분산환경에서 데이터 큐브와 신경망을 이용한 데이터마이닝기법

박민기*, 바비제라도*, 이재완*
*군산대학교

Data Mining mechanism using Data Cube and Neural Network in distributed environment

Mingi Park*, Bobby D. Gerardo*, Jaewan Lee*
*Kunsan National University
E-mail : sopiru@kunsan.ac.kr

요 약

본 논문에서는 분산환경에서 데이터마이닝을 효율적으로 하기 위해 데이터 일반화과정으로 데이터 일반화 기법과 데이터 큐브구성 기법을 제안하였다. 그리고 일반화 과정 이후 생성된 데이터 큐브로부터 가장 유용한 데이터를 찾기 위한 방법으로 신경망의 전통적인 자기형상화기법을 응용한 동적 자기구성 지도기법을 제안하였고 이를 위한 시스템 구조를 설계하였다.

ABSTRACT

In this paper, we proposed data generalization and data cube mechanism for efficient data mining in distribute environment. We also proposed active Self Organization Map applying traditional Self Organization Map of Neural network for searching the most informative data created from data cube after the generalization procedure and designed the system architecture for that.

키워드
데이터마이닝, 데이터일반화, 데이터큐브, 신경망, 분산환경

Ⅰ. 서 론

고객의 선호도나 성향을 분석하여 고객이 만족하는 서비스를 제공하지 않으면 경쟁력을 잃게될 정도로 기업 간 시장 경쟁은 가속화되고 있고 대용량의 데이터와 다양한 형태의 데이터 분석에 한계가 있음에도 불구하고 동적으로 지식을 추출해 내는 방안에 대한 요구가 급속히 증가하고 있다. 대용량 데이터로부터 유용한 정보를 찾아내는 과정이 마치 광산에 묻혀 있는 금을 캐내는 것과 유사하다고 해서 대용량 자료로부터 정보를 얻어내는 과정에 사용되는 시스템과 분석 기법을 데이터 마이닝(Data Mining)이라 한다.[1]

데이터 마이닝을 적용하기 앞서 데이터 일반화 과정을 통해 해결해야 할 일들이 있으며 이는 접근 데이터에 대한 모델 생성, 관련 없는 데이터나 잡음을 삭제하는 데이터 정제와 여과, 개별적인 데이터를 트랜잭션 단위로 묶는 그룹화, 이용자 등록정보와 같은 다른 데이터와의 통합 등의 일반화 과정이 요구되며 이와 같은 일반화 과정을 통해 우리는 더욱 효율적으로 데이터 마이닝을 할 수 있으므로 효율적인 데이터 일반화 과정에 대한 연구가 필요하다. [2][3][4]

이를 위해 본 논문에서는 기존의 일반화 과정을 두 개의 단계로 구성하여 수많은 데이터로부터 데이터를 일반화하는 과정과 일반화 된 데이터를 데이터 큐브로 구조를 구성하여 능률적으로 데이터 마이닝을 할 수 있게 하였고 신경망을 이용하여 동적으로 데이터 마이닝을 하기 위해 동적 자기구성 지도기법을 제안하였다.

II. 관련 연구

1. 데이터 일반화 및 데이터 큐브

데이터 마이닝 알고리즘을 적용하기 앞서 데이터 일반화 과정을 통해 해결해야 할 일들로는 접근 데이터에 대한 모델 생성, 관련 없는 데이터나 잡음을 삭제하는 데이터 정제와 여과, 개별적인 데이터를 트랜잭션 단위로 묶는 그룹화, 이용자 등록정보와 같은 다른 데이터와의 통합 등이 있다.[5]

데이터 큐브는 데이터 베이스로부터 추출한 데이터를 처리할 때 유용한 여러 가지 방법과 데이터 마이닝을 위한 연산작업을 빠르게 수행할 수 있도록 한다.[6]

데이터 큐브는 데이터 일반화 과정에서 생성된 데이터로부터 생성할 수 있으며 데이터 큐브는 여러 개의 차원으로 구성될 수 있고 각각은 데이터 큐브에 따라 큐보이드들로 재구성되어 데이터 마이닝을 할 때 여러 가지 데이터 큐브들을 구성하며 데이터 마이닝을 위한 최적의 상태를 갖는다.

