Journal of the Korea Academia-Industrial Cooperation Society
Vol. 12, No. 9 pp. 4183-4191, 2011

http://dx.doi.org/10.5762/KAIS.2011.12.9.4183

# 데이터 스트림 마이닝에서 정보 중요성 차별화를 위한 퍼지 윈도우 기법

장중혁[1*]
[1]대구대학교 컴퓨터IT공학부

# A Fuzzy Window Mechanism for Information Differentiation in Mining Data Streams

**Joong-Hyuk Chang**[1*]
[1]**Dept. of Computer & Information Technology, Daegu University**

**요 약** 구성요소가 지속적으로 생성되고 시간 흐름에 따라 변화되기도 하는 데이터 스트림의 특성을 고려하여 데이터 스트림 구성요소의 중요성을 발생 시간에 따라 차별화하기 위한 기법들이 활발히 제안되어 왔다. 기존의 방법들은 최근에 발생된 정보에 집중된 분석 결과를 제공하는데 효과적이나 보다 유연하게 다양한 형태로 정보 중요성을 차별화하는데 한계가 있다. 퍼지 개념에 기반한 정보 중요성 차별화는 이러한 한계를 보완하는 좋은 대안이 될 수 있다. 퍼지 개념은 기존의 뚜렷한 경계를 갖는 접근법의 문제점을 극복하고 실세계의 요구에 보다 부합되는 결과를 제공할 수 있는 방법으로 여러 데이터 마이닝 분야에서 널리 적용되어 왔다. 본 논문에서는 퍼지 개념을 적용하여 데이터 스트림 마이닝에서 정보 중요성 차별화에 효율적으로 활용될 수 있는 퍼지 윈도우 기법을 제안한다. 퍼지 캘린더를 포함한 기본적인 퍼지 개념에 대해서 먼저 기술하고, 다음으로 데이터 스트림 마이닝에서 퍼지 윈도우 기법을 적용한 가중치 패턴 탐색에 대한 세부 내용을 기술한다.

**Abstract** Considering the characteristics of a data stream whose data elements are continuously generated and may change over time, there have been many techniques to differentiate the importance of data elements in a data stream by their generation time. The conventional techniques are efficient to get an analysis result focusing on the recent information in a data stream, but they have a limitation to differentiate the importance of information in various ways more flexible. An information differentiation technique based on the term of a fuzzy set can be an alternative way to compensate the limitation. A term of a fuzzy set has been widely used in various data mining fields, which can overcome the sharp boundary problem and give an analysis result reflecting the requirements in real world applications more. In this paper, a fuzzy window mechanism is proposed, which is adapting a term of a fuzzy set and is efficiently used to differentiate the importance of information in mining data streams. Basic concepts including fuzzy calendars are described first, and subsequently details on data stream mining of weighted patterns using a fuzzy window technique are described.

**Key Words :** Fuzzy window mechanism, Data stream mining, Information differentiation, Data streams, Fuzzy data mining

## 1. 서론

컴퓨터 기술의 발달과 이를 활용한 컴퓨터 응용 환경의 변화에 따라 근래 들어 많은 분야에서 데이터 스트림 형태로 정보를 발생시키고 있다. 일반적으로 데이터 스트림은 구성요소가 지속적으로 생성되는 무한집합으로 정

이 논문은 2009년도 정부(교육과학기술부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임 (2009-0064582)
*교신저자 : 장중혁 (jhchang@daegu.ac.kr)
접수일 11년 08월 08일 수정일 (1차 11년 09월 01월, 2차 11년 09월 05일) 게재확정일 11년 09월 08일

의되며, 따라서 데이터 스트림의 단순 처리 및 특성 분석 등에 있어서는 지속적으로 생성되는 방대한 양의 데이터에 대한 빠른 처리 및 구성 요소의 발생 시간 특성 등이 고려되어야 한다. 데이터 스트림 관리 시스템(DSMS), 데이터 스트림에 대한 질의 처리 등은 물론 이거니와 응용 분야에서 발생되는 데이터 스트림의 활용도를 높이기 위한 데이터 스트림 마이닝에 있어서도 해당 특성을 고려한 연구들이 활발히 제안되어 왔다[1-6].

지속적으로 확장되고 변화되는 데이터 스트림의 특성을 고려할 때 데이터 스트림 마이닝에서는 분석 대상 데이터 스트림 구성요소들의 시간 특성을 반영한 분석 결과를 얻는 것이 중요한 관심 사항 중의 하나이다. 즉, 하나의 데이터 스트림에 있어서 시간 흐름에 따라 지속적으로 생성되는 구성요소들의 중요성을 시간 특성을 고려하여 차별화함으로써 보다 유용한 마이닝 결과를 얻을 수 있다. 이러한 목적을 위해서 데이터 스트림에 대한 빈발패턴 탐색 또는 관심 순차패턴 탐색 등의 분야에서 이동 윈도우(Sliding window) 기법[1,2]이나 감쇠(decay) 기반 기법[3,4,5]을 적용하여 분석 대상 데이터 스트림의 최근 특성을 효율적으로 반영한 분석 결과를 얻기 위한 방법들이 제안되어 왔다. 하지만, 해당 방법들은 하나의 데이터 스트림을 구성하는 방대한 정보 중에서 최근 발생 정보에 집중된 분석 결과를 얻는 것으로서 다양한 형태의 시간 기준을 고려한 정보의 중요성 차별화를 지원하는데 한계가 있다.

한편, 데이터 마이닝 과정에서 실세계의 특성이나 요구를 보다 효율적으로 반영하기 위한 방법으로 퍼지 개념에 기반한 마이닝 방법들이 활발히 제안되어 왔으며[7-11], 퍼지 빈발패턴 탐색 및 퍼지 관심 순차패턴 탐색 등의 분야에서도 다양한 연구들이 진행되어 왔다. 빈발패턴 및 관심 순차패턴 탐색을 위한 일반적인 데이터 마이닝 방법들에서는 분석 대상이 되는 데이터 집합에서 출현빈도 수 계산 등과 같이 하나의 상황을 만족하는지를 판단함에 있어서 명확히 구분되는 기준을 활용한다. 이러한 접근 방법은 경계점 문제(sharp boundary problem)을 야기시킬 수 있다. 특히, 시간 정보를 활용한 데이터 마이닝에서 해당 문제가 더욱 두드러지게 나타날 수 있으며, 이로 인해 실세계의 상황이나 요구를 효과적으로 반영하는데 한계가 있다. 퍼지 개념을 적용하는 경우 경계점 문제를 상쇄시킴으로써 이러한 한계점을 극복할 수 있다[8].

