Dynamically Expanding Context as Committee Adaptation Method in On-Line Recognition of Handwritten Latin Characters

Jorma Laaksonen, Matti Aksela, Erkki Oja
Helsinki University of Technology
Laboratory of Computer and Information Science
P.O.Box 5400, Fin-02015 HUT, Finland
{jorma.laaksonen,matti.aksela,erkki.oja}@hut.fi

Jari Kangas
Nokia Research Center
P.O.Box 100
Fin-33721 Tampere, Finland
jari.kangas@research.nokia.com

Abstract

We have developed an adaptive handwriting recognizer for isolated Latin characters in which the adaptive behavior is based on the Dynamically Expanding Context (DEC) algorithm. In our current system, the outputs of a set of static classifiers are combined in a committee machine, whose rules are adapted. Every misclassified character gives rise to adding a new DEC rule to the rule set of the committee. When the existing rules fail to produce correct recognition output, more and more context information is utilized in forming the new DEC rules. Not only the first-ranking outputs from the member classifiers but also the second-ranking ones can be taken into account when forming the DEC rules.

In the experiments described in this paper, various options in the implementation of the DEC committee classifier are evaluated. The results of the experiments show that the system is capable of fast adaptation to the user's handwriting and leads to lowered recognition error rates.

1. Introduction

On-line adaptation in a handwriting recognizer can be implemented in various ways. The most common approach is to have a single classifier which is adapted to the user's writing style during the training phase of the system. Another alternative, experimented with in this paper, is to have a set of classifiers whose outputs are combined in a committee machine. This committee then makes the classification decisions and is adapted to match the user's writing style. The member classifiers themselves can be either static or adaptive.

A simple committee classifier can perform majority voting on its inputs and give the most-voted class as the output. This majority-voting action may be seen as only a default rule for combining the inputs. When this simple rule fails, adaptation takes place by adding a new rule which produces the correct classification result. The committee adaptation studied in this paper is based on the Dynamically Expanding Context (DEC) principle of Kohonen [1, 2]. In DEC, new symbol transformation rules are added when the existing rule set is unable to produce unambiguous mapping from the input symbols to an output symbol. When new rules are formed, the context in which the input symbols are examined is expanded and, therefore, the new rules always utilize more contextual information and are thus more specific than the original ones. In our system, the context upon which the rules operate is formed from the set of outputs from the committee members. The members are first ranked and used in the order of decreasing individual recognition accuracy. It is then possible to use only the first-ranking outputs of all members or also the second-ranking ones.

In our experiments, the initial classifier has been user-independent. The addition of user-specific DEC rules makes the system user-dependent in the course of the adaptation. The results of our experiments show that for a set of static member classifiers, the adaptive committee is able to decrease the original recognition error rate considerably. The speed of adaptation is also quite fast.

2. Dynamically Expanding Context

The principle of the Dynamically Expanding Context (DEC) was introduced by Kohonen in 1986 [1, 2] and further elaborated, e.g., by Torkkola [6]. It was originally developed for speech recognition in which it was used to correct the coarticulation effects between adjacent phonemes recognized as a series of quasiphoneme strings.

The application of DEC can be formulated as a set of context-sensitive production rules $x(A)y \rightarrow (B)$, where A and B are the input and output symbols, respectively, and x and y are the left and right contexts, respectively, of the input symbol. The combined length of the x and y contexts determine the level of the rule. Each time a rule is found to be in conflict with the actual transformation needed, a new higher-level rule is added. This new rule is, due to the increased amount of context involved, not conflictive.

The DEC principle has been somewhat modified for our current purposes in on-line recognition of handwritten characters. In our setting, there are a set of individual classifiers for recognition of handwritten characters. The classifiers have been first initialized and then ranked in the order of decreasing recognition performance. These classifiers are then used to form a committee classifier and the modified DEC principle is used to create the production rules for the committee. The outputs of the member classifiers as well as the second-ranking recognition results from each of them are used as a one-sided context when forming the DEC rules.

