

論文94-31B-9-14

동적으로 출력 뉴런을 생성하는 경쟁 학습 신경회로망

(Competitive Learning Neural Network with Dynamic Output Neuron Generation)

金鍾完*, 安濟晟**, 金宗相*, 李興浩***, 趙成元****

(Jong Wan Kim, Je Sung Ahn, Chong Sang Kim, Heung Ho Lee and Seong Won Cho)

要約

기존의 경쟁 학습 알고리즘들은 미리 정해진 모든 출력 뉴런들에 대하여 승자 뉴런 선정을 위하여 유클리디안 거리를 계산하였다. 이 경우에는 초기 대표(가중치) 벡터들에 의하여 학습 알고리즘의 성능이 영향을 받는다는 단점이 있었다. 본 논문에서는 동적으로 출력 뉴런들을 생성하는 새로운 경쟁 학습 알고리즘을 제안한다. 제안된 방법은 생성되는 출력 뉴런마다 클래스 임계치란 패러미터를 사용하여 이 값을 동적으로 변경시키면서 출력 뉴런을 생성한다. 즉 입력 벡터와 생성된 출력 뉴런들의 대표벡터들과의 유사성을 검사하여, 이 값이 큰 경우에는 유사한 대표벡터를 학습시키고, 작은 경우에는 입력 벡터값을 새로운 출력 뉴런의 대표벡터로 삼는다. 제안된 방법은 입력 패턴들의 분포에 따라 출력 뉴런의 대표벡터를 동적으로 지정하므로, 불필요한 출력 뉴런 생성에 따라서 학습이 조기에 결정되는 현상을 해결한다. 음성 데이터에 적용하여 실험한 결과 제안한 방법은 기존의 방법들에 비하여 우수한 결과를 보였다.

Abstract

Conventional competitive learning algorithms compute the Eucliden distance to determine the winner neuron out of all predetermined output neurons. In such cases, there is a drawback that the performance of the learning algorithm depends on the initial reference(= weight) vectors. In this paper, we propose a new competitive learning algorithm that dynamically generates output neurons. The proposed method generates output neurons by dynamically changing the class thresholds for all output neurons. We compute the similarity between the input vector and the reference vector of each output neuron generated. If the two are similar, the reference vector is adjusted to make it still more like the input vector. Otherwise, the input vector is designated as the reference vector of a new output neuron. Since the reference vectors of output neurons are dynamically assigned according to input pattern distribution, the proposed method gets around the phenomenon that learning is early determined due to redundant output neurons. Experiments using speech data have shown the proposed method to be superior to existing methods.

*正會員, 서울大學校 컴퓨터工學科
(Dept. of Computer Eng., Seoul Nat'l Univ.)

**正會員, 金星社 中央研究所 知能情報1室
(Gold Star Co. Central Research Lab.)

***正會員, 忠南大學校 電氣工學科

(Dept. of Computer Eng., Chungnam Nat'l Univ.)

****正會員, 弘益大學校 電氣·制御工學科
(Dept. of Electrical and Control Eng.,
Hong Ik Univ.)

接受日字: 1993年 10月 21日

1. 서론

패턴 인식(Pattern Recognition)은 값싸고 성능이 향상된 워크스테이션(workstation)들이 대량으로 보급됨에 따라 문자 인식(character recognition), 음성 인식(speech recognition), 컴퓨터 시각(computer vision)등을 중심으로 연구가 활발하게 진행되고 있다. 패턴 인식 문제의 해결방안으로는 통계적 방식(statistical approach), 구조적 방식(structural approach), 신경회로망 방식(neural network approach)이 사용된다.^[1]

먼저 통계적 방식이란 입력 데이터로부터 정해진 특징(feature)을 추출하여 이들 특징 벡터들을 분류할 클래스로 지정하고, 판별 함수(decision function)를 사용하여 인식하는 방법이다. 이 방법은 주어진 패턴 집합으로부터 특징의 선정 및 추출의 어려움과 함께 많은 계산량을 필요로 한다. 구조적인 방식은 인식 대상의 패턴들을 각 패턴을 구성하는 기본 원소(primitive element)들간의 연결관계에 따른 생성 규칙(production rule)으로 표현하고, 이들의 집합으로 표현된 패턴을 정해진 문법(grammar)에 따라 인식하는 방법이다. 따라서 이 방법은 주로 문자나 도형등과 같이 시각적인 패턴 인식에 많이 이용된다. 하지만 기본 원소의 정의 및 추출이 쉽지 않으며 새로운 데이터에 대한 생성 규칙의 확장 능력이 제한적이다. 따라서 최근에는 주어진 패턴들을 학습 시킴으로써 특징의 선정, 기본 원소의 추출, 새로운 입력에 대한 일반화 능력 및 하드웨어 구현의 어려움 등의 문제점들을 해결하기 위해 신경회로망 방식이 많이 사용되고 있다.^[2]