이와 같은 퍼지 개념을 데이터 스트림 마이닝에 적합한 형태로 변형하여 적용하는 경우 다양한 형태의 시간 기준을 고려한 정보의 중요성 차별화를 지원할 수 있다. 즉, 데이터 스트림 구성요소의 발생 시간 정보를 기준으로 퍼지 개념을 적용하여 실제 응용 분야의 요구를 보다 효율적으로 반영한 정보 중요성 차별화를 지원할 수 있다. 본 논문에서는 퍼지 개념을 적용하여 데이터 스트림을 분석하기 위한 접근 방법을 제시한다. 즉, 데이터 스트림의 구성요소 발생 시간 정보에 효율적으로 적용될 수 있는 퍼지 집합에 대해서 기술하고, 이를 바탕으로 데이터 스트림 마이닝시 퍼지 개념을 활용하여 정보의 중요성을 차별화 할 수 있는 퍼지 윈도우 기법(FWM: **F**uzzy **W**indow **M**echanism)을 제시한다.

본 논문의 구성은 다음과 같다. 먼저 2장에서는 논문에서 다루는 주제와 연관된 이전 연구들을 간략히 정리한다. 3장에서는 데이터 스트림 마이닝에 효율적으로 적용될 수 있는 퍼지 윈도우 기법에 대해서 상세히 기술하고, 4장에서는 실험을 통해 제안된 기법의 유용성을 검증한다. 끝으로, 5장에서 논문의 결론을 맺는다.

## 2. 관련 연구

구성 요소가 지속적으로 생성되고 시간 흐름에 따른 가변성이 큰 데이터 스트림의 특성을 고려하여 마이닝 과정에서 시간 축을 기준으로 정보의 중요성을 차별화하기 위한 다양한 방법들[1,2,4,5]이 연구되어 왔다. 해당 방법들 중에서 대표적인 것으로 이동 윈도우 기법 및 감쇠 기반 기법이 제안되었다. 이동 윈도우 기법[1,2]은 일정 크기의 시간 윈도우를 정의하여 해당 범위 내에 포함되는 정보만 유효한 것으로 간주하고 범위에 포함되지 않는 정보는 무효한 것으로 간주하는 방법이다. 일반적으로 해당 기법에서는 시간 흐름에 따라 윈도우를 이동하면서 윈도우 크기만큼의 최근 시간 범위를 유효 범위로 정의한다. 감쇠 기반 기법[4,5]은 하나의 데이터 스트림을 구성하는 구성요소들 중에서 최근에 발생한 구성요소는 상대적으로 높은 중요성을 갖는 것으로 간주하고 과거에 발생한 구성요소는 그 중요성이 시간 흐름에 따라 점차적으로 감쇠되도록 하는 기법이다. 이를 통해 일정 시점에서 발생한 정보가 해당 시점에서는 매우 중요한 정보로 간주되지만 시간 흐름에 따라 그 중요성이 감쇠되고 충분히 오랜 시간이 지난 후에는 사실상 무효한 정보로 간주되도록 한다. 이동 윈도우 기법 및 감쇠 기반 기법은 최근에 발생된 정보 혹은 최근에 가까운 시점에 발생한 정보의 중요성을 높게 간주하고 이외의 정보는 무효하거나 중요성이 낮은 것으로 간주한다. 따라서, 최근 정보에 집중된 분석 결과를 얻기 위한 데이터 마이닝에서는 매우 효과적으로 적용될 수 있다. 하지만, 하나의 데이터 스트림에서 현재까지 발생된 정보 전체에 대해 특정한 시

간 범위나 조건 등을 보다 유연하게 지정하여 중요성을 차별화 하는데 한계가 있다.

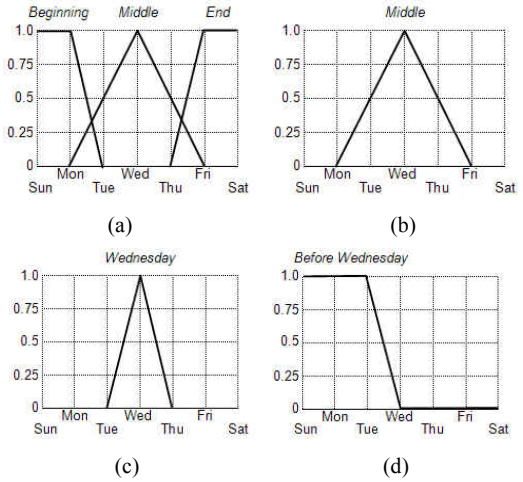
데이터 마이닝 분야에서 경계점 문제(sharp boundary problem)를 보완하기 위한 방법으로 퍼지 개념은 여러 세부적인 데이터 마이닝 분야에서 널리 활용되고 있으며, 퍼지 빈발패턴 탐색 및 퍼지 관심 순차패턴 탐색 등의 분야에서도 활발한 연구들이 진행되어 왔다[7-11]. [10] 및 [11]에서는 구성요소의 발생 여부뿐만 아니라 양적인 정보를 추가적으로 고려하는데 있어서 퍼지 개념에 기반한 가중치를 활용하고 있다. 예를 들어, 유통 소매점에서 발생한 구매 내역 데이터에 대한 분석 과정에서 각 품목별 구매 여부는 1(구매) 또는 0(비구매)으로 표현되는 이진(binary)으로 나타낼 수 있으나 구매 물량 정보는 단순히 이진으로 표현하는 것이 불가능하며, 이때 퍼지 개념을 적용하여 구매 물량에 따른 가중치를 부여할 수 있다. 퍼지 개념은 순차정보 데이터 집합에서 보다 흥미로운 관심 순차패턴을 탐색하는데 이용 되기도 하였다. [8]에서는 순차정보 데이터 집합에서 관심 순차패턴을 탐색하는데 있어서 순차정보를 구성하는 단위항목의 출현빈도뿐만 아니라 하나의 순차정보를 구성하는 단위항목들 사이의 발생 시간 간격(time-interval)도 고려하여 관심 순차패턴을 탐색한다. 이때, 다양한 값을 갖는 발생 시간 간격을 효과적으로 분석하기 위한 방법으로 퍼지 개념을 활용하고 있으며, 단위항목의 발생 시간 간격에 따른 최적의 퍼지 가중치 부여 기법에 대한 연구[12]도 진행되어 왔다. 데이터 마이닝에서 퍼지 개념을 적용한 대부분의 이전 연구들은 한정적인 데이터 집합에서 단위항목의 양적인 정보 및 발생 시간 간격 등과 같이 다양한 값으로 표현될 수 있는 정보의 중요성을 효율적으로 차별화 하는데 관심을 두고 있다.

