DOI : 10.5392/JKCA.2010.10.9.097

시계열 분류를 위한 PIPs 탐지와 Persist 이산화 기법들을 결합한 시계열 표현

Time Series Representation Combining PIPs Detection and Persist Discretization Techniques for Time Series Classification

박상호, 이주홍
인하대학교 컴퓨터 정보공학부

Sang-Ho Park(parksangho@datamining.inha.ac.kr), Ju-Hong Lee(juhong@inha.ac.kr)

요약

시계열 데이터를 효율적이고 효과적으로 처리하기 위해 다양한 시계열 표현 방법들이 제안되었다. SAX *(Symbolic Aggregate approXimation)*는 단편화와 이산화 기법들을 결합한 시계열 표현 방법으로, 시계열 분류 문제에 성공적으로 적용되었다. 그러나 SAX는 시계열의 움직임을 평활하여 시계열의 중요한 동적 패턴들을 정확히 표현하기 위해 세그먼트 수를 크게 해야 한다. 본 논문은 PIPs *(Perceptually Important Points)*탐지 기법과 Persist 이산화 방법을 결합한 시계열 표현 방법을 제안한다. 제안된 방법은 시계열의 중요한 변곡점들을 나타내는 PIP 들을 탐지하여 고차원 시계열의 동적 움직임을 저차원 공간에서 표현한다. 그리고 시계열의 자기 전이와 주변 확률 분포를 KL 다이버전스에 적용하여 최적의 이산화 영역들을 결정한다. 제안된 방법은 시계열의 차원 축소과정에서 정보 손실을 최소화하여 시계열 분류의 성능을 향상시킨다.

■ 중심어 : | 시계열 표현 | PIPs 탐지 | KL 발산 |

Abstract

Various time series representation methods have been suggested in order to process time series data efficiently and effectively. SAX is the representative time series representation method combining segmentation and discretization techniques, which has been successfully applied to the time series classification task. But SAX requires a large number of segments in order to represent the meaningful dynamic patterns of time series accurately, since it loss the dynamic property of time series in the course of smoothing the movement of time series. Therefore, this paper suggests a new time series representation method that combines PIPs detection and Persist discretization techniques. The suggested method represents the dynamic movement of high-diemensional time series in a lower dimensional space by detecting PIPs indicating the important inflection points of time series. And it determines the optimal discretizaton ranges by applying self-transition and marginal probabilities distributions to KL divergence measure. It minimizes the information loss in process of the dimensionality reduction. The suggested method enhances the performance of time series classification task by minimizing the information loss in the course of dimensionality reduction.

■ keyword : | Time Series Representation | PIPs Detection | KL Divergence |

접수번호 : #100831-006
접수일자 : 2010년 08월 31일
심사완료일 : 2010년 09월 16일
교신저자 : 박상호, e-mail : inhattt@naver.com

1. 서론

시계열 분류는 레이블된 시계열을 모형화하여 새로운 시계열에 레이블을 부여하는 것으로, 생물 의학 신호 [12], 연속 시스템 진단 [6], 금융 정보 시스템 [5]등 시계열의 다양한 응용 분야들에 적용되었다.

시계열의 고차원성 (high-dimensionality)과 연속성 (continuity)은 시계열 분류의 학습 과정에서 계산량을 증가시킨다. 시계열을 빠르고 정확하게 처리하기 위해 *Discrete Fourier Transformation* (DFT) [13], *Singular Value Decomposition* (SVD) [3], *Discrete Wavelet Transformation* (DWT) [9], *Piecewise Aggregate Approximation* (PAA) [1], *Adaptive Piecewise Constant Approximation* (APCA) [4], *Symbolic Aggregate approXimation* (SAX) [7]등 다양한 시계열 표현 방법들이 제안되었다. 제안된 방법들은 고차원 시계열을 저차원 공간으로 매핑하는 차원 축소 기법들로, 전처리 과정에서 수행되어 시계열 분류의 학습 과정에서 계산량을 줄인다.

SAX는 가장 대표적인 시계열 표현 방법으로, 단편화와 이산화 방법을 결합하여 연속 시계열을 축소된 차원 공간에서 이산 형태로 변환한다. SAX는 시계열의 차원 수를 줄이기 위해 PAA 단편화 기법을 이용하였다. PAA는 각 세그먼트에 속한 시계열의 평균을 PAA 계수 값으로 결정한다. 그리고 PAA 계수 값들을 이용하여 시계열을 표현한다. SAX의 이산화는 정규성 (normality) 가정하에 가우시안 분포 곡선에 의한 전체 영역을 동일한 크기의 영역들로 분할하여 이산화 영역들을 결정한다. 그리고 각 이산화 영역에 위치한 PAA 계수 값들을 이산화한다.

