

USN Middleware에서 센서데이터 패턴분석을 위한 센서데이터 관리 계층*

조운형^o 김영국^{**}

충남대학교 컴퓨터공학과
lunareclipse@cnu.ac.kr ykim@cnu.ac.kr

Sensor Data Management Tier for Analyzing Sensor Data Pattern in USN Middleware

Wn-hyoung Cho^o Young-kuk Kim

Department of Computer Engineering Chungnam National University

요 약

USN 환경은 다양한 센서들을 이용하여 사용자에게 보다 다양한 서비스를 제공하는 것을 목표로 한다. 이러한 USN 환경에 존재하는 센서들은 이종이며 무수한 응용들과 요구사항을 충족하기 위한 다양한 센서 네트워크들과 응용들 사이에서 동작하는 미들웨어가 반드시 필요하다. 기존의 미들웨어에서는 이기종 센서들 간의 통합에 중점을 두고, 센서데이터를 통한 정보 획득 부분은 미흡한 상태이다. 본 연구에서는 이러한 미흡한 부분인 정보 획득 부분을 센서 데이터 패턴분석을 이용한 이상치 탐색을 통하여 중요한 여러 응용에서 유용하게 제공됨을 예를 통하여 보인다.

1. 서 론

유비쿼터스 센서 네트워크(USN) 환경은 다양한 센서들을 이용하여 다양한 서비스를 제공하는 것을 목표로 한다. 그러나 USN 환경에서 존재하는 센서들은 서로 다르며 이들로 구성된 네트워크 또한 서로 다른 특성을 지닌다. 또한 이를 기반으로 제공되는 서비스 또한 무수히 많으며 요구사항 또한 다양하다. 그러므로 원활한 USN 환경을 구축하기 위해서는 이종의 센서들과 응용들 사이에서 동작하는 미들웨어가 반드시 필요하다. USN 미들웨어는 물리적으로 응용서비스와 센서 네트워크 인프라의 중간에 위치한다. 또한 응용 서비스를 효율적으로 지원하기 위하여 센서 노드와 게이트웨이, 그리고 서버 노드에 분산하여 위치하고, 상호 연동을 통해 서비스를 제공한다. USN Middleware의 주요 기능으로는 다양한 질의 유형 지원, 센싱정보 관리, 메타정보 관리, 이기종의 센서 네트워크 통합 지원, 상황정보 생성 및 관리, QoS보장, 센서노드 미들웨어의 갱신, 센싱정보의 보안, 센서노드의 위치인식[1] 등이 있다. 현재 MiLAN[2], DSWare[3], Impala[4], TinyDB[5] 및 cougar[6], COSMOS[7,8] 등이 있으며, u-Healthcare, u-silvercare, u-Transportatio

n, u-hospital, 재난·재해 방지 및 범죄예방, 환경 감시 시스템 등 서비스가 발달함에 따라 다양한 미들웨어가 등장하게 되었다[9]. 기존의 USN의 미들웨어는 단순히 이기종 센서들의 통합에 중점을 두고 있으며 패턴 분석 등 센서데이터 가공을 통한 정보 추출은 미흡한 실정이다. 패턴 분석을 통한 예측은 이상치 탐색이 중요한 산불 감시, 녹조 감시, 교통 상황 감시 등의 응용에서 사용될 수 있다[10].

본 논문에서는 센서 데이터 패턴 분석을 통한 예측이 유용하게 사용됨을 녹조 감시 응용을 통하여 보인다.

2. 관련 연구

센서 데이터를 통하여 새로운 정보를 추출하는 시스템은 이전에도 존재했다. 그림 1에서 보듯이 COSMOS는 센서 네트워크 지능화 컴포넌트에 센서 데이터 마이닝을 하는 기능이 들어가 있으나 구체적으로 기능이나 방법에 대해 언급이 없다[7]. 또한 그림 2에서 보듯이 SENSOR 또한 데이터 분석을 통한 서비스를 제안하고 있으나 단순히 개념적인 측면에서 제안하고 있을 뿐 구체적인 방법에 대해선 언급이 없다[11].

* 본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 육성·지원사업(NIPA-2010-C1090-1031-0002)의 연구결과로 수행되었음.