## 3. 퍼지 윈도우 기법

본 절에서는 퍼지 개념을 데이터 스트림 마이닝에 적합한 형태로 변형하여 적용함으로써 다양한 형태의 시간 기준을 고려한 정보의 중요성 차별화를 지원하는 **퍼지 윈도우 기법**(FWM)에 대해서 기술한다. 이를 위해서 3.1절에서는 퍼지의 기본 개념 및 퍼지 캘린더(fuzzy calendar)에 대해서 기술하고, 3.2절에서는 데이터 스트림 마이닝에서 FWM을 적용할 때 퍼지 가중치를 고려한 관심 패턴 및 퍼지 가중치의 기본 속성 등에 대해서 기술한다.

### 3.1 퍼지 캘린더

일반적으로 캘린더(calendar)는 시간 구간(time interval)들의 집합으로 정의된다. 예를 들어 ‘during’, ‘overlap’ 및 ‘meet’ 등도 각 의미에 맞는 시간 구간으로 표현될 수 있다[8,13]. 캘린더를 구성하는 시간 구간들은 서로 명확히 구분되는 시간 범위로 정의되기도 하지만 일상 생활에서 사용되는 대부분의 표현들은 일반적으로 시간 구간을 명확히 구분하는 것이 매우 어렵다. 예를 들어 대형 마트에서 발생된 상품 판매 기록에 대한 분석에서 ‘일요일 판매 기록 분석’과 같이 명확히 구분되는 특정 시간 구간을 지정하는 경우는 전자에 해당되고, ‘월 초 판매 기록 분석’과 같이 시간 구간을 명확히 구분하여 지정하기 힘든 경우는 후자에 해당된다. 실제 응용 분야에서 생성되는 다양한 데이터에 대한 분석 과정에서 해당 분야의 특성이나 요구를 보다 효과적으로 반영하기 위해서는 명확히 구분되는 시간 구간으로 구성되는 캘린더뿐만 아니라 다른 형태의 캘린더를 활용할 필요가 있다. 이러한 요구를 반영하여 하나의 시간 일람표를 구성하는 시간 구간들을 보다 유연하게 정의하기 위한 방법으로 퍼지 집합(fuzzy set) 이론을 적용한 퍼지 캘린더가 제안되었다.




[그림 1] 주 단위의 기준 시간에 대한 기본 퍼지 캘린더 : (a) 일반형, (b) 단순형, (c) 특정요일 지정, (d) 특정요일 이전

[Fig. 1] Basic fuzzy calendars for the time granularity of week: (a) General, (b) Simple, (c) Crisp time interval, (d) Before

퍼지 캘린더를 통해 일상적으로 많이 사용하는 다양한 표현들을 캘린더로 표현할 수 있다. 예를 들어, ‘월 중반’, ‘주 초반’ 또는 ‘한 주의 끝 무렵’ 등과 같은 표현들은 시간 구간이 명확히 구분되는 단순 캘린더에서는 정의할 수 없으나 퍼지 캘린더에서는 이를 효과적으로 정의할 수 있다. 퍼지 캘린더는 일 단위, 주 단위, 월 단위 또는

년 단위 등과 같이 일반적으로 사용되는 다양한 시간 단위에 대해서 적용될 수 있으며, 기본적인 퍼지 캘린더는 [정의 1]에서와 같이 정의된다[8].

**[정의 1. 기본 퍼지 캘린더]** 하나의 기준 시간 단위를 $U$라 할 때 기본적인 퍼지 캘린더 $A$는 해당 시간 단위 $U$에서 존재하는 시간 구간들의 집합에 대한 퍼지 속성을 나타내며, $U$에 속하는 각 시간 구간 $T_i(T_i \in U)$에 대한 소속함수(membership function) 형태로 다음과 같이 정의된다.

$$\mu_A : U \rightarrow [0, 1]$$

이때, $U$의 각 시간 구간 $T_i$에 대한 퍼지 연산 결과인 $\mu_A(T_i)$는 퍼지 캘린더 $A$에 대한 시간 구간의 $T_i$의 퍼지 일치 정도(fuzzy matching degree)를 나타낸다. ■

그림 1은 '주 단위'로 표현되는 몇 가지 퍼지 캘린더를 보여준다. 그림 1-(a)는 일반적인 퍼지 캘린더로서 기준 시간 단위에 속하는 모든 시간 구간들이 소속함수에 의해 정의되는 퍼지 가중치를 갖는다. 그림 1-(b) 및 1-(c)는 보다 단순한 형태의 퍼지 캘린더이며, 특히 그림 1-(c)는 기준 시간 단위에 속하는 시간 구간들 중에서 '수요일'이라는 명확히 구분되는 특정 요일에만 가중치가 부여된 형태로서 단순 캘린더를 적용한 경우와 동일한 효과를 얻을 수 있다. 그림 1-(d)에서와 같이 일정 시점 이전 또는 이후 범위 전체를 표현할 수도 있다. 한편, [정의 1]에서와 같이 하나의 기본 시간 단위에 정의되는 퍼지 캘린더를 조합하여 보다 다양한 형태를 표현할 수 있으며, [정의 2]에서와 같이 정의된다[8].

