

자가 조직화 지도의 커널 공간 해석에 관한 연구

정 성 문[†] · 김 기 범[†] · 홍 순 좌^{**}

요 약

Kohonen SOM(Self-Organizing Map)이나 MLP(Multi-Layer Perceptron), SVM(Support Vector Machine)과 같은 기존의 인식 및 클러스터링 알고리즘들은 새로운 입력 패턴에 대한 적응성이 떨어지고 학습 패턴 자체의 복잡도에 대한 학습률의 의존도가 크게 나타나는 등 여러 가지 단점이 있다. 이러한 학습 알고리즘의 단점은 문제의 학습 패턴 자체의 특성을 잃지 않고 문제의 복잡도를 낮출 수 있다면 보완할 수 있다. 패턴 자체의 특성을 유지하며 복잡도를 낮추는 방법론은 여러 가지가 있으며, 본 논문에서는 커널 공간 해석 기법을 접근 방법으로 한다. 본 논문에서 제안하는 kSOM(kernel based SOM)은 원 공간의 데이터가 갖는 복잡도를 무한대에 가까운 초 고차원의 공간으로 대응시킴으로써 데이터의 분포가 원 공간의 분포에 비해 상대적으로 성긴(spase) 구조적 특징을 지니게 하여 클러스터링 및 인식률의 상승을 보장하는 메커니즘을 제안한다. 클러스터링 및 인식률의 산출은 본 논문에서 제안한 새로운 유사성 탐색 및 갱신 기법에 근거하여 수행한다. CEDAR DB를 이용한 필기체 문자 클러스터링 및 인식 실험을 통해 기존의 SOM과 본 논문에서 제안한 kSOM과 성능을 비교한다.

키워드 : SOM, 자가조직화 지도, 커널 공간 해석, 학습

A New Self-Organizing Map based on Kernel Concepts

Sung-moon Cheong[†] · Ki-bom Kim[†] · Soon-jwa Hong^{**}

ABSTRACT

Previous recognition/clustering algorithms such as Kohonen SOM(Self-Organizing Map), MLP(Multi-Layer Perceptron) and SVM(Support Vector Machine) might not adapt to unexpected input pattern. And it's recognition rate depends highly on the complexity of own training patterns. We could make up for and improve the weak points with lowering complexity of original problem without losing original characteristics. There are so many ways to lower complexity of the problem, and we chose a kernel concepts as an approach to do it. In this paper, using a kernel concepts, original data are mapped to hyper-dimension space which is near infinite dimension. Therefore, transferred data into the hyper-dimension are distributed sparsely rather than originally distributed so as to guarantee the rate to be risen. Estimating ratio of recognition is based on a new similarity-probing and learning method that are proposed in this paper. Using CEDAR DB which data is written in cursive letters, 0 to 9, we compare a recognition/clustering performance of kSOM that is proposed in this paper with previous SOM.

Key Words : SOM, Self-Organizing, Kernel Concept, Learning

1. 서 론

21세기는 첨단 기술 사회로 인간 생활의 질적 향상을 위해서 기계와 인간의 대화가 가능할 정도의 음성 인식, 얼굴 인식, 필기체 인식 등의 발전이 필수적인 요소의 하나이다. 한 가지 예로, 귀에 문제가 있어 소리를 듣는데 장애를 가진 사람들이나 나이가 많이 들어 소리를 잘 듣지 못하는 노인분들께 휴대용 음성 인식 시스템을 통해 전해진 소리를 액정에 문자들로 표현할 수 있다면 그들에게 있어서 삶의 질은 이전보다 훨씬 향상될 것이다. 또한 자동차와 같이 운

전을 하는 중에 다른 곳에 시선을 두게 되면 사고의 위험이 높은 상황에서 단순하게 말로써 명령을 내림으로써 라디오를 켜거나 걸려온 전화를 받을 수 있는 명령 인식 시스템 등은 삶의 질을 향상 시키게 될 것임은 두말할 나위가 없다.

이와 같은 학습 및 인식의 연구는 사람의 학습 과정을 모방하는데서 출발한다. 사람은 크게 두 가지 학습 방법을 통해 지식을 습득한다. 교사학습(Supervised Learning)과 비교사학습(Unsupervised Learning)이 그것이다. 초기의 학습은 책과 같은 객관적인 정보를 통해 수행되지만 일정 수준의 정보가 축적되면 사람은 누군가의 조언이 없더라도 스스로 갖고 있는 정보들을 토대로 판단을 내리게 된다. 기존인 학습 방법론 연구는 이 두 가지 분류를 따른다. MLP(Multi-Layer Perceptron)나 RBF(Radial-Basis Function), HMM(Hidden

[†] 정 회 원 : 국가보안기술연구소 연구원

^{**} 정 회 원 : 국가보안기술연구소 팀장

논문접수 : 2006년 2월 8일, 심사완료 : 2006년 6월 26일

Marcov Model), SVM(Support Vector Machine)[4, 6] 등 다양한 알고리즘들이 교사학습 유형의 연구이고, PCA(Principal Component Analysis), ICA(Independent Component Analysis), SOM(Self Organizing Map), k-means, clustering[11] 등이 비교사학습의 범주의 연구이다.

문제는 기존의 학습 방법이 학습 패턴의 범주 내에서는 성공적인 반응을 보이지만, 예상치 못한 입력이나 예외적인 상황에 직면할 경우 정상적인 처리를 수행하지 못한다는 단점이 존재한다. 또한 학습 패턴의 복잡도에 따라 문제를 해결할 수 없거나, 지나치게 학습패턴에 정형화된 학습 결과를 보이기 때문에 잡음과 같은 실생활 데이터에 적용하기 어려운 것이 현실이다.

SOM은 비교사학습 이론의 한가지로 사람의 신경세포의 동작 원리를 모사한다. 여러 비교사학습 이론들 중에서도 알고리즘의 단순함에 비해 높은 성능과 시각적 상관관계의 표현 가능성 때문에 널리 사용된다. 그러나 SOM역시 데이터 분포의 복잡도나 차원의 크기, 잡음 환경 등에 많은 영향을 받는다. 낮은 차원의 단순한 문제에서는 높은 클러스터링 결과를 얻을 수 있으나, 문제가 복잡해지면 학습이 이루어지지 않거나 수렴한 특징 지도의 클래스 간 경계에서 모호한 분류 결과를 보이기 때문에 응용이 어렵다.

SVM에서는 이러한 문제의 복잡도로 인해 해결의 어려움을 극복하기 위해 커널 개념을 이용한다[4, 6]. 일반적으로 문제의 복잡도를 낮추기 위해 특징 추출 기법을 사용한다. 기존의 특징 추출 방법이 순수 데이터의 크기를 줄임으로써 상대적으로 잡음 요소를 제거하고 차원의 복잡도를 낮추는 반면 SVM 연구에서는 차원을 무한대로 증가시켜 상대적인 성김(spase)을 확보함으로써 커널 기교(Kernel Trick)을 통해 데이터 분포의 복잡도를 상대적으로 낮추면서 비약적으로 인식률을 높인다.

본 논문에서는 커널 함수의 특징을 이용하여 기존의 SOM의 학습 법칙을 커널 공간상의 벡터 내적으로 치환함으로써 새로운 커널 공간 kSOM을 유도한다. 유도된 학습 법칙의 수렴성 여부를 증명하고 필기체 문자 인식 실험을 통해 비교사학습의 인식률을 교사학습에 준하는 수준으로 향상시킬 수 있음을 증명한다.