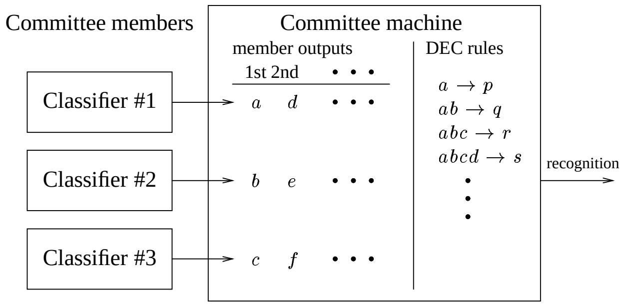

Figure 1. Block diagram of a DEC-based adaptive committee classifier.

Figure 1 displays a schematic diagram of the DEC-based adaptive committee classifier. In the illustration, there are three member classifiers. The first-rank outputs from the classifiers are denoted by a, b, and c. Likewise, the second-rank outputs are d, e, and f, respectively. The DEC rules can in this case be written symbolically as $A \to x$, where A is a string of member classifier outputs and x is the demanded recognition result.

Each time a new character has been input to the system, the string of the member classifiers' outputs is matched to the existing DEC rules. If no match is found, a default decision is applied. This default action can be, e.g., to use the first output of the best individual classifier. If one or more rules match the situation, the highest-level one, i.e., the one with the largest context, is applied and the output symbol specified by the rule is used. If the recognition result is then found to be incorrect, a new rule with more context is included in the rule base.

Table 1 illustrates this process. The columns of the table show the correct class of the input character, the first outputs of the member classifiers, the existing rules, the output of the committee, and the action taken. Let us assume that

Table 1. Generation of DEC rules in the case of hardly distinguishable letters 't' and 'f'.

round	input	members	rules	output	result	action
1.	t	t f t	—	t	ok	—
2.	f	t f t	—	t	err	add rule: t f - → f
3.	f	t f t	t f - → f	f	ok	–
4.	t	t f t	t f - → f	f	err	add rule: t f t → t
5.	t	t f t	t f - → f t f t → t	t	ok	—

1) the first output of the first classifier is the default decision, 2) no additional rules have been included yet, and 3) a series of hardly distinguishable letters 't' and 'f' are input to the system.

On the first round, a 't' is input and because there are no additional rules, the output of the first member is used as the committee's decision. This happens to be correct and no further actions are taken. On the second round, the member outputs are the same but the correct output is now 'f' instead of 't'. Due to the error, a rule "t f - → f" is inserted. Its application is demonstrated on the third round where the classification is again correct in a similar situation. On the next two rounds, a 't' is input and recognized first incorrectly and then correctly after the rule "t f t → f" has been included. This process is continued as more characters are input to the system. In the case of off-line training, the training set could be reiterated but, in our experiments, we have assumed that the system is in on-line use where the previous input characters cannot be preserved.

In the above example, only the first outputs from the member classifiers were displayed. In our present experiments, the second best guesses from all classifiers are also allowed to be taken in the context. Referring to Figure 1, this can be implemented either horizontally, i.e., by consuming first both recognitions of the best-ranking classifier and obtaining the symbol sequence "$a\ d\ b\ e\ c\ f$"; or vertically, when the resulting sequence with three member classifiers is "$a\ b\ c\ d\ e\ f$".

As new DEC rules are being added, all the available context information will eventually be used by the rules. All error situations thereafter call for additional rules but the context cannot be expanded anymore. Therefore, it is allowed that there exist more than one highest-level rule for a single context. In this case, the number of correct applications of the rule is maintained for each and the one with the highest value is realized.

We experimented with some variations in the settings of the committee classifier. These included: 1) The default rule for cases when there were no applicable rules in the rule set was either (a) to obey the opinion of the best individual classifier, or (b) to perform majority voting among the members. 2) It could be demanded that in every DEC rule $A \rightarrow x$, x should be included in A. This means that at least one of the symbols in the context has to be correct when a new transformation rule is created. In the opposite case, this constraint was not enforced. 3a) Either only the first-ranking outputs from the committee classifiers were used, or also the second-ranking results were utilized, either (b) vertically or (c) horizontally. 4) The size of the context could be fixed. This means that when the number of committee members was four and the context size was fixed to four, the intermediate level two and three rules were not generated at all. Instead, the level four rules were used right from the first error. Context sizes smaller than four were also allowed, which reduced the actual number of committee members.