**[정의 2. 퍼지 캘린더의 재귀적 정의]** $A$와 $B$를 하나의 기준 시간 단위 $U$에 대해서 [정의 1]에서와 같이 정의되는 퍼지 캘린더라 할 때 '$A$ **AND** $B$'($A \wedge B$), '$A$ **OR** $B$'($A \vee B$) 및 '**NOT** $A$'($\neg A$) 또한 퍼지 캘린더이며, $U$에 속하는 하나의 시간 구간 $T_i(T_i \in U)$의 각 퍼지 캘린더에 대한 퍼지 일치 정도를 나타내는 $\mu_{A \wedge B}(T_i)$, $\mu_{A \vee B}(T_i)$ 및 $\mu_{\neg A}(T_i)$는 각각 다음과 같이 정의된다.

$$\mu_{A \wedge B}(T_i) = \mathbf{min}\{\mu_A(T_i), \mu_B(T_i)\}$$
$$\mu_{A \vee B}(T_i) = \mathbf{max}\{\mu_A(T_i), \mu_B(T_i)\}$$
$$\mu_{\neg A}(T_i) = 1 - \mu_A(T_i)$$

여기서, $\mathbf{min}\{\mu_A(T_i), \mu_B(T_i)\}$ 및 $\mathbf{max}\{\mu_A(T_i), \mu_B(T_i)\}$는 $\mu_A(T_i)$ 및 $\mu_B(T_i)$ 중에서 작은 값, 큰 값을 각각 의미한다. ■

[정의 2]에서 제시된 퍼지 캘린더 이외에도 기본 퍼지 캘린더를 조합하여 반복적으로 정의되는 여러 형태의 퍼지 캘린더 정의할 수 있으며, 반복적으로 정의된 퍼지 캘린더에 대한 하나의 시간 구간의 퍼지 일치 정도(fuzzy matching degree)도 응용 분야의 특성 등을 고려하여 다양한 형태로 정의할 수 있다. 이에 대한 상세한 내용은 [8]에서 확인할 수 있다.

## 3.2 FWM을 적용한 가중치 패턴

퍼지 캘린더를 적용하여 가중치 패턴을 구하는 과정은 다양한 데이터 마이닝 분야에서 활용될 수 있다. 그 중에서도 구성 요소가 지속적으로 확장되는 데이터 스트림에서 발생 시간 등에 따라 구성요소의 중요성을 차별화 하고자 하는 경우에 매우 효율적으로 활용될 수 있다. 이를 통해 감쇠율 기법[4,5]이나 이동 윈도우 기법[1,2]을 적용하여 정보 중요성을 차별화 하던 기존의 데이터 스트림 마이닝 방법들에 비해 보다 유연한 형태의 정보 중요성 차별화를 기법을 제공할 수 있다.

### 3.2.1 FWM 적용시 트랜잭션의 가중치

FWM을 적용한 데이터 스트림에 대한 가중치 패턴 탐색에서는 하나의 데이터 스트림에 포함되는 각 트랜잭션(transaction)의 가중치를 구하여 활용하며, 각 트랜잭션의 생성 시간과 FWM에서 주어진 퍼지 캘린더를 활용하여 해당 트랜잭션의 가중치를 구한다. 이와 같이 구해진 트랜잭션의 가중치는 패턴의 출현빈도 수 계산 및 해당 데이터 스트림의 크기 등을 계산하는데 활용되며, 본 논문에서는 이를 해당 트랜잭션의 **FW-가중치**(FW-weight: fuzzy window weight) 라 지칭한다.

데이터 스트림 $DS_k$에 대해서 관심 가중치 패턴을 탐색하는데 필요한 퍼지 캘린더 $FC$가 주어졌을 때, 해당 데이터 스트림에 속하는 하나의 트랜잭션 $T$의 FW-가중치 $W_{FW}(T)$는 다음과 같이 구해진다: 먼저, 해당 트랜잭션의 발생 시간 정보 $t$를 구하고, 이어서 주어진 퍼지 캘린더 $FC$상에서 해당 발생 시간 정보의 일치 정도(matching degree)로부터 해당 트랜잭션의 FW-가중치를 얻는다. 즉, $W_{FW}(T)$는 다음과 같이 구해진다.

$$W_{FW}(T) = \mu_{FC}(t)$$

이때, 주어진 퍼지 캘린더 $FC$가 두 개 이상의 기본 퍼지 캘린더들의 조합으로 정의된 경우에는 [정의 2]에서와 같이 해당 트랜잭션의 FW-가중치를 구할 수 있다.

| TID | Generation time | Transaction |
|---|---|---|
| *T1* | Jul/04/2011, Mon., 11:30:40 | <*a, b, c, d*> |
| *T2* | Jul/05/2011, Tue., 15:10:20 | <*b, c, d*> |
| *T3* | Jul/06/2011, Wed., 11:30:10 | <*a, b, c*> |
| *T4* | Jul/07/2011, Thu., 09:20:25 | <*b, c*> |
| *T5* | Jul/09/2011, Sat., 19:50:09 | <*a, b, d*> |

**[그림 2]** 예제 데이터 스트림 $DS_5$
**[Fig. 2]** An example data stream $DS_5$

실제 적용 예제로서 그림 2의 예제 데이터 스트림에 대해서 그림 1-(b)의 퍼지 캘린더를 적용하는 경우를 고려해 보자. 일반적으로 데이터 스트림은 매우 빈번히 발생되는 방대한 양의 트랜잭션으로 구성된다. 하지만 해당 예제에서는 보다 쉽게 이해할 수 있도록 다섯 개의 트랜잭션으로 구성되는 간단한 데이터 스트림을 고려하였으며, 각 트랜잭션의 생성 시간은 그림에서와 같다. 이러한 상황에서 각 트랜잭션의 FW-가중치는 그림 1-(b)의 퍼지 캘린더에 의해 다음과 같이 구해진다.

$$W_{FW}(T_1)=0,\ W_{FW}(T_2)=0.5,$$
$$W_{FW}(T_3)=1,\ W_{FW}(T_4)=0.5,\ W_{FW}(T_5)=0$$

### 3.2.2 FW-가중치 패턴

트랜잭션들로 구성되는 하나의 데이터 스트림 $DS_k$에서 가중치를 고려한 관심 패턴(interesting weighted patterns)을 탐색하는 일반적인 데이터 마이닝의 경우 해당 데이터 스트림에 출현한 패턴 A의 가중치 출현빈도수(weighted count)는 해당 패턴 A를 포함하는 모든 트랜잭션들의 가중치 합으로 정의된다. 또한, 데이터 마이닝에서 패턴의 지지도에 대한 정의에 따라 패턴 A의 가중치 지지도는 $DS_k$에 포함되는 모든 트랜잭션들의 가중치 합에 대한 A의 가중치 출현빈도 수의 비율로 정의된다. 이와 마찬가지로, FWM을 적용하여 데이터 스트림에서 가중치 패턴을 탐색하는 경우에는 각 패턴의 **FW-가중치 출현빈도수**(FW-weighted count) 및 **FW-가중치 지지도**(FW-weighted support)가 활용되며, 이들 값은 해당 데이터 스트림에 포함되는 트랜잭션들의 FW-가중치로부터 구해진다. 본 절에서는 이러한 과정에서 데이터 스트림에 출현한 하나의 패턴에 있어서 FW-가중치 지지도 계산 방법에 대해서 기술한다.

