combination techniques have been investigated: Majority Vote, Weighted Majority Vote, Maximum Probability, Sum Rule, and Product Rule.

The paper is organized as follows. Section 2 presents an overview on the state-of-the-art in semi-supervised learning. Section 3 describes in detail the two feedback-based strategies. The operating conditions are presented in Section 4. The experimental results and conclusion are reported, respectively, in Section 5 and 6.

II. Semi-Supervised learning: Overview

Semi-supervised learning uses a large amount of unlabeled data, together with a limited amount of the labeled data, to build better classifiers.

Self-training (or Self-update) is a commonly used technique for semi-supervised learning. In self-training the classifier is first trained on the set of labeled data and, subsequently, those unlabeled data classified with a high confidence are moved to enlarge the original training set. The whole process iterates until a stopping criterion is satisfied. A typical stopping criterion is that either there is no unlabeled instances left or the maximum number of iterations has been reached. Note that the classifier uses its own predictions to retrain itself. One can imagine that a classification mistake can reinforce itself. Some algorithms try to avoid this by "unlearn" unlabeled data points if the prediction confidence drops below a threshold [2].

Co-training (or Co-update) is the situation under which two classifiers improve each other. Co-training assumes that: 1) features can be split into two sets; 2) each sub-feature set is sufficient to train a good classifier; 3) the two sets are conditionally independent given the class. Each classifier then classifies the unlabeled data, and "teaches" the other classifier with the few unlabeled examples (and the prediction labels) they recognize confidently. Each classifier is retrained with the additional training examples given by the other classifier. Moreover, co-training can be easily extended from two to *n* classifiers. The entire process iterates until stopped by a similar criterion to self-training.

Co-training and Self-training have a very strong role, in the current state-of-the-art, in biometrics template updating processes [9,10] or the field of handwriting recognition [11,12].

Other instance selection methods in Semi-Supervised Learning can be studied in the literature survey [2,3].

III. Retraining system

A. Instance selection from ensemble behavior

Let:

- C_j, for $j = 1,2,\dots,M$ be the set of pattern classes;
- $P = \{x_k | k = 1,2,\dots,K\}$ be the set of patterns. In this work P is considered to be partitioned into 3 subsets: P_1, P_2 and P_3. In particular, P_1 is used for learning only and its data are labeled, whereas P_2 is used both for classification and learning (when necessary) and their data are unlabeled. Finally, P_3 is used for testing set.
- $y_s \in \Omega$ be the label of the pattern x_s, $\Omega = \{C_1, C_2, \dots, C_M\}$;
- A_i be the i-th classifier, $i = 1,2,\dots,N$;
- $F_i(k) = (F_{i,1}(k), F_{i,2}(k), \dots, F_{i,r}(k), \dots, F_{i,R}(k))$ be the numeral feature vector used by A_i for representing the pattern $x_k \in P$ (for the sake of simplicity it is here assumed that each classifier uses the same R numeral features);
- KB_i be the knowledge base of A_i after the processing of the set P_2. In particular $KB_i = \{KB_i^1, KB_i^2, \dots, KB_i^M\}$;
- E be the multi expert system which combines the individual classifier decisions in order to obtain the final classification result.

Initially, in the first stage (s=1), the classifier A_i is trained using the patterns $x_k \in P_1$. Therefore, the knowledge base KB_i of A_i is initially defined as:

$$KB_i = \{KB_i^1, KB_i^2, \dots, KB_i^M\} \quad (1)$$

where, for $j = 1,2,\dots,M$:

$$KB_i^j = \{F_{i,1}^j(k), F_{i,2}^j(k), \dots, F_{i,r}^j(k), \dots, F_{i,R}^j(k)\} \quad (2)$$

being $F_{i,r}^j(k)$ the set of the r^{th} feature of the i^{th} classifier for the patterns of the class C_j that belongs to P_1.