신경회로망의 대표적인 학습 알고리즘들은 장단점을 가지고 있다. 가장 잘 알려진 오류 역전파(Error Backpropagation : EBP) 학습 규칙은 지도 학습(supervised learning) 모델로서 경쟁 학습 방식에 비하여 대체로 높은 클래스 인식률을 보이고 있다.^[3] 하지만 이 모델은 신경망의 실용화란 관점에서 인식률과 함께 중요한 요소인 학습 속도가 느리고, 적절한 학습 자료의 수집을 위한 시간이 필요하다는 문제점이 있다. 이에 비해 경쟁 학습(competitive learning)은 인식률은 EBP보다 대체로 낮지만 학습 규칙이 단순하므로 학습 속도가 훨씬 빠른 장점을 가지고 있다.^[4,6] 따라서 하드웨어 구현이 용이하여 실제 응용에서는 많이 사용되고 있다.^[7] 또한 패턴의 인위적인 구분이 모호하여 자율 학습이 필요하고 신속한 처리가 요구되는 문제에 대해서는 경쟁 학습이 유리한 것으로 알려져 있다.^[5,8] 특히 경쟁 학습은 빠

른 학습 속도때문에 화상(image) 데이터, 음성(speech) 데이터와 같이 많은 데이터를 다루는 패턴 인식 문제에 많이 사용되고 있다.^[5,8]

일반적으로 경쟁 학습 알고리즘들은 고정된 출력 뉴런들에 대하여 입력 벡터와 출력 뉴런의 가중치 벡터들간의 거리를 계산하여 승자 뉴런(winner neuron)의 가중치를 갱신하였다. 하지만 이 방법들은 안정된 학습을 수행하지만 초기 가중치 벡터들의 분포에 따라 성능 향상에 한계가 있었다.^[2,8] 이와 반해 Carpenter와 Grossberg는 통계적 방식에서 사용되는 leader clustering algorithm^[9]을 신경회로망 방식으로 변용한 ART(Adaptive Resonance Theory) 모델을 제안하였다.^[10] 이 모델은 패턴의 분포에 따라 출력 뉴런을 동적으로 늘려가면서 학습을 시킨다. 하지만 패턴 정합(pattern matching) 검사의 기준인 경계치(vigilance parameter)의 선정이 어렵고, 입력 패턴과 대표벡터(exemplar vector)의 내적(inner product) 연산에 의한 이전에 학습된 패턴의 감쇄(pattern attenuation) 현상이 문제가 된다. 한편 동적으로 출력 뉴런을 생성한 후 클러스터 병합(cluster merging)을 수행하는 2단계 자율 구성 신경회로망(2-Phase Self-Organizing Neural Network) 알고리즘이 최근에 제안되었다.^[11] 이 방법은 1회의 on-line 학습을 수행하므로 벡터 양자화(Vector Quantization : VQ)와 같은 클러스터링에 적합한 알고리즘이지만 학습의 안정성이 문제가 되었다.

이에 본 논문에서는 클래스내 분산을 최소로 줄이면서 동적으로 출력 뉴런을 생성하는 새로운 경쟁 학습 알고리즘을 제안한다. 제안한 방법은 먼저 생성되는 출력 뉴런마다 동일한 초기 클래스 임계치를 준다. 입력 패턴들이 생성된 출력 뉴런으로 분류됨에 따라 출력 뉴런의 클래스 임계치를 동적으로 변화시켜서, 입력 패턴을 학습을 위해 사용할 것인지 아니면 새로운 출력 뉴런의 대표벡터로 사용할 것인지를 검사한다. 제안한 신경회로망 모델은 기존의 경쟁 학습 방식들 중에서 구조면에서는 ART 모델과 같이 동적으로 출력 뉴런들을 정해진 최대 갯수까지 생성시키지만, 학습 방식면에서는 양방향 학습을 수행하는 ART와 달리 전방향 학습만을 수행하고, 학습 패러미터로서 클래스 임계치 1개만을 사용하므로 훨씬 단순하다. 또한 가중치 변경시에는 기존의 경쟁 학습을 따르지만 출력 뉴런마다 고유한 클래스 임계치를 사용한다는 점이 다르다.

본 논문의 II 장에서는 경쟁 학습 알고리즘을 소개하고, III 장에서는 제안된 동적 출력 뉴런 생성 알고

리즘을 기술하고, IV장에서는 실험 결과를 보인 후, V장에서 결론을 맺는다.

II. 경쟁 학습 알고리즘

경쟁 학습이란 입력 벡터와 가장 유사한 출력 뉴런을 자율적으로 학습시키는 방법이다. 경쟁 학습의 승자 뉴런 선정 기준과 대표적인 알고리즘들을 간략히 설명하겠다.

일반적으로 경쟁 학습에서는 입력 벡터와 출력 뉴런의 가중치 벡터간의 거리가 최소인 뉴런을 승자 뉴런으로 선정한다. 이를 수식으로 표현하면 식(1)과 같다.

$$d[X, W_i(t)] = \min_{1 \leq i \leq M} \{ d[X, W_i(t)] \}. \quad (1)$$

이때 X : 입력 패턴, $W_i(t)$: 출력 뉴런 i 의 가중치 벡터, M : 출력 뉴런들의 수.

