적응적 IOLIN시스템을 사용한 Concept Drift가 있는 데이터 스트림의 분류

김재우*, 이주홍*, 홍준식**
*인하대학교 컴퓨터 · 정보공학과
**청주대학교 전자정보공학부
e-mail: *crazynad@datamining.inha.ac.kr, juhong@inha.ac.kr,
**jmskhong@cju.ac.kr

Concept-Drifting Data Streams classification using Adapted IOLIN System

Jae-Woo Kim*, Ju-Hong Lee*, Jun-Sik Hong**
*Dept of Computer Science & Information Engineering, Inha University
**School of Electronic Information Engineering, Cheongju University

요 약

스트림 데이터를 분류하는 문제는 데이터 스트림 마이닝 분야에서 가장 넓게 연구되고 있는 항목이다. 실세계에서의 데이터 스트림을 분류하는데 있어서 본질적인 문제점들이 있다 : 1)많은 양의 데이터가 불규칙적으로 빠르게 입력되는 것과, 2)유동적 컨셉트로 알려진, 데이터의 분류가 시간에 따라서 유동적으로 변하는 문제이다. 본 논문에서는 위와 같은 문제를 해결하기 위해서 적응적 OLIN시스템을 제안한다. 제안된 시스템은 지역적인 유동적 컨셉트뿐만 아니라 전역적인 유동적 컨셉트 문제까지 고려하여, 기존의 시스템보다 향상된 성능을 보였다.

1. 서론

기존의 대부분 데이터 마이닝 기법들은 정적 데이터를 이용하여 분류, 군집, 연관, 예측 방법들에 대한 연구를 진행하였다. 그러나 최근 네트워크와 통신기술의 발달로 데이터의 양이 방대해짐에 따라, 데이터의 실시간 처리에 대한 연구가 요구되었다. 이러한 데이터의 실시간 처리는 데이터 마이닝 작업을 위한 데이터 스트림에 대한 사전작업을 의미하며, 많은 시간을 요구하는 데이터 마이닝 작업의 학습 능률을 향상시킬 수 있다. 기존 데이터 스트림에 대한 연구는 다음과 같다[1][2][3][4][5][6][8][10].

그중에서 데이터 스트림을 분류하는 문제는 데이터 스트림 마이닝 분야에서 가장 넓게 연구되고 있는 항목이다. 데이터 스트림을 분류하는데 있어서 현재 두 가지 중요한 문제점이 연구되고 있다. 1)데이터 스트림이 거대한 양으로 끊임없이 불규칙적으로 들어오기 때문에 이러한 데이터를 실시간적으로 빠르게 처리해야하는 문제와 2)데이터가 시간이 경과함에 따라서 그 목적하는 방향이 변화하는 문제이다. 유동적 컨셉트(concept dirft) [10], 혹은 데이터 스트림 진화(data stream evolution) [1]로 불리는 이 변화는, 데이터 스트림에서 이전의 데이터들로부터 생성된 모델을 이용하여 새롭게 유입되는 데이터를 분류할 경우, 갑작스럽게 기존의 정확률에 비해 현저하게 낮은 정확률을 갖는 것을 의미한다.

유동적 컨셉트를 다루는 기법은 크게 세 가지로 구분 된다[10]. 첫째, 인스턴스 선택(instance selection)학습은 현재 컨셉트에 적합한 인스턴스를 선택하는 기법이다. 보통 슬라이딩 윈도우를 사용하여 현재의 컨셉트를 학습한 후 다음의 데이터를 예측한다. 둘째, 인스턴스 가중치(instance weighting)학습을 사용하여 인스턴스에 가중치를 주는 기법이다. 가중치를 주는 요건으로는 시간이 지남에 따라 이전의 데이터에 가중치를 작게 주는 방법을 사용하거나, 현재 컨셉트에 가장 적합한 인스턴스에 가중치를 주는 방법을 사용한다. 가장 일반적인 예로 지지기반벡터(Support Vector Machine)학습 방법을 들 수 있다. 셋째, 앙상블 학습(ensemble learning)은 여러 개의 학습 기계를 두어서 나온 각각의 결과를 가지고 투표를 통하여 최적의 결과를 결정하는 학습기법이다. 인스턴스 가중치기법은 인스턴스 선택기법보다 유동적 컨셉트를 학습하는데 있어서 더 안 좋은 결과를 내는 경우가 많은데, 이는 과최적화(overfitting)가 발생 할 확률이 높기 때문이다[8]. 또한, 앙상블 학습은 높은 정확도를 보여주는 대신에, 학습에 걸리는 시간이 너무 길다.

