

승자 노드의 빈도 수를 이용한 개선된 SOM 알고리즘

이준행*, 김재용**, 김광백**

*신라대학교 컴퓨터정보공학부

**신라대학교 컴퓨터공학과

Enhanced SOM Algorithm by Using Frequency Number of Winner Node

Jun-Hang Lee*, Jae-Young Kim**, Kwang-Baek Kim**

* Division of Computer and Information Engineering, Silla University

** Dept. of Computer Engineering, Silla University

E-mail : leejh5406@hanmail.net, arioner@webmail.silla.ac.kr, gbkim@silla.ac.kr

요 약

SOM 알고리즘에서 가중치 조정은 입력 벡터와 승자 노드의 대표 벡터간의 차이만큼 조정되고 승자 노드의 대표 벡터에 입력 벡터의 정보를 반영하게 된다. 여기서 그 정보를 반영할 때 입력 벡터와 승자 노드의 대표 벡터간에 차이가 크면 승자 노드의 대표 벡터에 입력 벡터를 기억시키기 위해 입력 벡터의 정보를 더 많이 반영해야 한다. 이러한 문제점을 개선하기 위해 본 논문에서는 승자 노드의 대표 벡터와 입력 벡터간의 출력 오류를 0과 1사이의 정규화된 값으로 출력 오류를 계산하여 학습률을 조정하고 승자 노드의 저 활용 문제를 개선하기 위해 학습 중에 각 승자 노드의 대표 벡터들이 수정되고 선택되어지는 횟수가 가능한 동등해지도록 각 노드의 승자 빈도수를 가중치 조정에 반영하는 개선된 SOM 알고리즘을 제안하였다. 제안된 방법의 인식 성능을 평가하기 위해 주민등록증에서 추출한 숫자 패턴 50개를 대상으로 실험한 결과, 제안된 방법의 인식 성능이 기존의 SOM 알고리즘보다 개선된 것을 확인하였다.

키워드

SOM 알고리즘, 승자 노드, 대표 벡터, 승자 빈도수

1. 서 론

오늘날 신경망 학습 알고리즘의 학습 시간과 인식성의 능력을 향상시키기 위한 연구가 진행되어 왔다. SOM(Self Organizing Map) 알고리즘은 인간의 자율적인 학습과 유사한 형태로 학습이 이루어지는 알고리즘이며 음성인식과 문자인식 등 다양한 분야에 적용되고 있다[1]. SOM 알고리즘은 자기 조직화하는 Network으로서 입력층과 경쟁층의 2개의 층으로 이루어져 있다. 연결강도와 입력 벡터와의 거리를 계산하여 승자 벡터를 선정하고 승자 벡터와 이웃뉴런들만이 학습한다. SOM 알고리즘은 주어진 입력에 대해 올바른 출력값이 제공되지 않는 자율학습 방식이다[2,3]. 또한 반응하는 순서나 위치를 통해 위상이 보존(topology preserving)되는 특성을 가지고 있어 많은 분야에 용

용되고 있다. 그러나 SOM 알고리즘은 학습이 되기 전에 위상을 미리 고정시켜야 하는 문제점이 있다. 그리고 SOM 알고리즘은 자기 구성 지도의 고정된 구조 때문에 발생하는 문제를 해결하기 위해 지도 구조를 학습 중에 적절하게 변경시켜야 하고 가중치를 조정하는 과정에서 학습률에 따라 클러스터의 수와 인식률이 달라지는 문제점이 있다[4,5]. SOM 알고리즘에서 가중치 조정은 입력 벡터와 승자 노드의 대표 벡터간의 차이만큼 조정되고 승자 노드의 대표 벡터에 입력 벡터의 정보를 반영하게 된다. 여기서 그 정보를 반영할 때 입력 벡터와 승자 노드의 대표 벡터간의 차이가 크면 승자 벡터에 입력 벡터를 기억시키기 위해 입력 벡터의 정보를 더 많이 반영해야 한다. 따라서 기존의 SOM 알고리즘은 학습이 진행됨에 따라 학습률을 일정한 비율로 감소시키므로 입력 벡터들을 정확

히 분류되지 않는 경우가 발생한다. 따라서 제안된 SOM방법에서는 승자 노드의 대표 벡터와 입력벡터간의 출력오류를 0과1사이의 정규화된 값으로 출력오류를 계산하여 학습률을 조정하고 승자노드의 저 활용 문제를 개선하기 위해 학습 중에 각 승자 노드의 대표 벡터들이 수정되고 선택되어지는 횟수가 가능한 동등해지도록 각 노드의 승자 빈도수를 가중치 조정에 적용하는 개선된 SOM 알고리즘을 제안한다.