초기에 경쟁 학습 알고리즘은 승자 뉴런만을 학습시키는 단순 경쟁 학습(Simple Competitive Learning : SCL), 일명 Winner-Take-All 학습을 수행하였다.^[2] 이를 수식으로 표현하면,

$$W_i(t+1) = W_i(t) + \alpha(t) [X - W_i(t)] \quad (2)$$

이때 i : 승자 뉴런, $\alpha(t)$: 학습률(learning rate)

하지만 이 알고리즘은 승자 뉴런만을 계속 학습시키므로 초기 가중치 벡터의 분포에 따라 학습이 많이 되는 뉴런들과 함께, 학습이 거의 안되는 뉴런이 생기는 학습 불균형 현상이 발생하였다. 이를 해결하기 위한 여러 시도가 있었다.^[5, 8, 12]

Kohonen은 승자 뉴런과 위상적으로 이웃한 뉴런들을 함께 학습시키는 자기 구성 특징 지도(Self-Organizing Feature Map : SOFM) 알고리즘을 제안하였다.^[5] 이 방식은 단순 경쟁 학습에 비하여 학습의 안정화를 이루지만 추가적인 계산이 필요하다. 즉 승자 뉴런의 이웃을 계산하고 이들의 가중치들을 함께 학습시키므로, 간혹 불필요한 계산으로 인하여 학습이 어려워지기도 한다. Kangas등은 SOFM을 변형하여 이웃 뉴런 개념을 기존의 위상적으로 인접한 뉴런 대신에, 승자 뉴런과의 벡터 거리의 상대적인 크기에 따라 이웃 뉴런을 정의하였다.^[13] 이들은 이러한 동적 이웃 개념을 최소 신장 트리(Minimal Spanning Tree : MST)^[14]를 사용하여 정의하였다. 이렇게 하면 특징 지도의 최종 상태가 학습 데이터의 분포를 더 잘 보여준다. 이때 최소 신

장 트리란 모든 노드들을 단일 링크(link)로 연결하면서, 링크들간의 거리 합이 최소가 되도록 하는 알고리즘이다.

DeSieno^[12]와 Ahalt등^[8]은 빈번하게 승자가 되는 출력 뉴런들에 벌을 줌으로써 승리 정도(winning rate)를 낮추어 모든 출력 뉴런들이 고르게 학습되도록 하는 conscience(공평) 개념을 도입했다. Ahalt등은 이를 FSCL(Frequency-Sensitive Competitive Learning)이라 명명했다.^[8] 이 방식은 벡터 양자화에서는 좋은 결과를 얻는 것으로 알려져 있으나 승리 정도의 조정이 데이터에 따라 일정하지 않다는 문제를 가지고 있다.

또한 Grossberg등이 제안한 ART 모델은 출력 뉴런의 동적 생성을 검사하기 위한 기준으로서 식 (3)과 같은 정합 검사를 수행하였다.^[10]

$$\frac{\|T \cdot X\|}{\|X\|} \geq \rho \quad (3)$$

X 이때 X : 입력 벡터, T : 대표 벡터, ρ 경계치 유사성 검사(similarity test)는 입력 벡터에 대한 대표 벡터(exemplar vector)와 입력 벡터와의 곱의 놈(norm)의 비율로서 정의된다. 이 유사성이 경계치(vigilance parameter)보다 크거나 같으면 정합된 것으로 판정하여 가중치 벡터를 갱신하고, 작으면 새로운 출력 뉴런을 생성하여 입력 벡터를 새로운 대표 벡터로 삼는다. 하지만 경계치의 선정이 쉽지 않고, 이진 패턴이 아닌 아날로그 패턴의 경우에는 훨씬 더 많은 패러미터를 가지게 되므로 적절한 학습을 수행하기가 더 힘들어진다.

III. 동적으로 출력 뉴런을 생성하는 경쟁 학습 알고리즘

1. 제안된 신경회로망

제안된 신경회로망은 크기가 고정된 입력층(input layer)과 정해진 최대 갯수에 도달할 때까지 가변적으로 증가하는 출력층(output layer)을 가지는 2층(two-layer) 구조로 이루어진다. 출력층 뉴런들은 각각 카운터(F_j) 및 분산(S_j), 그리고 클래스 임계치(σ_j)를 가지고 있다. 제안된 신경회로망의 구조는 그림 1과 같다.

제안된 신경회로망 모델은 전방향으로만(feed forward) 가중치를 변경하고, 학습자가 없이 입력 패턴을 출력 클래스로 분류하는 비지도 학습(unsupervised learning)을 수행한다. N차원 공간 상에서 주어진 패턴들을 군집화(clustering)하면서

학습이 이루어지며, 기존의 경쟁 학습들과 마찬가지로 반복 학습을 수행함으로써 안정된 학습을 이룬다. 제안된 모델과 기존의 모델과의 가장 큰 차이는 동적으로 출력 뉴런을 생성한다는 점이다. 이를 위해 본 논문에서는 입력 패턴이 학습된 출력 뉴런의 클래스 임계치안에 들어오면 가중치를 변경함으로써 학습을 시키고, 그렇지 않으면 입력 패턴을 새로운 출력 뉴런으로 생성한다. 즉 생성된 출력 뉴런의 가중치 벡터로 입력 패턴을 사용하는 것이다. 이렇게 하면 유사한 입력 패턴들이 서로 다른 출력 뉴런으로 중복 학습되는 현상을 막고, 출력 뉴런마다 고유한 클래스 임계치를 가지게 되어 학습 패턴을 수용할 수 있는 능력이 출력 뉴런마다 다르게 되므로, 임의의 분포를 가지는 패턴 인식에 적합하게 된다.

