퍼지 및 지능시스템학회 논문지 2004, Vol. 14, No. 7, pp. 816-820

고차원 데이터 처리를 위한 SVM기반의 클러스터링 기법

SVM based Clustering Technique for Processing High Dimensional Data

김만선*&**, 이상용***

Man-Sun Kim and Sang-Yong Lee

* 한국표준과학연구원(KRISS) 정보전산그룹 ** 공주대학교 컴퓨터공학과

*** 공주대학교 정보통신공학부 교수(교신저자)

요 약

클러스터링은 데이터 집합을 유사한 데이터 개체들의 클러스터들로 분할하여 데이터 속에 존재하는 의미 있는 정보를 얻는 과정이다. 클러스터링의 주요 쟁점은 고차원 데이터를 효율적으로 클러스터링하는 것과 최적화 문제를 해결하는 것이다. 본 논문에서는 SVM(Support Vector Machines)기반의 새로운 유사도 측정법과 효율적으로 클러스터의 개수를 생성하는 방법을 제안한다. 고차원의 데이터는 커널 함수를 이용해 Feature Space로 매핑시킨 후 이웃하는 클러스터와의 유사도를 측정한다. 이미 생성된 클러스터들은 측정된 유사도값과 Δd 임계값에 의해서 원하는 클러스터의 개수를 얻을 수 있다. 제안된 방법을 검증하기 위하여 6개의 UCI Machine Learning Repository의 데이터를 사용한 결과, 제시된 클러스터의 개수와 기존의 연구와 비교하여 향상된 응집도를 얻을 수 있었다.

Abstract

Clustering is a process of dividing similar data objects in data set into clusters and acquiring meaningful information in the data. The main issues related to clustering are the effective clustering of high dimensional data and optimization. This study proposed a method of measuring similarity based on SVM and a new method of calculating the number of clusters in an efficient way. The high dimensional data are mapped to Feature Space ones using kernel functions and then similarity between neighboring clusters is measured. As for created clusters, the desired number of clusters can be got using the value of similarity measured and the value of Δd. In order to verify the proposed methods, the author used data of six UCI Machine Learning Repositories and obtained the presented number of clusters as well as improved cohesiveness compared to the results of previous researches.

Key words : SVM, 클러스터링(Clustering), 고차원 데이터(high dimensional data)

1. 서 론

클러스터링이란 한 군집에 포함된 데이터들이 유사한 성질을 갖도록 데이터들을 묶는 것이다. 패턴인식, 영상처리 등의 공학 분야에 널리 적용되고 있을 뿐 아니라, 최근 많은 관심의 대상이 되고 있는 데이터 마이닝의 주요 기술로서 활발히 응용되고 있다.

클러스터링은 입력 데이터 집합을 유사한 관찰값들의 군집들로 구분하여 데이터 집합 속에 존재하는 의미 있는 정보를 얻는 과정이다[1]. 즉, 군집내의 유사성은 최대화하고, 군집들 간의 유사성은 최소화하도록 데이터 집합을 분할하는 것이다[2].

기존의 클러스터링 방법에서 이용되었던 신경망 기반의 방법, k-means 와 같은 알고리즘들은 지역적 최적해에 수렴하는 것과, 사전에 클러스터 개수를 미리 결정해야 하는 문제점을 갖고 있다. 또한 지역적 최소값(local minimum)을 피하기 위한 초기화 작업이 거의 경험적으로 이루어지며, 수렴속도의 지연 그리고 근사화 및 수렴율에 영향을 미치는 커널 함수(kernel function)의 선택 등이 여전히 어려운 문제로 남아 있다. 그리고 FCM(Fuzzy C-means) 기반의 클러스터링 알고리즘 또한 클러스터의 표현이 하나의 점, 선, 타원 등으로 고정되며 무정형의 클러스터는 제대로 표현될 수 없다. 특히, 고차원 데이터를 다루는 문제에서는 원 데이터가 왜곡되지 않도록 차원을 축소하는 문제가 가장 어렵다. 생성된 클러스터의 수가 많다면 그 중에서 바람직하지 못한 클러스터가 생성될 수도 있다. 이에 관한 기존의 연구에서 고차원의 데이터를 저차원으로 변환하기 위해 2차원 프로젝션을 이용하여 데이터의 잡음을 효과적으로 제거할 수는 있었으나 알고리즘의 효율성과 최적화 문제를 해결하지 못했다.