SAX의 시계열 표현은 시계열의 가독성(readability)을 향상시키고, 시계열의 분류 모델을 빠르게 구축할 수 있다 [7]. 그리고 이산 시계열을 입력으로 요구하는 의사 결정 트리와 같은 기계 학습 알고리즘들에 유용하다 [8]. 그러나 SAX는 다음과 같은 문제점들을 가진다. 첫째, SAX는 시계열의 움직임을 평활(smoothing)하여 시계열의 다양한 동적 패턴들을 정확히 표현하지 못한다. 둘째, SAX는 정보 손실을 최소화하기 위해 세그먼트 수를 크게 해야한다. 셋째, 타임 시프팅과 타임 스케일링된 시계열 데이터 집합이 주어졌을 때, SAX는 시계열 분류자의 학습 과정에서 DTW (Dynamic Time Warping)과 같이 높은 계산 복잡도의 거리 척도를 요구한다.

그러므로 본 논문은 새로운 시계열 표현 방법을 제안한다. 제안된 방법은 다음과 같은 특징을 가진다. 첫째, 시계열의 중요한 변곡점들을 나타내는 PIP 들을 탐지하여 시계열의 동적 특성을 저차원 공간에서 표현한다. 둘째, 시계열의 최적화된 이산화 영역들을 결정하기 위해 Persist 이산화 방법을 이용한다. Persist 이산화 방법은 시계열의 최적화된 이산화 영역들을 찾는다. 그러므로 제안된 시계열 표현 방법은 SAX보다 시계열의 정보 손실을 최소화하여 시계열의 분류 성능을 향상시킨다. 실험을 통하여, 제안된 방법이 SAX보다 시계열 분류에 적합한 시계열 표현임을 보인다.

본 논문의 구성은 다음과 같다. 2장에서는 SAX의 시계열 표현 방법을 설명하고, 3장에서는 제안된 시계열 표현 방법을 설명한다. 4장에서는 제안된 시계열 표현 방법을 평가하고, 5장에서 결론을 맺는다.

2. 관련 연구

시계열 데이터, T, 는 $\{(t_1,v_1),\cdots,(t_i,v_i),\cdots,(t_n,v_n)\}$와 같이 기술되며, n차원 공간에서 하나의 실수 벡터 $(v_1,\cdots,v_i,\cdots,v_n)$에 의해 표현된다. 이 때, v_i는 t_i 시간에서 시계열 값이고, t_i는 $i<j \Leftrightarrow t_i<t_j$조건을 만족해야 한다. 시계열 T 는 큰 값의 n차원 공간에서 연속 데이터 형식의 v_i 에 의해 표현된다. 이것은 시계열 분류의 계산량을 증가시킨다.

2.1 SAX의 시계열 표현

SAX는 단편화와 이산화 기법들을 결합한 시계열 표현 방법으로, 시계열의 차원 수를 줄이기 위해 PAA 단편화 기법을 적용하였다 [7]. 시계열의 단편화는 n차원 시계열 T를 w차원의 벡터, $C=(c_1,...,c_w)$로 변환하는 방법이다 ($n>>w$). PAA기법은 n차원 시계열T 를 n/w크

기의 w개 세그먼트들로 분할한다. 그리고 각 세그먼트에 위치한 n/w개 시계열 값들을 평균하여 PAA 계수 c_i의 값를 얻는다. 식(1)은 c_i의 계산식이다.

$$c_i = \frac{w}{n} \sum_{j=\frac{n}{w}(i-1)+1}^{\frac{n}{w}i} v_i \quad (1)$$

시계열의 이산화는 실수 벡터 $C=(c_1,\cdots,c_w)$를 정수 벡터 $D=(d_1,\cdots,d_w)$ 혹은 심볼 시퀀스로 변환하는 방법이다 [15]. 이때, 1과 w사이의 모든 i와 j에 대해서, $c_i \le c_j$이면 $d_i \le d_j$ 조건을 충족해야 한다. SAX의 이산화는 가우시안 분포 곡선에 의한 전체 영역을 동일한 크기의 영역들로 분할하여 이산화 영역들을 결정한다. 그리고 실수 벡터 C의 각 요소 c_i를 c_i가 위치한 이산화 영역의 대표치 혹은 심볼로 대치하여 정수 벡터D 혹은 심볼 시퀀스(sequence)로 결정한다. 예를 들어, 이산화 수가 5로 주어졌다고 가정하자. SAX는 가우시안 분포 곡선에 의한 영역을 동일한 크기의 영역들로 분할하여, 5개 이산화 영역 집합 {(0.84,+∞), (0.25,0.84), (-0.25,0.25), (-0.84,-0.25), (-∞,-0.84)을 결정한다. 그리고 벡터 C = (0.1, 0.6, 1.1, 0.7, -0.1,-1.0, -0.3)의 각 요소 c_i를 이산화 영역들의 대표 심볼들 A, B, C, D, E중 하나의 심볼로 변환한다.

[그림 1]은 세그먼트와 이산화 수가 각각 7과 5일때, 길이 70의 시계열을 길이7의 심볼 시퀀스 (C, B, A, B, C, E, D)로 변환하는 SAX의 시계열 표현을 보여준다.