II. 관련 연구

SOM 알고리즘은 자율학습의 한 형태로 자기조직화(self-organizing)하여 경쟁 구조를 가진다. 자기 조직화란 주어진 입력패턴에 대하여 정확한 해답을 미리 주지 않고 자기 스스로 학습할 수 있는 능력을 말한다. SOM 알고리즘은 계층적인 시스템이 아니라 두 개의 층으로 이루어져 있다. 이 네트워크의 첫 번째 층은 입력층을 의미하고, 두 번째 층은 경쟁층(competitive layer)인데 2차원의 격자 또는 1차원의 격자로 이루어져 있다. 모든 연결들은 첫 번째 층에서 두 번째 층으로의 연결되고 완전연결 구조를 가진다.

SOM 알고리즘의 학습 방법은 “승자-독점”이다. 승자만이 출력을 낼 수 있으며 승자와 그 이웃들만이 그들의 연결 강도를 조정할 수 있기 때문이다. 따라서 각 뉴런들은 연결 강도 벡터와 입력 벡터가 얼마나 가까운가를 계산하고, 각 뉴런들은 학습할 수 있는 특권을 부여받기 위해 경쟁하는데, 입력층과 경쟁층 사이의 거리가 가장 가까운 뉴런이 승리하게 된다. 이 승자 뉴런이 출력신호를 보낼 수 있는 유일한 뉴런이 된다. 승자 뉴런을 결정하고 난 후에는 SOM 알고리즘의 학습 규칙에 따라 뉴런의 연결 강도를 조정한다. 이 규칙은 다음식과 같다.

$$W(t+1) = W(t) + \eta(X - W(t))$$

여기서 $W(t)$ 는 조정되기 이전의 연결 강도 벡터이며, $W(t+1)$ 는 조정된 후의 새로운 연결 강도 벡터이다. 그리고 X 는 입력벡터이고, η 는 학습률이다. 승자 연결 강도 벡터는 기하학적으로 입력패턴 벡터에 가장 가깝다. SOM 알고리즘의 학습은 단순히 연결 강도 벡터와 입력패턴 벡터의 차이를 구한 다음 그것의 일정한 비율을 원래의 연결강도 벡터에 더하는 것이다. 이 때 승자 뉴런만이 그것과 관련된 연결 강도 벡터를 조정하는 것이 아니라 그 이웃 반경 안에 드는 모든 뉴런들도 유사한 조정을 하게 된다. 승자 뉴런은 +1출력으로 내며, 승자 뉴런과 그것의 이웃 뉴런들은 각각의 연결 강도 벡터를 입력 벡터에 접근하게 된다. 그리고 η 는 0과 1사이의 값을 가지는 학습률로 시간이 경과함에 따라 일정한 비율로 점차 감소한다.

III 빈도 수를 이용한 개선된 SOM 알고리즘

기존의 SOM 알고리즘에서 가중치 조정은 입력 벡터와 승자 노드의 대표 벡터간의 차이만큼 조정되고 승자 노드의 대표 벡터에 입력벡터의 정보를 반영하게 된다. 여기서 그 정보를 반영할 때 입력 벡터와 승자 노드의 대표 벡터간에 차이가 크면 입력 벡터를 기억시키기 위하여 입력 벡터의 정보를 더 많이 반영해야한다. 이러한 문제점을 개선하기 위하여 제안된 SOM 알고리즘에서는 승자 노드에서의 오류 값을 0과 1사이의 값으로 정규하고 학습률에 적용한다. 따라서 제시된 출력 오류에 의한 가중치 조정으로 오류의 크기에 따라서 그 비율만큼 가중치를 조정한다.