** 교신저자

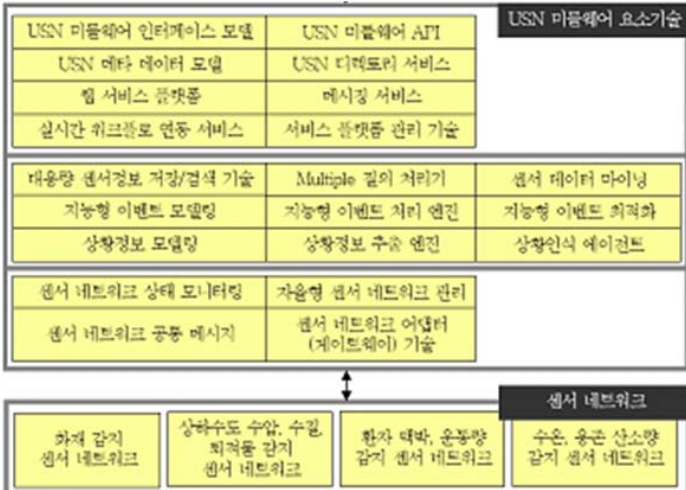


그림 1 Functional Architecture of COSMOS[7]

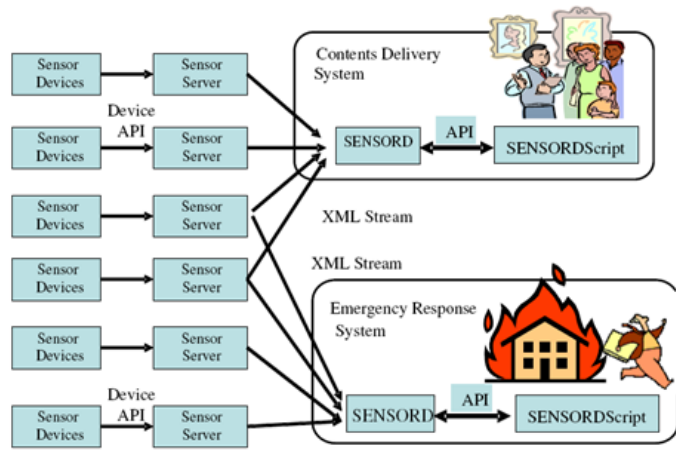


그림 2 Architecture of SENSORD[11]

3. 제안 시스템 설계 및 구현

본 연구에서는 패턴 분석을 통한 예측을 녹조 탐색에 적용한다.

3.1 USN 상황인지 미들웨어

본 연구에서는 자체개발한 USN 상황인지 미들웨어를 사용한다[14]. 미들웨어는 그림 3과 같이 3계층으로 구성되며, 각 계층에서 하는 일은 다음과 같다.

1) USN Service Tier

USN을 기반으로 하는 다양한 서비스를 지원하는 부분이며, USN 인프라 및 응용과 관련된 메타 데이터 관리 및 검색, 지능형 서비스 제공, OPEN API를 통한 USN 응용서비스를 위한 개방형 인터페이스를 제공한다.

2) Sensor Information Intelligence Tier

센서 정보를 지능화하는 부분이며, 센서정보에 대한 질의 처리 기능, USN 인프라로부터 제공되는 센서 정보의 실시간 관리 기능을 제공 한다.

3) Sensor Network Abstraction Tier

실제 네트워크 어댑터와 통신하여 필요한 정보를 취하는 기능을 제공한다.

본 연구에서는 3개의 계층 중 Sensor Information Intelligence Tier에 중점을 두며, Sensor Data Management

컴포넌트에 패턴분석을 위한 기능을 추가한다. 그림4에서 녹색으로 표시한 부분이 추가된 부분이다.