데이터 스트림에 속하는 하나의 패턴에 대해서 해당 패턴의 FW-가중치 출현빈도 수 및 FW-가중치 지지도는 3.2.1절에서 기술한 트랜잭션의 FW-가중치를 이용하여 구한다. $k$ 개의 트랜잭션으로 구성되는 데이터 스트림 $DS_k$에 대해서 퍼지 캘린더 $FC$가 주어졌을 때, 해당 데이터 스트림에 출현한 패턴 $X$의 FW-가중치 출현빈도수 $C_{FW}(X)$는 $\Sigma_{T:(X\subseteq T)\wedge(T\in DS_k)} W_{FW}(T)$로 구해지며, 따라서 해당 패턴의 FW-가중치 지지도 $S_{FW}(X)$는 $\frac{\Sigma_{T:(X\subseteq T)\wedge(T\in DS_k)} W_{FW}(T)}{\Sigma_{T:T\in DS_k} W_{FW}(T)}$로 정의된다. 이를 바탕으로 FWM을 적용한 데이터 스트림 마이닝을 위한 관심 FW-가중치 패턴(interesting FW-weighted pattern)이 정의된다. 즉, 데이터 스트림 $DS_k$에 대해서 관심 가중치 패턴 탐색을 위한 지지도 임계값 $minS(0<minS\leq 1)$가 주어졌을 때, 해당 데이터 스트림에서 발생한 하나의 패턴 $X$의 FW-가중치 지지도 $S_{FW}(X)$가 $minS$보다 크거나 같은 경우 해당 패턴을 **관심 FW-가중치 패턴**이라 정의한다.

표 2는 그림 2의 예제 데이터 스트림에서 유도된 세 개의 패턴들에 대해서 그림 1-(b)의 퍼지 캘린더를 적용했을 때 지지도 변화를 보여준다. 각 패턴의 단순 지지도는 0.6으로서 서로 동일하다. 그러나, 패턴 <*b, d*>의 경우 $T_1$ 및 $T_5$와 같이 FW-가중치가 작은 트랜잭션들에 주로 출현하였기 때문에 FW-가중치 지지도는 매우 작은 값을 갖는다. 만약 해당 데이터 스트림에 대한 관심 패턴 탐색을 위한 지지도 임계값이 0.5로 설정되었다면, 단순 지지도를 적용하는 경우 표 2에서 제시된 세 개의 패턴들이 모두 관심 패턴으로 탐색되나, FW-가중치 지지도를 적용하는 경우 패턴 <*b, d*>의 FW-가중치 지지도는 주어진 임계값보다 작으므로 해당 패턴은 관심 패턴으로 탐색되지 못한다.

**[표 2]** 단순 지지도와 FW-가중치 지지도의 비교
**[Table 2]** Comparing of supports (Simple support vs. FW-weighted support)

| *Patterns* | 단순 지지도 | FW-가중치 지지도 |
|---|---|---|
| <*a, b*> | 0.6 | 0.5 |
| <*b, c*> | 0.6 | 0.75 |
| <*b, d*> | 0.6 | 0.25 |

### 3.2.3 퍼지 가중치 지지도의 anti-monotone 속성

일반적인 관심 패턴 탐색을 위한 단순 지지도와 마찬가지로 FW-가중치 패턴 탐색을 위한 FW-가중치 지지도 또한 anti-monotone 속성을 갖는다. 하나의 데이터 스트림 $DS_k$에서 발생한 두 개의 패턴 $A$와 $B$에 있어서 패턴 $B$가 $A$의 확대패턴(super-pattern)이라 가정하자(즉, $A\subseteq B$). 이 경우 $A\subseteq B$ 관계를 만족하기 때문에 $\Sigma_{T:(A\subseteq T)\wedge(T\in DS_k)} W_{FW}(T)$ 값은 항상

$\Sigma_{T:(B\subseteq T)\wedge(T\in DS_k)} W_{FW}(T)$ 값보다 크거나 같다. 따라서, 앞서 기술한 FW-가중치 지지도의 정의에 의해 두 패턴 $A$와 $B$의 지지도 사이에는 다음과 같은 관계가 성립된다.

$$S_{FW}(A)=\frac{\Sigma_{T:(A\subseteq T)\wedge(T\in DS_k)} W_{FW}(T)}{\Sigma_{T:T\in DS_k} W_{FW}(T)}$$

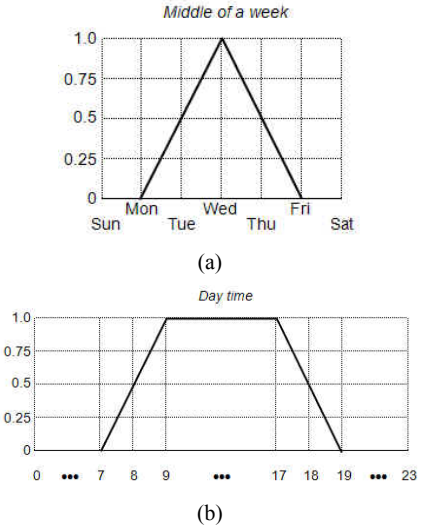
$$\geq$$

$$\frac{\Sigma_{T:(B\subseteq T)\wedge(T\in DS_k)} W_{FW}(T)}{\Sigma_{T:T\in DS_k} W_{FW}(T)}=S_{FW}(B)$$

이러한 관계로 인해, 만약 $S_{FW}(A)$값이 관심 패턴 탐색을 위해 주어진 지지도 임계값보다 작다면 $S_{FW}(B)$값 또한 해당 임계값보다 작다. 즉, 확대패턴 관계에 있는 두 패턴 $A$ 및 $B$ 사이의 이러한 관계로부터 FW-가중치 지지도가 anti-monotone 속성을 만족함을 알 수 있다. 단순 지지도 기반의 일반적인 관심 패턴 마이닝에서와 마찬가지로 FW-가중치 지지도의 anti-monotone 속성은 FWM을 적용한 관심 가중치 패턴 분석에 있어서 데이터 스트림에 대한 탐색 범위를 줄임으로써 마이닝 성능을 향상시키는데 활용된다.