본 논문의 구성은 다음과 같다. 2장에서는 전통적 Kohonen SOM 알고리즘의 개념과 메커니즘 및 커널 공간 해석 기법을 설명하고, 3장에서는 커널 공간 해석기법과의 접목을 통해 기대하는 복잡도의 해결에 대해 설명하며, 4장에서는 실제로 커널 공간 해석기법을 통해 전통적 Kohonen SOM을 재해석한 kSOM을 제안한다. 5장에서는 제안한 알고리즘의 수학적 증명을 통해 학습 법칙으로써의 활용 가능성을 기술하며 6장에서 CEDAR DB와 TI46 DB를 이용한 장난감 문제에 대해 클러스터링을 수행함으로써 기존의 Kohonen SOM과 kSOM의 성능을 비교한다.

2. 관련 연구

본 장에서는 kSOM(kernel-based Self Organizing Map)

의 기반 구조가 되는 전통적 SOM 알고리즘의 개념을 기술한다. 또한 전통적 SOM의 한계와 이를 개선하기 위해 기계 학습에서 도입된 커널 공간 해석 기법에 대한 개념을 기술한다.

2.1 Self-Organizing Map(SOM)

본 절에서는 Kohonen이 제안했던 전통적 SOM[11]의 알고리즘을 기술한다.

Kohonen의 SOM 학습 알고리즘은 생물학적 특성에서 출발한 학습 이론이다. 학습 결과는 학습에 참여한 여러 데이터들 간의 구조적 관계(유사성)를 표현해주게 된다. 그러나 학습 패턴의 수가 많아지고 분류해야 하는 클러스터의 수가 많아지면 학습 시간이 기하급수적으로 상승하고, 수렴성 역시 감소하게 되어 효율적 학습이 이루어지지 않는다. 즉 원(original) 공간에서 문제의 복잡도(complexity)가 커지면 커질수록 학습의 효율이 떨어지게 된다.

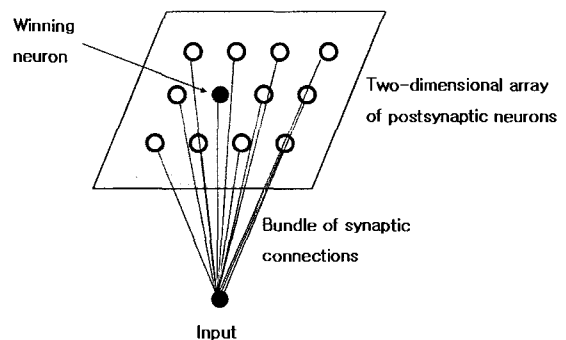
Kohonen의 SOM 알고리즘은 학습 단계 측면에서 크게 세 가지 과정(샘플링, 유사성 탐색, 갱신)으로 나눌 수 있으며, 학습 상태 측면에서는 크게 두 가지 단계(정렬, 수렴)로 나눌 수 있다. 학습은 샘플링, 유사성 탐색, 갱신과정을 반복함으로써 비교사학습을 수행한다. 이때 학습은 특징지도(feature map)의 형태가 더 이상 움직이지 않는 임계점 안에 들어올 때까지 세 과정을 반복한다. 임계점에 이르기까지의 특징지도의 상태는 학습 전반기의 정렬단계를 지나 학습 후반기의 수렴단계로 진행되며 아래와 같은 특징을 지닌다[11].

2.1.1 정렬 단계(ordering phase)

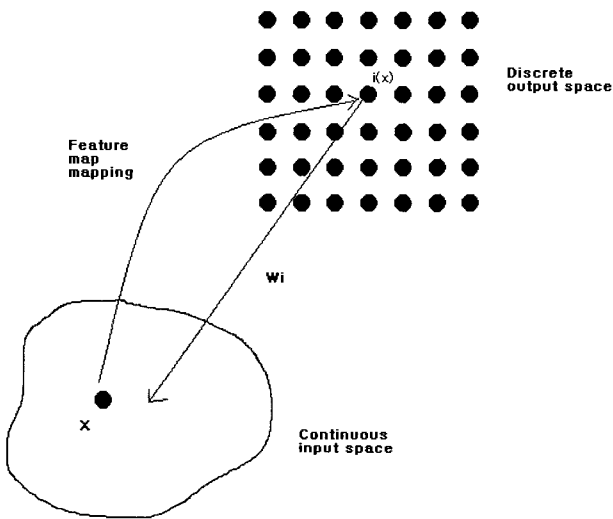
- 학습률(learning rate): 초기에는 큰 값(0.1~1.0)으로 설정하고, 점진적으로 크기를 줄여(0.001~0.01)로 설정한다.
- 이웃 영역(neighborhood): 초기에는 특징 지도의 모든 노드에 영향을 줄 수 있을 만큼 큰 값으로 설정하고 점진적으로 크기를 줄이며, 최종적으로 가장 유사한 노드 값만을 갱신하는 형태, 즉 이웃 영역이 0(자기 자신)이 될 때 까지 줄여간다.

2.1.2 수렴 단계(convergence phase)

- 학습률: 매우 작은 값(0.001~0.0001)으로 일정하게 설정한다.



(그림 1) 유사성 탐색과 이웃 영역



(그림 2) 원(original) 데이터 공간과 특징 지도

- 이웃 영역: 가장 유사한 노드(자기 자신)만 갱신하도록 0으로 설정한다.

2.1.3 알고리즘

전통적 Kohonen SOM의 알고리즘은 다음과 같다.

(1) 초기화

- 초기 무게(weight) 벡터 $\vec{w}_j(0)$ 는 랜덤 값이다. 유일하게 제한된 조건은 $\vec{w}_j(0)$ 는 $j=1,2,3,\dots,l$ (l 은 특징 지도의 노드 개수)에 대해 모두 다르게 선택해야 한다는 것이다. 이때 각각의 무게 벡터는 특징 지도의 노드 하나하나에 일대일 대응되어 있으며 이 벡터의 크기는 학습 패턴의 벡터 길이와 동일하다.

(2) 샘플링

- 학습 패턴 집합으로부터 임의의 확률 모델에 따라 샘플 (\vec{x}) 을 하나 선택하고 이 샘플을 특징 지도에 적용한다.

(3) 유사성 탐색

- 유클리디안 거리 개념을 이용하여 (2)에서 선택된 샘플과의 거리가 최소가 되는 특징 지도 노드, 즉 식(1)을 만족하는 위닝 노드(winning node)를 찾는다.

$$i(\vec{x}) = \arg \min_j \|\vec{x} - \vec{w}_j(n)\|, j = 1, 2, \dots, l \quad (1)$$

(4) 갱신

- 위닝 노드로부터 일정 영역 안에 속해 있는 모든 노드의 벡터들에 대해 식(2)의 학습 법칙에 따라 무게 벡터를 갱신한다.

$$\vec{w}_j(n+1) = \vec{w}_j(n) + \eta(n)h_{j,i(\vec{x})}(n)(\vec{x}(n) - \vec{w}_j(n)) \quad (2)$$

이때 $\eta(n)$ 는 현재 학습 단계(n)에 설정된 학습률이고, $h_{j,i(\vec{x})}$ 는 유사성 탐색 과정에서 선택된 위닝 노드 $i(\vec{x})$ 를 중

심으로 하는 이웃 영역을 결정 함수이다. 앞서 설명한 바와 같이 이 두 변수는 정렬단계 동안 점진적으로 변화시키며 학습을 수행한다.