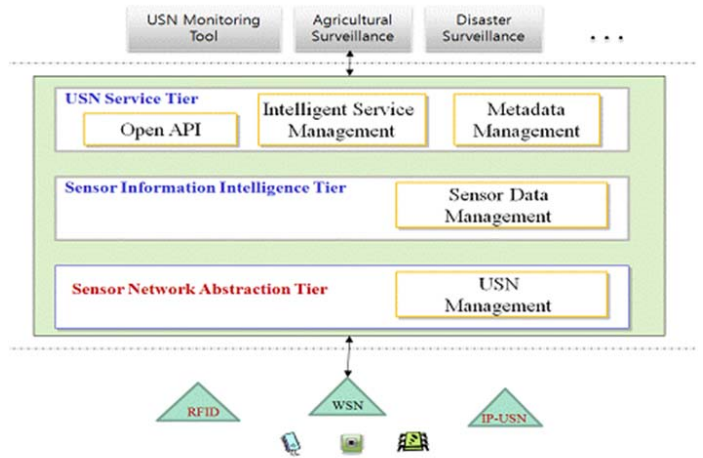


그림 3 Architecture of Intelligent USN Middleware[14]

3.2 Sensor Data Management 컴포넌트

그림 4와 같이 총 4개의 모듈로 구성되며 각각의 모듈이 하는 일은 다음과 같다.

1) Sensor Data Service Manager

사용자가 요청하는 서비스의 분석 및 질의 변환 기능을 제공한다.

2) Stream Query Manager

쓰레드를 이용하여 질의를 처리하고 정규화 등 가공된 센서 데이터를 DB에 저장하는 기능을 제공한다.

3) Sensor Data Manager

수집된 센서 데이터의 정규화, 필터링 등의 기능을 제공한다.

4) Sensor Data Acquisition

USN 인프라로부터 들어오는 다양한 센서들의 데이터를 수집하는 기능을 제공한다.

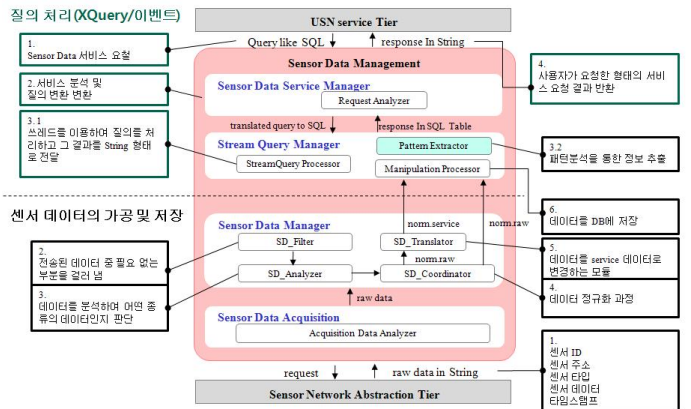


그림 4 Architecture of Sensor Data Management Component After insert Pattern Extractor

3.3 Pattern Extractor

USN 인프라로부터 들어오는 원시 센서데이터에 대한

가공이 아닌 정제된 센서 데이터의 가공이기 때문에 Sensor Query Manager 모듈에 Pattern Extractor를 새롭게 추가하고, Manipulation Processor를 통해 데이터를 주고 받는다. 센서의 종류는 온도, 습도, 조도의 3가지로 가정하고 센서의 수는 계산의 용이성을 위하여 각 센서 별로 30개, 합 90개로 하며 센서 데이터는 1초에 한 번씩 들어온다. Pattern Extractor은 3개의 기능으로 분류하며 각 기능은 다음과 같다.

1) PE_getData

CQ(Continous Query)를 이용하여 manipulation processor를 통하여 DB에 저장된 정제된 센서 데이터를 가져온다.

2) PE_groupingData

수집한 센서 데이터들을 마이닝을 위해 일정 기간 단위(60초)로 분류한다.

3) Pattern Extract

DB에 저장된 센서 데이터를 이용하여 패턴 추출 pattern extractor는 데이터를 받는 기능, 데이터를 분류하는 기능, 패턴을 분석하는 기능으로 구성되며, 분석된 패턴은 다시 manipulation processor에 전달을 하여 DB에 업데이트 한다.

그림 5에서 Pattern Extractor가 어떻게 동작하는지 보인다.