본 논문에서는 위와 같은 문제점들을 극복하기 위하여 SVM을 이용하여 지역적 최적해에 수렴하는 것을 방지하고, 불필요하게 생성된 클러스터들을 Δd 임계값을 통해 병합하거나 제거하여 빠르고 정확한 거리 측정 방법을 제시한다. SVM의 특성상 입력 벡터의 차원에 무관하므로 높은 차원의 데이터에 대하여 좋은 수행 결과를 얻을 수 있다.

본 논문의 나머지 부분은 다음과 같이 구성되어 있다. 2절 관련 연구에서는 기존의 연구들의 문제점들을 알아보고, 3절에서는 SVM에 대한 특징을 알아보고, 제안하는 방법을 논한다. 4절에서는 몇 개의 데이터 집합들에 대한 결과를 제시

접수일자 : 2004년 8월 27일
완료일자 : 2004년 11월 4일.

하고, 다른 알고리즘과 비교 분석하여 제시한 알고리즘의 우수성을 보인다.

2. 관련 연구

잘 알려진 대부분의 클러스터링 알고리즘들은 고차원 데이터 공간에서 클러스터를 탐색하는데 실패하는 경향이 있다. 이와 같은 문제점은 데이터 점들이 갖는 자체의 희소성(sparsity)으로 인한 것으로 차원 전체가 클러스터 탐색에 관련되지 않을 수 있다는 것이다. 이와 같은 문제를 다루는 방법으로서 연관성 있는 차원을 선택하고 대응하는 부분공간에서 클러스터를 탐색하고자 하는 연구가 진행되고 있다. 관련성 있는 차원만을 고려하는 알고리즘으로는 CLIQUE, PROCLUS 등이 있으며, 이들은 데이터 공간상의 점들은 전체차원의 부분집합 즉, 일부차원에 대하여 보다 효과적으로 클러스터를 탐색 할 수 있다.

이 밖의 연구로는 CLARANS, DBSCAN, OPTICS, BIRCH 이 있는데, 고차원 데이터의 공간에서 거리를 기반으로 모든 차원을 고려하여 클러스터링을 수행한다.

CLARANS(Clustering Large Applications based upon Randomized Search) 는 전체 데이터 집합에서 임의의 샘플 데이터를 뽑아 K개의 클러스터를 대표할 수 있는 k-medoid 들의 집합인 그래프를 검색하는 과정으로 처음으로 소개된 분할기법 알고리즘이다. 데이터 집합의 패턴이나 분포도가 복잡해질수록 충분한 공간 정보를 제공하지 못하고, 임의 탐색을 사용하므로 최적의 클러스터링 결과를 보장할 수 없다[3][4].

DBSCAN은 잡음을 고려하는 대표적인 알고리즘으로서 클러스터의 밀도기반 개념을 이용하여 임의 형태의 클러스터를 탐사한다. 그러나 DBSCAN은 R*-트리 기반으로 구현되므로 고차원 공간에서 R-트리 기반 인덱스의 성능 저하로 말미암아 효율적으로 수행하지 못한다. 모든 인덱스-기반 방법들이 그렇듯이 DBSCAN도 효율성에 있어서 심각한 성능저하와 잡음을 포함하는 데이터 집합에 대해 효과성 문제를 보이고 있다[5].

BIRCH(Balanced Iterative Reducing and Clustering using Hierarchies)는 잡음 데이터를 다룬 첫 번째 알고리즘으로서, 클러스터를 탐사하거나 잡음으로부터 클러스터의 구별을 위해 몇 가지 경험적 정보를 사용한다. BIRCH는 클러스터 특징을 저장하는 균형 트리인 CF-트리라 불리는 계층적 데이터 구조를 사용한다. 아직까지 BIRCH는 가장 효율적인 알고리즘의 하나이며 데이터베이스를 오직 한번 스캔하는데, 이것은 고차원 데이터에 대해서도 그렇다. 그러나 BIRCH는 요약될 데이터 항목을 정의하는데 반지름이나 지름 등의 유사성 개념을 사용하므로 오직 구형의 클러스터만을 탐색하는 한계가 있으며, 또한 데이터의 입력순서에 민감한 단점을 갖고 있다[6].