(5) 반복

- 특징 지도의 각 노드에 할당된 벡터의 변화율이 임의의 임계점 이하로 수렴할 때 까지(2),(3),(4)의 과정을 반복한다.

앞서 언급한 바와 같이 Kohonen SOM 알고리즘의 문제점은 학습 패턴의 복잡도에 따라 학습 효율이 달라진다는 점이다. 반면 학습 패턴의 복잡도 문제가 충분히 해결될 수 있다면 작은 크기에 비해 높은 학습률을 보이는 SOM 알고리즘은 다양한 어플리케이션과 연동되어 활용될 수 있다.

2.2 커널 공간 해석

이 절에서는 커널 공간의 개념을 설명하고, 적절한 커널 함수를 선택함으로써 비선형적 특성을 갖는 높은 복잡도의 데이터 분포를 선형적 특성을 갖는 데이터 분포로 재구성하는 커널 공간 해석 기법을 기술한다.

2.2.1 머서의 이론(Mercer's theorem)

임의의 데이터 공간을 커널 공간으로 할당하는 커널 함수는 머서의 이론에 의해 정의된다.

다음의 수식

$$\int g(x)^2 dx < \infty \quad (3)$$

을 만족하는 임의의 함수 $g(x)$ 가 존재하여

$$\int k(x,y)g(x)g(y)dx dy \geq 0 \quad (4)$$

를 만족시키기만 한다면, 전사함수 Φ 가 존재하여 $k(x,y)$ 는 식(5)와 같이 기술될 수 있다.

$$k(x,y) = \sum_{j=1}^{N_F} \lambda_j \psi_j(x) \psi_j(y) \quad (5)$$

이때 N_F 는 임의의 유한한 정수를 말한다.

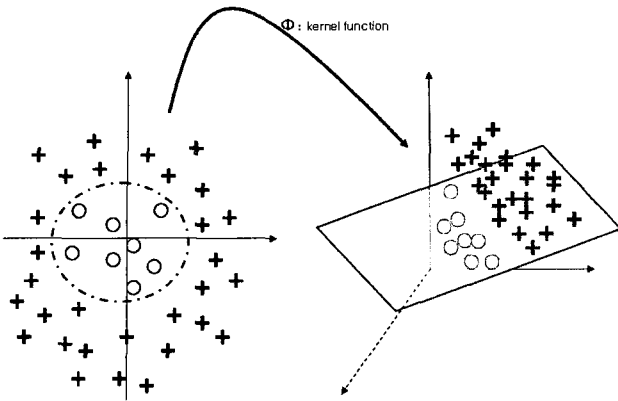
결국 $k(x,y)$ 는

$$k(x,y) = \vec{\Phi}(x) \cdot \vec{\Phi}(y), \quad \vec{\Phi}(x) = (\sqrt{\lambda_1}\psi_1(x), \sqrt{\lambda_2}\psi_2(x), \dots, \sqrt{\lambda_{N_F}}\psi_{N_F}(x)) \quad (6)$$

와 같이 전개될 수 있으며[1], 이때 함수 $k(x,y)$ 를 커널 함수라 부르고, $\vec{\Phi}(x)$ 에 의해 할당된 공간을 커널 공간이라 한다.

2.2.2 커널 공간 해석

커널 공간은 원 데이터 공간과 다음의 관계를 갖는다.



(그림 3) 원 공간 vs. 커널 공간 데이터 분포

$$\begin{aligned} \Phi: \mathbb{R}^N \rightarrow \mathbb{F} \\ \vec{x} \in \mathbb{R}^N, \mathbb{F} = \mathbb{R}^M, N \ll M \end{aligned} \quad (7)$$

식(7)과 같이 $\vec{x}_1, \dots, \vec{x}_i \in \mathbb{R}^N$ 는 \mathbb{F} 에 의해 잠정적으로 훨씬 높은 차원의 공간 \mathbb{F} 로 할당된다. 이때 높은 차원으로 할당된 데이터는 원 데이터 공간에서의 분포에 비해 상대적으로 성긴(sparse) 특성을 갖기 때문에 차원의 상승으로 인해 데이터 자체의 크기는 증가하였으나, 데이터들 간의 복잡도는 상대적으로 감소하게 된다.

문제는 데이터 자체의 크기가 증가하였기 때문에 원 데이터 공간의 계산량의 $\frac{M}{N}$ 배에 해당하는 문제점이 발생한다. 극단적인 경우 $M \approx \infty$ 라면 무한대의 계산량 증가가 발생하게 된다. 이러한 높은 차원에서의 전사 문제에 대해 소위 '차원의 저주(Curse of dimensionality)' 현상이 발생했다고 말한다.

학습을 위한 고차원의 특징 공간으로의 변환이 좋은 접근 방법인가에 대한 문제에 대해, 확률적 학습 이론 연구는 \mathbb{F} 공간에서의 계산이 충분히 해결 가능한 수준의 문제라면 오히려 원 데이터 공간에서의 해석보다 더 간단하게 문제가 해결 될 수 있음을 증명했다[1]. 직관적으로 이 생각의 이해를 도울 수 있는 간단한 예가(그림 3)에 나타나있다.

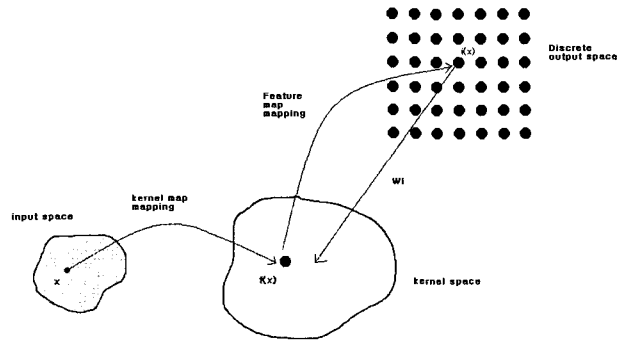
2차원 공간에서 분류 결정자가 직선임을 감안할 때 다소 복잡한 비선형 분류방법으로 해결해야 하는 문제가 커널 공간(3차원 공간)에서는 분류 결정자, 소위 선형 초월 평면(hyperplane)에 의해 간단하게 나누어지게 된다.

비록 확률적 학습 이론 연구에 의해 고차원 공간으로의 변환이 문제 해결의 효율성을 높일 수 있다는 사실이 증명되었다 하더라도 '차원의 저주'가 해결된 것은 아니다.

그러나 다행스럽게도, 특별한 조건(머서의 이론)을 만족하는 커널 공간과 이에 상응하는 커널 함수 $k(x, y)$ 는 식(6)의 정의에 의해 원 데이터 공간에서의 연산만으로 커널 공간에서의 벡터 내적을 수행할 수 있다.

만약 임의의 주어진 학습 식을 커널 공간에서의 학습식으로 치환하고, 치환된 커널 공간 학습 식을 벡터 내적을 통해 커널 함수의 연산으로 대신 할 수 있다면 '차원의 저주'에 의한 연산량의 상승문제를 성공적으로 해결할 수 있게 된다.

본 논문에서는 커널 공간에서 새롭게 정의된 Kohone SOM



(그림 4) 커널 데이터 공간과 특징 지도

학습 모델을 커널 함수의 특성을 이용하여 벡터 내적으로 재구성함으로써 커널 공간의 성긴(sparse) 구조적 특성을 활용하여 향상된 SOM 알고리즘을 제안하고, 제안된 알고리즘의 수학적 증명을 통해 학습 알고리즘으로서의 수렴성을 증명한다.