Successively, the set P_2 of unknown samples is provided to the multi-expert system both for classification and for learning. P_3 is just considered to be the testing set in order to avoid biased or too optimistic results. When considering new data (samples of P_2), in order to inspect and take advantage of the common behavior of the ensemble of classifiers, the following two simple strategies are proposed and evaluated in this work:

The first rule chooses the elements for retraining are

$$\{x_t | x_t \in P_2 \wedge s_{ME}(E(x_t)) > \tau \wedge A_i(x_t) \neq E(x_t)\} \quad (3)$$

while in the second rule, the set or elements is

$$\{x_t | x_t \in rP_2 \wedge s_{ME}(E(x_t)) > \tau\} \quad (4)$$

with rP_2=1/3 of (unknown) data in P_2, chosen randomly.

In Eq. (3), hereafter called *error-based (ERR) strategy*, A_i is updated with those patterns where the output of A_i disagrees with the output of the multi-expert system, if the multi-expert system classifies the pattern with a confidence greater than a threshold τ. In contrast, in Eq. (4), hereafter called *redistribution-based (RED) strategy*, A_i is updated with 1/N of unknown data in P_2, chosen randomly, classified with an high confidence measure by multi-expert system; successively the N different knowledge bases are redistributed randomly in the whole system.

It is worth noting that these two strategies, dubbed error-based and redistribution-based, take into account the performance of the individual classifier as well as the performance of the multi-expert system. They are able to select not only samples to be used for the updating process, but also classifiers to which those samples must be returned in order to improve the multi-expert performance.

B. Algorithm

In order to describe these two different feedback-based strategies, they are detailed in Algorithm 1. Each expert is trained on an initial set of labeled data, named P_1 and a set P_2 of unlabeled samples. Among all the samples in P_2, only those classified with high confidence by the multi-expert system are used to enlarge the knowledge base of those experts.

Algorithm 1: Feedback-based Retraining Process

1. Given:
   - $P_1=\{x_1,x_2,..., x_L\}$: the initial training set
   - $P_2=\{x_{L+1},\ldots, x_T\}$: the unlabeled data set
   - KB_i: the knowledge base of the expert A_i, i=1,2,..,N

2. For h=L+1,…,T
   3. For each expert:
      $A_i(x_h, P_1)$ is the classification of the sample $x_h \in P_2$ given P_1 as training data
      End for
   4. Apply the *combination rule*: $E(x_h,P_1)$ is the final decision of the sample $x_h \in P_2$ on P_1 combining all experts.
   5. Determine for the sample $x_h \in P_2$ a classification score of the Multi-Expert system $s_{ME}(E(x_h,P_1))$
   6. For each expert
      Eq. (3) OR Eq. (4)
      End for

End for.

Obviously, in the error-based strategy, many new samples will not give any feedback to a specific expert. This phenomenon depends on the acceptance threshold, the classifiers performance and the ratio between new and old data. As a matter of fact, given a specific classifier, the misclassifications could be attributed to the fact that the classifier is unable to represent the specific class or sample, and no improvement would be obtained by introducing unnecessary new data in its knowledge base. Also, if each classifier in the ensemble were able to recognize exactly the same set of patterns, their combination would be not useful [13,14]. In order to avoid this phenomenon, the Similarity Index [13] is used to estimate the similarity between experts.

Consequently, the final goal of this paper is to find a good compromise between these two phenomenons to investigate on the better retraining rule in order to obtain successful results.

IV. Experimental Setup

In this paper the handwritten digits (classes from "0" to "9") of the CEDAR database are used. Each digit image is split into 16 regular regions [15,16], successively for each region the occurrences of the following set of features have been considered:

1. *Geometric features*: hole, up cavity, down cavity, left cavity, right cavity, up end point, down end point, left end point, right end point, crossing points, upper extreme points, down extreme points, left extreme points, right extreme points;
2. *Contour profiles*: max/min peaks, max/min profiles, max/min width, max/min height;
3. *Intersection with lines*: 5 horizontal lines, 5 vertical lines, 5 slant -45° lines and 5 slant +45° lines.