유동적 컨셉트는 지역적인 발생과 전역적인 발생으로 나눌 수 있다[10]. 지역적인 유동적 컨셉트는 어느 한 속성의 변화가 급격히 일어나 전체적인 정확률을 떨어뜨리는 현상으로, 유동적 컨셉트는 일반적으로 지역적으로 발생한다. 전역적인 유동적 컨셉트는 모든 속성에 대해서 변화가 발생하는 현상이다.

본 논문에서는 인스턴스 선택학습의 한 방법인 온라인 정보망(On-Line Information Network) 시스템[9]을 기반으로 한 점진적 온라인 정보망(Incremental On-Line Information Network)을 사용하여 유동적인 컨셉트 비정형 스트림 데이터(nonstationary

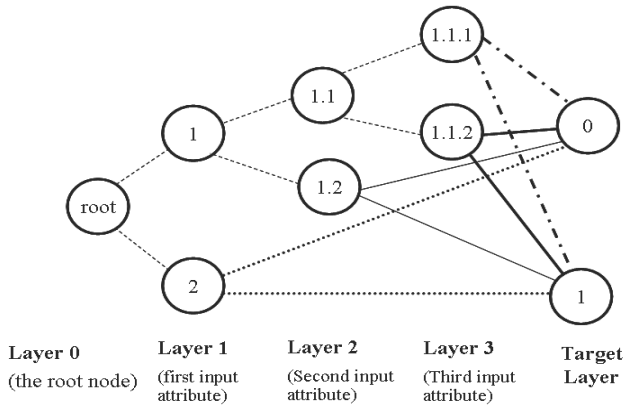
stream data)를 분류하는 새로운 방법을 제안한다. 본 논문에서 제안하는 방법은 Info-Fuzzy Network[9](이하 IFN)를 기반으로 한 온라인 학습 방법으로, 기존의 일반적인 트리알고리즘보다 더 적은 층을 구성하는 장점을 갖는다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 데이터 스트림에서의 분류방법과 유동적 컨셉트를 적용한 분류방법을 소개하고 각각의 장단점을 기술한다. 3장에서는 본 논문에서 제안하는 컨셉트에 적응적인 점진적 온라인 정보망 알고리즘을 기술한다. 4장에서는 성능평가를 기술하고, 그에 따른 결론 및 향후연구는 5장에서 보여준다.

2. 관련 연구

데이터 스트림이라는 연구 분야가 새롭게 제시되었을 때에는 데이터의 빠른 처리가 중요한 연구주제였다. Doming & Hulten [5]은 VFDT(Very Fast Decision Tree)시스템을 제안하였는데, VFDT 시스템은 트리를 기반으로 한 정형(stationary) 스트림 데이터의 부분 추출로 이루어진다. 원패스(One-Pass) 알고리즘을 적용하기 때문에 매우 빠른 실행시간(run-time)을 가진다. 하지만 유동적 컨셉트에 의해 갑작스럽게 정확률이 떨어지는 문제가 스트림 데이터 분류에서 제기되었고, 이에 대한 많은 논문들이 연구주제로 다루어지게 되었다[3][4][6][7][10]. 스트림 데이터의 유동적 컨셉트를 다루기 위해서 Hulten 등은 [6]은 VFDT알고리즘을 확장한 CVFDT(Concept adapting Very Fast Decision Tree)알고리즘을 제안하였다. 이 알고리즘은 새로운 데이터가 입력될 때 기존의 의사결정나무를 데이터에 맞게 수정하는 방법으로, 좀 더 정확한 데이터 분류를 한다.