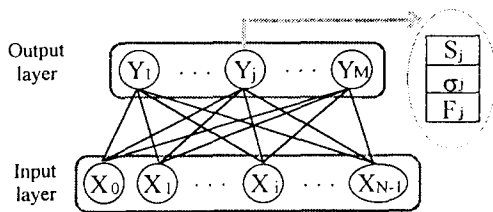


그림 1. 제안된 신경회로망의 구조
Fig. 1. The architecture of the proposed neural network.

2. 학습 알고리즘

제안된 신경회로망의 학습 규칙은 기존의 경쟁 학습 알고리즘에 출력 뉴런 생성 검사가 추가된다. 학습 규칙은 클래스내 분산을 줄이는 방향으로 동적으로 출력 뉴런들을 증가시켜 가면서 출력층 뉴런들을 자율 구성한다. 자세한 학습 과정은 다음과 같다.

단계 1. 신경회로망 변수들을 초기화한다.

$$w_{ij}(0) = 0, S_j(0) = r_0^2, \sigma_j(0) = r_0^2, F_j = 0, \alpha_0 \text{를 초기화} \quad (4)$$

단. $0 \leq i \leq N-1, 1 \leq j \leq M, 0 < r_0 < 1, 0 < \alpha_0 < 1$

이때 w_{ij} : i번째 입력 벡터 성분과 j번째 출력 뉴런 간의 가중치 벡터 원소.

S_j : 출력 뉴런 j의 분산.

σ_j : 출력 뉴런 j의 클래스 임계치.

F_j : 출력 뉴런 j의 카운터값.

r_0 : 초기 클래스 임계치.

α_0 : 학습률 초기값

단계 2. 학습 패턴 $x(t)$ 를 입력한다.

단계 3. 모든 출력층 뉴런들과의 거리를 계산한다.

$$d_j = \sum_{i=0}^{N-1} [x_i(t) - w_{ij}(t)]^2, 1 \leq j \leq m \quad (5)$$

이때 m : 현재까지 생성된 출력 뉴런들의 수.

$$1 \leq m \leq M$$

M : 최대 출력 뉴런들의 수

단계 4. 최소 거리를 가지는 뉴런을 후보 뉴런으로 선택한다.

$$j^* = \min_j \{d_j\} \quad (6)$$

만일 최소 거리를 가지는 뉴런이 2개 이상일 경우에는 순차 리스트 L에 저장한 후, 번호순으로 다음 단계의 소속 기준 검사를 수행한다.

단계 5. 소속 기준 검사(Acceptance test)

입력 패턴과 후보 뉴런과의 분산 $S_j^*(t)$ 를 계산하여 전상태 분산과의 변화량 ΔS_j^* 를 계산한다. 또한 분산 변화량 ΔS_j^* 를 사용하여 클래스 임계치 $\sigma_j^*(t)$ 를 갱신한다.

$$S_j^*(t) = \sum_{i=0}^{N-1} [x_i(t) - w_{ij}^*(t)]^2 \quad (7)$$

$$\Delta S_j^* = S_j^*(t) - S_j^*(t-1) \quad (8)$$

$$\sigma_j^*(t) = \sigma_j^*(t-1) + \frac{\Delta S_j^*}{F_j^*} \quad (9)$$

분산 변화량과 갱신된 클래스 임계치의 크기를 비교하여 입력 패턴이 선택된 후보 뉴런에 속할 것인지를 검사한다. 또한 최대 개수의 출력 뉴런이 이미 생성되었으면 가중치 변경을 수행한다.

If $\Delta S_j^* \leq \sigma_j^*(t)$ or $m = M$, then 단계 7로 분기 (10)
else 단계 6으로 분기

단계 6. 후보 뉴런 리스트를 검사한다.

후보 뉴런 리스트 L을 검사하여 원소가 남아있는 경우에는 단계 5로 분기하여 소속 기준 검사를 반복한다. 만일 후보 뉴런이 없다면 입력 패턴을 새로운 출력 뉴런으로 생성하여 클래스내 분산과 클래스 임계치를 초기화하고, 단계 8로 분기한다.

If $L \neq \{ \}$, then 단계 5로 분기 (11)

$$j = j + 1 \quad (12)$$

$$w_{ij}(t) = x_i(t), \text{ for } 0 \leq i \leq N-1. \quad (13)$$

$$S_j(t) = r_0^2, \sigma_j(t) = r_0^2, F_j = 1$$

단계 8로 분기한다.

단계 7. 가중치를 변경한다.

$$F_j^* = F_j^* + 1 \quad (14)$$

$$\Delta w_{ij}^* = \alpha(t) \cdot [x_j(t) - w_{ij}^*(t)] \quad (15)$$

이때 $\alpha(t) = \alpha_0 \cdot \frac{1}{t}$, t: learning step.

단계 8. 단계 2로 분기하여 반복 수행한다.

제안된 방법은 비지도 패턴 인식 알고리즘이므로 학습된 뉴런들이 어느 클래스를 대표하는 지 학습 과정만으로는 알 수가 없다. 따라서 신경회로망의 학습이 종료되면 학습 패턴들을 재입력시켜서 각 출력 뉴런이 어느 클래스를 대표하는 지를 알아내어 실제 인식률을 계산한다.