3. 동 기

2.1에서 설명한 Kohonen의 SOM 알고리즘에 커널 개념을 적용하여 새로운 학습 법칙을 유도한다.

기존의 SOM은 원 공간의 데이터를 조작 없이 해석하기 때문에 원 공간 데이터가 갖는 문제의 복잡도를 그대로 갖는다. 데이터의 차원이 높고 데이터의 양이 많으며 클래스가 다양할수록 비교사학습인 SOM을 통해서 클러스터링을 효율적으로 수행할 수 없다.

앞서 언급한 바와 같이 이와 같은 문제의 복잡도를 해결하기 위한 접근법은 두 가지이다. 문제 자체의 복잡도를 낮추어 해결 가능한 문제로 바꾸거나, SOM의 논리 특징 지도에 더 많은 노드를 할당함으로써 경계 영역에서의 클러스터링의 가능성을 높이는 방법이다. 경계 영역 노드를 삽입하는 방법은 문제 자체가 갖는 복잡도가 전혀 해결되지 않고, 상대적으로 노드 수의 증가로 인한 계산량의 증가를 가져오기 때문에 근본적인 해결책이 되지 못한다. 반면 문제 자체의 복잡도를 낮추는 방법은 계산량의 문제가 발생하지만 많았다면 성능의 향상을 보장한다.

본 논문에서는 문제의 복잡도를 해결하기 위한 접근법으로 커널 공간 해석 기법을 적용한다. 원 공간에서의 데이터 분포가 복잡하더라도 커널 공간으로 대응시킴으로 선형적으로 분리 가능한 형태로 데이터의 분포가 변환이 된다면 높은 수준의 클러스터링 결과를 기대할 수 있다. 물론 이 과정에서 데이터의 차원이 극심하게 증가하지만 실제로 얼마나 증가하는가는 논외이다. 왜냐하면 변환된 데이터는 커널 공간이라 불리는 초고차원공간에 존재하지만 실제로 연산 과정에서는 원 공간의 데이터를 사용하기 때문이다[2]. 이를 일컬어 커널 기교(kernel trick)이라 부른다.

일례로 SVM은 커널 공간의 특성을 활용하여 원 공간에서 선형적으로 분리 불가능한 문제를 커널 공간으로 대응시켜 선형 분리 가능에 가까운 형태로 재구성하여 분류 신뢰

도를 비약적으로 높인다.

본 논문에서 제안하는 kSOM역시 다차원으로 데이터를 대응 시킴으로써 데이터의 분포를 원 공간에 비해 상대적으로 성기게 만듦으로써 경계 영역에서의 모호성을 제거하고, 커널 기교를 이용하여 차원의 상승으로인한 계산량 증가 문제를 해결함으로써 비교사학습인 클러스터링에서도 SVM과 같은 결과를 이끌어 낸다.

4. Kernel-based SOM

본 논문에서 제안하는 kSOM(kernel-based SOM)은 Kohonen의 SOM 알고리즘에 대해 초기화 단계와 샘플링 단계는 동일 하며 유사성 탐색과 갱신의 단계에서 차이점이 존재한다.

Kohonen의 SOM에서는 데이터의 원 공간에서 문제를 해결하고, 무게 벡터 자체를 학습하여 클러스터링을 수행한다. 그러나 kSOM에서 다루는 모든 데이터간의 관계와 해석은 커널 공간으로 변환된 학습패턴을 대상으로 하며, 무게 벡터를 식(12)와 같이 학습 데이터들의 커널 공간 선형 조합으로 표현함으로써 갱신 단계에서의 학습이 무게 벡터 자체가 아닌 무게 벡터 구성을 위한 계수에 의해 이루어진다.

4.1 유사성 탐색

갱신의 단계에 앞서 주어진 특정 지도로부터 입력 패턴과 가장 유사한 노드를 찾아야 한다. 이를 위해서는 커널 공간에서의 거리 개념을 정의할 필요가 있다.

기존의 Kohonen SOM에서 유사성 탐색을 위해 사용된 식(1)을 커널 공간에서의 표현 방법으로 재 정의하면 식(8)과 같다.

$$i(\vec{x}_s) = \arg \min_j \|\vec{\Phi}(\vec{x}_s) - \vec{w}_j(n)\|, j = 1, 2, \dots, l \quad (8)$$

이때 유클리디안 거리는 $\|\vec{\Phi}(\vec{x}_s) - \vec{w}_j(n)\|^2$ 이기 때문에 이를 정리하면

$$\begin{aligned} & \|\vec{\Phi}(\vec{x}_s) - \vec{w}_j(n)\|^2 \\ &= k(\vec{x}_s, \vec{x}_s) + \vec{\alpha}_j^{(n)} \mathbb{K}^T \vec{\alpha}_j^{(n)} - 2\vec{\alpha}_j^{(n)} \vec{k}_s^T \end{aligned} \quad (9)$$

이고, 여기서 \mathbb{K} 는

$$\begin{aligned} \mathbb{K} &\equiv \begin{bmatrix} k(\vec{x}_1, \vec{x}_1) & k(\vec{x}_2, \vec{x}_1) & \dots & k(\vec{x}_L, \vec{x}_1) \\ k(\vec{x}_1, \vec{x}_2) & \ddots & \dots & \vdots \\ \vdots & k(\vec{x}_j, \vec{x}_i) & \ddots & \vdots \\ k(\vec{x}_1, \vec{x}_L) & \dots & \dots & k(\vec{x}_L, \vec{x}_L) \end{bmatrix} \equiv \begin{bmatrix} \vec{k}_1 \\ \vec{k}_2 \\ \vdots \\ \vec{k}_L \end{bmatrix} \\ &= [\vec{k}_1^T \ \vec{k}_2^T \ \dots \ \vec{k}_L^T] \end{aligned} \quad (10)$$

이기 때문에 커널 공간에서 식(9) 유사성 탐색 법칙은

$$\begin{aligned} i(\vec{x}_s) &= \arg \min_j k(\vec{x}_s, \vec{x}_s) + \vec{\alpha}_j^{(n)} \mathbb{K}^T \vec{\alpha}_j^{(n)} - 2\vec{\alpha}_j^{(n)} \vec{k}_s^T \\ j &= 1, 2, \dots, l \\ s &= \text{selected sample index} \end{aligned} \quad (11)$$

와 같이 정리될 수 있다.

4.2 갱신

(그림 4)에서와 같이, 샘플 \vec{x}_s 가 적절한 커널 함수에 의해 커널 공간에 존재하는 $\vec{\Phi}$ 로 변환되고, SOM의 알고리즘에 의해 다시 토폴로지가 규정된 특징 공간의 $i(\vec{x})$ 로 할당된다. 따라서 식(2)의 Kohonen SOM은

$$\begin{aligned} \vec{w}_j(n+1) &= \vec{w}_j(n) + \eta(n) h_{j,i(\vec{x}_s)}(n) (\vec{\Phi}(\vec{x}_s) - \vec{w}_j(n)), \\ \vec{w}_j \text{ and } \vec{\Phi} &\in \mathbb{R}^M \end{aligned} \quad (12)$$

로 변환되며, 여기서 $\vec{\Phi}(\vec{x})$ 는 원 공간의 학습 데이터를 식 (7)을 만족하는 커널 공간으로 변환하는 함수이다.