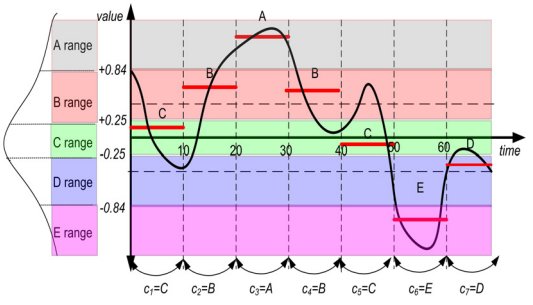


그림 1. SAX 시계열 표현의 예

2.2 SAX의 문제점

SAX의 시계열 표현은 다음과 같은 문제점을 가진다. 첫째, SAX는 서로 다른 형태의 시계열 패턴들을 명확히 구분하지 못한다. 왜냐하면, SAX는 각 세그먼트에 속한 시계열 값들을 평균하여 시계열의 움직임을 평활한다. 이로 인해 시계열의 동적 정보가 손실된다. 둘째, SAX는 시계열의 정보 손실을 최소화하기 위해 세그먼트 수를 크게 해야 한다.

[그림 2]는 SAX가 구분하지 못하는 시계열의 패턴들의 집합을 보여준다. [그림 2] (A), (B), (C), (D)에서 4개 시계열들은 서로 다른 움직임 형태들을 가진다. 그러나 SAX의 PAA 단편화 기법은 4개 시계열들에게 동일한 PAA 계수 값을 부여하여 그들을 동일한 패턴의 시계열로 간주한다.

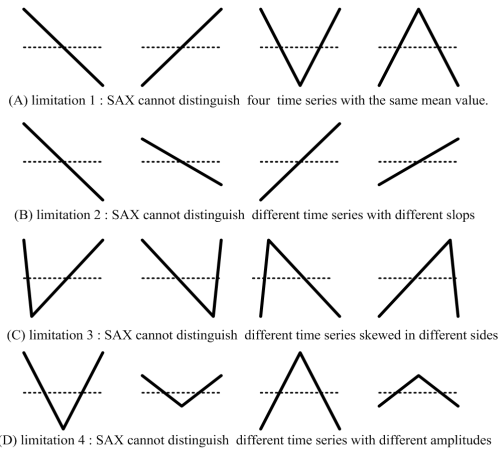


그림 2. SAX 시계열 표현의 문제 패턴들

[그림 2]와 같이, 다른 움직임 형태의 시계열들을 구분하기 위해서는 세그먼트 수를 크게 해야 하는 문제점을 가진다. [그림 3]은 주기성을 갖는 두 개의 시계열 T_α와 T_β 에 SAX의 PAA 단편화 기법을 적용한 예이다. 이 때, 세그먼트 수는 5로 동일하다.

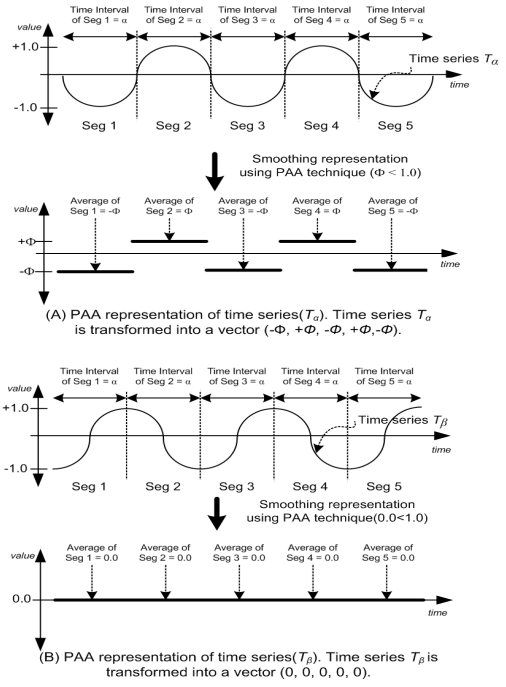


그림 3. 주기성 시계열 데이터에 대한 SAX 시계열 표현

[그림 3]의 T_a 의 경우, PAA 단편화 기법은 시계열의 움직임을 근사적으로 잘 표현한다. 그러나 T_β 의 경우, 5개 PAA 계수들이 동일한 값 0 을 가져, 시계열의 정보 손실을 최대화한다. 그러므로 다양한 형태를 갖는 시계열의 정보 손실을 최소화하기 위해서 PAA 단편화 기법은 단편 수를 크게 해야 한다.

SAX 의 이산화는 시계열의 정규성을 가정하여 이산화 영역들을 결정한다. 그러나 [그림 4]와 같은 분포의 시계열이 주어졌을 때, SAX의 이산화는 시계열의 정보를 왜곡할 수 있다. 왜냐하면, 중심 극한 정리에 의해 [그림 4]와 같은 분포의 시계열은 근사적 정규 분포 형태를 보일 수 있기 때문이다.