그리고 승자 노드의 저 활용 문제를 개선하기 위해 학습 중에 각 승자 노드들이 수정되고 선택되어지는 횟수가 가능한 동등해지도록 각 노드의 승자 빈도수를 가중치 조정에 적용한다. 즉, 학습이 끝날 때 각각의 노드들이 승자 노드로 선정된 빈도수가 유사하도록 한다. 이를 위해서는 승자 노드의 대표 벡터들과 학습 벡터간의 실제적인 왜곡 정도도 충분히 고려되어야 한다. 따라서 경쟁학습 알고리즘에서 승자노드를 선정하기 위한 왜곡의 척도를 승자노드로 선정된 빈도수와 유클리디안 거리의 곱으로 수정한다. 오류값을 이용한 방법은 현재의 가중치 변화량만 고려하고 그 전의 변화량은 고려되지 않는다. 따라서 비록 적용적으로 가중치 조정이 되지만, 현재의 변화량에 의해 가중치의 값이 많이 좌우된다. 이러한 문제점을 해결하기 위해 가중치 조정시 현재의 가중치 변화뿐만 아니라 그 이전의 가중치 변화량에 비례하여 가중치를 조정하도록 한다. 제안된 SOM 학습 알고리즘을 단계별로 정리하면 다음과 같다.

단계 1 : N개의 입력으로부터 M개의 출력노드 사이의 연결강도(w_{ij})를 임의의 적은 값으로 초기화한다.

단계 2 : 새로운 입력 패턴을 제시한다.

단계 3 : 입력 노드와 출력 노드들 간의 거리(d_j)를 계산한다.

$$d_j = \sum_{i=0}^n (x_i - w_{ij})^2$$

단계 4 : 최소거리에 있는 출력 노드(j^*)를 승자 노드로 선택한다.

단계 5 : 선택된 승자 노드(j^*)와 그 이웃 반경 내의 노드들의 연결 가중치와 학습률을 다음과 같이 조정한다.

$$w_{ij}(n+1) = w_{ij}(n) + \delta(n+1)$$

$$\delta(n+1) = \eta(n+1)(x_i(n) - w_{ij}(n)) + \eta(n+1)\delta(n)$$

η 는 학습률로서 승자노드와 입력노드 간의 출력오류를 0과1사이의 정규화된 값과 승자노드가 된 빈도수를 적용하여 동적으로 조정한다.

$$\eta = f(e_{j^*}) + \frac{1}{f_{j^*}}$$

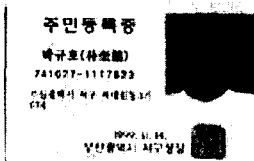
$$e_{j^*} = \frac{1}{1 + e^{-(x_i - w_{ij^*})}}$$

여기서 j^* 는 j 번째 출력 노드가 승자로 선택된 것을 의미하고 f_{j^*} 는 승자 노드로 선택된 빈도수이다. 그리고 $f(e_{j^*})$ 는 시그모이드 함수이다.

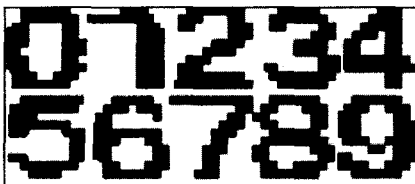
단계 6 : 단계 2로 가서 반복한다.

IV. 실험 및 분석

제안된 알고리즘의 학습 및 인식 성능을 분석하기 위하여 Intel Pentium IV 2.0 GHz CPU와 256 MB RAM이 장착된 IBM 호환 PC 상에서 Visual C++ 6.0으로 구현하였다. 실험에 사용된 입력 데이터는 800 × 600 픽셀 크기를 가진 주민등록증 영상에서 추출한 주민등록 번호를 학습 및 인식에 적용하였다. 실험에 적용된 영상은 그림 2와 같고, 추출된 주민등록증 번호는 그림 3과 같다.



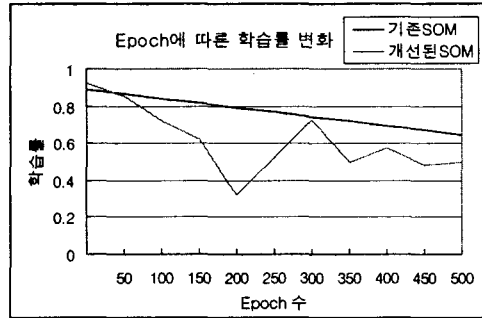
[그림 2] 주민등록증 영상



[그림 3] 추출된 주민등록 번호

그림 3과 같이 추출된 80개의 주민등록 번호 중에서 학습에 적용된 주민등록 번호는 30개이고 인식 실험에 적용된 주민등록 번호는 50개이다.