Lee가 제안한 CLIP(CLustering based on Incremental Projection)은 k-차원 데이터 공간에서 한 차원씩 점진적으로 프로젝션하면서 클러스터를 탐색한다. 즉 하나의 차원에서 시작하여 밀집영역을 구한 뒤, 그 차원의 밀집영역에 의존적인 그 다음 차원의 밀집영역을 찾아내는 방법으로 최종 k차원까지 반복해 나가는 것이다. 클러스터가 존재할 후보영역을 제공함으로써 클러스터 탐색 공간을 크게 감소시킬 수 있으며, 데이터 점들의 평균값을 중심으로 즉 공간 지역성을 고려하여 점들을 탐색해 나감으로서 보다 효과적으로 클러스터 형태를 식별할 수 있다. 효율성을 위해 다양한 실험을 통한 결과가 입증되지 않았다[7].

Jang은 고차원 데이터의 잡음을 제거하는 클러스터링 기법을 제안하였다. 고차원의 데이터를 저차원으로 변환하기 위해 반복적인 2차원 프로젝션을 이용하여 데이터의 잡음을 효과적으로 제거할 수는 있었으나 알고리즘의 효율성과 최적화 문제를 해결하지 못했다[8].

3. SVM 기반의 새로운 클러스터링 방법

일반적인 SVM(Support Vector Machine)의 특징과 커널 함수의 매핑을 통한 차원 문제 해결, 클러스터의 크기를 결정짓게 하는 변수를 두어 불필요한 클러스터의 제거, 거리 측정 방법에 대해 논한다.

3.1 SVM 특징

SVM은 고차원 가상 특징 공간을 이용한 통계적 학습 모델이다[9][10]. 기존의 통계적 학습 방법들이 훈련과정에서의 오류를 최소화하는 경험적 오차(Empirical Risk Minization:ERM)를 이용하는 것과는 달리 구조적 오차 최소화 방법(Structural Risk Minization:SRM)을 이용하여 일반화 오차(generalization error)를 감소시킨다.

SVM은 기본적으로 2 클래스 분류기이다. 따라서 그림 1.과 같이 SVM은 최종적으로 두 그룹의 데이터를 분리할 수 있는 최적 분리 경계면(optimal hyperplane)을 구한다. 이 때 최적 분리 경계면과 각 그룹의 가장 근접한 데이터를 support vector라고 하며, 각 그룹의 support vector간의 거리 $2/\|W\|$가 최대가 되는 지점에서 최적 분리 경계면이 설정된다.

아래의 그림 1.은 선형분리가 된 상태이고, 그림 2-a.는 선형분리가 되지 않는 상태이다. 선형분리가 되지 않는 상태는 노이즈가 포함된 도메인일수도 있고, 오분류가 포함된 데이터일수도 있지만 실세계의 대부분 응용 문제는 선형 분리가 불가능하다. 그림 2-b.는 그림 2-a.의 상태에 커널 함수를 이용하여 선형적으로 투영하여 해석할 수 있도록 하며, 각 Feature 사이의 최적의 경계면을 생성한다. 현재 RBF, Polynomial등의 커널 함수가 주로 이용된다. 또한 slack variable를 두어 에러를 허용하고, C 파라미터를 두어 마진과 에러의 트레이드오프를 조절한다.

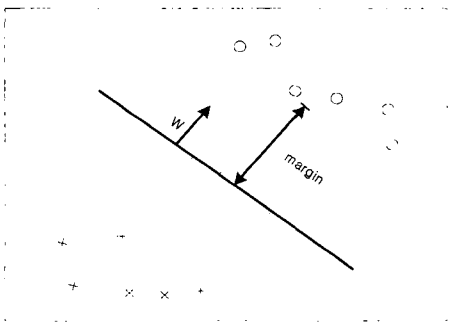

그림 1. 선형 분리되는 문제

3.2 커널 함수의 매핑을 통한 차원 문제 해결

신경망 같은 경우 입력 변수(차원), 즉 input node 수가 많을수록 계산시간이 증가하지만, SVM은 커널함수를 통해 특징 공간으로 매핑할 때 원 데이터의 차원수가 높더라도 그 계산은 단순하다.

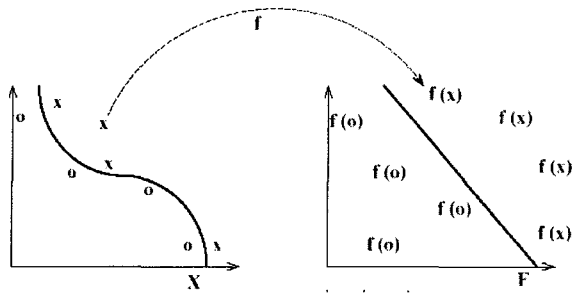

그림 2-a 그림 2-b

RBF(Radial Basis Function)커널을 사용하면 수식 (3.1)과 같이 표현 할 수 있다. 수식 (3.1)에서 σ는 가우시안 윈도우의 너비이다. x는 입력 벡터이고, y는 입력 패턴이다.

$$K(x,y)=\exp(-\| x-y \| 2/2\sigma 2) \quad (3.1)$$

위와 같은 방식으로 진행되기 때문에 입력 차원, input feature수에 영향을 받지 않는다는 장점을 갖고 있다.