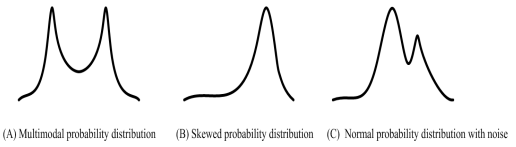


그림 4. 정규성 위반 사례

3. 제안된 시계열 표현 방법

본 논문에서 제안한 시계열 표현 방법은 다음과 같은 특징을 가진다. 첫째, 시계열의 PIP들을 탐지하여 시계열의 차원 수를 줄인다. 둘째, Persist 이산화 방법을 적용하여 시계열을 이산화한다. 알고리즘1은 본 논문이 제안한 시계열 표현 방법의 의사 코드이다.

알고리즘1. 시계열 표현의 의사 코드

Input : A n-dimensional continuous time series set T={T_1,⋯,T_i,⋯,T_m} where T_i={(t_1,v_1),⋯, (t_j,v_j) ,⋯,(t_n,v_n)}, w the number of PIPs, k the degree of discretization

Output : A w-dimensional discrete time series set, D = {D_1,⋯,D_i,⋯,D_m} where D_i={d_1,⋯,d_w}

Find k discretization ranges from T
For i = 1 ⋯ m
  Detect w number of PIPs from T_i
  Discretize PIPs detected by menas of k discretization ranges
Endfor

본 논문이 제안한 시계열 표현은 시계열 집합 T를 Persist 이산화 알고리즘에 적용하여 k개 이산화 영역들의 경계치들을 결정한다. 그리고 각 시계열 T_i 에 대해, w개 PIP들을 탐지하여 k차원의 PIP벡터를 구성한다. 이때, PIP의 각 요소들은 연속형 수치 값을 가진다. 그리고 k개 이산화 영역들의 경계치들을 이용하여 PIP 벡터의 각 요소를 이산화하여 D_i를 얻는다.

3.1 PIP 탐지 기법을 적용한 시계열의 단편화

본 논문은 시계열의 PIP들을 탐지하여 시계열의 움직임을 저차원 공간에서 표현한다. PIP는 인접한 두 개의 PIP들을 연결한 직선과 시계열과의 수직 거리에 의해 탐지된다. 알고리즘 2는 시계열 T_i 가 주어졌을 때, w개 PIP들을 탐지하는 과정을 보여준다.

알고리즘 2. PIPs 탐지

// T_i and w denote the i-th time series of time series set T and the number of PIPs, respectively.

//*start* and *end* indicate the first and last time slot of T_i, respectively.

// MIN is set to a smallest number.

```
typedef Pnode struct {start,end,MaxTime,MaxDistance} ;
int PIP[w] ;
void PIPDetectionWholeTimeRange(T_i,w)
{
Pnode T ;
    T = FindMaxVD(start,end) ;
    insertHeap(T, T.MaxDistance) ;
    PIPDetectionSplittedTimeRanges(1,w) ;
}
void PIPDetectionSplittedTimeRanges(k,w)
{
    if (k>w) return ;
    T = deleteHeap( ) ;
    PIP[k]=T.MaxTime ;
    LeftT = FindMaxVD(T.start, T.MaxTime) ;
    RightT = FindMaxVD(T.MaxTime, T.end) ;
    if (LeftT=!NULL)
       insertHeap(LeftT, LeftT.MaxDistance) ;
    if (RightT=!NULL)
       insertHeap(RightT, RightT.MaxDistance) ;
    PIPDetectionSplittedTimeRanges(k+1,w) ;
}
Pnode FindMaxVD(start,end)
{
    if (end-start<2) return NULL ;
    MaxDistance=MIN ;
    for( t=start+1 ; t<end ; i++) {
      VD=CalcuateVD(t) ;
      if (VD > MaxDistance) {
        MaxDistance = VD ;
        MaxTime=t ;
      }
    }
    Return (start, end, MaxTime, MaxDistance)
}
```

알고리즘 2에서, w개 PIP들은 시계열의 중요한 변곡점들을 나타내고, 벡터 PIP = {pip_1,…,pip_k,…,pip_w}로 기술된다. *PIPDetectionWholeTimeRange* 함수는 시계열 T_i 에서 가장 중요한 하나의 PIP를 탐색하고 *PIPDetectionSplitted- TimeRanges* 함수는 재귀적 호출 과정을 통하여 k-1개의 PIP들을 탐색한다. 본 논문은 k개 PIP들을 탐색하는 과정에서 최대 수직 거리(*MaxDistance*)의 시계열 구간을 결정하기 위해 최대 힙(max-heap)을 이용하였다. 왜냐하면, 최대 힙은 수직 거리의 가장 큰 값이 루트에 저장되기 때문이다. *FindMaxVD* 함수는 시계열의 시작과 마지막 시점에서 시계열을 연결한 직선과 시계열과의 수직 거리를 계산하여 최대 수직 거리 값을 갖는 시계열의 위치를 결정한다. 시계열과 인접한 PIP들을 연결한 직선과의 수직 거리 (*CalcuateVD*)는 식(2)에 의해 계산된다.

$$VD(t) = |y_t - y_t'| = |y_t - (y_s + (y_e - y_s) \cdot \frac{x_t - x_s}{x_e - x_s}| \quad (2)$$

식(2)에서, y_t' 과 y_t 는 시점 t 에서 인접한 PIP를 연결한 직선과 시계열 T_i의 값을 각각 나타낸다. 그리고 첨자 s와 e는 시계열의 시작(*start*)과 마지막(*end*)시점을 각각 나타낸다. [그림 5]는 PIP 탐지(A)와 탐지된 PIP에 의한 시계열 표현(B)의 간단한 예를 보여준다.