그림 3과 같은 30개의 주민등록 번호를 기존의 SOM 알고리즘과 제안된 SOM 알고리즘에 적용하여 학습의 반복 횟수에 따른 학습률의 변화를 그림 4로 나타내었다. 여기서 초기 반복은 4개로 설정하였다.



[그림 4] Epoch수에 따른 학습률 변화

그림 4에서와 같이 기존의 SOM 알고리즘 보다 제안된 SOM 알고리즘이 입력 벡터와 승자 노드의 대표 벡터간의 차이가 많이 나면 승자 노드의 대표 벡터에 입력 벡터의 정보를 더 많이 반영하는 것을 확인할 수 있다.

<표 1> Epoch수에 따른 인식을 비교

알고리즘	반경	Epoch	인식률
SOM알고리즘	4	100	17/50
		200	34/50
		300	46/50
		400	47/50
		500	47/50
개선된SOM 알고리즘	4	100	19/50
		200	37/50
		300	47/50
		400	49/50
		500	50/50

제안된 SOM 알고리즘은 가중치 조정에 출력 오류 개념을 도입하여 승자 노드의 대표 벡터와 입력 벡터의 오류 값에 비례하도록 가중치를 조정하고 현재의 가중치 변화량뿐만 아니라 이전의 가중치 변화량도 고려하여 학습하므로 표 1에서와 같이 개선된 SOM 알고리즘이 기존 SOM 알고리즘보다 인식률이 개선되었다.

IV. 결론

SOM 알고리즘에서 가중치 조정은 입력 벡터와 승자 노드의 대표 벡터간의 차이만큼 조정되고 승자노드의 대표벡터에 입력벡터의 정보를 반영하게

된다. 여기서 그 정보를 반영할 때 입력벡터와 승자 노드의 대표 벡터간에 차이가 크면 승자 노드의 대표 벡터에 입력벡터를 기억시키기 위해 입력 벡터의 정보를 더 많이 반영해야 한다. 이러한 문제점을 개선하기 위하여 개선된 SOM 알고리즘에서는 출력오류 개념을 도입하여 승자 노드의 대표 벡터와 입력 벡터의 오류값에 비례하도록 가중치를 조정하였다. 또한 가중치 조정시 현재의 가중치 변화량뿐만 아니라 이전의 가중치 변화량도 영향을 주게 하였다.

제안된 방법의 인식 성능을 평가하기 위해 주민 등록증에서 추출한 숫자 패턴 50개를 대상으로 실험한 결과, 제안된 방법의 인식 성능이 기존의 SOM 알고리즘보다 개선된 것을 확인하였다.

향후 연구과제는 제안된 방법에서 승자 노드의 출력 오류 값을 정규화 하는데 적용된 시그모이드 함수 대신에 입력 벡터와 승자 노드의 대표 벡터의 차이를 정확히 반영할 수 있는 퍼지 이론을 적용한 함수에 대한 연구가 필요하고 제안된 방법을 컬러 이미지의 벡터 양자화에 적용하여 다른 방법들과 압축 성능을 비교 분석 할 것이다.

참고 문헌

- [1] A. S. Pandya, R. b. Macy, Pattern Recognition with Neural Networks in C++, IEEE Press, 1995.
- [2] T. Kohonen. "Self-Organizing Mpa, Springer, Berlin Heidelberg, 1995.
- [3] 김광백, 노영욱, "동적인 임계화 방법과 코호넨 알고리즘을 이용한 차량 번호판 인식에 관한 연구," 한국통신학회논문지, 제26권, 12A호, pp.2019-2026, 2001.
- [4] 김현돈, 조성배, "유전자 알고리즘을 사용한 구조 적용 자기구성 지도의 최적화," 퍼지 및 지능 시스템학회논문지, 제11권 제3호, pp.223-230, 2001.
- [5] K. B. Kim, E. Y. Cha," A Fuzzy Self Organizing Vector Quantization for Image," Proceedings of Soft Computing, Vol.2, pp.757-760, 1997.