For our purpose, the DB has been partitioned into 3 subsets: P_1, P_2 and P_3 for training, feedback (unlabeled data), and test set.

More specifically, the training set contained 1000 samples, 100 for each class, while feedback and test set consist of 17223 and 2128 digits, respectively. In particular, $P_1 \cup P_2$ represents the set usually adopted for training when considering the CEDAR (handwritten digits) DB. P_3 is the testing dataset.

Two different experimental setup have been. In the first setup, we use only P_1 as the initial training. In the second setup, training was done on data, randomly chosen from $P_1 \cup P_2$. For the latter, five different test runs were conducted.

The multi-expert parallel system uses three different SVM classifiers, named A_1, A_2, and A_3. Each classifier uses the same set of features, but different samples in each knowledge base. The kernel function adopted in this work is the radial basis function (rbf). The performance is certainly influenced by the number of features, the kernel parameter (gamma) and by the tolerance of classification errors in learning (C) [17].

TABLE I. RETRAINING RESULT USING P1 AS TRAINING SET

	First Iteration		*Second Iteration*		*Third Iteration*	
	ERR_Strategy	***RED_Strategy***	***ERR_Strategy***	***RED_Strategy***	***ERR_Strategy***	***RED_Strategy***
ER	5.58	4.28	5.11	4.27	5.00	**4.22**
SI	44.72	76.94	67.26	80.12	**76.64**	81.45

TABLE II. RETRAINING RESULT USING A SUBSET OF P1∪P2 AS TRAINING SET

	First Iteration		*Second Iteration*		*Third Iteration*	
	ERR_Strategy	***RED_Strategy***	***ERR_Strategy***	***RED_Strategy***	***ERR_Strategy***	***RED_Strategy***
ER	4.72	4.05	4.14	3.70	4.04	**3.59**
SI	43.44	74.96	59.94	77.51	**72.41**	80.36

Finally, in a multi-expert scenario both, combination technique and acceptance threshold, play a crucial role in the selection of new patterns to be added to the classifiers' knowledge base in the proposed approach. In this work the following combination rules have been considered and compared: Maximum Probability (MP), Majority Vote (MV), Weighted Majority Vote (WMV), Sum Rule (SR) and Product Rule (PR). Threshold τ is set to 0.25 as proposed in the literature [18]. After an output is produced by the multi-expert system, min-max normalization is applied. All those samples classified with a score greater than 0.25 are then used to enlarge the training set.

Finally, up to three iterations for both strategies, Eq. 3 and Eq. 4, have been performed.

V. RESULTS

In Tables I and II, the results obtained by the five combination rules discussed above are averaged for each strategy and the mean values of error rate (ER) and Similarity Index (SI) are considered.

The label "*ERR_Strategy"* indicates the use of the error-based strategy proposed in Eq. 3 for instance selection and the label "*RED_Strategy*" shows the results of the redistribution-based strategy described in Eq. 4.

Table I presents the error rate and similarity index considering the first setup. In other words, P_1 is used for training and P_3 for testing. P_2 is used for feedback learning. In this setup, the best error rate after three iterations is obtained using the *redistribution-based strategy*. More specifically, an improvement of 0.78% compared to the other strategy is achieved. The best result with respect to the similarity index is obtained using the *error-based strategy*.

Table II reports results related to the second setup. In particular, the training set is assigned randomly in $P_1 \cup P_2$, the remaining data for feedback learning. P_3 is used for testing. In this second case, the best error rate after three iterations is obtained using the *redistribution-based strategy*, which means an improvement of 0.45% compared to the other strategy. Again, the best result with respect to the similarity index is obtained using the *error-based strategy*.

To combine both measures into a single score, we propose the use of a new correlation score (CS) that combines the error rate and similarity measure. The idea behind this score is the observation that a similarity index of an ensemble of recognizers is not independent of the recognition rate. For low error rates, most of the samples are recognized correctly and hence, all recognizers produce the same output. To counteract this effect, we multiply the similarity index with the error rate, thus CS = ER*SI/100. A lower correlation score indicates a higher diversity between the classifiers. This can be beneficial for further self-training iterations as well as possible improvements using ensemble methods.