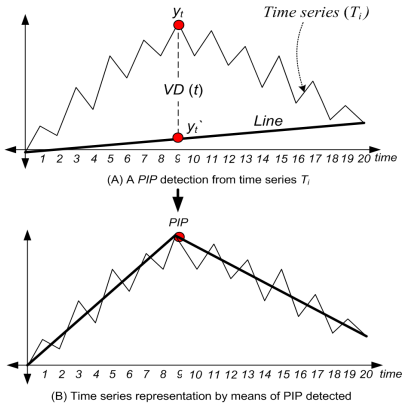


그림 5. PIP 탐지에 의한 시계열 표현

[그림 5] (A)에서, 시계열 T_i= {(t_0,v_0),⋯,(t_j,v_j),⋯,(t_{20},v_{20})}이 주어졌을 때, T_i의 첫 번째(t_0)와 마지막(t_{20})시점에서의 시계열 값 v_0와 v_{20}를 연결하여 하나의 직선(Line)을 그린다. 그리고 1부터 19까지 각 t 시점에서 시계열 값과 직선과의 수직 거리를 계산하여 가장 큰 수직 거리의 시점(t_9)에서의 v_9을 시계열T_i의 PIP로 결정한다.

3.2 Persist 알고리즘을 적용한 시계열의 이산화

Fabian M.은 최적의 이산화 경계치들을 결정하기 위해 Persist 알고리즘을 제안하였다 [11]. Persist는 두 개의 확률 분포를 비교하는 척도(measure)인 KL (Kullback-Leibler) 발산을 이용하여 시계열의 시간적 구조(temporal structure)를 고려한 최적의 이산화 경계치들을 결정한다. 즉, 자기 전이 확률 분포 P={p_1,..., p_i,...,p_k}와 주변 확률 분포 Q={q_1,...,q_i,...,q_k} 가 주어졌을 때, Persist는 KL 발산 이론을 이용하여 두 개의 확률 분포 P와 Q의 차이를 수치화하여 최적의 이산화 경계치들을 결정한다. Persist 이산화 방법은 기존 이산화 방법들보다 높은 정확도를 가지며, 잡음에 강한 특성을 가진다. 알고리즘 3 은 탐지된 연속형 수치형의 PIP 벡터의 각 요소를 이산화하기 위한 k-1개 이산화 경계치들을 결정하는 과정을 보여준다.

알고리즘 3. 최적의 이산화 경계치 탐색
B = Ø
For i=1 ... k-1
  P = Ø
  Foreach c∈C
    P = P∪{*Persistence*(D(T, B∪{c}))}
  End
  b=max$_i$(P)
  B=B∪{c_b}
Endfor

알고리즘 3 에서, C는 Equal frequency binning 방법에 의해 결정된 이산화 경계치들의 후보 집합으로 C={c_j|c_j∈R,j=1...m}로 기술된다. 그리고 Persistence( ) 는 k개 이산화 영역들의 집합 S={s_1,...,s_k}에 의한 시계열 T 의 Persistence score를 나타내며, 식(3)에 의해 계산된다. 그리고 D는 이산화 함수이다. 식(4)에서, sgn() 은 Persistence score의 부호를 결정하는 지시자(indicator)이며, 식(3)의 Persistence(s_i)는 식(4)에 의해 계산된다.

$$Persistence(S) = \frac{1}{k}\sum_{i=1}^{k} persistence(s_i) \quad (3)$$

$$persistence(s_i) = sgn(A(j,j) - P(s_j))SKL(A(j,j), P(s_j)) \quad (4)$$

식(4)에서, A(j, j)와 P(s_j)는 각각 자기 전이 확률과 주변 확률을 각각 나타낸다. 예를 들어, 시계열 D_i={d_j|d_j∈S, j=1,..,n}가 이산화 영역들의 집합 S에 의해 표현될 때, k x k 전이 확률 행렬이 계산된다. A(j, j)는 k x k 전이 확률 행렬에서 대각선에 위치한 확률 P(d_i=s_j|d_{i-1}=s_j)이다. 그리고 P(s_j)는 심볼 s_j 의 발생 빈도를 시계열의 길이 n으로 나눈 값으로 심볼 s_j 가 일어날 확률이다. Persistence()는 sgn()이 양이고 A(j,j)와 P(s_j)의 SKL(*Symmetric Kullback Leibler divergence*)이 클 때, 큰 값을 가진다. Persistence()의 SKL은 식(5)에 의해 계산되고, 식(6)은 KL 계산식이다.