## 4. 실험 결과 고찰

본 절에서는 FWM의 유용성 평가 실험 결과를 제시한다. 앞서 기술한 바와 같이 FWM은 시간 흐름에 따라 지속적으로 발생되는 구성 요소의 중요성 차별화 기법으로서 다양한 형태의 데이터 스트림 마이닝 기법에 적용될 수 있다. 본 절에서 제시되는 실험에서는 FWM을 데이터 스트림에 대한 관심 순차패턴 탐색 기법인 eISeq[14]에 적용하여 이의 유용성을 분석하였다. eISeq 방법은 기본적으로 하나의 순차 데이터 스트림을 구성하는 각 순차정보가 서로 동일한 중요성을 갖는 것으로 간주하지만, FWM을 적용하는 경우 각 순차정보에 대한 탐색 과정에서 주어진 퍼지 캘린더를 활용하여 해당 순차정보의 발생 시간 정보로부터 이의 가중치를 구한다. 논문의 나머지 부분에서 기술되는 실험에서 eISeq의 기본 매개 변수 설정은 다음과 같다: 중요 패턴 지지도(significant support)는 관심 패턴 탐색을 위한 지지도 임계값(support threshold)의 30%로 설정되었으며, 강제 전지 작업은 매번 1000개의 순차정보가 처리된 후 수행된다. 해당 매개변수들[14]은 FWM의 특성에는 영향을 미치지 않는 것으로 본 논문에서는 상세 설명은 생략한다.




[그림 3] 실험을 위한 기본 퍼지 캘린더 : (a)주 단위 퍼지 캘린더, (b) 시 단위 퍼지 캘린더

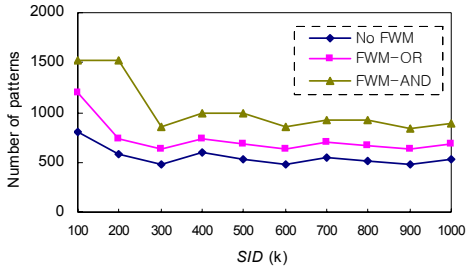
[Fig. 3] Fuzzy calendars for experiments: (a) The time granularity of week, (b) The time granularity of hour

본 절에서 기술되는 실험에서는 그림 3에서 제시된 기본 퍼지 캘린더에 대해 'AND' 및 'OR' 연산을 적용한 두 개의 퍼지 캘린더를 활용한다. 그림에서 제시된 기본 퍼지 캘린더 중 하나는 주 단위 퍼지 캘린더로서 주 중반(middle of a week)을 나타내는 것이며, 다른 하나는 시간 단위 퍼지 캘린더로서 일과시간(day time)을 나타낸다. 두 개의 기본 퍼지 캘린더에 대한 'AND' 연산에서는 순차정보의 발생 시점이 '화/수/목 요일이면서 8시부터 18시까지'인 경우 유효한 가중치를 갖게 되는 반면 '화/수/목 요일이라도 7시 이전 또는 19 이후'는 가중치가 0으로 부여된다. 'OR'연산에는 순차정보의 발생 시점이 '화/수/목 요일'인 경우 일과시간 범위에 무관하게 유효한 가중치를 갖게 되며, 또한 '8시부터 18시까지' 범위인 경우 요일에 무관하게 유효한 가중치를 갖게 된다.

제안된 기법의 특성을 보다 명확히 분석하기 위해서 실험에서는 서로 다른 두 개의 기본 데이터 집합을 조합하여 생성된 *abSDS* 데이터 집합을 사용하였다. 해당 데이터 집합 *abSDS*는 *part_A* 및 *part_B*라는 두 부분으로 나눠진다. *p*art_A는 단위항목(item) 집합 *set_A*를 기반으로 생성된 순차정보들의 집합이며, *p*art_B는 단위항목 집합 *set_B*를 기반으로 생성된 순차정보들의 집합이다. *part_A*및 *part_B*는 동일한 데이터 생성 도구[15]에 의해 생성되었으나, 서로 다른 단위항목 집합을 기반으로 생성

된 것으로서 둘 사이에 공통되는 단위항목은 존재하지 않는다. 해당 데이터 집합 *abSDS*는 총 100만개의 순차정보로 구성되며 각 순차정보는 '년월일' 및 '시분초'로 구성되는 발생 시간 정보를 갖는다. 이때, 발생 요일(day)이 수요일인 순차정보는 *part_B*에 속하며, 다른 요일인 경우는 part_A에 속한다. 즉, 생성 요일이 수요일인 순차정보들은 단위항목 집합 *set_A*를 기반으로 생성된 순차정보이며, 그 밖의 순차정보들은 단위항목 집합 *set_B*를 기반으로 생성된 순차정보이다. 순차정보의 발생 요일은 '년월일' 정보로부터 구할 수 있다.

FWM을 적용한 데이터 스트림 마이닝에서 마이닝 결과 집합의 변화를 분석하기 위해서 *abSDS* 데이터 집합에 대한 실험을 수행하였다. 본 실험에서 관심 순차패턴을 위한 지지도 임계값은 0.1%로 설정되었다. 그림 4는 실험 결과로 얻어지는 순차패턴들 중에서 에서 유도된 순차패턴들의 개수를 보여준다. 앞서 기술한 바와 같이 *part_B*에 속하는 순차정보는 순차정보의 발생 요일이 수요일인 경우로서 그림의 결과는 수요일에 발생한 순차정보로부터 유도된 순차패턴들의 수를 나타낸다. FWM 적용시 사용된 퍼지 캘린더에 따라 'FWM-AND' 및 'FWM-OR'로 구분되며, 두 경우 모두 FWM을 적용하지 않은 경우('No FWM'으로 표시)에 비해 *part_B*에서 유도된 순차패턴들의 개수가 많음을 알 수 있다. 그림 3-(a)에서 알 수 있듯이 실험에서 사용된 퍼지 캘린더는 수요일에 발생된 정보가 가장 큰 가중치를 가지며 화/수/목요일을 제외한 나머지 요일에 발생된 정보는 가중치가 0이며 무효한 것으로 간주된다. 해당 퍼지 캘린더에 기반한 FWM을 적용하는 경우 수요일에 발생된 순차정보들이 상대적으로 보다 높은 중요성을 갖는 것으로 간주되고, 따라서 해당 순차정보들에서 기인한 순차패턴들이 마이닝 결과 집합에 더 많이 포함되므로 그림 4와 같은 결과를 얻게 된다. 한편, 두 개의 기본 퍼지 캘린더에 대한 'OR' 연산의 경우 시간 축 기준의 퍼지 캘린더에 의해 수요일 이외에 발생한 순차정보들도 유효한 가중치를 갖는다. 반면, 둘 사이의 'AND' 연산의 경우 수요일 이외의 요일에 발생한 순차정보들은 무효한 것으로 간주되어 수요일에 발생된 순차정보 혹은 해당 순차정보에서 발생한 순차패턴들이 보다 큰 중요성을 갖는 것으로 평가된다. 따라서 그림에서 보는 바와 같이 두 개의 기본 퍼지 캘린더에 대한 'AND' 연산인 경우에 더 많은 수의 *part_B*에서 유도된 순차패턴들이 마이닝 결과로 탐색된다.



