

Adaptive Pen User Interface with Supervised Competitive Learning

Takayuki Dan Kimura

Laboratory for Pen-Based Silicon Paper Technology (PenLab)
Department of Computer Science
Washington University in St. Louis
St. Louis, MO 63105, USA
<http://www.cs.wustl.edu/~tdk/>

Abstract

An Adaptive user interface for a pen-based educational application and the related technical difficulties are discussed. The Supervised Competitive Learning (SCL) model is proposed as a mechanism for a pen-based adaptive user interface. It offers a solution for one of the fundamental problems of using neural for adaptive user interface; the Grossberg's stability-plasticity dilemma.

1. Introduction

By *adaptive user interface* we mean a human-computer interaction mechanism that can learn. Novice end users expect computers to be adaptive and be able to learn as humans do. When a kindergarten child learning the alphabet scribes a character on a pen computer, the child assumes that the computer can recognize it correctly as much as the teacher can do, once the computer is told what character the handwriting represents. When a computer fails to learn as humans, it is no longer a friendly tool.

There is another, more subtle need for adaptive user interface. When a child is learning the alphabet or arithmetic facts using a pen computer, the handwriting tends to become larger and sloppier when the child gets frustrated by the difficulty of the assignment or for other various reasons. When that happens, it is pedagogically imperative for the computer to reduce the difficulty level of the next assignment. If the pen computer fails to detect the change of handwriting and is unable to adapt to the children's frustration, it falls short of being an effective education tool.

Adaptive user interfaces are not limited to pen computers. Programming by demonstration is an example of adaptive graphical user interface applied to the scripting level of interaction. In this paper, however, we will focus on the handwriting recognition problem specific to an educational application. The problem of handwriting recognition in real time, transcribing continuous pen scribbles to discrete category names, is not new. But satisfactory solutions are yet to be found. One of the difficulties lies in the fact that people write differently under different circumstances. Any practical solution for the recognition problem must accommodate mechanisms of adaptation in two levels; one is to learn user's consistent handwriting in a stable environment, and the other is to learn to adapt to changing handwriting in different environments, without losing what was already learned.

Recognition of young children's handwriting exemplifies the problem, where the target of adaptation (learning) is transitory. In this situation both the computer and the child are adaptive. The computer tries to learn how the child writes, while the child is learning how to write and the child improves their writing continuously. When the computer tells the child that the handwriting is illegible, the child tries to write neater and changes the way of handwriting. The computer, in turn, tries to adapt to the newly improved handwriting of the child, even before the previous handwriting is learned

Detection of sloppiness also requires more than just straight handwriting recognition. It requires to learn the child's normal handwriting first and then tolerable deviations from the norm. When the child writes a sloppy character '4' which looks like '6', the computer must discriminate aberrations caused by sloppiness from normal writings of '6'.

In the above both cases, either for young children or for detection of sloppiness, the computer has to learn not only children's normal handwriting but also adaptation to changing environments.

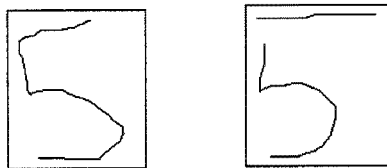
In this paper we present a learning algorithm, called *supervised competitive learning* (SCL), that can adapt to changing environments. It is incorporated in a pen-based computer system developed for schoolchildren[1]. The previous version of SCL was implemented in C [2]. An enhanced version of SCL is currently being implemented in Java for Windows 95 running on the Mitsubishi Amity-SP pen computer.

In the next section we define SCL. Section 3 discusses one possible way of measuring the handwriting sloppiness using a slightly modified SCL.

2. Supervised Competitive Learning Model

Neural networks and hidden Markov models are common techniques used for solving handwriting recognition problems[3]. Due to its heavy computational demand, hidden Markov models are not suitable for real time adaptive user interfaces where repeated reconstruction of new models are required. Neural networks such as back-propagation nets, on the other hand, suffer from the stability-plasticity dilemma formulated by Grossberg [4]: How can a learning system remain plastic in response to a new environment and yet remain stable in response to previous environments? How can it maintain previous knowledge while continuing to gain new?

Let us assume that a back-propagation net is already trained successfully to recognize a child's single stroke numeral '5' such as the one shown in Figure 1(a). When the child learns a two-stroke '5' such as shown in Figure 1(b) and the net is retrained to recognize it, can the net recognize a single stroke '5' as well as it used to? The answer is most probably no because, due to the fundamental nature of global distributed con-



(a) Single Stroke '5' (b) Two Strokes '5'

Figure 1: Sample Handwriting

nectivity in the neural network architecture, there is no innate mechanism to isolate the impact of retraining from the previously established recognition capabilities.