IV. 실험 결과

제안된 방법을 음성인식에 적용하여 실험하였다. 음성인식은 일반적인 패턴인식과는 달리 시간적 변화를 고려해야 하므로 더욱 어렵다고 할 수 있다. 이 실험에서는 응용분야를 음성인식의 가장 단순한 형태인 음소(단모음)인식에만 한정하여, 시간적 변화를 어느 정도 무시할수 있도록 하였다. 본 논문에서는 동적 뉴런 생성 알고리즘을 SCL, SOFM, FSCL등에 각각 적용하여 고정된 출력 뉴런을 가지는 표준의 SCL, SOFM, FSCL등과 비교하였다.

1. 실험 데이터

실험 데이터는 한국어 음소들 중에서 8개의 단모음을 사용하였으며, 기계잡음이 포함된 실험실 환경에서 서로 다른 3명의 남성화자로부터 수집되었다. 실험에 사용된 단모음들과 해당 음성기호¹⁵⁾는 표 1과 같다.

표 1. 실험에 사용된 한국어 단모음

Table 1. Korean monophthongs used in the experiment.

한글 표기	ㅏ	ㅓ	ㅗ	ㅜ	ㅡ	ㅣ
국제 음성 기호(IPA)	a	e	o	u	ɨ	i

음성 데이터는 증폭 및 5kHz 저역(Low-pass) filtering을 거쳐서 10kHz로 sampling되었다. 입력된 음성은 pre-emphasis와 hamming window를 거친 후, DFT(Discrete Fourier Transform)를

통하여 주파수 대역별 에너지가 구해지고 이 값의 대수 에너지(log energy)에 IDFT(Inverse Discrete Fourier Transform)를 적용하여 12개의 cepstrum 계수를 계산한다.¹⁶⁾ 그림 2는 실험 데이터의 생성을 위한 음성 데이터의 전처리 과정을 보여준다.

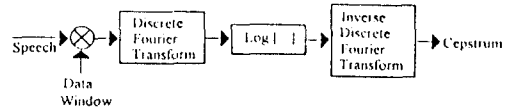


그림 2. 음성 데이터의 전처리 과정

Fig. 2. Preprocessing of speech data.

신경회로망으로의 입력벡터는 12개의 cepstrum 계수와 전체 에너지를 포함하여 13차의 벡터가 된다. Cepstrum은 유성음의 기본주기를 측정하는데 뛰어난 역할을 하며¹⁷⁾, 순환 신경망(Recurrent Neural Network)을 이용한 음성인식에 가장 적합한 음성특징임이 실험적으로 밝혀지기도 했다.¹⁸⁾ 입력벡터는 단위길이를 정규화를 거쳐서 각 원소의 값은 0에서 1.0까지의 범위를 갖게 된다.

신경회로망들을 학습시키고 테스트하기 위하여 실험 데이터는 각 화자별 음소별로 200개 및 400개씩의 서로 다른 패턴으로 이루어져 있다. 따라서 총 학습 패턴 및 테스트 패턴의 갯수는 각각 4800개와 9600개이다.

2. 시뮬레이션

신경회로망의 입력으로는 13차원 실수값을 사용하므로 입력층의 뉴런수는 13개로 고정된다. 인식하고자 하는 대상이 8개 단모음이므로 신경회로망의 출력층은 클래스 수의 10배인 최대 80개의 출력 뉴런을 사용한다. 또한 신경회로망의 가중치 벡터들의 초기 값은 학습 패턴들의 샘플들로부터 계산된다. 따라서 표준 SCL, SOFM, FSCL등은 출력 뉴런을 80개로 고정하고, 각 클래스별로 처음 10개씩의 패턴이 초기 가중치 벡터로 사용되었다. 하지만 동적 뉴런 생성 알고리즘의 경우에는, 첫번째 학습 패턴만을 첫번째 출력 뉴런의 초기 가중치 벡터로 제공하고, 학습을 진행하면서 최대 80개까지 출력 뉴런을 생성하면서 가중치 벡터들을 초기화한다. 따라서 제안된 방법은 기존의 방법과 달리 출력 뉴런의 가중치 벡터를 학습시키면서 새로운 출력 뉴런의 초기화를 수행함으로써 학습 패턴의 특성을 동적으로 수용할 수 있다.