식(12)에서 \vec{w}_j 의 개념은 비슷한 패턴 무리들의 대표값으로 해석할 수 있기 때문에

$$\begin{aligned} \vec{w}_j(n) &= \sum_{k=1}^L \alpha_{j,k}^{(n)} \vec{\Phi}(\vec{x}_{k,j}) \\ j &= 1, 2, \dots, l \\ k &= 1, 2, \dots, L \end{aligned} \quad (13)$$

와 같이 주어진 입력 패턴들에 대한 커널 공간 데이터의 선형 결합으로 사용해도 무방하다. 이때 l 은 특정 지도의 노드 개수이며, L 은 학습 패턴의 개수이다.

식(13)를 식(12)에 대입하고, 양변에 $\vec{\Phi}(\vec{x}_v), v \in 1, 2, \dots, L$ 의 내적을 적용하면

$$\begin{aligned} & \sum_{k=1}^L \alpha_{j,k}^{(n+1)} \vec{\Phi}(\vec{x}_k) \cdot (\vec{x}_v) \\ &= (1 - \eta(n) h_{j,i(\vec{x}_s)}(n)) \sum_{k=1}^L \alpha_{j,k}^{(n)} \vec{\Phi}(\vec{x}_k) \cdot (\vec{x}_v) \\ & \quad + \eta(n) h_{j,i(\vec{x}_s)}(n) \vec{\Phi}(\vec{x}_s) \cdot (\vec{x}_v) \end{aligned} \quad (14)$$

를 얻을 수 있다.

식(6)에 의해 커널 함수 $k(\vec{x}_k, \vec{y}_v)$ 는 $k(\vec{x}_k, \vec{x}_v) = \vec{\Phi}(\vec{x}_k) \cdot \vec{\Phi}(\vec{x}_v)$ 라는 사실을 이용하면, 식(14)은 다음과 같이 행렬을 이용한 형태로 표현할 수 있다.

$$\begin{aligned} & \vec{\alpha}_j^{(n+1)} \mathbb{K} \\ &= (1 - \eta(n) h_{j,i(\vec{x}_s)}(n)) \vec{\alpha}_j^{(n)} \mathbb{K} + \eta(n) h_{j,i(\vec{x}_s)}(n) \vec{k}_s \end{aligned} \quad (15)$$

이때 $\vec{\alpha}_j^{(n)} \equiv [\alpha_{j,1}^{(n)}, \alpha_{j,2}^{(n)}, \dots, \alpha_{j,L}^{(n)}]$ 이며, \vec{k}_s 는 \mathbb{K} 의 s 번째 열벡터를 의미하기 때문에 $\vec{k}_s = [0 \ 0 \ \dots \ 1 \ \dots \ 0] \mathbb{K}$ 와 같이 s 번째 값만이 1인 벡터와의 내적 형태로 치환이 가능하다.

$\vec{k}_s = [0 \ 0 \ \dots \ 1 \ \dots \ 0] \mathbb{K}$ 사실을 이용하여 식(15)을 정리하면 최종적으로 커널 공간에서의 kSOM의 갱신 학습식을 얻을 수 있다.

$$\begin{aligned} & \alpha_{j,s}^{(n+1)} \\ &= (1 - \eta(n) h_{j,i(\vec{x}_s)}(n)) \alpha_{j,s}^{(n)} + \eta(n) h_{j,i(\vec{x}_s)}(n) \end{aligned} \quad (16)$$

커널 공간 해석 기법을 통해 유도한 갱신 학습식 식(11)과 유사성 탐색식 식(16)은 커널 함수 ϕ 에 의해 초 고차원의 공간으로 변환된다. 이에 따라 원 공간의 학습 패턴 데이터가 변환된다 하더라도, 실제 학습을 하는 과정에서 ‘차원의 저주’는 발생하지 않음을 확인할 수 있다.

5. 수렴성 증명

커널 공간에서 새롭게 정의된 식(11)과 식(16)을 기반으로 하는 kSOM 알고리즘의 궁극적인 목적은 주어진 입력데이터의 분포를 커널 공간에 대응시키고, 초 고차원의 커널 공간에서의 구조적 유사성을 SOM의 논리 공간에 투영하여 클러스터링을 수행하는 것이다. 즉, Kohonen의 학습 법칙과 마찬가지로 반복적인 학습과정을 통해 주어진 토폴로지 상에서 각 노드의 대표값이 특정 기대값에 수렴하여야 한다.

따라서 새롭게 정의된 kSOM의 학습식에 대한 수렴성을 검증할 필요가 있다.

수렴성 증명을 위해 문제를 단순화 시키면, 주어진 학습 패턴에 대해 특정 지도상의 임의의 노드는 학습이 되거나 유지된다. 즉 임의의 노드가 학습되는 경우라면 식(16)은

$$\alpha_{j,s}^{(n+1)} = (1 - C)\alpha_{j,s}^{(n)} + C, \quad C = \eta(n)h_{j,i(\vec{x}_n)}(n) \quad (17)$$

와 같이 해석될 수 있으며, 임의의 노드가 이웃 영역에 포함되지 않아 학습 되지 않는다면 식(16)은

$$\begin{aligned} \alpha_{j,s}^{(n+1)} &= (1 - C)\alpha_{j,s}^{(n)} + C, \\ C &= \eta(n)h_{j,i(\vec{x}_n)}(n) = 0 \end{aligned} \quad (18)$$

과 같이 해석 될 수 있다. 이때 $\alpha_{j,s}^{(n+1)}$ 는 스칼라 값이기 때문에 이를 α_{n+1} 로 치환하고 식(17)과 식(18) 통합하면

$$\begin{aligned} \alpha_{n+1} &= (1 - C)\alpha_n + C \\ 0 \leq C &= \eta(n)h_{j,i(\vec{x}_n)} < 1 \end{aligned} \quad (19)$$

와 같이 단순화 시킬 수 있다.

일반적으로 α_n 에 대해

$$\lim_{n \rightarrow \infty} \alpha_{n+1} - \alpha_n = 0 \quad (20)$$

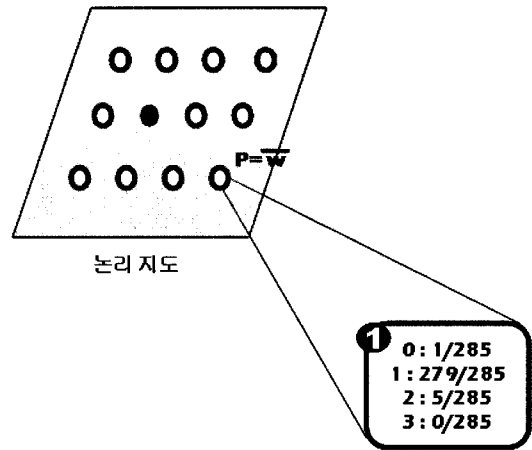
를 만족하기만 한다면 수렴한다고 말한다.

따라서 식(19)을 정리하면 $\alpha_{n+1} - \alpha_n = C(1 - \alpha_n)$ 와 같기 때문에 식(19)의 수렴성 증명은 결국

$$\lim_{n \rightarrow \infty} C(1 - \alpha_n) = 0 \quad (21)$$

를 증명하는 것과 같은 문제이다.