Carpenter and Grossberg [5][6] proposed *adaptive resonance theory* (ART) as a possible solution for the stability-plasticity dilemma in the unsupervised competitive learning paradigm. It assumes no teaching input and performs unsupervised learning. It organizes itself to group "similar" input patterns into the same category. Category proliferation is controlled by the vigilance parameter. An ART system with low vigilance will permit grouping of only grossly similar patterns, and a system with high vigilance will try to form separate categories for patterns that have only minor differences. The ART model consists of two sets of processing nodes: the *attention* subsystem and the *orienting* subsystem. The nodes in the attention subsystem compete with each other when activated by an input pattern. The winning node represents the learned category of the input pattern and also carries the prototype (attention) pattern associated with the category. The orienting subsystem compares the prototype with the input, and if the two are significantly different, it resets (disables) the winning node for a new round of

competition, with the assumption that the input pattern does not belong to a category represented by the current winner. If all prototype patterns in the attention subsystem are sufficiently different from the input, the input pattern itself becomes the prototype of a new node representing a new category. The degree of similarity is controlled by the vigilance parameter.

Supervised Competitive Learning (SCL) is a scheme of compounding a set of learning modules into a supervised learning system that associates each input pattern with its category name. Each category may consist of several subcategories (prototypes). For example, a set of single stroke numeral of '5' entered by the same child may form a subcategory of the category of '5'. Similarly a set of two stroke numeral of '5's form another subcategory. We use each learning module as a similarity detector for one prototype and adopt a prototype resetting mechanism to create new prototypes. It uses the ART's resetting mechanism in the context of supervised learning. Any learning can be used for component modules as long as it is capable of both positive (encouraging) and negative (discouraging) training; back-propagation nets and pattern classification models based on fuzzy logic are two natural candidates.

The schematic definition of SCL is given in Figure 2.

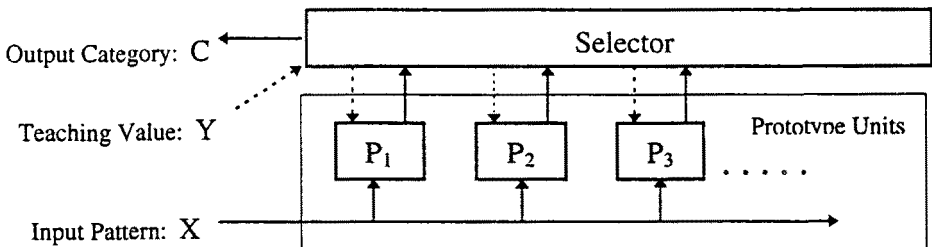


Figure 2: Schematic Definition of Supervised Competitive Learning (SCL)

SCL receives the input pattern X and outputs the category name C . If the system fails to produce the correct category name, then the correct name is given to the system as the teaching value Y . The system learns the association between X and Y so that it may respond correctly to the input X next time.

An SCL system consists of a set of prototype units and a Selector. Each prototype unit is responsible for identifying all the input patterns that belong to a particular subcategory (prototype) and the unit carries the category name to which the subcategory belongs. When the input pattern X is presented, each prototype unit, P_i , produces the certainty of X belonging to the subcategory of the unit as its output value. The Selector selects, as the winning unit, the unit whose output is larger than a threshold (*vigilance*) value, p , and the largest among those above the threshold. Then it produces the name of the category, to which the winning unit belongs, as the output of the SCL system. If the output is correct, then nothing changes. If there is no winner, then the input X is considered to be a member of a new subcategory, and a new prototype unit is created with the teaching value Y as its category name. If the winning unit is wrongly selected, the teaching value, Y , is used to train the prototype units as follows: The units of the category Y that respond to X with low outputs will be trained to increase their outputs for X up to a *high-confidence value*, σ_H . Those units representing categories other than Y that respond to X with high outputs will be trained to decrease their outputs down to a *low-confidence value*, σ_L . Losing units with category name Y are not trained. We assume that each unit is trainable to produce high output values for members of its subcategory and to produce low output values for non-members.

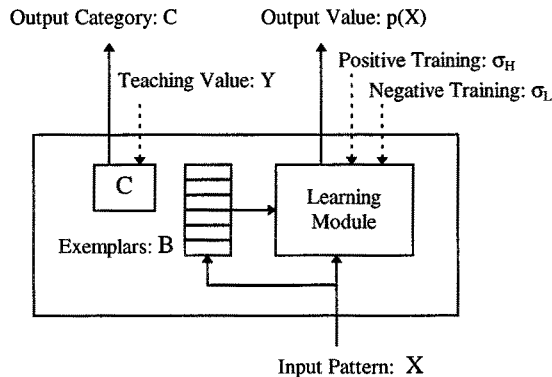


Figure 3: Prototype Unit

Figure 3 shows the structure of the prototype unit. It maintains, in the buffer B , a collection of exemplar patterns of the subcategory of the unit. The learning module, typically a neural net, is capable of learning to produce a high value if the input pattern is similar to the exemplar patterns, and a low value otherwise. The learning module is responsible for computing the output value, $p(X)$, of the prototype unit for the input X .

If the output is wrongly too low for the input X , the learning model re-trains itself positively so that it can produce the output value higher than σ_H , for the input pattern as well as for all the exemplar patterns. Similarly, if the output is wrongly too high, it re-trains itself both negatively for the input pattern until the output value becomes lower

than σ_L , and positively for the exemplar patterns so that it may keep the correct responses to the exemplar patterns.