Widmer & Kubat [11] 은 데이터 분류의 정확도에 따라서 슬라이딩 윈도우의 크기를 조절하는 프레임워크를 제안했다. FLORA라고 불리는 이 시스템은 유동적 컨셉트가 발생했을 때, 윈도우의 크기를 줄이고 안정화 되어 있다면 윈도우의 크기를 유지하거나 늘린다. 유동적 컨셉트가 발생하지 않는 환경에서 더 좋은 성능을 보여준다.



(그림 1) 3계층 구조의 Info-fuzzy 망

Last [8] 는 분류 모델로서 IFN을 온라인상에서 구현한 온라인 정보망(Online Information network)시스템을 제안하였다. (그림 1)은 온라인 정보망 시스템의 기본 구성요소인 IFN 의 구성을 보여준다. IFN은 정보 이론(Information Theory)을 사용한 방법들 중의 하나로서 Last & Maimom [9]이 발전시켰다. 예측되는 속성을 최소화하도록 설계된 이 시스템은 입력 값들과 출력 값들 사이의 상호정보(Mutual Information)를 계산하기 위하여 다중 계층 망(multi-layered network)을 구축한다. 이 알고리즘의 결과물은 네트워크로서 의사결정나무(decision tree)와 같이 목표 속성(target attribute)을 예측하는데 사용된다. 의사결정나무와 IFN의 차이점은 모든 단말노드(leaf node)들은 목표 층(target layer)의 모든 노드들과 네트워크로 연결된다는 점이다. (그림 1)에서 노드 2, 1.2, 1.1.1, 1.1.2는 목표 층의 노드 0, 1과 연결되어 있어서 데이터를 분류하는데 중요한 역할을 한다.

IFN을 기반으로 한 OLIN시스템은 비정형 윈도우를 사용하여 연속적인 스트림 데이터를 처리한다. 유동적 컨셉트(예기치 못하게 분류 정확도가 떨어지는 현상)가 발생하면 동적으로 윈도우의 크기를 줄여서 대처한다. 윈도우 크기의 계산은 정보 이론(Information Theory)과 통계적 측정을 통해 계산한다. 그 결과 비정형 데이터 스트림에서, 동적인 윈도우 크기를 사용한 시스템이 고정된 크기를 가진 윈도우보다 더 높은 정확률을 보였다. 하지만 온라인 정보망 시스템의 단점은 새로운 데이터가 입력될 때마다 유동적 컨셉트에 상관없이 새로운 네트워크를 구축하기 때문에 실행시간이 긴 단점이 있다.

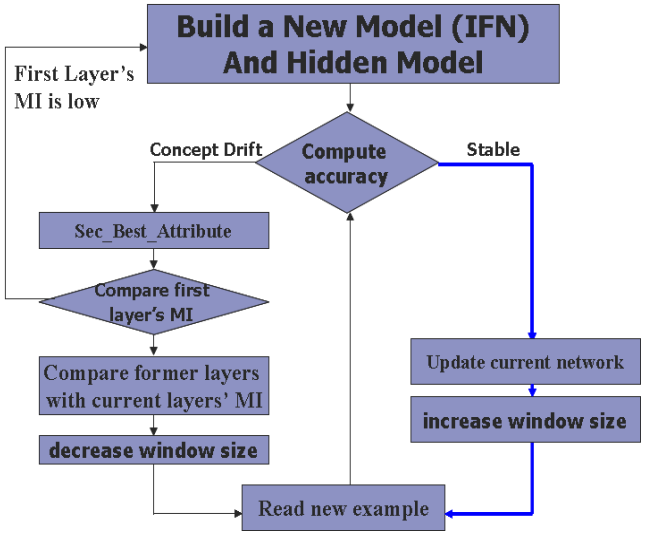
Cohen [4]은 OLIN 시스템의 성능을 향상시키기 위하여 점진적 온라인 정보망을 제안하였다. 이 시스템은 기존 시스템의 단점인 실행시간을 단축시키는 알고리즘을 제안하였는데, 유동적 컨셉트가 발생한다면 새로운 네트워크를 구축하는 점은 같지만, 유동적 컨셉트가 발생하지 않는다면 기존의 네트워크를 가지고 새로운 데이터를 통해 갱신함으로써 실행시간과 정확률 사이의 교환을 통해 상대적으로 낮은 정확률의 저하와 더욱 빠른 실행시간을 가짐을 보였다.