여기서 $C=0$ 이면 식(21)이 0이 됨은 자명하다. 또한 $0 < C < 1$ 인 경우 α_n 은



(그림 9) 논리 지도의 각 노드와 투표 벡터의 관계

$$\alpha_n = (1 - C)^n \alpha_0 + C(1 - C)^{n-1} + \dots + C(1 - C) + C \quad (22)$$

와 같이 정리할 수 있기 때문에

$$\lim_{n \rightarrow \infty} \alpha_n = \alpha_0 \lim_{n \rightarrow \infty} (1 - C)^n + C \frac{1}{1 - (1 - C)} = 1 \quad (23)$$

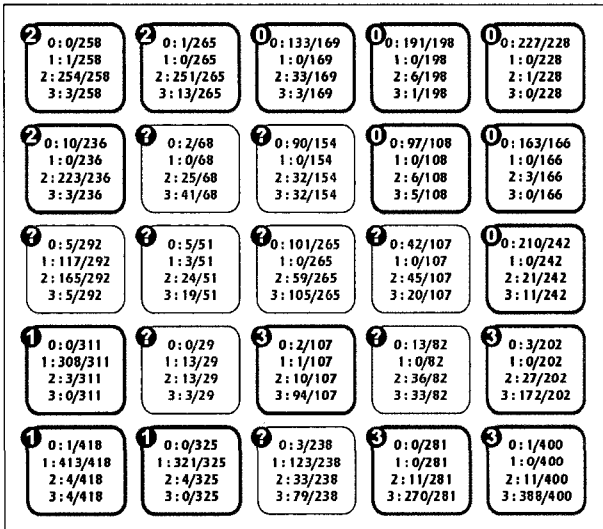
에 의해 식(20)은 0으로 수렴한다.

즉, 하나의 노드가 계속적으로 경쟁에서 승리하여 선택 학습되거나, 경쟁에서 항상 밀려나 학습이 전혀 이루어지지 않더라도 발산하거나 진동하는 경우는 발생하지 않게 됨을 확인할 수 있다.

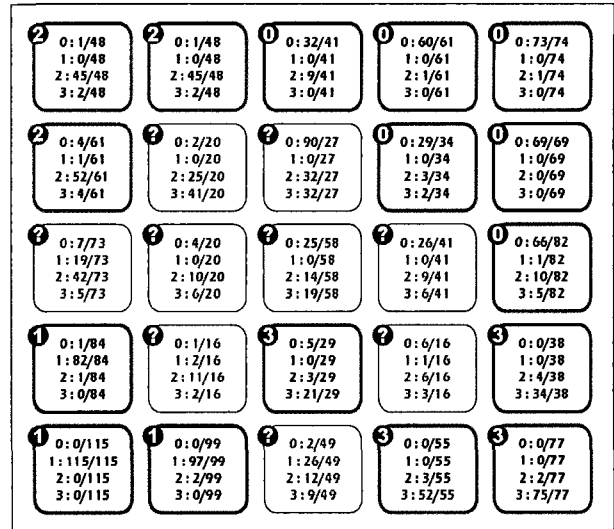
6. 클러스터링 실험

본 장에서는 전통적인 Kohonen의 SOM과 본 논문에서 제안한 kSOM의 성능을 비교하는 실험을 수행한 결과를 분석한다. 사용한 데이터는 CEDAR DB의 0~9 숫자에 대한 필기체 이미지와 TI46DB의 0~9까지의 음성데이터이다. 필기체 숫자는 18000개의 학습 패턴과 3000개의 테스트 패턴으로 이루어져 있으며 12*12 크기의 픽셀과 각 픽셀은 0~15까지의 정수 값을 갖도록 정규화 하여 실험하며, 음성 데이터는 0~9 영어 숫자 발음과 Enter, Space등의 10 명령어, A~Z의 알파벳 발음으로 이루어져 있으며 MFCC에 기반한 음성 특징 추출 방법으로 이용하여 64프레임으로 10ms 씩 이동하며 각 프레임은 30ms로 이루어지도록 하였다. 이때 각 프레임으로부터 12개씩의 MFCC 계수를 추출하여 768차원(64프레임*12MFCC계수)을 구성하여 실험한다.

학습에 있어 특징 지도의 크기는 중요한 요소 중 하나이다. 장난감 문제의 경우 참여하는 클래스의 개수가 4이기 때문에 각 노드가 하나의 클래스를 표현하는 SOM의 경우 최소 4개의 노드, 즉 2*2의 특징 지도가 필요하다. 그러나 노드의 개수가 적으면, 클러스터간의 경계 영역에서의 중첩이 심하게 발생하기 때문에 Kohonen SOM 뿐만 아니라 kSOM의 경우 역시 그 성능을 확인하기 어렵다. 둘 간의 성능 비교를



(그림 5) Kohonen SOM에 기반한 트레이닝 패턴 학습결과



(그림 6) Kohonen SOM에 기반한 테스트 패턴 인식 결과

위해 정상적인 학습이 이루어질 수 있는 최소한의 특징 지도 크기를 설정할 필요가 있으며, 2×2, 3×3, 4×4형태로 점진적으로 지도의 크기를 확대하며 실험한 결과 5×5이상인 경우 Kohonen SOM과 kSOM간의 성능차가 최대화되기 시작하므로, 실험은 5×5환경에서 수행한다.

기존의 Kohonen SOM을 이용하여 트레이닝 패턴에 대해 클러스터링을 수행하고, 그 결과 각 노드를 대표하는 값을 선정하여, 임의의 입력에 대해 클러스터링 결과의 신뢰도를 테스트하고, 동일한 문제에 대해 kSOM을 적용하여 클러스터링을 수행함으로써 기존 결과와 비교한다. 이때 인식률을 계산하는 방법은 다음과 같다.

논리 지도의 각 노드는 학습에 참여한 클래스 개수 만큼의 투표 벡터(voting vector)를 갖는다.

학습이 종료된 시점에 클래스 K로부터 추출된 데이터가 논리지도 노드 P에 대응되는 \vec{w} 와 유사도가 가장 높을 경우 노드 P에 대응되는 투표 벡터 \vec{v}_P 의 $\vec{v}_{P,K}$ 항목을 1 증가시킨다. 모든 학습 패턴에 대하여 유사 노드를 탐색하고 각 원소를 증가시킴으로써 하나의 완성된 투표 벡터 논리 지도(그림 5)를 획득할 수 있다. 이때 각 노드의 투표 벡터의 원소가 최대가 되는 인덱스를 해당 노드의 대푯값으로 설정함으로써 클러스터링이 종료된다. (그림 5)에서는 이 결과를 ①, ①, ②, ③으로 표기하고, 복잡도가 높아 특정 노드를 대푯값으로 설정하기 어려운 경우 ②으로 표기한다.

테스트 패턴에 대해서는 완성된 투표 벡터 지도에 테스트 패턴을 적용함으로써 테스트 패턴에 가장 가까운 노드 \vec{w} 의 대푯값과 테스트 패턴이 추출된 클래스의 인덱스 정보가 일치할 경우 인식률은 100%가 되고, 일치하지 않을 경우 감소하게 된다.

6.1 Kohonen SOM을 이용한 클러스터링

(그림 5), (그림 6)은 CEDAR DB 중 0~3까지의 장난감 문제에 대해 기존의 Kohonen SOM을 적용한 결과를 시각

적으로 표현한 결과이다.