$$SKL(A(j,j), P(s_j)) = \frac{1}{2}(KL(A(j,j), P(s_j)) + KL(P(s_j), A(j,j))) \quad (5)$$

$$KL(A(j,j),P(s_j)) = A(j,j)\log(\frac{A(j,j)}{P(s_j)}) \quad (6)$$

4. 실험 결과

본 논문은 시계열 분류 실험을 통하여 SAX와 제안한 시계열 표현 방법의 성능 비교 실험을 수행하였다. 실험은 RAM 1.87M, CPU 2.50GHz, Window XP 시스템에서 수행되었다.

실험을 위해, 본 논문은 UCI Time Series repository [2]에서 SCC (Synthetic Control Chart)와 EEG (Electro- EncephaloGram)시계열 데이터 집합을 수집하였다. 그리고 세그먼트 수와 이산화 차원(Degree)를 증가시키면서, SCC와 EGG 시계열의 분류 모델들의 성능을 각각 측정하였다. 본 논문은 시계열 표현 방법들의 분류 성능을 비교하기 위해 SVM (*Support Vector Machine*)을 분류자로 선택하였다. 왜냐하면, SVM은 선형 대수 이론에 기반하여 학습 속도가 빠르고 데이터의 양이 적을 때에도 높은 일반화 성능을 보장하기 때문이다. 시계열 데이터의 학습과 테스트 비율은 50%대 50%이고, SVM의 커널 (kernel) 함수로 Gaussian Radial Basis Function를 사용하였다. 그리고 Tay와 Cao의 실험 결과에 근거하여, SVM의 파라미터 C와 δ^2를 78과 25로 설정하였다 [5][10].

SCC 시계열 데이터 집합은 6개 클래스로 구성되며, 각 클래스는 100개의 길이 60의 유사한 시계열 패턴들을 가진다. [그림 6]은 SCC의 6개 클래스들의 시계열 패턴들을 보여준다.

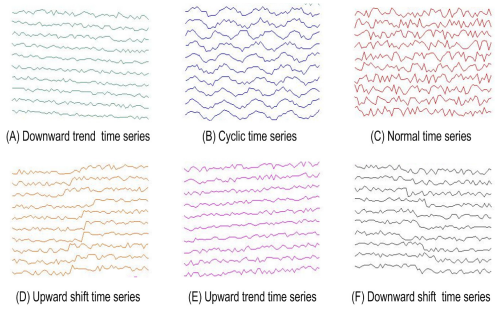


그림 6. SCC 시계열 데이터 집합의 6가지 패턴들

본 논문에 사용된 EEG 시계열 데이터는 알코올 중독자 (alcoholic)와 정상인 (control)의 두피에서 측정된 64개 전극 (electrode)의 뇌파 데이터로서, 1초 동안 256Hz의 3.9msec epoch를 가진다. [그림 7]의 (A)와 (B)는 알코올 중독자와 정상인의 EEG 데이터를 각각 보여준다.

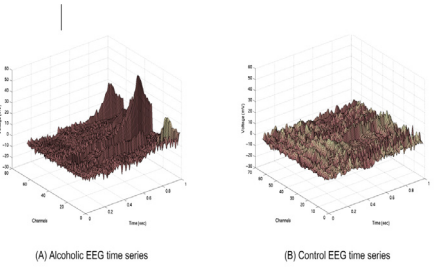


그림 7. 알코올 중독자와 정상인의 EEG 시계열 데이터

시계열 표현 방법들의 성능을 비교하기 위해, 본 논문은 SCC와 EGG 시계열 데이터에 대해 다음과 같은 실험들을 수행하였다. 첫째, 세그먼트 수를 증가시키면서, SAX의 PAA 기법과 PIP 탐지 기법에 의한 시계열의 분류 정확도를 측정하였다. [그림 8]은 SVM 분류자를 이용하여, SCC와 EGG 시계열 데이터에 PAA와 PIP 탐지 기법을 각각 적용하여 얻은 시계열 데이터의 분류 정확도를 보여준다.

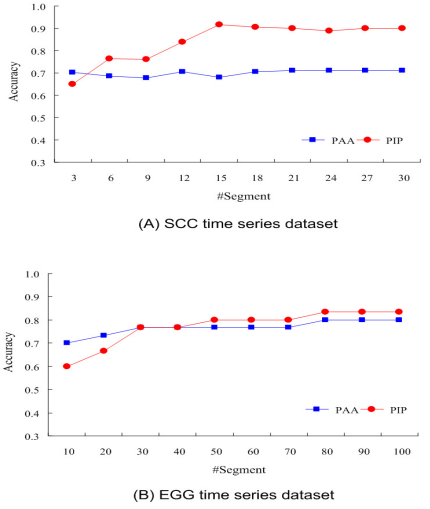
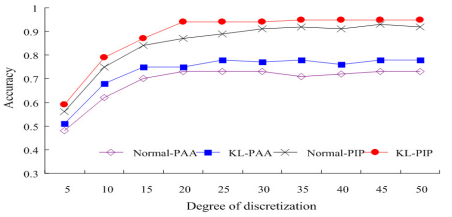