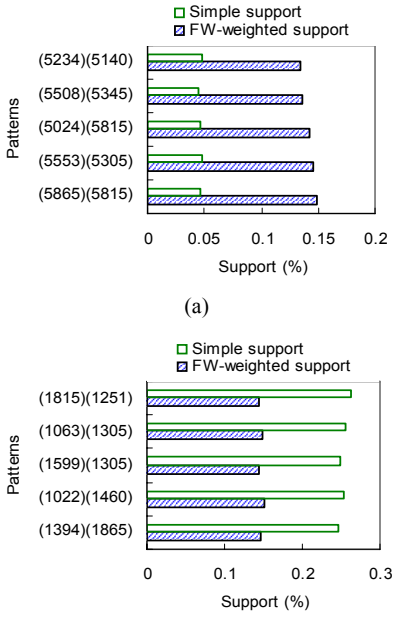

**[그림 4]** *abSDS* 데이터 집합에 대한 실험에서 *part_B*에서 유도된 패턴의 수

**[Fig. 4]** Number of patterns derived from *part_B*

그림 5에서는 같은 실험으로 얻어진 마이닝 결과 집합에 포함된 몇 개의 순차패턴에 대해서 FW-가중치 지지도(FW-weighed support)와 일반적인 단순 지지도(simple support)를 비교하였다. 단순 지지도라 함은 기존의 일반적인 관심 순차패턴 탐색 시 활용되는 단순 출현빈도수에 기반한 지지도를 의미한다. 해당 그림에서는 두 개의 기본 퍼지 캘린더에 대한 'AND' 연산의 결과만 제시하였으며, 둘 사이의 'OR' 연산의 경우에도 지지도 증감 정도는 다소 차이가 있으나 전체적으로 비슷한 경향을 보인다. 그림에서 Y축은 비교 대상이 되는 순차패턴을 의미하며, 각 숫자는 단위항목을 의미한다. 예를 들어, '(5234)(5140)'은 두 개의 단위 항목 5234와 5140으로 이루어진 순차패턴을 의미한다. 그림에서 알 수 있듯이 part_B에서 유도된 순차패턴들의 경우에는 퍼지 가중치 지지도가 일반 단순 지지도에 비해 크게 증가되는 반면, *part_A*에서 유도된 순차패턴들의 퍼지 가중치 지지도는 일반 단순 지지도에 비해 상당히 감소되었다. FWM에서는 앞서 기술한 바와 마찬가지로 주어진 퍼지 캘린더에 의해 *part_B*에 해당되는 순차정보들(즉, 수요일에 발생된 순차정보들)은 가중치가 상대적으로 증가되는 반면 part_A에 해당되는 순차정보들(즉, 수요일 이외의 요일에 발생된 순차정보들)은 가중치가 상대적으로 감소되기 때문이다.

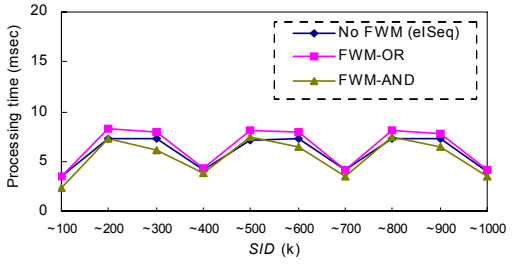
FWM을 적용한 데이터 스트림 마이닝의 효율성 검증의 수단으로 데이터 마이닝 기법의 기본 성능 중의 하나인 마이닝 수행 시간을 비교하였다. 일반적으로 마이닝 수행 시간은 하나의 단위 정보를 처리하는데 필요한 시간을 비교하며, 본 실험에서는 FWM을 적용하지 않은 기본적인 eISeq[14]와 FWM을 적용하는 경우의 마이닝 수행 시간을 비교하였다. [14]에서 제시된 바와 같이 eISeq 방법은 감쇠율 기법을 적용하는 경우에도 수행시간 등의 성능이 거의 동일하게 유지된다. 그림 6은 *abSDS* 데이터

집합에 대한 실험에서 각 10만 개의 순차정보로 구성되는 열 개의 구간으로 나누고, 각 구간에서 하나의 순차정보를 처리하는데 소요된 시간의 평균값을 보여준다. 그림에서 보듯이 각 구간에서 'FWM-AND' 및 'FWM-OR' 두 경우 모두 마이닝 수행 시간이 eISeq의 경우와 거의 유사하며, 10 msec보다 작음을 알 수 있다. 즉, 본 논문에서 제안된 FWM을 적용하는 경우에도 효율적으로 마이닝 작업을 수행함을 알 수 있다.




[그림 5] FW-가중치 지지도와 단순 지지도의 비교 : (a) *part_B*에서 유도된 패턴, (b) *part_A*에서 유도된 패턴

[Fig. 5] Support comparison between an FW-weighted support and a simple support: (a) Patterns derived from *part_B*, (b) Patterns derived from *part_A*.




[그림 6] 수행 시간

[Fig. 6] Processing time

## 5. 결론

데이터 마이닝에서 경계점 문제를 보완하고 실세계의 특성이나 요구를 보다 효율적으로 반영하기 위한 방법으로 퍼지 개념을 적용한 마이닝 방법들이 다수 제안되어 왔다. 대부분의 이들 연구들은 한정적인 데이터 집합에서 단위항목의 양적인 정보 및 발생 시간 간격 등과 같이 다양한 값으로 표현될 수 있는 정보의 중요성을 효율적으로 차별화 하는데 초점을 맞추고 있다. 본 논문에서는 이러한 퍼지 개념을 데이터 스트림 마이닝에서 정보의 중요성 차별화에 적용하였다. 즉, 구성요소가 지속적으로 생성되고 그 특성도 시간 흐름에 따라 매우 가변적인 데이터 스트림에 있어서 구성요소의 중요성을 생성 시간을 고려하여 차별화하기 위한 방법으로 FWM을 제안하였다. 데이터 스트림에 대한 분석에서 최근에 발생된 정보에 집중된 분석 결과를 제공하는 이동 윈도우 기법이나 감쇠 기반 기법과 달리 FWM은 보다 유연하고 다양한 형태의 정보 중요성 차별화를 지원한다.