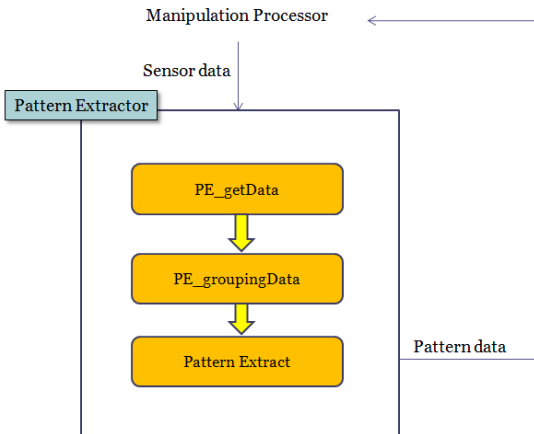


그림 5 Process of Pattern Extractor

패턴 추출을 위하여 순차 패턴 마이닝[12]를 사용한다. 순차 패턴 마이닝은 연관 규칙(Association Rule)에 시간의 개념을 첨가하여 시간의 흐름에 따른 항목들의 상호 연관성을 탐색하는 방법이며, 사용자가 정의한 최소지지도(minsup)를 갖는 시퀀스(large sequence)를 추출하고 이들 가운데 최대 시퀀스(max sequence)를 탐색한다. 본 연구에서는 Apriori-like[13] 알고리즘을 사용하여 패턴 분석기를 구현한다.

3.4 패턴 추출 과정

센서 데이터는 stream 형식이기 때문에 Apriori-like 알고리즘을 적용시키기가 어렵다. 본 연구에서는 이러한 센서 데이터를 알고리즘에 반영하기 위하여 각 센서에서 들어오는 값 들을 일정시간 동안 취합하여 평균값을 내

어 그 값을 하나의 아이템으로 취급한다. 센서의 종류는 3가지이며, 각 아이템은 센서의 종류(온도, 습도, 조도)에 따라서 각각 T(온도), H(습도), I(조도)로 표시하며, 평균값에 따라 기준에 따라서 위험도를 표시한다.

기준은 녹조 현상이 일어날 수 있는 환경을 임의로 가정하여 위험도를 3가지로 분류해서 위험도 별로 랭크를 부여하며 랭크가 높을수록 위험도가 올라가는 것이다. 랭크표시는 센서의 약자 뒤에 숫자로 표시한다. 센서 별로 기준이 다르며, 녹조 현상의 원인은 수온이 섭씨 20도 이상인 더운 날씨가 7일 이상 계속될 때 발생한다고 가정한다.

1. 온도(T) - 수온

if T < 10 then T => T1,
else if T < 20 => T2,
else T => T3

2. 습도(H) - 습도가 낮으면 수온 상승(가정)

if H > 50 then H => H1,
else if H > 25 => H2,
else H => T3

3. 조도(I) - 조도가 높으면 수온 상승(가정)

if I < 500 then I => I1,
else if I < 1000 => I2,
else I => I3

데이터가 들어오는 순서는 매 초마다 T, H, I 순서대로 들어오고 한 번에 3개의 센서에서 동시에 들어온다고 가정한다. 그러므로 매 초 들어오는 아이템 셋은 ({Tx,Hx, Ix} X 총 센서의수/3)의 모양이 되며, 각 트랜잭션은 1분 동안 수집된 센서 n개의 센서 데이터(60 X n)라고 했을 때, 표1과 같이 나타낼 수 있다.

표 1 Sensor Data Transactions

TID	Items
1	{T1, H3, I3, T1, H2, I1, T1, H3, I3, ... I2}
2	{T2, H3, I1, T1, H3, I2, T1, H3, I1, ... I1}
3	{T3, H2, I1, T1, H2, I1, T3, H3, I3, ... I2}
4	{T1, H1, I3, T1, H2, I2, T1, H1, I1, ... I2}
5	{T2, H2, I2, T2, H3, I1, T3, H1, I2, ... I3}
...	...
60	{T2, H2, I2, T3, H1, I3, T1, H2, I2, ..., I1}

본 연구에서 사용한 센서의 개수는 15개이고, minsup은 3으로 한다. 센서들로부터 추출한 트랜잭션을 이용하여 최소 지지도를 만족하는 60초 동안 발생한 위험도에 대한 패턴을 추출한다.

표2는 랜덤한 Transaction의 1-itemsets이다.

표 2 1-itemsets

Item	Count
T1	3
H1	4
I1	3
T2	0
H2	1
I2	2
T3	1
H3	1
I3	0

가지치기가 완료된 추출된 패턴은 표3과 같이 $P_{min} = \{Tx, Hx, Ix\}$ 의 형태가 된다.