Cohen [3]은 점진적 온라인 정보망 시스템에 기반을 둔 다중 모델(Multi Model)의 점진적 온라인 정보망과 진보한(Advanced) 점진적 온라인 정보망 시스템을 제안하였다. 다중 모델의 점진적 온라인 정보망은 데이터의 잠재적인 주기성을 이용하여 이전의 만들어진 모델들을 재사용하는 것이다. 유동적 컨셉트가 발생했을 때, 이전에 만들어진 모델에서 목표 속성(Target Attribute)의 상호 정보 값을 비교하여 유사한 모델로 대체하고 만약 유동적 컨셉트가 다시 발생된다면, 새로운 모델을 만드는 방법이다. 하지만 잠재적인 주기성을 이용하기 위하여 이전에 만들어졌던 모델들을 메모리에 저장하기 때문에 메모리의 요구량이 많아지므로 메모리에서 모델들의 효율적인 삭제 작업이 필요하다. 진보한 점진적 온라인 정보망은 하향식 접근방법(top-down approach)을 사용하여 각각의 층에서 이전의 시스템과 현재의 데이터로 만든 시스템의 조건적인 상호 정보(conditional mutual information)를 비교하여 이전의 시스템의 층보다 현재의 시스템의 층이 조건적인 상호 정보가 더 크다면 그 층의 조건적인 상호 정보를 교체하는 알고리즘이다. 그러나 유동적 컨셉트에 상관없이 층간의 조건적인 상호 정보를 비교하기 때문에 그만큼 평균 실행시간이 늘어난다. 또한, 유동적 컨셉트가 전역적으로 발생한다면 각 층

의 노드를 교체하는데 드는 시간이 새로운 모델을 만드는 시간보다 더 길어질 수도 있다.

3. 컨셉트에 적응적인 점진적 온라인 정보망

본 논문에서 제안하는 방법은 점진적 온라인 정보망[4]에 기반을 둔다. 점진적 온라인 정보망의 기본 방법은 컨셉트가 안정화되어 있는 동안에는 현재의 모델을 갱신시킨다. 만약 유동적 컨셉트가 발생한다면 새로운 네트워크를 구축한다. 점진적 온라인 정보망은 유동적 컨셉트가 발생하면 모델을 재구축하기 때문에 실행시간이 길어지는 단점이 있다.



(그림 2) 컨셉트에 적응적인 점진적 온라인 정보망

(그림 2)는 컨셉트에 적응적인 점진적 온라인 정보망을 나타낸다. 본 논문에서 제안하는 방법은 유동적 컨셉트가 전역적으로 발생하지 않고 지역적으로 발생했을 때, 기존의 모델을 향상시키는 방법이다. 만약 유동적 컨셉트가 발생했을 때, 첫 번째 층의 조건적인 상호정보가 다른 속성들의 조건적인 상호정보보다 작다면 새로운 모델을 만들고, 반대로 첫 번째 층의 조건적인 상호정보가 기존의 속성들보다 크다면, 기존의 각 층의 상호정보를 현재 데이터에 기반한 상호정보와 비교하여 교체하는 알고리즘이다. 아래의 <표 1>은 유동적 컨셉트가 발생했을 때 동작하는 알고리즘이다.