2. 데이터 마이닝을 위한 신경망

신경망 모형은 데이터 마이닝에 대한 관심이 모아지면서 가장 일반적으로 언급되어지고 또한 다양한 응용 분야를 가지고 있는 기법이다. 신경망 모형은 인간이 경험으로부터 학습해 가는 두뇌의 신경망 활동을 흉내내어 자신이 가진 데이터로부터의 반복적인 학습 과정을 거쳐 패턴을 찾아내고 이를 일반화함으로써 특히 향후를 예측(Prediction)하고자 하는 문제에 유용하다.

매우 복잡한 구조를 가진 데이터들 사이의 관계나 패턴을 찾아내는 유연한 비선형 모형(Flexible nonlinear Model)의 하나로, 신경생리학과의 유사성 때문에 일반적으로 다른 예측모형에 비해 흥미롭게 여겨지고 있다. 비교사학습에서 코호넌 맵(Kohonen maps)을 이용하여 데이터의 클러스터링과 분류 작업을 수행하는데 쓰이기도 한다.[7][8]

신경망 모형은 상당히 다양한 산업분야의 다양한 문제에 적용될 수 있고, 입력변수와 목적변수의 관계를 그리기가 어려운 복잡한 데이터에 대해서도 좋은 결과를 주는 것으로 알려져 있다.

III. 데이터 마이닝 시스템 구조

본 시스템은 데이터마이닝을 위해 크게 두기지 부분으로 나누어지고 각각은 아래 그림1에서와 같이 데이터 일반화 과정과 데이터 마이닝 과정으로 나누어진다. 또한 데이터 일반화 과정에서는 다량의 데이터를 일반화시키는 단계와 일반화된 데이터를 효율적으로 처리하기 위해 데이터 큐브로 구조를 생성하며 두 번째 단계인 데이터 마이닝에서는 구조화된 데이터 큐브들로부터 데이터 특성에 맞는 값을 선택해 사용자가 원하는 정보를 제공한다.

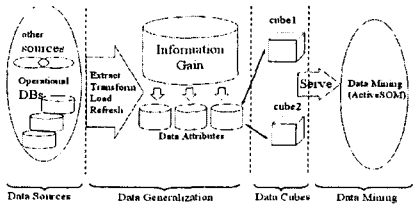

그림 4 . 데이터 마이닝 시스템 구조

이 과정은 데이터마이닝의 일반화 과정으로 각 단계를 거쳐 지식을 발견해 낸다. 적용하는 두 가지 기법은 정보획득기법(*Information Gain*)과 데이터 큐브(Data cube) 생성기법 이며, 데이터 큐브생성에 앞서 일반화를 먼저 수행시켜 전체 문서집합의 개요를 획득하고 데이터 큐브를 위한 판단기준을 얻어낸다.

1. 데이터 일반화(Data Generalization)

데이터 베이스로부터 데이터를 일반화하는 과정은 다음과 같다 . 먼저 훈련 샘플을 S라 하고 각각의 샘플은 튜플에 영향을 미치며 하나의 속성(attribute)은 훈련 샘플의 클래스를 결정하는데 사용된다. 속성 상태로 각각의 클래스 레벨로 결정할 수 있으며 이를 위한 순서는 다음과 같다.

(1) m개의 클래스들이 있다고 가정하자. S는 클래스 Ci의 si샘플들을 포함하며 속성들은 확률 $\frac{s_i}{s}$를 갖는 클래스 Ci에 포함되고 이 때 s는 S의 전체 샘플 수이다.

(2) 주어진 샘플을 분류하기 위한 정보는 다음과 같다.

$$I(s_1, s_2, \ldots, s_m) = -\sum_{i=1}^{m} \frac{s_i}{s} \log_2 \frac{s_i}{s}$$

(3) $\{a_1, a_2, \ldots, a_v\}$값을 지닌 속성 A는 S를 $\{S_1, S_2, \ldots S_v\}$집합으로 분할하는데 사용되며 S_j는 A의 a_j값을 갖는 S의 샘플들을 포함하고 S는 C_i 클래스의 s_{ij}샘플들을 포함한다. 이때 A에 의해 분할된 것을 기초로 하여 생긴 정보는 A의 엔트로피라고 하고 다음 식으로 구한다.

$$E(A) = \sum_{j=1}^{v} \frac{s_{1j} + \ldots + s_{mj}}{s} I(s_{1j}, \ldots, s_{mj})$$