3. Experiments

Two data sets were used in the experiments. The first set was contributed by 21 writers and contained approximately 8400 lower case letters and digits written in isolation. These characters were used to form the user-independent member classifiers of the committee. There were four classifiers, each based on point-wise elastic matching [5] between the input character and a set of 273 stored prototypes. The prototypes were selected with a semiautomatic clustering algorithm [4].

The difference between the four classifiers was in the normalization of the characters. In two of them, the characters were centered by their bounding boxes and in the other two, by their mass centers. On the other hand, in two classifiers the size of the characters was normalized and in the other two it was not. Thus, every classifier shared some normalization properties with two others. It can be argued that each of the described classifiers can be expected to perform better than the others in some particular classification tasks. Therefore, it can be assumed that a committee formed from the classifiers would outperform the best single classifier. The average recognition error rates for the four classifiers are shown in Table 2. The accuracies of the first three classifiers are quite close to each other, whereas the last one is clearly worse.

The second data set was collected from 16 writers not included in the first set. Each subject wrote either 500 or 1000 characters and when the writing was started, the system was initialized so that there were no rules in the DEC rule base. During adaptation, every misclassification was recorded and DEC rules were generated. The overall performance of the committee classifier for every writer was evaluated as the total percentage of misclassified characters among all characters. This measure takes into account to some extent both the speed of adaptation and the final level of recognition accuracy. The decisive figure of merit for each committee classifier was calculated as the average of the total error percentages of the 16 writers.

Table 2. Recognition error rates of the four committee member classifiers. The data set contained lower case letters and digits.

bounding box	mass center	size scaling	errors
•		•	15.3%
	•	•	16.0%
	•		16.1%
•			18.9%

The results of the experiments are shown in Table 3. The first column (**def**) displays the default decision when no explicit rules were available, either the best individual classifier (b) or majority voting (v). Bullets in the second column (**inc**) indicate when the output symbol x was demanded to be included in the context A. Vertical (v) and horizontal (h) consumption of the second-ranking outputs from the member classifiers are shown in the third column (**2nd**). A dash is draw when the second opinions were not used at all. For every combination of the parameters in the first three columns of Table 3, we used both contexts which were allowed to expand to any size and fixed-sized contexts of sizes from one to eight. The best results of these experiments are recorded in the last three columns. The **size** column shows the context size that yielded the best accuracy, "exp" standing for expanding context size. The next column (**err**) displays the resulting recognition error rate. The rightmost column (**tail**) shows the average error rate after 300 input characters measured for the next 200 characters.

The last line in Table 3 displays the performance of a reference recognizer which was formed from the same member classifiers but without using the DEC rules. Instead, a count of the number of correct recognitions was maintained for each member classifier. Every input character was then classified according to the opinion of that classifier which at that particular moment had the highest success count. This lead to a sort of adaptive selection of the best single classifier for each individual test subject. The number of member classifiers in the reference recognizer was varied from one to four, and the classifiers were used in the order of increasing error rate.

The results show that majority voting as the default decision seems to outperform the use of the best individual classifier. This may result from the fact that the three best committee members were quite similar to each other in recognition accuracy. Second, the requirement for the output sym-

Table 3. Classification error rates for the DEC-based adaptive committee when the members were non-adaptive.

def	inc	2nd	size	err	tail
b		–	exp	10.7%	8.4%
b		v	exp	10.0%	8.5%
b		h	exp	10.4%	8.3%
b	•	–	exp	10.9%	9.1%
b	•	v	exp	11.4%	9.6%
b	•	h	exp	10.6%	8.8%
v		–	exp	9.7%	7.8%
v		v	exp	9.7%	7.9%
v		h	exp	10.1%	7.7%
v	•	–	exp	10.0%	8.5%
v	•	v	exp	11.1%	7.9%
v	•	h	exp	10.2%	8.3%
reference			3	14.0%	12.5%