주어진 l개의 패턴 쌍(xi, yi) 들에 대하여, 최대 마진 평면 <wx> + b는 다음의 quadratic programming problem을 푸는 것에 의해서 얻어질 수 있다.

minimize <wx>+b <w · w>,

subject to yi(<w · xi>+b)≥1, i=1,…,l,

여기에서 파라미터들인 w, b와 yi=1는 각각 가중치 벡터(weight vector), 바이어스(bias), 클래스 라벨(class label)을 나타낸다.

3.3 SVM을 이용한 클러스터간 거리 측정 방법

일단 초기의 클러스터가 생성된 후 아래의 그림 3처럼 두 클러스터간 최근접한 꼭지점을 사용하여 클러스터간 거리를 나타낸다. 임의의 꼭지점으로 연결하면 클러스터가 확장되거나 결합할 때 잘못된 거리를 나타낼 수 있다.

아래의 그림 4.은 최적의 클러스터간 거리를 구할 수 있다. SVM에 의한 최대 마진 평면의 가중치 벡터 w*와 바이어스 b*를 사용하여, 두 클러스터간 Ai와 Aj사이의 거리는 수식 (3.2)과 같이 표현된다.

SVM의 목표는 $d_{svm}(A^i, A^j)$의 거리를 최대화 하는 것임으로 $| d_{svm}(A^i, A^j) |$의 값이 클수록 좀 더 신뢰성 있는 거리 측정 결과라고 할 수 있다.

$$d_{svm}(A^i, A^j)=2 \cdot \frac{\min(\| w^* x_i + b^* \|)}{w} \quad (3.2)$$

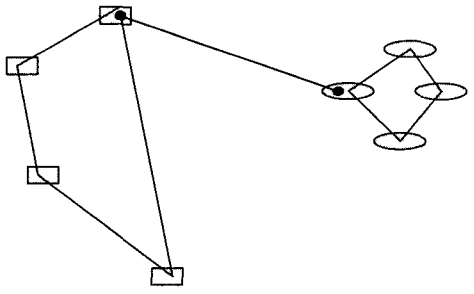

그림 3. 꼭지점을 사용한 클러스터간 거리

3.4 거리기반의 테이블 생성 및 클러스터 결합

클러스터의 크기를 결정짓게 하는 변수를 두어 크기가 결정된 후 필요에 의해 거리가 가까운 클러스터는 병합한다.

SVM을 이용한 클러스터간 거리 측정 방법에 의해 측정된 거리, 즉 $d_{svm}(A^i, A^j)$를 기반으로 각 클러스터들 간의 거리를 정렬하여 테이블로 생성한다. 즉, n개의 클러스터가 이미 형성된 경우, n(n-1)/2 개의 클러스터간 거리를 모두 계산하여 오름차순으로 정렬한 후 테이블을 생성한다.

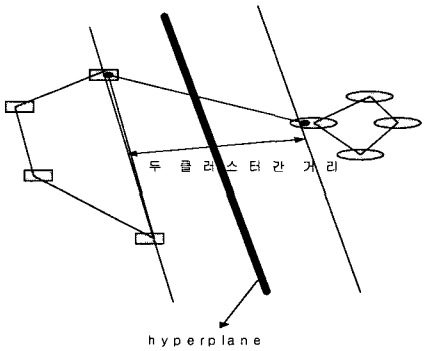

그림 4. SVM을 이용한 거리 측정 방법

또한 Δd 임계값을 주어 적절한 클러스터의 개수를 얻을 수 있다. 따라서 Δd 임계값의 선택이 매우 중요하다. SVM 거리측정 기반으로 생성된 d_{ij}값을 비교하면서 적절한 임계값을 입력해 주어야 하며, 중요한 이유는 그 값에 따라서 클러스터를 병합할 수도 있고 그대로 둘 수도 있기 때문이다. 제안하는 클러스터 기법은 다음과 같이 요약할 수 있다.

```
initialize Δd.
for all cluster do.
        calculate weight vector  w* by SVM.
        set dij = dsvm(Ai, Aj)
store table.
sort cluster in increasing order according
for all cluster of table do to dij.
        if (dij <Δd) then
        merge the cluster
        end if
end for
```