(4) A에 분할을 하여 얻은 정보획득(Information Gain)은 다음과 같이 정의할 수 있다.

$$Gain(A) = I(s_1, s_2, \ldots, s_m) - E(A)$$

이 접근방법으로 우리는 S에 있는 샘플들을 정의하고 있는 각각의 속성에 대한 정보 획득을 계산할 수 있고 가장 정보 획득이 높은 속성을 주어진

환경에서 가장 식별할 수 있는 속성으로 간주한다. 그러므로 우리는 각각의 속성에 대해 정보획득을 계산하여 속성들의 순위를 얻을 수 있고 이 순서는 개념 설명에 사용되는 속성을 선택하기 위한 일반화에 사용될 수 있다.

데이터 일반화 과정에서 데이터 큐브를 생성할 수 있으며 생성된 데이터 큐브로부터 다시 큐보이드들을 재구성한다. 이 때 큐보이드들은 서로 다른 그룹에 의해 표현된 값을 갖고 있으며 가장 아래에 있는 기저큐보이드 (base cuboid) 또는 3-D큐보이드는 다음 그림2에서와 같이 생산품, 날짜, 지역을 포함하고 있다. 그리고 가장 위에 있는 큐보이드는 정점큐보이드(apex cuboid) 또는 0-D 큐보이드로서 공백으로 그룹이 생성된 경우이고 모든 판매의 전체 합을 나타낸다.

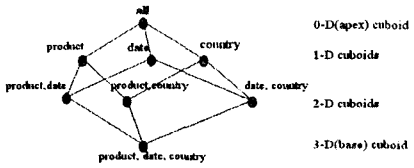

그림 5 . 데이터 큐브 구조

IV. 신경망을 이용한 데이터 마이닝 구조

데이터 일반화 과정을 통해 생성된 데이터 큐브로부터 신경망의 자기구성지도(SOM)를 이용하여 다음 그림과 같이 동적 데이터 마이닝을 할 수 있으며 이를 위한 데이터 마이닝 시스템 구조는 그림 3과 같다.

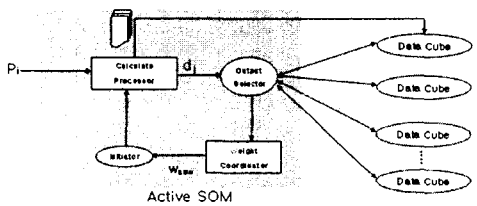

그림 6 . 데이터 마이닝 구성도

동적 데이터 마이닝 순서는 먼저 초기자가 일반화 과정에서 생긴 데이터 큐브와의 연결강도를 임의의 수로 초기화한다. 계산처리기에서는 동적인 구조를 위해 참조 리스트를 이용해 이미 처리된 데이터 마이닝 과정이 있을 경우 이를 참조하여 직접 데이터 큐브를 선택하고 처음 입력 과정일 경우에만 이 입력 데이터 패턴과 연결강도를 이용해 입력 데이터 패턴을 처리할 데이터 큐브 패턴의 거리를 계산하고 출력선택자는 계산된 데이터 큐브 패턴의 거리들 가운데 최소 거리에 있는 것을 최종 데이터 큐브로 선택한다. 데이터 큐브 패턴과 그 이웃들의 연결강도를 재조정하기 위해 연결강도 조정자에 선택된 데이터 큐브 패턴을 전달한다. 마지막으로 조정된 연결강도값을 초기자에게 전달하여 연결강도를 새로이 조정하며 새로운 입력 데이터 패턴이 들어왔을 경우 위 과정을 반복한다.

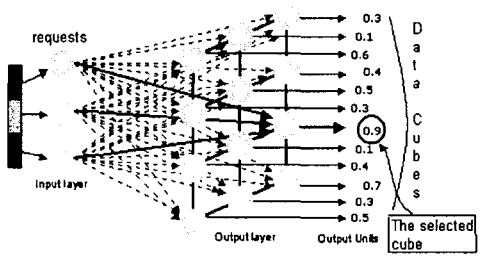

그림 7 . 학습을 통해 선택된 데이터 큐브

이 과정을 큰 변화가 없을 때까지 반복한다. 이 학습과정을 통해 그림4와 같이 사용자 요구에 대해 가장 연결 강도가 큰 값을 지닌 데이터 큐브를 최종적으로 선택하게 된다.

데이터 마이닝을 위한 각각의 구체적인 구성요소는 다음과 같다.