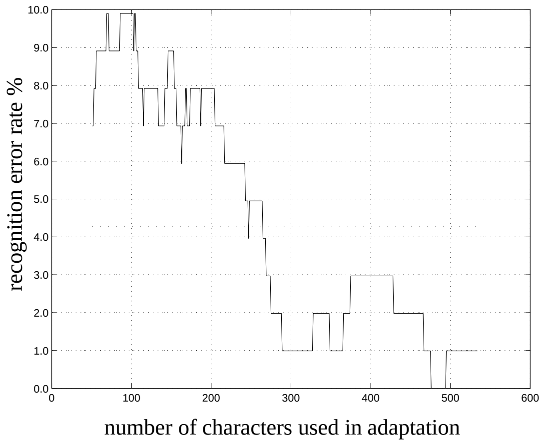

Figure 2. The evolution of recognition error rate during the creation of the DEC rules for one writer.

bol to be included in the context seems to be disadvantageous. Also, it seems that the use of the second-ranked outputs from the member classifiers does increase the recognition accuracy. However, it is not clear whether the vertical ordering is better than horizontal, as the total error percentage seems to somewhat favor the former while the tail error rate favors the latter. Finally, it is evident that the expanding formation of context rules outperforms the use of fixed-sized contexts.

When the DEC results are compared with the accuracies of the individual classifiers in Table 2 and with the reference classifier in Table 3 , it is clear that the DEC approach is beneficial. Figure 2 illustrates the evolution of the recognition error rate within a sliding window of 100 characters for one writer during the adaptation. The average error rate for the writer was 4.3% but, as can be seen in the figure, the initial error level is about 9% whereas the final level less than 1%. The final level of accuracy can be seen to be achieved after approximately 300 input characters.

4. Conclusions and Future Directions

We have described a handwriting recognition system in which a set of static classifiers is organized in a committee with adaptive decision rules. The results of the performed experiments show that this approach produces recognition results comparable with our earlier studies with adaptive classifiers [3, 4, 7]. The principle of DEC, the Dynamically Expanding Context, was thus proven to be suited for the implementation of adaptive symbol transformation rules also in the case of handwriting recognition.

We are currently experimenting with a set of adaptive classifiers as the members of the adaptive committee. This doubly-adaptive approach seems to offer even better recognition accuracy and faster adaptation speed than the previous methods.

References

[1] T. Kohonen. Dynamically Expanding Context, with application to the correction of symbol strings in the recognition of continuous speech. In *Proceedings 8th International Conference on Pattern Recognition (8th ICPR)*, pages 1148–1151, Paris, France, Oct. 27–31 1986.

[2] T. Kohonen. Dynamically expanding context. *Journal of Intelligent Systems*, 1(1):79–95, 1987.

[3] J. Laaksonen, J. Hurri, E. Oja, and J. Kangas. Comparison of adaptive strategies for on-line character recognition. In *Proceedings of the International Conference on Neural Networks*, volume 1, pages 245–250, Skvde, Sweden, Sept. 1998.

[4] J. Laaksonen, V. Vuori, E. Oja, and J. Kangas. Adaptation of prototype sets in on-line recognition of isolated handwritten Latin characters. In S.-W. Lee, editor, *Advances in Handwriting Recognition*, pages 489–497. World Scientific Publishing, 1999.

[5] C. C. Tappert. Speed, accuracy, and flexibility trade-offs in on-line character recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 5(1&2):79–95, 1991.

[6] K. Torkkola. *Short-Time Feature Vector Based Phonemic Speech Recognition with the Aid of Local Context*. PhD thesis, Helsinki University of Technology, Espoo, Finland, 1991.

[7] V. Vuori. Adaptation in on-line recognition of handwriting. Master's thesis, Helsinki University of Technology, 1999.