The new correlation score for all different strategies as a function of the self-training iterations is given in Fig. 1 and Fig. 2. It is easy to observe that the *redistribution-based strategy* is the best among the investigated combination rules. In particular, Fig. 1 shows that the *redistribution-based strategy* using the combination rules: MP, WMV, SR and PR better adapt the system. Also in Fig. 2, it is possible to observe that with the combination rules MP and SR the *redistribution-based strategy* outperforms the *error-based strategy*.

VI. CONCLUSION

Two new feedback-based strategies for instance selection in semi-supervised multi-expert systems have been introduced, according to five different combination rules.

The experimental results show that the better retraining rule depends not only to the classifier structure and the combination strategy of the multi-expert which is responsible for sample selection, but also on data distribution of the initial training set as well as similarity between samples in the feedback and testing set. The results also show that multiple self-training iterations on the same data set are able to improve performance at multi-expert level. Finally, the use of a correlation score helps to understand why some retraining rule perform better with respect to error rate and similarity index.

In the future, these strategies will be investigated on the task of unsupervised learning.

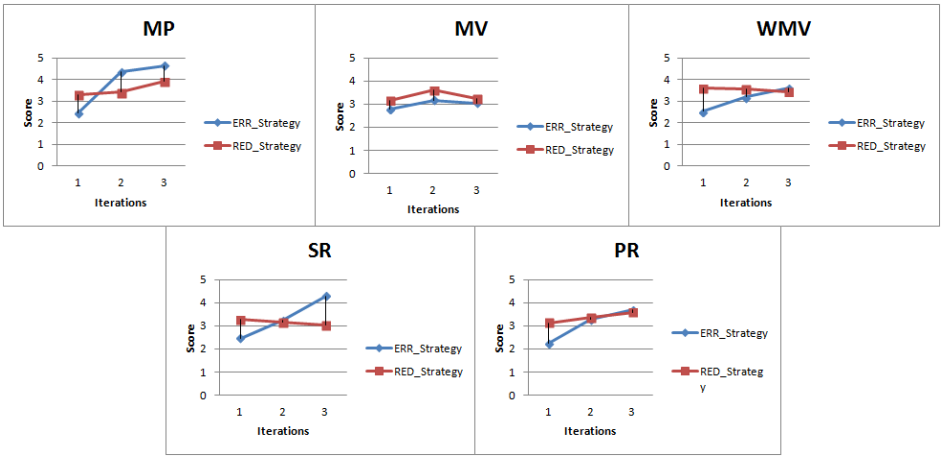

Fig. 1. Retraining result using P1 as training set

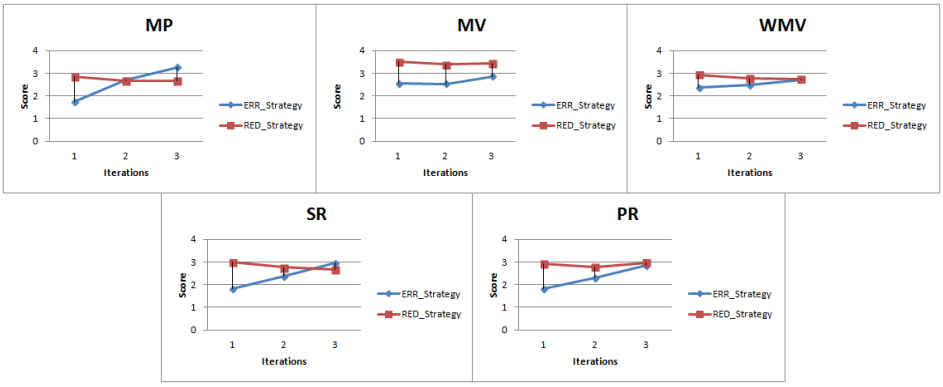

Fig. 2. Retraining result using A Subset of $P_1 \cup P_2$ as Training Set

REFERENCES

[1] R. Plamondon and S.N. Srihari, "On-line and Off-line Handwriting Recognition: A comprehensive survey", IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 22, n.1, pp. 63-84, 2000.