The algorithm for the Selector is given in Figure 4 (next page). We assume that the output is normalized to $[-1, 1]$ and that $-1 < \sigma_L < \rho < \sigma_H < 1$.

3. Detection of Sloppy Handwriting

A particular handwritten character is *sloppy* if it deviates from the normal handwriting enough to be noticeable, but not enough to form a new prototype. The sloppiness can be measured by the size of the handwriting (the area of the bounding box) and by the deviation from the 'normal' handwriting of the same character. We propose to use SCL for measuring the deviation.

When an SCL system recognizes an input pattern X correctly, the output value of the winning prototype unit represents the similarity between the input X and the prototype patterns of the X 's subcategory. The higher the value is, the closer X is to the prototype patterns.

We propose to modify the Selector of the SCL model to compute the following value as the deviation (dissimilarity) measure:

$$d = (1 - p(X)) / (p(X) - \rho)$$

where $p(X)$ is the output value of the winning prototype unit for the input pattern X .

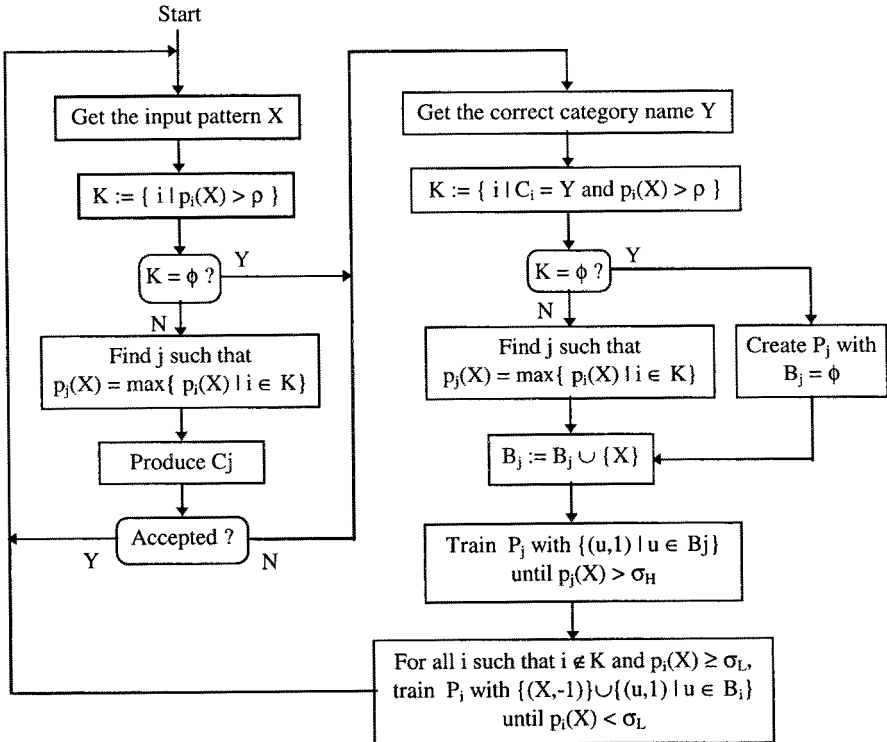


Figure 4: Algorithm for the Selector

Note that since it is always the case that $p \leq p(X) \leq 1$ for the winning unit, d is 0 (no sloppiness) when $p(X)$ is 1, and d is ∞ (very sloppy) when $p(X)$ is p , the smallest value allowed.

The change of the handwriting size can be measured by comparing the area of the bounding box of X and the average area of the bounding boxes of all the exemplar patterns owned by the winning prototype unit.

4. Conclusions

We expect to complete the current Java implementation of an SCL system in the near future. The system uses a single-layer Perceptron (back-propagation net without hidden units) as the prototype units. This is based on our experiences obtained in the previous implementation of SCL and its performance measurements [2].

The sloppiness measure proposed in Section 3 is a very preliminary idea which is to be refined through a sequence of SCL experiments.

5. References

- [1] Kimura, T.D. "Learning Math with Silicon Paper," Technical Report WUCS-92-11, Department of Computer Science, Washington University in St. Louis, 1992.
- [2] Fuller, Jr., T.H. and T.D. Kimura. "Supervised Competitive Learning," *Journal of Intelligent Material Systems and Structures and Mathematical and Computer Modeling*, 5:2 (1994), pp. 232 - 246.
- [3] Tappert, C.C., C.Y. Suen and T. Wakahara. "The State of the Art in On-Line Handwriting Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:8 (1990), pp. 787 - 808.
- [4] Grossberg, S. "Competitive Learning: From Interactive Activation to Adaptive Resonance," *Cognitive Science* 11(1986), pp. 23 - 63.
- [5] Carpenter, G. A. and S. Grossberg. "ART2: Self-Organizing of Stable Category Recognition Codes for Analog Input Patterns," *Applied Optics* (1987) pp. 4919 - 4930.
- [6] Carpenter, G. A. and S. Grossberg. "The ART of Adaptive Pattern Recognition by a Self-Organizing Neural Network," *IEEE Computer* 21:3 (1988), pp. 77 - 88.