<표 1> 컨셉트에 적응적인 점진적 온라인 정보망

Concept Adapted IOLIN
Input: *Training window, current network model, conditional mutual information of each network layer i(Former_Conditional_MI(i)*
Output: *Update or re-generated network model*

```
For each new training window
  If concept drift is detected
    Calculate Sec_Best_Attribute
      If first layer's attribute's is higher than Sec_Best_Attribute
        For each Layer i in existing network
          Calculate Cond_MI(i)
            IF Cond_MI(i)≥Former_Conditional_MI(i)*0.95
              Keep i^th layer as is and move to the next layer
            Else
              continue the network construction by adding new layers
        If reached the last layer
          New_Split_Validity(IFN)[4]
            Save value: Former_Condtional_MI(i) = Conditional_MI(i)
        Else
            create new IFN Model
    Else //concept is stable
        Update_Current_Network (IFN)[4]
  Calculate New_Training_Window_Size(W)[8]
```

Sec_Best_Attribute 함수는 현재 네트워크 모델의 모든 속성들 중에서 가장 높은 상호정보를 갖는 속성을 찾는다.

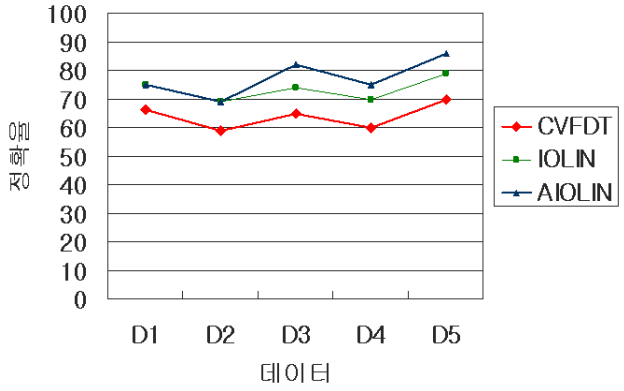
*Update_Current_Network(IFN)*함수는 가장 마지막 층의 속성보다 더 높은 상호정보 값을 갖는 속성이 있다면 더 높은 속성으로 교체하고 *Check_Split_Validity*함수를 실행한다.

*Check_Split_Validity*함수는 층이 더 나누어 질 수 있는지를 검사하여 더 나누어진다면 새로운 층을 갱신한다. 마지막으로 *New_Training_Window_Size*함수는 새로운 슬라이딩 윈도우의 크기를 계산하는 함수이다.

본 논문에서 제안하는 알고리즘은 유동적 컨셉트 상황에서 정보네트워크의 첫 번째 층의 상호정보를 비교함으로써 모델을 재구축할 것인지 부분적인 갱신을 할 것인지를 판단한다. 첫 번째 층을 이루는 속성은 정보네트워크를 이루는 정확률의 근간이라 할 수 있다. 유동적 컨셉트 상황에서 각 층의 상호정보를 비교함으로써 지역적인 유동적 컨셉트에 대처할 수 있고, 정확률을 높일 수 있다. 또한 Concept 가 안정화되어 있는 상황에서 모델을 갱신함으로써 정확률을 유지할 수 있다.

4. 성능평가

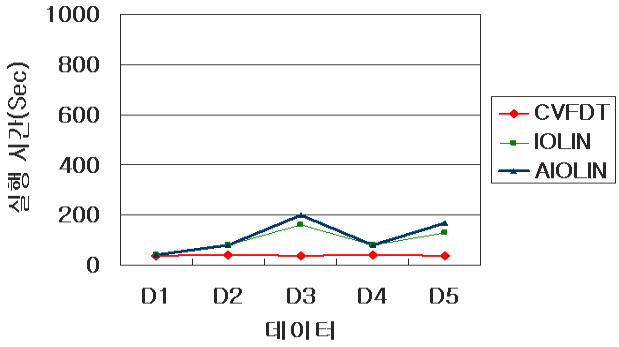
본 장에서는 컨셉트에 적응적인 점진적 온라인 정보망의 성능을 평가한다. 평가방법은 점진적 온라인 정보망과 CVFDT알고리즘을 수행하여 정확률과 실행시간의 비교를 통하여 이루어진다. 실험 환경은 Pentium 4 D 3.2G프로세서와 1GB 램, Windows XP Professional 운영체제와 150GB 하드를 사용하였다. 개발언어는 자바를 사용하였다.