표 3 Sensor Data Pattens after pruning

P_{min}	Pattern
P_{min1}	T1, H1, I1
P_{min2}	T2, H1, I3
P_{min3}	T1, H2, I2
P_{min4}	T3, H1, I1
...	T3, H3, I3
P_{min60}	T2, H3, I1

모인 패턴들에 대한 패턴 분석을 한 번 더 수행하여 $P_{hour} = \{P_{min}(Tx, Hx, Ix) \times 60\}$ 을 추출한다. 추출한 패턴들을 DB에 저장하고 응용에서는 같은 시기의 지난 패턴과 현재 패턴을 비교하여 평균 수치를 계산하여 사용자에게 전체 위험도와 종류별 위험도를 알려주며 다음과 같이 나타낼 수 있다.

$$P(H) = \{(T3, H1, I2) | M\}, (T | high), (H | Low), (I | Medium)\}$$

4. 제안 시스템 실험 및 결과

녹조 현상은 호수나 하천 등에서 식물성 플랑크톤인 녹조류가 크게 늘어나 물빛을 녹색으로 변화시키는 현상을 뜻한다. 녹조 현상이 발생되면 수중으로 햇빛이 차단되고 물에 산소가 유입되지 않아 용존산소량을 줄임으로써 물고기 등 수중생물이 폐사하며 악취가 나는 등 피해가 매우 크기 때문에 즉각적인 대처보다는 예방이 중요하기 때문에 패턴분석을 통한 예측이 적절하게 사용될 수 있다.

4.1 시나리오

1. 녹조 현상에 대한 상황 입력 - 수온 섭씨 20도 이상 7일동안 지속
2. CQ를 통하여 센서 데이터 수집 - 60초 동안의 센서 데이터를 수집
3. 데이터의 패턴 분석 - 1분마다 분석한 센서들의 패턴집합을 매 시간마다 분석하여 시간 패턴을 DB에 저장
4. 응용은 DB를 통하여 매년 비슷한 시기의 패턴과,

현재 패턴을 분석 - 현재의 환경 패턴과 매년 같은 시기의 패턴을 비교 후 평균값 산정 후 사용자에게 전체 위험도와 종류별 위험도를 알림

5. 2010년 7월 1일 ~ 7월 7일의 양식장의 환경 패턴이 녹조 발생 환경에 근접 - 응용에서 수온의 증가량이 20도 이상 7일 동안 발생하여 용존산소량, 미생물량이 꾸준한 증가세를 보이고, 매년 6월 25일 ~ 7월 10일 사이에 녹조 발생확률이 높고, 발생했던 적이 있으므로 3일 이내에 녹조 발생 가능성이 있다고 예측
6. 양식장 주인은 녹조 현상이 일어나기 전에 예방

4.2 실험 결과

미들웨어를 통한 센서 정보의 추출과 패턴분석을 통한 상태 값을 나타내도록 하였다. 그림 6은 미들웨어의 인터페이스를 통한 패턴 분석의 결과를 나타낸다.

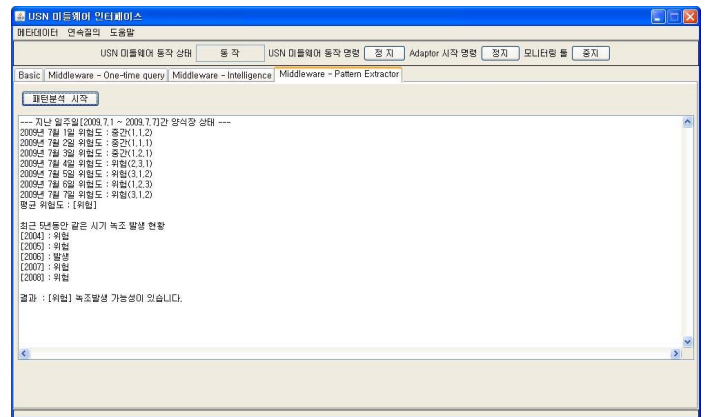


그림 6 패턴분석 결과

5. 결론 및 향후과제

본 연구에서는 패턴 분석이라는 센서 데이터의 가공을 활용하여 기존의 실시간 질의에만 의존했던 정보 추출 외에 새로운 정보 창출을 구현하였다. 이 방법은 미들웨어를 이용한 새로운 서비스와 정책 수립에 도움이 되며 미들웨어의 기본적인 효율성 증가에 도움이 될 것이다. 앞으로 다양한 응용 분야에서 다양한 패턴 분석을 통하여 범위를 넓혀갈 수 있다.