확률 추정(stochastic approximation)론에서, 학습률 $\alpha(t)$ 는 혼돈 상태는 억제하면서 성능 척도로 사

용되는 평균 제곱 오차(Mean Square Error : MSE)의 국부 최소값을 보장하기 위하여 시간에 따라 감소해야 한다고 말한다.^[19]

$$MSE = \frac{1}{N} \sum_{i=1}^N [X_i - E(X_i)]^2 \quad (16)$$

이때 X_i : 학습 벡터,

$E(X_i)$: X_i 의 기대값 또는 평균 벡터,

N : 학습패턴들의 총수

따라서 학습률은 천천히 감소해야 하지만 너무 느려서는 안된다. 조화 수열(harmonic-series) 계수, $\alpha(t) = 1/t$ 은 이들 제약조건을 만족한다.^[6] 본 실험에서는 초기값 0.9를 가지면서 선형 감소하는 값이 공통적으로 사용되었다. 사용된 학습률은 식 (17)과 같다.

$$\alpha(t) = 0.9 \cdot \frac{1}{t}, 1 \leq t \leq \text{Iterations} \quad (17)$$

t 이때 t : 현재 학습 횟수,

Iterations : 최대 학습 횟수

학습이 종료되면 인식률을 계산하기 위해 클래스 지정 작업을 수행한다. 경쟁 학습 알고리즘은 비지도 학습 알고리즘이므로 학습된 출력 뉴런을 최고 빈도수를 가지는 클래스로 단순히 지정하는 것은 무리가 있을 수 있다. 따라서 본 실험에서는 출력 뉴런의 최고 빈도수를 가지는 클래스가 전체 빈도수의 50% 이상일 때만 해당 뉴런을 최고 빈도수의 클래스로 지정하여 인식률을 계산할 때 사용한다. 만일 50% 미만인 경우에는 그 출력 뉴런은 인식률 계산에서 제외하고, 입력 패턴과 다음으로 거리가 가까운 뉴런을 사용하여 인식률을 계산한다.

우리는 Sun Microsystems사의 Sun 4/75 워크스테이션과 호환인 기계를 사용하여 Sun OS 4.1의 환경에서 C 언어로 프로그래밍하였다.

3. 성능 평가

실험에서는 표준 SCL, SOFM, FSCL 알고리즘을 제안된 동적 출력 뉴런 생성 경쟁 학습 알고리즘에 적용시킨 DSCL(Dynamic Simple Competitive Learning), DSOFM(Dynamic Self-Organizing Feature Map), DFSCL(Dynamic Frequency-Sensitive Competitive Learning) 알고리즘들과 비교하였다. 제안된 알고리즘의 출력 뉴런 초기 클래스 임계치 $r_0 = 0.1$ 이 공통적으로 사용되었다. 특히 SOFM 알고리즘의 경우에는 위상적으로 인접한 이웃 뉴런 대신에 Kangas등이 제안한 최

소 신장 트리(MST)를 계산하여 이웃 뉴런을 정의하는 SOFM의 변형을 사용하였다.^[13] 이들이 제안한 방식이 우리가 제안한 동적 뉴런 생성 알고리즘에 보다 적합하기 때문이다. 또한 이들이 실험한 방식과 비슷하게 우리도 학습 횟수가 10회이내일 때는 학습 횟수가 증가할 때마다 매번 최소 신장 트리를 새로 계산하였으나, 10회이상일 때는 어느 정도 학습의 안정화가 이루어지므로 매 10회마다 갱신하고, 100회 이상일 때는 매 100회마다 신장트리를 수정하여 실행 시간을 줄였다.

표준 SCL과 DSCL에 대하여 학습 데이터와 테스트 데이터를 사용한 인식률(recognition rate) 및 학습 데이터의 MSE를 표 2에 보인다. 실험 결과는 제안된 동적 출력 뉴런 생성 알고리즘을 적용하여 수행시킨 DSCL은 SCL에 비하여 MSE가 약간 크지만, 학습 데이터의 인식률과 테스트 데이터 인식률 모두에서 우수함을 보인다. 특히 테스트 데이터의 인식률이 매우 우수하다.

표 2. 한국어 단모음 데이터에 대한 표준 SCL과 DSCL의 인식률과 MSE 비교

Table 2. Recognition rates and MSE's of SCL and DSCL for Korean monophthongs data.

학습 횟수		10	50	100	300	500	1000
실험 방법							
SCL	학습	95.38	95.62	95.25	95.19	95.17	95.17
	테스트	84.97	83.77	83.51	83.26	83.21	84.52
	MSE $\times 10^2$	10.79	7.72	7.46	7.34	7.33	7.32
DSCL	학습	95.44	95.60	95.62	95.65	95.65	95.65
	테스트	85.71	86.24	86.08	86.03	86.03	86.02
	MSE $\times 10^2$	11.18	7.78	7.51	7.38	7.38	7.37

학습 데이터와 테스트 데이터에 대한 SOFM과 DSOFM의 인식률 및 학습 데이터에 대한 MSE를 표 3에, FSCL과 DFSCL에 대한 결과는 표 4에 보인다. 표 3의 실험 결과로부터, 제안된 동적 출력 뉴런 생성 알고리즘을 적용하여 수행시킨 DSOFM은 SOFM에 비하여 MSE가 매우 우수하므로 학습 데이터의 인식률이 뛰어나며, 테스트 데이터의 경우에도 약간 우수함을 알 수 있다. 특히 제안된 방법은 기존의 방법보다 MSE가 우수할 뿐만 아니라 MSE가 빠르게 감소하여, 입력 패턴의 분포에 빠르게 그리고 더욱 정확하게 적응함을 보여주고 있다. 표 4의 실험 결과는 FSCL이 다른 경쟁 학습 알고리즘에 비해 빠

르게 수렴하며 성능이 우수함을 보인다. 수렴속도면에서는 기존의 방법과 제안된 방법이 크게 차이가 없지만, 학습 데이터와 테스트 데이터 모두에 대해서 제안된 방법은 약간 우수한 인식률을 보인다.