- Initiator : 연결강도 초기자는 연결강도를 작은 값의 임의의 수로 초기화한다. 그리고 계산처리기에 초기화한 연결강도를 계산 처리기에 전달한다.
- Calculator Processor : 계산 처리기는 입력된 데이터 패턴을 조사하여 이미 데이터 마이닝 과정이 이루어 졌을 경우 목적지데이터 큐브로 전송되며 그렇지 않고 새로운 입력이었을 경우 이 패턴과 연결강도를 이용하여 모든 패턴간의 거리를 구하는 식에 의해 계산하고 그 결과를 출력 선택자에 전송한다.
- Output Selector : 출력 선택자는 최소 거리에 있는 서버 패턴을 선택하여 입력 데이터 패턴을 선택한 서버로 전송하며 이 때 선택된 서버와 이웃들간의 연결강도를 재조정하기 위해 연결강도 중재자에 전송한다.
- Weight Coordinator : 연결강도 중재자는 선택된 서버 패턴과 이웃들의 연결강도를 구하는 다음 식에 의해 재조정하고 새로운 연결강도(Wnew)를 연결강도 초기자에 전송하여 연결 강도를 재 설정한다.

SOM의 가장 큰 특징은 뉴런들이 규칙적인 격자상에 분포되어 있고, 학습되는 과정에서 특정한 입력에 대하여 어떤 하나의 뉴런과 그 주변의 뉴런들이 가진 가중치들이 갱신되어, 뉴런들간에 어떤 구별을 만들어 간다는 것이다. 이와 같은 자기 구성지도는 학습 단계에서 전방 전달(feedforward) 방법을 사용하므로 연속적인 학습이 가능하며 동적

인 변화에 대응할 수 있다는 장점이 있다.

V. 결론 및 향후 연구 방향

본 논문에서는 분산환경에서 효율적으로 데이터 일반화 과정을 위한 기법을 제안하였으며 일반화 과정을 통해 생긴 데이터 큐브로부터 데이터를 마이닝 할 수 있도록 신경망을 이용한 동적 자기구성 지도 기법을 설계하였다. 이러한 신경망 모형은 고객의 신용평가, 불량거래의 색출, 의료진단예측, 우량고객의 선정, 타깃 마케팅의 여러 주제 등을 비롯한 여러 분야에 적용될 수가 있는데, 주로 교사학습에 적용되어 목적변수(target)에 대한 예측(Prediction)이나 분류(Classifica tion)를 목적으로 감춰진 패턴을 찾고 이를 일반화하는데 이용된다.

향후 연구 방향으로는 시뮬레이션을 통해 본 논문에서 제시한 일반화기법과 데이터 큐브구성 그리고 동적 자기구성 지도의 기법을 평가 할 것이며 평가항목으로는 데이터의 특성에 따른 정확도와 가장 적합한 데이터를 산출할 때까지의 총 동작 시간을 산출하는 연구가 필요하다.

참고 문헌

[1] Mark W. Craven, Jude W. Shavlik. "Using Neural Networks for Data Mining" Futer Generation Computer Systems.

[2] 데이터베이스연구회, 데이터 마이닝, pp. 103-127, 1998.

[3] 나민영, 대규모 지식데이터베이스에서 유용한 지식 추출하는 기법, http://www.dp c.or.kr/dbworld/document/9709/spec.html.

[4] 박민기, 김귀태, 이재완 "분산환경에서 신경망을 응용한 데이터서버 마이닝", 춘계해양정보통신학회 학술지 Vol.7, No.1 pp.473-476, 2003.05.

[5] Apte, C., and Hong, S. J. Predicting Equity Reurns from Securities Data with Minimal Rule Generation. In Advances in Knowledte Discovery and Data Mining, eds. U. Fayyad, G. Piatetsky-Shapiro, P.Smyth, and R. Uthurusamy, Menlo Park, Calif.:AAAI Press.pp.514-560, 1996.

[6] Jiawei Han and Micheline Kamber, "Data Mining : Concepts and Techniques" Morgan Kaufmann, Academic press.pp.71-79, 2001.

[7] Mark W. Craven, Jude W. Shavlik. "Using Neural Networks for Data Mining" Futer Generation Computer Systems.

[8] Bishop, C. M. Neural Betworks for Pattern Recognition. Oxford University Press, Oxford, England, 1996.