4. 모의 실험 결과

본 논문에서는 제안하는 방법을 검증하기 위하여 실세계 데이터인 6개의 UCI Machine Learning Repository[11]의 데이터를 사용한다. UCI 데이터는 분류나 클러스터를 할때 많이 이용되는 데이터이며 데이터의 수(instances), 특징(attributes), 부류개수(class)가 데이터베이스 안에 모두 제시되어 있습니다. 이 중에서 Car Evaluation Database는 자동차의 각 요구 사양에 맞추어 자동차를 평가해 놓은 것들이다. 요구 사항은 속성들로 포함되어 있고 가격(사는 가격과 유지비)과 기술(편암함과 안정도 등)의 6가지 속성으로 구성되어 있다.

6가지 실험 데이터의 특징은 다음 표 1.과 같다.

표 1. 6가지 실험 데이터의 특징

실험데이터	특징수	샘플수	부류개수
car evaluation	6	1,728	4
covertype	54	581,012	8
diabetes	9	768	2
letter recognition	17	20,000	1
nursery DB	5	12,960	8
solar flare	13	1,389	3

4.1 거리에 기반한 테이블 생성 표 만들기

만약 solar flare 데이터의 경우 최종 3개의 클러스터의 수가 결정되어야 하는데, 이보다 많은 수의 클러스터가 생성되었다면 표 2.와 같은 거리에 기반한 테이블을 생성하여 거리가 가까운 클러스터를 병합하면 된다.

표 2. SVM 거리에 기반한 테이블 생성 표

n:이미 형성된 클러스터의 개수	
n=2	dsvm(Ai, Aj)
n=3	dsvm(Ai, Aj) dsvm(Ai, Ak) dsvm(Aj, Ak)
n=4	dsvm(Ai, Aj) dsvm(Ai, Ak) dsvm(Ai, Al) dsvm(Aj, Ak) dsvm(Aj, Al) dsvm(Ak, Al)

SVM의 커널로는 RBF커널을 사용하였으며 거리의 제곱을 시그마로 나눈값으로 시그마 σ는 1로 설정하였다. 시그마 σ값에 따라서 가우시간의 모양이 평퍼짐한지 뾰족한지가 결정된다. 경험적인 문제가 따르겠지만 여기서는 1의 값을 default로 결정한다.

4.2 응집도 Q값

클러스터링 결과의 성능은 식(4.1)과 같이 Di의 평균인 Q로 측정하였다. Di는 클러스터 i에 속하는 모든 데이터 개체들 사이의 평균거리로 클러스터가 얼마나 잘 응집되어 있는가를 나타낸다.

Q값이 적을수록 잘 응축된 클러스터를 얻는 것이다.

$$Q = \sum_{i=1}^{k} \frac{Di}{k} \qquad (4.1)$$

$$Di = \sum_{a=1}^{n_i} \sum_{b=1}^{n_i} \sqrt{(x_a - x_b)^2}$$

제안하는 방법(NEW)과 PPC, k-means, 최단연결법, 최장연결법, 평균연결법, 중심연결법 등 총 7가지 방법으로 나누어 실험하였다.

표 3.은 실험 결과값인 응축도(Q)를 보여준다. 군집화는 크게 통계적 군집화 기법과 인공신경망 모델을 기반으로 구분된다. 통계적 군집화 방법은 다시 계층적 군집화 방법(최단연결, 최장연결, 평균연결, 중심연결)과 분할적 군집화 방법(k-means)으로 나누어진다[12][13].

최단 연결법과 최장 연결법은 군집간의 유사도를 측정하기 위해 특정 데이터 개체 하나에만 의존하기 때문에 잡음에 민감하고, 평균 연결법과 중심 연결법은 하나 이상의 데이터 개체를 기반으로 중심값을 갖기 때문에 잡음에 덜 민감하였다. PPC는 자기조직화지도로 전처리 하는 과정에서 데이터를 요약, 압축했고 두 개의 데이터 개체를 기반으로 인접거리, 연결거리를 기반으로 좋은 응축도를 보였다. 제안하는 방법으로 실험한 결과 PPC만큼이나 좋은 응축도를 보였고, SVM을 이용한 거리 기반의 유사도를 측정해 원하는 클러스터의 개수를 얻을 수 있었다. 따라서 양질의 군집을 얻는데 매우 효과적이다.