표 3. 한국어 단모음 데이터에 대한 표준 SOFM 과 DSOFM의 인식률과 MSE 비교

Table 3. Recognition rates and MSE's of SOFM and DSOFM for Korean monophthongs data.

학습 횟수		10	50	100	300	500	1000
실험 방법							
SOFM	학습	87.23	92.73	93.27	93.33	93.29	93.27
	테스트	79.76	84.92	84.80	84.56	84.58	84.95
	MSE $\times 10^2$	30.37	11.71	10.93	10.63	10.61	10.52
DSOFM	학습	92.83	94.71	95.44	95.19	95.08	94.38
	테스트	88.34	85.64	85.08	85.38	84.58	85.19
	MSE $\times 10^2$	11.18	7.87	7.31	7.07	7.00	7.08

표 4. 한국어 단모음 데이터에 대한 표준 FSCL 과 DFSCL의 인식률과 MSE 비교

Table 4. Recognition rates and MSE's of FSCL and DFSCL for Korean monophthongs data.

학습 횟수		10	50	100	300	500	1000
실험 방법							
FSCL	학습	95.60	94.50	94.79	94.85	94.85	94.85
	테스트	86.14	84.61	84.22	85.47	85.27	85.38
	MSE $\times 10^2$	8.97	7.12	7.03	7.00	7.02	7.02
DFSCL	학습	94.85	94.79	95.00	95.06	95.08	95.08
	테스트	85.15	85.64	85.80	85.75	85.67	85.60
	MSE $\times 10^2$	9.41	7.11	7.02	7.01	7.02	7.01

결론적으로 화자독립 한국어 단모음 인식 실험에서, 제안된 방법은 표준 방법들에 비하여 높은 인식률을 보인다. 또한 학습횟수가 증가할수록 MSE가 감소하여 학습이 안정되게 진행됨을 보여 준다. 이는 기존의 방법들과 마찬가지로 반복 학습을 수행함으로써 알고리즘이 수렴함을 실험적으로 입증하고 있다.

V. 결론

본 논문에서 제안한 경쟁 학습 알고리즘은 먼저 출력 뉴런의 클래스 임계치가 분산값 변화량에 따라 동

적으로 변하므로 가중치 벡터의 초기 학습 결정 현상을 극복하고, 입력 패턴의 분포 특성을 보다 잘 반영하면서 안정된 학습을 수행한다. 특히 입력 패턴의 분포에 따라 출력 뉴런을 생성하므로 불필요한 출력 뉴런을 생성하지 않고, 생성된 뉴런의 미학습 현상은 발생하지 않는다.

제안된 알고리즘은 패턴 인식에 응용되는 대부분의 경쟁 학습 알고리즘들에 적용시킬 수 있다. 따라서 우리는 제안된 알고리즘을 SCL, 변형된 SOFM, 그리고 FSCL 알고리즘들에 적용시켜 표준의 방법들과 비교하여 성능을 평가하였다. 실험 데이터로는 화자 독립 음소 데이터를 사용하였다. 실험 결과는 제안한 방법이 기존의 방법들에 비해 인식률이 향상되고, 학습 횟수가 증가할수록 학습 오차인 MSE는 감소하므로 기존의 방식들과 마찬가지로 수렴함을 보여준다.

제안된 방법은 초기에 입력 패턴들의 분포에 적절히 적용하면서 동적으로 출력 뉴런들을 생성하고, 최대 갯수의 출력 뉴런이 생성될때까지는 승자 뉴런 계산시간이 단축되므로 초기 학습 속도가 기존의 방식들에 비하여 빠르다는 장점이 있다. 하지만 제안한 방법에서도 ART와 같은 기존의 동적 출력 뉴런 생성 방식들과 마찬가지로 신경망의 초기 패러미터 선정이 문제가 된다. 즉 출력 뉴런마다 클래스 임계치라는 패러미터를 사용하므로, 이 패러미터의 적절한 초기값 선정이 중요하다. 그러나 학습이 진행되면서 분산값 변화량에 따라 이 값이 동적으로 증감된다는 점이 기존의 방식들과 다르다. 이 값의 선정을 주어진 학습 패턴의 특성으로부터 적절히 선택하기 위한 방식의 개발이 추후 과제로서 필요하고, 실험으로 입증된 알고리즘의 학습 수렴성에 대한 수학적 증명이 요구된다.

參考文獻

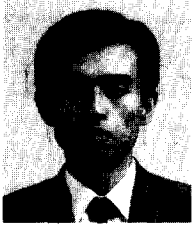
[1] Schalkoff, R.J., *Pattern Recognition: Statistical, Structural and Neural Approaches*, John Wiley & Sons, New York, 1992.

[2] Zurada, J.M., *Introduction to Artificial Neural Systems*, West Publishing Company, St. Paul, MN, 1992.

[3] Rummelhart, D.E., Hinton, G.E., and Williams, R.J., "Learning Internal Representations by Error Propagation." In Rummelhart, D.E. and McClelland, J.L eds *Parallel Distributed Processing*.