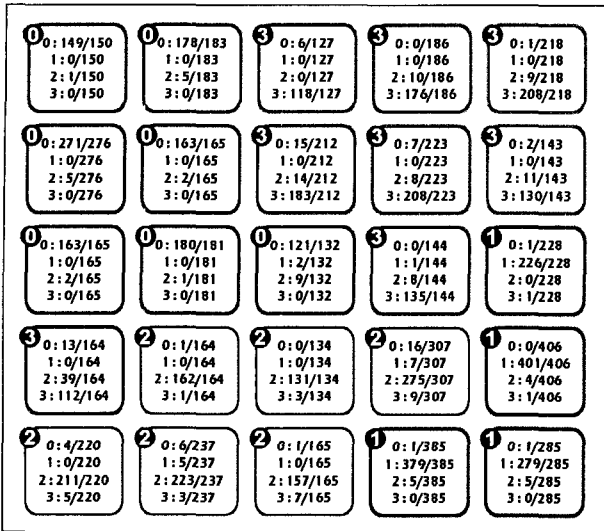
①과 ②, ①과 ③ 등은 수학적으로 계산을 하지 않더라도 근접한 공간에 존재할 것임을 예상할 수 있다. 실제 학습결과를 보면 0-3, 0-2, 1-3, 1-2 쌍 특징 지도에서 서로 접하는 공간에 놓여있다. 그러나 원 데이터의 유사성으로 인한 문제의 복잡도가 전혀 해결되지 않았기 때문에 각 클래스가 접하는 부분을 비롯한 중앙 영역에서 클러스터링이 거의 이루어지지 않음을 확인할 수 있다.

6.2 kSOM을 이용한 클러스터링

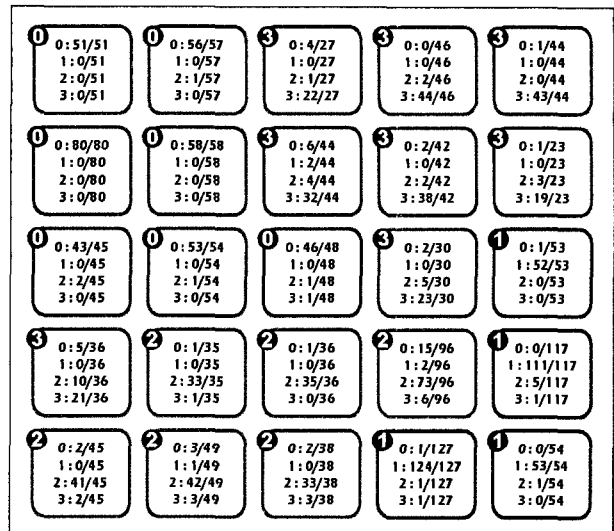
(그림 7), (그림 8)은 본 논문에서 제안한 kSOM을 이용하여 클러스터링을 수행하고 인식률을 산출한 결과이다. 전통적 Kohonen SOM의 경우 중앙 부분에서의 모호성으로 인해 클러스터링 자체가 잘 수행되지 않고, 대표값을 세울 수 없기 때문에 인식률을 산출할 없다. 반면, kSOM의 경우 클래스와 클래스가 접하는 경계 영역에서의 모호성이 해결되어 명확하게 분류 작업이 수행된다.

(그림 5)와(그림 7)의 결과에서 시계방향으로(그림 5)는 2-0-3-1의 분포이고, (그림 7)은 0-3-1-2인 이유는 무작위로 할당된 초기값의 분포에 따라 다르게 나타나는 것으로 일견 다른 결과처럼 보이나 시계방향으로 회전을 시키면 동일한 관계를 표현하고 있음을 알 수 있다.

기본적으로 (그림 5)와 (그림 7)은 동일한 문제에서 시작하였지만, 내부적으로 데이터의 분포는 완전히 다르다. 즉, 둘은 서로 다른 데이터를 사용한 SOM이라고 봐도 무방하다. 제안한 알고리즘의 목표가 N차원 공간의 데이터를 M차원 공간으로 전사(mapping)하여 데이터들 간의 분포에 변화가 발생하기를 기대하는 방법이다. 이때 데이터들의 분포는 사용한 커널 함수에 따라 다르기 때문에 정확히 묘사할 수 없으나, 고 차원으로의 전사가 데이터 분포에 성김 특성을 부여하여 Kohonen SOM의 경우에 비해 상대적으로 명확한 클러스터링 결과를 보인다.



(그림 7) kSOM에 기반한 트레이닝 패턴 학습결과



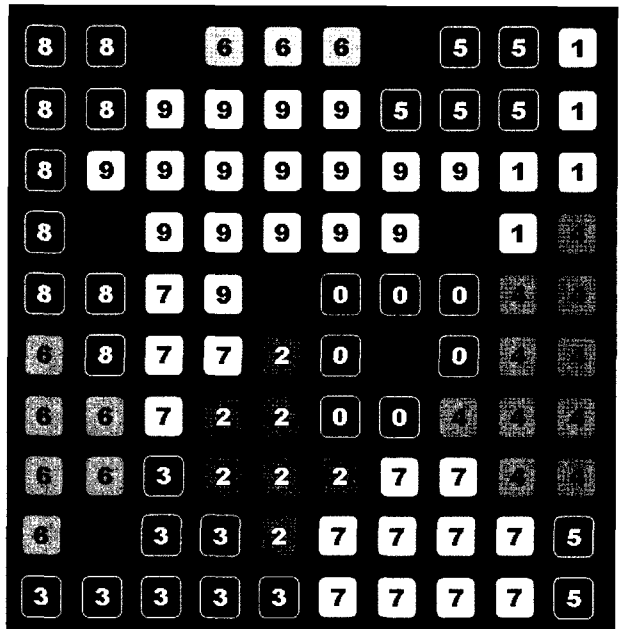
(그림 8) kSOM에 기반한 테스트 패턴 인식 결과

<표 1> kSOM 클러스터링 결과(CEDAR DB : 0~3)

	학습 패턴 결과	테스트 패턴 결과
클래스 0	94.38%(1227/1300)	89.40%(388/434)
클래스 1	98.84%(1285/1300)	98.55%(340/345)
클래스 2	88.77%(1154/1300)	88.18%(257/296)
클래스 3	97.69%(1270/1300)	93.08%(242/260)
전체	94.92%(4936/5200)	91.91%(1233/1335)

<표 2> kSOM 클러스터링 결과(CEDAR DB : 0~9)

클래스	0	1	2	3	4
학습결과	85.23%	96.46%	90.69%	81.23%	66.92%
클래스	5	6	7	8	9
학습결과	84.08%	92.08%	63.46%	81.85%	73.23%



(그림 9) kSOM에 기반한 TI46DB 학습 결과

또한 kSOM의 경우 데이터의 분포뿐만 아니라 각 노드에서 “Winner Takes All”의 원칙을 충분히 따르고 있기 때문에 인식기로서도 충분히 좋은 성능을 보인다.

<표 1>은 이와 같은(그림 7)과(그림 8)의 결과로 부터 산출된 클러스터링 신뢰도와 해당 클러스터링 결과에 기반한 테스트 패턴의 인식률을 산출한 결과이다. 클래스 1과 클래스 3의 경우 95%가 넘는 높은 학습률을 보이며 전체의 학습 신뢰도 역시 94.92%에 이른다. 본 논문에서 제안한 비교사학습의 성능 향상을 위한 접근법처럼, 문제의 복잡도를 해결할 수만 있다면 비교사학습을 통해서도 교사학습에 준하는 수준의 높은 인식률을 얻을 수 있을 것으로 예상할 수 있다.

반면에 기존의 Kohonen SOM을 적용한 클러스터링 실험

의 경우, (그림 5), (그림 6)와 같이 중앙 영역에서의 복잡도가 너무 높아 어느 하나의 값을 대표값으로 설정하고 인식률을 계산하는 것 자체가 무리가 있기 때문에 인식률을 산출할 수 없다.