(그림 3) 정확률 측정 실험 결과 비교

첫 번째 실험 평가에서는 데이터 스트림 분류의 정확률

을 측정 하였다. (그림 3)은 정확률의 결과이다. D2, D4에서 유동적 컨셉트가 발생한 후, 오히려 정확률이 증가하는 것을 볼 수 있다. 이는 기존의 점진적 온라인 정보망보다 좀 더 향상된 성능을 보여준다. CVFDT알고리즘은 실험에서 가장 낮은 평균 정확률을 보였다.



(그림 4) 실행시간 측정 실험 결과 비교

두 번째 실험 평가에서는 데이터 스트림 분류의 실행시간을 비교하였다. (그림 4)는 실행시간의 결과를 보여준다. CVFDT 알고리즘이 가장 빠른 실행시간을 보여주었는데, 이는 데이터가 중복실험이 아닌 원패스(one-pass) 알고리즘을 사용했기 때문이다. 유동적 컨셉트가 발생했을 때, 점진적 온라인 정보망이 컨셉트에 적응적인 점진적 온라인 정보망보다 실행시간이 긴 이유는 각각의 층의 상호정보를 비교하기 때문에 좀 더 많은 데이터 처리가 필요하기 때문이다.

5. 결론 및 향후 연구

본 논문에서는 유동적 컨셉트를 처리하는 적응적인 점진적 온라인 정보망 알고리즘을 제안하였다. 이는 기존의 문제점들을 개선하고 유동적 컨셉트가 발생했을 때, 더 좋은 성능을 가짐으로서, 보다 정확하고 빠른 데이터 스트림 분류를 제공한다. 성능평가에서 기존의 점진적 온라인 정보망보다 빠르고 높은 정확률을 보여주었다.

향후 연구는 좀 더 정확률을 높인 연구가 진행되어야 할 것이며, 데이터의 원패스 처리방법도 연구가 필요하다. 현재 IFN을 이용한 앙상블 모델도 구현 중에 있다.

참고문헌

[1] C. Aggarwal, A Framework for Diagnosing Changes in Evolving Data Streams. Proceedings of the ACM SIGKDD Conference, 2003.

[2] C. Aggarwal, Data Streams: Models and Algorithms, 354Page, Springer, 2007.

[3] L. Cohen, G. Avrahami, M. Last, A. Kandel, and O. Kipersztok, “Incremental Classification of Nonstationary Data Streams”, Proceedings of the Second International Workshop on Knowledge Discovery in Data Streams, pp. 117-124, October 7, 2005, Porto, Portugal.

[4] L. Cohen, M. Last, G. Avrahami, “Incremental Info-Fuzzy Algorithm for Real Time Data Mining of Non-Stationary Data Streams”, TDM Workshop, Brighton UK, 2004.

[5] P. Domingos and G. Hulten. “Mining high-speed data streams” In Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Pages 71-80, Boston, MA, 2000. ACM Press.

[6] G. Hulten, L, Spencer, and P. Domingos, “Mining Time-Changing Data Streams”, Proc. of KDD 2001, pp. 97-106, ACM Press, 2001.

[7] R. Klinkenberg, Learning drifting concepts: example selection vs. example weighting, Intelligent Data Analysis, Special Issue on Incremental Learning Systems Capable of Dealing with Concept Drift, 8 (3), 2004.

[8] M. Last, “Online Classification of Nonstationary Data Streams”, Intelligent Data Analysis, Vol. 6, No. 2, pp. 129-147 2002.

[9] O. Maimon and M. Last, Knowledge Discovery and Data Mining - The Info-Fuzzy Network (IFN) Methodology, Kluwer Academic Publishers, December 2000.

[10] A. Tsymbal The problem of concept drift: definitions and related work, Technical Report TCD-CS-2004-15, Department of Computer Science, Trinity College Dublin, Ireland, 2004.

[11] G. Widmer and M. Kubat, Learning in the Presence of Concept Drift and Hidden Contexts, Machine Learning, Vol. 23, No. 1, pp. 69-101, 1996.