- MIT Press, Cambridge, MA, 1988.
- [4] Rummelhart, D.E. and Zipser, D., "Feature Discovery by Competitive Learning." In Rummelhart, D.E. and McClelland, J.L. eds *Parallel Distributed Processing*, MIT Press, Cambridge, MA, 1988.
- [5] Kohonen, T., *Self-Organization and Associative Memory*, 3rd ed., Springer-Verlag, Berlin, 1989.
- [6] Gong, S.G. and Kosko, B., "Differential Competitive Learning for Phoneme Recognition." In Kosko, B. ed *Neural Networks for Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1992.
- [7] Lee, J.C., Sheu, B.J., Fang, W.C., and Chellappa, R., "VLSI Neuroprocessors for Video Motion Detection." *IEEE Trans. on Neural Networks*, Vol. 4, PP.178-191, 1993.
- [8] Ahalt, S.C., Krishnamurthy, A.K., Chen, P., and Melton, D.E., "Competitive Learning Algorithms for Vector Quantization." *Neural Networks*, Vol. 3, pp.277-290, 1990.
- [9] Hartigan, J.A., *Clustering Algorithms*, John Wiley & Sons, New York, 1975.
- [10] Carpenter, G. and Grossberg, S., "A Massively Parallel Architecture for a Self-Organizing Neural Pattern Recognition Machine." *CVGIP*, Vol.37, pp.54-115, 1987.
- [11] Kim, J.W. and Hwang, H.Y., "Improved VQ Design Method using 2-Phase Self-Organizing Neural Network." *Proc. Int'l Conf. on Neural Network Applications to Signal Processing*, Singapore, pp.370-375, 1993
- [12] DeSieno, D., "Adding a Conscience to Competitive Learning." *Proc. IEEE Int'l Conf. on Neural Networks*, San Diego, Vol.I, pp.117-124, 1988
- [13] Kangas, J.A., Kohonen, T.K., and Laaksonen, J.T., "Variants of Self-Organizing Maps." *IEEE Trans. on Neural Networks*, Vol.1, pp.93-99, 1990.
- [14] Sedgewick, R., *Algorithms in C++*, Addison-Wesley, Reading, MA, 1992.
- [15] 김 승곤, "음성학." 정음사, pp.141-142, 1983
- [16] Schafer, R.W., Rabiner, L.R., "Digital Representations of Speech Signals." In Waibel, A. and Lee, K. eds *Readings in Speech Recognition*, Morgan Kaufmann, San Mateo, CA, pp.49-64, 1990.
- [17] Noll, A.M., "Cepstrum pitch determination." *J. Acoust. Soc. Amer.*, Vol. 41, pp.293-309, 1967.
- [18] 김 기석, 임 은진, 황 회용, "음성인식 신경망을 위한 음성 파라미터들의 성능 비교." 한국음향학회 논문지, Vol.11, No.3, pp.61-66, 1992
- [19] Robbins, H. and Monro, S., "A Stochastic Approximation Method." *Annals of Mathematical Statistics*, Vol. 22, pp.400-407, 1951.

著者紹介



金鍾完(正會員)

1964年 11月 13日生. 1987年 2月 서울대학교 컴퓨터공학과 졸업 (학사). 1989年 2月 서울대학교 대학원 컴퓨터공학과 졸업 (석사). 1994年 8月 서울대학교 대학원 컴퓨터공학과 졸업 (박사). 1991年

3月 ~ 현재 서울대학교 연구처 조교. 주관심 분야는 신경회로망, 패턴인식, 컴퓨터비전 등임.



安濟晟(正會員)

1992年 서울대학교 컴퓨터공학과 졸업 (학사). 1994年 서울대학교 컴퓨터공학과 졸업 (석사). 1994年 ~ 현재 금성사 중앙연구소 연구원. 주관심 분야는 인공지능망, 유전자 알고리즘 등임.

金宗相(正會員)

1960年 2月 서울대학교 전자공학과 졸업 (학사). 1965年 2月 서울대학교 전자공학과 졸업 (석사). 1975年 2月 서울대학교 전자공학과 졸업 (박사). 1983年 ~ 현재 서울대학교 컴퓨터공학과 교수. 1992年 ~ 1994年 컴퓨터신기술공동연구소 소장. 1986年 ~ 1988年 한국정보과학회 회장. 주관심 분야는 컴퓨터통신 프로토콜, 컴퓨터구조 등임.



李興浩(正會員)

1973年 2月 서울대학교 공업교육과(전기전공)졸업. 1977年 2月 서울대학교 대학원(석사). 1994年 8月 서울대학교 대학원 컴퓨터공학과 졸업(공학박사). 1983年 12月 ~ 1984年 12月 미국 FIT 대학 전기공학과 교수. 주관심 분야는 디지털 시스템, 영상 패턴 인식, 신경망, 전력전자 전기 설비 등임.



趙成元(正會員)

1959年 9月 7日生. 1982年 서울대학교 전기공학과 졸업 (학사). 1987年 미국 Purdue 대학교 전기공학과 졸업 (석사). 1992年 미국 Purdue 대학교 전기공학과 졸업 (박사). 1992年 ~ 현재 홍익

대학교 전기 제어공학과 조교수. 주관심 분야는 신경회로망, 퍼지시스템, 인공지능, 컴퓨터비전 및 음성인식 등임.