향후 연구 과제로는 본 연구에서는 센서의 개수에 제한을 두었지만 센서의 개수에 관계없이 필요한 데이터를 추출하는 방법이 필요하며, 연구에서 사용한 apriori-like 알고리즘 외에 순차 패턴 알고리즘을 변형하여 스트림 센서데이터(stream sensor data)에 맞게 최적화할 수 있는 연구가 필요하다. 또한 이에 맞는 새로운 저장 정책도 필요하고 자주 사용되는 응용에 대한 OPEN API를 제공함으로써 사용자에게 보다 편리한 환경 제공이 필요하다.

6. 참고문헌

[1] 김민수, 이용준, 박종현, "USN 미들웨어 기술개발 동향", 전자통신동향분석, 제22권, 제3호, 2007.
 [2] W. Heinzelman, A. Murphy, H. Carvalho, and M. Pe

- rillo, "Middleware to Support Sensor Network Applications", IEEE Network Magazine Special Issue Jan. 2004.
- [3] Shuoqi Li, Sang H. Son, and John A. Stankovic, "Event Detection Services Using Data Service middleware in Distributed Sensor Networks", Information Proc. In Sensor Networks, Apr. 2003, LNCS 2634, pp. 502-517.
- [4] T. Liu and M. Martonosi, "Impala: A Middleware System for Managing Autonomic," Parallel Sensor Systems, Proc. ACM SIGPLAN Symp. Principles and Practice of Parallel Programming, 2003, pp. 107-118.
- [5] S. R. Madden, M. J. Franklin, and J. M. Hellerstein, "TinyDB: An Acquisitional Query Processing System for Sensor Networks", ACM TODS, Vol. 30, No.1, 2005, pp. 122-173.
- [6] Yong Yao and J. E. Gehrke, "The Cougar Approach to In-Network Query Processing in Sensor Networks", SIGMOD RECORD. Vol. 31, No. 3, Sep. 2002.
- [7] Marie KIM, JUN WOOK LEE, YONG JOON LEE, RYOU Jae-Cheol, "COSMOS: A Middleware for Integrated Data Processing over Heterogeneous Sensor Networks", ETRI journal, vol. 30, no5, 2008, pp. 696-706.
- [8] Yung Bok Kim, Marie Kim, Yong Joon Lee, "COSMOS: a middleware platform for sensor networks and a u-healthcare service", Proceedings of the 2008 ACM symposium on Applied computing, Fortaleza, Ceara, Brazil, 2008.
- [9] 김관중, 김선지, 김내수, 표철식, "USN 서비스 및 시장 동향", 정보과학회지, 제25권, 제12호, 2007.
- [10] 김선진, 정우석, 박가람, 최연경, 김선중, "USN 응용서비스 동향", 전자통신동향분석 제22권 제3호, 2007.
- [11] 김대영, 김재언, 성종우, 이강우, "센서 네트워크 운영체제/미들웨어 기술동향", IITA 주간기술동향 통권 1221호, 2005.
- [11] Akio Sashima, Yutaka Inoue and Koichi kurumatan i, "Spatio-Temporal Sensor Data Management for Context-Aware Services", Proc. 2006 Advanced Data Processing in Ubiquitous Computing(ADPUC), Melbourne, Australia, 2006.
- [12] 박종선, 이진희, "순차 패턴 마이닝 알고리즘에 대한 고찰", 성균관대학교 통계학과, 2003.
- [13] R.Agrawal and R. Srikant, "Mining Sequential Patterns", In Proc. of the 11th Int'l Conference on Data Engineering(ICDE'95), Taipei, Taiwan, 1995.
- [14] "USN 상황인지 미들웨어 개발 결과보고서", 자체문서, 충남대학교 공과대학 실시간정보시스템 연구실, 2009.