유통 관련 회사의 구매/판매 정보, 전자 상거래 관련 데이터 및 웹 이용 로그, 센서 네트워크 환경에서 발생되는 센싱 정보 등 근래 대다수의 컴퓨터 응용 환경에서 발생되는 정보는 데이터 스트림 형태로 정보를 발생 시키고 있다. 본 논문에서 제안된 FWM은 해당 데이터 스트림의 특성 분석을 위한 데이터 마이닝 과정에 유연한 정보 차별화를 지원할 수 있으며, 다양한 퍼지 캘린더를 통해 분석 과정에서 실세계의 요구나 제한조건을 보다 정확하게 표현할 수 있다.

본 논문에서는 퍼지 개념을 활용한 데이터 스트림의 정보 차별화에 대한 기본적인 내용을 기술하였다. 제안된 기법을 실제 응용 분야에서 보다 널리 활용하기 위해서는 다양한 응용 분야의 특성에 맞는 최적의 퍼지 캘린더를 찾는 작업을 필요로 한다. 예를 들어 초 단위 혹은 그보다 작은 시간 단위로 수없이 많은 정보들이 발생되는 센서 네트워크 환경에서는 기준 시간 단위를 매우 작게 설정한 퍼지 캘린더를 필요로 하는 반면, 대형 마트의 판매 기록과 같이 상대적으로 긴 정보 생성 주기를 갖는 경우에는 기준 시간 단위를 보다 큰 단위로 설정한 퍼지 캘린더도 효율적으로 적용될 수 있다. 한편, 시간 흐름에 따라 지속적으로 변화될 수 있는 데이터 스트림의 특성을 고려하여 FWM을 적용한 데이터 마이닝 과정에서 동적으로 퍼지 캘린더를 갱신할 수 있도록 지원한다면 본 논문에서 제안된 기법의 활용도를 높일 수 있을 것이다. 이러한 것들은 흥미로운 향후 연구주제가 될 수 있을 것이다.

## References


[1] S.-C. Chiu, H.-F. Li, J.-L. Huang, and H.-H. You, " Incremental Mining of Closed Inter-Transaction Itemsets over Data Stream Sliding Windows," Journal of Information Science, Vol. 37, No. 2, pp. 208-220, 2011.

[2] H.-F. Li and S.-Y. Lee, "Mining Frequent Itemsets over Data Streams using Efficient Window Sliding Techniques," Expert Systems with Applications, Vol. 36, No. 2, pp. 1466-1477, 2009.

[3] H.-F. Li, "Pattern Discovery and Change Detection of Online Music Query Streams," Multimedia Tools and Applications, Vol. 41, No. 2, pp. 287-304, 2009.

[4] L. Jia, Z. Wang, N. Lu, X. Xu, D. Zhou, and Y. Wang, "RFIMiner: A Regression-based Algorithm for Recently Frequent Patterns in Multiple Time Granularity Data Streams," Applied Mathematics and Computation, Vol. 185, No. 2, pp. 769-783, 2007.

[5] J.H. Chang and W.S. Lee, "Finding Recently Frequent Itemsets Adaptively over Online Transactional Data Streams," Information Systems, Vol. 31, No. 8, pp. 849-869, 2006.

[6] Y.-H. Kim, W.-Y. Kim, and U.-M. Kim, "An Efficient Method for Mining Frequent Patterns based on Weighted Support over Data Streams," Journal of the Korea Academia-Industrial Cooperation Society, Vol. 10, No. 8, pp. 1998-2004.

[7] W.-J. Lee and S.-J. Lee, "Discovery of Fuzzy Temporal Association Rules," IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics, Vol. 34, No. 6, pp. 2330-2342, 2004.

[8] Y.-L. Chen, M.-C. Chiang, and M.-T. Ko, "Discovering Fuzzy Time-Interval Sequential Patterns in Sequence Databases," IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics, Vol. 35, No. 5, pp. 959-972, 2005.

[9] T. C.-K. Huang, "Developing an Efficient Knowledge Discovering Model for Mining Fuzzy Multi-level Sequential Patterns in Sequence Databases," Fuzzy Sets and Systems, Vol. 160, No. 23, pp. 3359-3381, 2009.

[10] D.L. Olson and Y. Li, "Mining Fuzzy Weighted Association Rules," Proc. of the 40th Hawaii International Conference on System Sciences, pp. 53-61, 2007.

[11] C.-J. Li and T.-Q. Yang, "Effective Mining of Fuzzy Quantitative Weighted Association Rules," Proc. of the Int'l Conf. on E-Business and E-Government, pp. 1418-1421, 2010.

[12] Y.-M. Wang and T. M.S. Elhag, "On the Normalization of Interval and Fuzzy Weights," Fuzzy Sets and Systems, Vol. 157, No. 18, pp. 2456-2471, 2006.

[13] S. Ramaswamy, S. Mahajan, and A. Silberschatz, "On the Discovery of Interesting Patterns in Association Rules," Proc. of the Int'l Conf. on Very Large Database, pp. 368-379, 1998.

[14] J.H. Chang and W.S. Lee, "Efficient Mining Method for Retrieving Sequential Patterns over Online Data Streams," Journal of Information Science, Vol. 31, No. 5, pp. 420-432, 2005.

[15] R. Agrawal and R. Srikant, "Mining Sequential Patterns," Proc. of the Int'l Conf. on Data Engineering, pp. 3-14, 1995.



**장 중 혁**(Joong-Hyuk Chang) **[정회원]**

- 1996년 2월 : 연세대학교 컴퓨터과학과 (이학사)
- 1998년 8월 : 연세대학교 컴퓨터과학과 (공학석사)
- 2005년 8월 : 연세대학교 컴퓨터과학과 (공학박사)
- 2006년 1월 ~ 2008년 7월 : UIUC, Wright State University 박사후 연구원
- 2008년 9월 ~ 현재 : 대구대학교 컴퓨터IT공학부 교수

<관심분야>
데이터 스트림, 데이터 마이닝, 데이타베이스