[2] X.J. Zhu, "Semi-supervised learning literature survey" (2008).

[3] Y. Guo, H. Zhang, X. Liu, "Instance Selection in Semi-supervised Learning". Canadian Conference on AI 2011: 158-169.

[4] D. Barbuzzi, D. Impedovo, F.M. Mangini, G. Pirlo, "Learning Iterative Strategies in Multi-Expert Systems Using SVMs for Digit Recognition". In: A. Petrosino. Image Analysis and Processing – ICIAP 2013. LECTURE NOTES IN COMPUTER SCIENCE, vol. 8156, p. 121-130.

[5] G. Pirlo, C.A. Trullo, D. Impedovo, "A Feedback-Based Multi-Classifier System", IEEE proc. of ICDAR 2009, pp. 713-717, 2009.

[6] D. Impedovo and G. Pirlo "Updating Knowledge in Feedback-based Multi-Classifier Systems", in IEEE proc. of ICDAR2011, pp. 227-231, 2011.

[7] D. Impedovo, G. Pirlo, D. Barbuzzi, "Supervised Learning Strategies in Multi-Classifier Systems". In: Proceedings of ISSPA 2012. Montreal, Canada, July 2-5, 2012, p. 1215-1220.

[8] D. Barbuzzi, D. Impedovo, G. Pirlo, "Benchmarking of Update Learning Strategies on Digit Classifier Systems". In: Proceedings of ICFHR 2012. Capitolo, Bari, September 18-20, 2012, p. 35-40.

[9] V. Vapnik, "Statistical learning theory". Wiley-Interscience, 1998.

[10] U. Uludag, A. Ross, A. Jain, "Biometric template selection and update: a case study in fingerprints", Pattern Recognition, Vol. 37, Issue 7, pp. 1533–1542, 2004.

[11] V. Frinken, H. Bunke "Evaluating Retraining Rules for Semi-Supervised Learning in Neural Network Based Cursive Word Recognition", proc. IEEE of ICDAR2009, pp. 31-35, 2009.

[12] V. Frinken, A. Fischer, H. Bunke, A. Fornes, "Co-Training for Handwritten Word Recognition", in IEEE proc. of ICDAR2011, pp. 314-318, 2011.

[13] G. Pirlo, D. Impedovo, D. Barbuzzi, "The Similarity Index lower and upper bounds: Theoretical Considerations and Experimental Verification", in INTERNATIONAL JOURNAL OF MATHEMATICAL MODELS AND METHODS IN APPLIED SCIENCES, Issue 7, Volume 7, pp. 682-691, 2013.

[14] S. Gunter, H. Bunke, "Feature selection algorithms for the generation of multiple classifier systems and their application to handwritten word recognition", Pattern Recognition Letters 25 (2004), pp. 1323-1336.

[15] G. Pirlo, D. Impedovo, "Fuzzy-Zoning-Based Classification for Handwritten Characters", IEEE Trans. on Fuzzy Systems, Vol. 19, Issue 4, pp. 780-785, 2011.

[16] D. Impedovo, G. Pirlo, "Zoning methods for handwritten character recognition: A survey", Pattern Recognition, Volume 47, Issue 3, March 2014, Pages 969–981.

[17] Chih-Chung Chang, Chih-Jen Lin, "LIBSVM: a library for support vector machines". ACM Transactions on Intelligent Systems and Technology, 2:27:1--27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[18] S. Impedovo, D. Barbuzzi, G. Pirlo, "Evaluating Threshold for Retraining Rule in Semi- Supervised Learning using Multi-Expert System". In Proceedings of 14th International Conference on Frontiers in Handwriting Recognition. p. 169-174 , Crete, September 1-4, 2014.