그림 8. PAA와 PIP 단편화 기법의 성능 비교

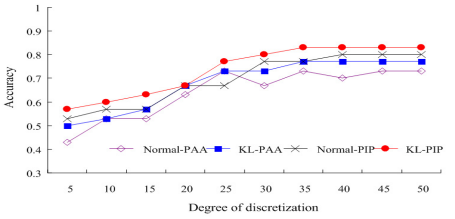
[그림 8]의 실험에서, PIP 탐지 기법에 의한 시계열 표현이 SAX의 PAA 기법에 의한 시계열 표현보다 시계열을 더욱 효과적 분류하였다. 그리고 PIP탐지 기법은 PAA 기법보다 세그먼트 수에 민감하였다. 이것은 PIP 탐지 기법은 시계열 분류를 위해 일정 수준 이상의 세그먼트 수를 요구함을 의미한다. 그러나 PIP탐지 기법은 세그먼트 수가 일정 크기 이상일때, PAA 기법보다 좋은 분류 성능을 보인다. 이것은 PIP 탐지 기법이 시계열의 정보 손실을 최소화함을 입증한다.

SCC 시계열 데이터의 경우, 세그먼트 수가 15이하에서 급격한 성능 향상을 보인반면, 세그먼트 수가 15이상 일때, 시계열 분류의 분류 성능은 큰 변화를 보이지 않았다. 이것은 SCC 시계열 데이터의 정보 손실을 최소화하는 최소 변곡점의 수가 15임을 나타내며, 차원을 (15/60)*100=25%까지 축소할 수 있음을 의미한다. 왜냐하면, 15번째 이후에 탐지된 PIP들은 시계열의 움직임 형태에 큰 영향을 주지 않기 때문이다. EGG 시계열 데이터의 경우, 세그먼트 수가 30이상 일 때, 시계열의 분류 성능은 완만하게 상승하였으며, 차원을 (30/256)*100=11.7%까지 축소 할 수 있음을 의미한다. 이러한 성능 향상은 SCC 시계열 데이터가 안정적 (stationary)인 EGG 시계열 데이터와 달리 비안정적 (nonstationary)이기 때문이다.

둘째, 이산화의 차수 (degree)를 증가시키면서, 다음과 같은 4가지 형태로 표현된 시계열 데이터들의 분류 정확도를 측정하였다. SAX에 의한 시계열 표현 (Normal-PAA), PAA 기법과 KL을 적용한 시계열 표현 (KL-PAA), SAX의 이산화와 PIP 탐지 기법에 의한 시계열 표현 (Normal-PIP), KL과 PIP 탐지에 의한 시계열 표현 (KL-PIP). [그림 9]는 4 가지 형태로 변환된 SCC와 EGG 시계열 데이터들의 분류 정확도를 각각 보여준다.



(A) SCC time series dataset



(B) EGG time series dataset

그림 9. 시계열 표현 방법들의 분류 성능 비교

[그림 6]의 실험에서, PIP 탐지 기법과 KL 이산화 기법을 결합한 시계열 표현이 가장 좋은 분류 성능을 보였다. SCC 시계열 데이터의 경우, KL-PIP, Normal-PIP, KL-PAA, Normal-PAA를 적용한 순으로 높은 분류 정확률을 보였다. EGG 시계열 데이터의 경우도, KL-PIP기법을 적용한 시계열 표현이 가장 높은 분류 성능을 보였다. 그리고 SAX의 이산화 방법을 적용한 시계열 표현보다 KL을 적용한 시계열 표현이 더 높은 분류 정확률을 보였다. 실험 결과는 본 논문이 제안한 시계열 표현이 SAX의 시계열 표현보다 시계열 분류에 더 적합함을 증명해준다. 그 이유는 다음과 같다. 대부분의 시계열의 유사한 패턴들은 타임 시프팅과 타임 스케일의 정도가 상이하다. 그러한 경우, 기존 시계열 표현은 시계열의 유사한 패턴들을 비유사도 (dissimilarity)가 높은 유사하지 않은 시계열 패턴으로 분류하여 성능을 저하시킨다. 그러나 본 논문에서 제안한 시계열 표현은 시계열 패턴의 타임 시프팅과 타임 스케일 문제를 해결하여 분류 성능을 향상시킨다. 그리고 제안된 시계열 표현은 시계열의 추세 표현에 적합하며, 세그먼트의 중복성을 제거할 수 있으며, 중요도가 높은 변곡점을 우선적으로 탐지할 수 있는 장점들을 가진다.