표 3. 7가지 방법에 대한 실험 결과

응축도 (Q)	NEW	PPC	k-means	최단 연결	최장 연결	평균 연결	중심 연결
Car evaluation	0.577	0.578	0.599	0.671	0.693	0.670	0.693
Covertype	0.200	0.209	0.315	0.522	0.510	0.436	0.511
Diabetes	1.800	1.872	1.018	1.986	1.981	1.936	1.967
Letter recognition	0.578	1.578	1.599	1.671	1.693	1.670	1.693
Nursery DB	2.23	2.209	2.315	2.522	2.510	2.436	2.511
Solar flare	2..00	2.255	2.772	2.645	2.535	2.435	2.235

PPC* : 학습율 0.3

5. 결론 및 향후 과제

본 논문에서는 SVM을 이용하여 지역적 최적해에 수렴하는 것을 방지하고, 새로운 거리 측정 방법을 제시한다. 이미 형성된 클러스터간의 거리를 오름차순으로 정렬하여 테이블을 생성하고, Δd 임계값을 주어 이웃한 클러스터들을 병합함으로서 적절한 클러스터의 개수를 얻을 수 있다.

이렇게 얻어진 클러스터의 응집도를 측정함으로서 양질의 군집을 얻는데 매우 효과적이라는 실험결과를 얻을 수 있다.

향후 연구 과제로는 SVM의 다양한 커널 함수를 적용한 다양한 실험이 필요하며, 제안된 클러스터 간 거리 측정법에 대하여 보다 더 잡음(noise)에 강한 특성과 효율적인 클러스터 병합을 위하여 방법이 제시되어야 하며, 적절한 Δd.임계값을 선택하는 문제 또한 연구될 수 있을 것이다.

참 고 문 헌

[1] Tian Zhang, Raghu Ramakrishnan, and Miron, "Birch : an efficient data clustering method for very large database," the ACM SIGMOD Conference on Management of Data, Montreal, Canada, June, 1996.

[2] R.. Pyle, D.E. Hart, "Pattern Classfication and Scene Analysis," A Wiley-Interscience Publication, NewYork, 1973.

[3] Raymond T.Ng,Jiawei Han, "Efficient and Effective Clustering Methods for Spatial Data Mining", Proc. of 20th Int.Conf. on VLDB, pp.144-155, 1994.

[4] 손은정, 강인수, 김태원, 이기준, "클러스터링 분석에

의한 공간 데이터마이닝 방법", 한국정보과학회 가을 학술발표논문집(2), 1998.
[5] M.Ester, H. Kriegel, Jorg Sander, and Xiaowei Xu, "A density-based algorithm for discovering clusters in large spatial database with noise", Proc. of Int .Conf. on Knowledge Discovery and Data Mining, 1996.
[6] Tian Zhang, Raghu Ramakrishnan, and Miron Livny, "BIRCH:An Efficient Data Clustring Method for Very Large Databases", Proc. of ACMSIGMOD Int. Conf. on Management of Data, pp.103-114, 1996.
[7] 이혜명, 박영배, "점진적 프로젝션을 이용한 고차원 글러스터링 기법", 한국정보과학회논문지:데이타베이스 Vol.28.No.4, pp.568-576, 2001
[8] 장미희,이혜명, 박영배, "고차원 데이터에서 2차원 프로젝션을 이용한 클러스터링", 한국정보과학회 가을학술발표논문집 Vol.28.No.2, 2001.
[9] http://www.kernel-machines.org
[10] http://svm.cs.rhbnc.ac.uk
[11] http://www.ics.uci.edu/

저 자 소 개

김만선(Man-Sun Kim)
2000년 : 홍익대학교 전자전기컴퓨터공학부 (공학사)
2002년 : 공주대학교 대학원 전자계산학과 (이학석사)
2002년~현재 : 공주대학교 대학원 컴퓨터공학과 박사과정

관심분야 : 데이터마이닝, 신경망, 게임과 인공지능, 기계학습
e-mail : mansun@kongju.ac.kr

이상용(Sang-Yong Lee)
1984년 : 중앙대학교 전자계산학과(공학사)
1988년 : 일본동경공업대학 총합이공학연구 (공학석사)
1988년~1989년 : 일본 NEC 중앙연구소 연구원
1993년 : 중앙대학교 일반대학원 전자계산학과 (공학박사)
1993년~현재 : 공주대학교 정보통신공학부 교수
1996년~1997년 : University of Central Florida 방문교수

관심분야 : 인공지능, 기계학습, 에이전트, 바이오인포매틱스
e-mail : sylee@kongju.ac.kr