<표 2>는 CEDAR DB의 0~9까지의 모든 데이터에 대해 비교사학습을 수행한 결과로, 문제의 크기가 커졌기 때문에 상대적으로 학습률 및 인식률이 떨어졌으나 4, 7, 9와 같이 사람이 직접 판단하더라도 문제의 소지가 있는 경우를 제외하고 80%~90%에 이르는 매우 높은 인식률을 보인다. 또한 문제의 특성에 맞는 커널 함수를 선택함으로써 인식률은 더욱 향상될 수 있다. 그러나 Kohonen SOM을 이용한 학습의 경우 학습 자체가 거의 이루어지지 않았기 때문에 인식률은 비교할 수 없다.

<표 3> kSOM vs. Kohonen SOM 클러스터링 결과(TI46 DB : 0~9)

클래스	0	1	2	3	4
kSOM	94.71%	98.36%	95.50%	98.03%	93.39%
SOM	93.69%	97.33%	95.50%	91.03%	89.99%
클래스	5	6	7	8	9
kSOM	76.23%	83.34%	97.55%	86.49%	96.48%
SOM	81.41%	81.73%	97.12%	85.21%	90.37%

(그림 9)와 <표 3>은 TI46 DB의 0~9까지의 영어 숫자 발음을 대상으로 Kohonen SOM과 kSOM을 적용한 실험 결과이다. TI46DB의 경우 데이터 자체가 갖는 특성이 선형적으로 매우 잘 분리되는 형태를 보이기 때문에 Kohonen SOM과 kSOM의 결과가 큰 차이를 보이지 않는다. 오히려 선형적으로 분리가 쉬운 문제에 대해 복잡한 커널 함수를 적용할 경우 성능이 하락하는 문제점을 나타냈다. 결국 kSOM의 성능을 결정하는 최대 요인은 어떤 커널 함수를 선택하여 원 공간의 데이터를 커널 공간에 효율적으로 대응 시키는가에 있다고 할 수 있다.

실제 실험에서는 Gaussian RBF 커널 함수와, Polynomial 커널 함수를 사용하여 실험하였으며 CEDAR DB의 경우 Gaussian RBF의 경우 좋은 성능을 나타낸 반면, TI46 DB의 경우 차원이 낮은 Polynomial 커널 함수일 경우 좋은 성능을 보였으며, Polynomial의 차원을 높일수록 급격히 성능이 하락하였다.

실험은 AMD 32bit 프로세서, 512M 메모리, 리눅스 OS, MATLAB 프로그래밍 환경에서 수행하였으며, 학습률은 0.1~0.0001로 변화하며 초기 node값은 랜덤으로 할당하고 이웃 함수는 rectangular를 사용하였다. Threshold를 0.001로 설정하였을 때 CEDAR DB의 4 클래스 문제의 경우 Kohonen SOM은 8분, kSOM의 경우 5분정도의 시간이면 충분히 학습이 이루어졌다. 실제로 학습률의 변화 정도와 초기값의 분포 MATLAB 프로그래밍 스킬에 따라 매우 큰 차이를 보이기 때문에 정량적으로 성능을 정의하기는 어렵다.

학습 특성은 Kohonen의 경우 선형(linear)적 특성을 보이고 kSOM의 경우 로그(log)적 특성을 나타내며, 동일한 시간에 kSOM이 Kohonen SOM보다 더 빠른 속도로 수렴한다.

7. 결론 및 향후 연구

본 논문은 기존의 Kohonen SOM에 커널 공간 해석기법을 접목함으로써 커널 공간에서의 kSOM 알고리즘을 제안하고, 동일한 복잡도를 갖는 문제에 대해 기존의 알고리즘과 비교 실험을 통해 성능의 향상을 검증하였다. 이는 기존의 비교사학습의 전통적 문제인 문제의 복잡도에 따른 클러스터링 성능 하락의 단점을 해결할 수 있는 획기적인 접근법이다.

본 논문에서는 비교사학습 이론중 하나인 SOM을 대상으로 커널 공간 해석기법을 적용하였으나, 향후 MLP, RBF 등

다양한 알고리즘에 대한 커널 공간 확장을 시도함으로써 추가적인 성능향상이 예상된다.

그러나 TI46 DB의 실험 결과와 같이 데이터 자체가 갖는 특성에 부적합한 커널 공간으로의 변형은 오히려 성능을 하락시키는 단점을 갖는다. 따라서 문제의 특성에 맞는 커널 함수 선정에 대한 연구를 수행함으로써 커널 공간 해석의 신뢰도를 높이기 위한 연구가 필요하다.

참고 문헌

- [1] V.N. Vapnik, "The nature of statistical learning theory", Springer Verlag, New York, 1995.
- [2] V.N. Vapnik, "Statistical Learning Theory", Wiley, New York, 1998.
- [3] K.-R. Muller, et al., "An Intoduction to Kernel-Based Learning Algorithms", IEEE Transactions on Neural Netwoks, Vol.12, No.2, March, 2001.
- [4] C.J.C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition", Data Mining and Knowledge Discovery, Vol.2, pp.121-167, 1998.
- [5] Bernhard Scholkopf, et. al., "Input Space vs. Feature Space in Kernel Based Methods", IEEE Transactions on Neural Netwoks, Vol.10, No.5, pp.1000-1017, 1999.
- [6] B. Scholkopf, "Support vector learning", Oldenbourg Verlag, Munich, 1997.
- [7] G. Baudat and F. Anouar, "Generalized discriminant analysis using a kernel approach", Neural Computation, Vol.12, No.10, pp.2385-2404, 2000.
- [8] B. Scholkopf, A.J. Smola, and K.-R. Muller, "Nonlinear component analysis as a kernel eigenvalue problem", Neural Computation, Vol.10, pp.1299-1319, 1998.
- [9] Y.A. LeCun, et. al., "Learning algorithms for classification: A comparison on handwritten digit recognition", Neural Networks, 1995.
- [10] Sungmoon Cheong, Sang Hoon Oh and Soo-Young Lee, "Support Vector Machines with Binary Tree Achitecture for Multi-Class Classification", Neural Information Processing-Letters and Reviews, Vol.2, No.3, March, 2004.
- [11] Simon Haykin, "Neural Networks", Prentice Hall, 2nd Ed., 1999.

정 성 문

e-mail : insomnia@etri.re.kr
 2002년 KAIST 전자전산학과(학사)
 2004년 KAIST 전자전산학과(석사)
 2004년~현재 국가보안기술연구소 연구원
 관심분야: 컴퓨터 보안, 데이터 마이닝, 기계학습 이론 등

김 기 범

e-mail : kibom@etri.re.kr

1994년 제주대학교 정보공학과(학사)

1996년 고려대학교 전산학과(석사)

2001년 고려대학교 전산학과(박사)

2004년(주)이씨오 개발 부장

2004년~현재 국가보안기술연구소 연구원

관심분야: 컴퓨터 보안, 정적분석, 유비쿼터스컴퓨팅 등

홍 순 작

e-mail : hongsj@etri.re.kr

1989년 숭실대학교 전산학과(공학사)

1991년 숭실대학교 전산학과(공학석사)

2005년 충남대학교 컴퓨터과학과(이학박사)

1991년~2000년 국방과학연구소 선임연구원

2000년~현재 국가보안기술연구소 팀장

관심분야: 컴퓨터 보안, Ad-hoc 네트워크, 무선 네트워크 등