5. 결론

SAX는 대표적 시계열 표현 방법으로, 시계열을 효율적이고 효과적으로 분류하기 위해서 단편화와 이산화 기법들을 결합하여 고차원 연속 시계열을 저차원 공간에서 이산 형태로 표현한다. 그러나 SAX는 각 세그먼트에 속한 시계열 값들을 평균화하여 시계열의 중요한 패턴들을 왜곡할 수 있다.

본 논문은 시계열을 정확하게 분류하기 위해서 PIP 탐지 기법과 Persist 방법을 결합한 새로운 시계열 표현 방법을 제안하였다. 제안된 방법은 PIP 탐지 기법을 이용하여 n차원의 연속 시계열의 중요도가 높은 w개 변곡점들을 탐지하여 시계열의 차원을 축소한다($n>w$). 그리고 시계열의 자기 전이와 주변 확률 분포의 KL 다이버전스에 적용하여 최적의 이산화 영역들을 결정한다. 제안된 시계열 표현 방법은 시계열의 차원 축소과정에서 타임 시프팅과 타임 스케일된 유사한 패턴의 시계열 들의 정보 손실을 최소화하여 시계열 분류의 성능을 향상시킨다.

참 고 문 헌

[1] B-K Yi and C. Faloutsos, “Fast Time Sequence Indexing for Arbitrary Lp Norms”, Proceedings of the VLDB, Cairo, Egypt, 2000(9).

[2] C. L. Blake and C. J. Merz, UCI Repository of Machine Learning Databases, http://www.ics.uci.edu/~mlearn/MLRepository.html, UC Irvine, Dept. Information and Computer Science, 1998.

[3] E. Keogh, K. Chakrabarti and M. Pazzani, S. Mehrotra, "Dimensionality reduction for fast similarity search in large time series databases," Journal of Knowledge and Information Systems, Vol.3, No.3, pp.263-286, 2001.

[4] E. Keogh, K. Chakrabarti, M. Pazzani, and S. Mehrotra, “Locally adaptive dimensionality reduction for indexing large time series databases," In proceedings of ACM SIGMOD Conference on Management of Data. Santa Barbara, CA, May 21-24, pp.151-162, 2001.

[5] F. E. H. Tay and L. Cao, "Application of support vector machine in financial time series forecasting," Omega 29, pp.309-317, 2001.

[6] J. Carlos, G. Alonso and J. R. Juan, "A graphical rule language for continuous dynamic systems," In Computational Intelligence for Modelling, Control and Automation. Masoud Mohammadian, Ed., Amsterdam, Netherlands, CIMCA-99, pp.482-487, IOS Press, 1999.

[7] J. Lin, E. Keogh, L. Wei, and S. Lonardi, "Experiencing SAX: A novel symbolic representation of time series," Data Mining and Knowledge Discovery, Vol.15, No.2, 2007.

[8] J. R. Quinlan, C4.5 : Programs for Machine Learning, Morgan Kaufmann Pub, LosAltos, Califoormia, 1993.

[9] K. Chan and W. Fu, "Efficient time series matching by wavelets," Proceedings of the 15th IEEE International Conference on Data Engineering, 1999.

[10] K. J. Kim, "Financial time series forecasting using support vector machines," Neurocomputing, Vol.55, pp.307-319, 2003.

[11] M. Fabian and U. Alfred, “Optimizing Time Series Discretization for Knowledge Discovery,” ACM SIGKDD, pp.660-665, 2005.

[12] M. Kubat, I. Koprinska, and G. Pfurtscheller, "Learning to classify biomedical signals", In Machine Learning and Data Mining, R.S. Michalski ,I.Bratko, M.Kubat, Eds., pp.409-428, John Wiley & Sons, 1998.

[13] R. Agrawal, C. Faloutsos, and A. Swami, "Efficient similarity search in sequence databases," Proceedings of the 4th Conference

on Foundations of Data Organization and Algorithms, 1993.

[14] T. Fu, T-c. Fu, F. I. Chung, V Ng, and R. Luk, "Pattern Discovery from Stock Time Series Using Self-Organizing Maps," Notes KDD 2001 Workshop Temporal Data Mining, pp.27-37, 2001.

[15] U. M. Fayyad and K. B. Irani, "Multi-Interval Discretization of continuous-valued Attributes for Classification Learning," Proc. 13th Int'l Joint Conference of Artificial Intelligence, pp.1022-1027, 1993.

저 자 소 개

박 상 호(Sang-Ho Park) 정회원

- 2002년 2월 : 인하대학교 전자전기컴퓨터 공학부(공학사)
- 2004년 2월 : 인하대학교 전자계산공학과(공학석사)
- 2004년 3월 ~ 현재 : 인하대학교 정보 공학과 박사과정

<관심분야> : 시계열 데이터 마이닝, 패턴 인식

이 주 홍(Ju-Hong Lee) 정회원

- 2001년 2월 : 한국과학기술원 정보 & 통신공학과(공학박사)
- 2001년 ~ 현재 : 인하대학교 IT 공과 대학 컴퓨터정보공학부 교수

<관심분야> : 데이터 마이닝, 데이터베이스, 정보검색