

# Naïve Bayesian 분류화 기법을 이용한 시간대별 평균 구간 속도 기반 주행 시간 예측 알고리즘<sup>†</sup>

## Travel Time Prediction Algorithm Based on Time-varying Average Segment Velocity using Naïve Bayesian Classification

엄정호\* / Jung-Ho Um, 니하드 카림 초우더리\*\* / Nihad Karim Chowdhury,  
이현조\*\*\* / Hyun-jo Lee, 장재우\*\*\*\* / Jae-Woo Chang, 김연중\*\*\*\* / Yeon-Jung Kim

### 요약

주행 시간 예측은 첨단 여행 정보 시스템(ATIS) 및 교통 관리 시스템(ITS)에서 필수적이다. 이를 위해 본 연구에서는 대용량의 데이터 분류에서 높은 정확도와 빠른 속도를 보장하는 Na ve Bayesian 분류화 기법을 기반으로 한 주행 시간 예측 알고리즘을 제안한다. 제안된 알고리즘은 도로 네트워크 상에서 사용자 지정 주행 경로에 대하여 주행 시간 예측이 가능하며, 또한 주어진 경로에 대해 시간대별 평균 구간 속도를 고려하여 보다 정확한 주행 시간 예측을 수행한다. 제안된 알고리즘을 기존의 링크-기반 예측(link-based prediction) 알고리즘[1] 및 Micro T\* 알고리즘[2]과 성능 비교를 수행하였다. 성능 비교 결과, 제안된 기법이 타 예측 기법에 비해 MARE (mean absolute relative error) 가 크게 감소하여 성능이 향상되었음을 보였다.

### Abstract

Travel time prediction is an indispensable to many advanced traveler information systems(ATIS) and intelligent transportation systems(ITS). In this paper we propose a method to predict travel time using Na ve Bayesian classification method which has exhibited high accuracy and processing speed when applied to classify large amounts of data. Our proposed prediction algorithm is also scalable to road networks with arbitrary travel routes. For a given route, we consider time-varying average segment velocity to perform more accuracy of travel time prediction. We compare the proposed method with the existing prediction algorithms like link-based prediction algorithm [1] and Micro T\* algorithm [2]. It is shown from the performance comparison that the proposed predictor can reduce MARE (mean absolute relative error) significantly, compared with the existing predictors.

**주요어** : 첨단 여행 정보 시스템(ATIS), Na ve Bayesian 분류화 기법, linear regression, 주행 시간 예측

**Keyword** : Advanced travelers information system (ATIS), Naïve Bayesian Classification, linear regression, travel time prediction

<sup>†</sup> 이 논문은 2008년도 정부(교육과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행된 연구임(No. R01-2008-000-11002-0)

■ 접수일 : 2008.04.02    ■ 수정일 : 2008.06.09    ■ 심사완료일 : 2008.06.13

\* 전북대학교 컴퓨터공학과 박사과정(jhum)@dmlab.chonbuk.ac.kr

\*\* 전북대학교 대학원 컴퓨터공학과 석사과정(nihad)@dmlab.chonbuk.ac.kr

\*\*\* 전북대학교 대학원 컴퓨터공학과 박사과정(hjlee)@dmlab.chonbuk.ac.kr

\*\*\*\* 교신저자 전북대학교 컴퓨터공학과 교수(jwchang)@chonbuk.ac.kr

\*\*\*\*\* 전북대학교 대학원 컴퓨터공학과 박사과정수료(yikim)@dmlab.chonbuk.ac.kr

## 1 서론

첨단 여행 정보 시스템(Advanced Travelers Information Systems: ATIS) [1]의 발전에 따라, 이를 위한 효율적인 주행 시간 예측 알고리즘의 필요성이 증대되었다. 뿐만 아니라 다양한 지능형 교통 시스템 (intelligent transportation system: ITS), 예를 들어 여행 계획 수립 및 동적 경로 안내, 주행 시간 예측 등은 사용자로 하여금 보다 효과적인 여행 일정 관리를 가능하게 한다. 사용자들의 만족도를 높이기 위해서는 주어진 주행 경로, 즉 출발지에서 목적지까지의 주행 시간을 보다 정확하게 예측해야 한다. 그러나 차량 속도 및 교통량 기반의 주행 시간 예측 알고리즘은 날씨, 교통 사고, 교통 체증과 같은 도로 구간별 교통 상황에 민감하다 [3]. 이는 전체 경로에서의 평균 차량 운행 시간만을 이용하여 주행 시간을 예측하므로, 각 도로 구간별 교통 상황을 반영할 수 없기 때문이다. 따라서 주행 시간 예측의 정확도가 저하된다. 도로 구간별 교통 상황의 영향 요인 중 하나는 일별 또는 주별 단위로 발생하는 상황이다. 예를 들어, 일별 교통 상황은 시간대에 따라 아침, 저녁, 그리고 러시아워로 구분할 수 있으며, 각 시간대별 교통량 및 차량의 평균 주행 속도가 현저히 다를 수 있다. 따라서 보다 효율적인 주행 시간 예측을 위해서는 이러한 도로 구간별 교통 상황을 고려해야 한다.

본 논문에서는 신뢰성 있는 보다 정확한 주행 시간 예측을 위해, Naïve Bayesian 분류화 기법 기반의 주행 시간 예측 알고리즘을 제안한다. 제안하는 알고리즘은 도로 네트워크 상의 사용자로부터 주어진 임의의 경로에서 주행 시간의 예측이 가능하다. 아울러 하루를 여러 시간 그룹으로 나누어 각 시간 그룹별 평균 구간 속도를 고려하여, 보다 정확도가 높은 주행 시간 예측이 가능하다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구를 살펴보고, 3장에서는 제안된 알고리즘에 대하여 설명한다. 4장에서는 제안된 알고리즘의 성능 평가를 수행하여, 마지막으로, 5장에서는 결론 및 향후 연구를 제시한다.

## 2 관련 연구

동적 경로 안내 시스템의 사용자 만족도 향상을 위해서는 질의 경로에 대한 정확한 주행 시간 예측이 매우 중요하다. 이에 따라 도로 네트워크 상에서 보다 정확한 주행 시간 예측을 지원하는 알고리즘에 대한 연구가 활발히 진행되고 있다. 대부분의 기존 연구들은 주행 시간 예측을 위해 경로-기반(path-based) 기법을 사용하였다[4, 5, 6, 7, 8, 9]. 경로-기반 기법이란, 시스템에 저장된 특정 경로에 대해서 주행 시간을 예측하는 기법이다. 시스템에 저장된 특정 경로란, 시스템에서 도로 네트워크 상의 출발 지점, 도착 지점 및 두 지점 사이의 전체 경로를 정의한 것으로, 두 지점 사이의 전체 경로는 사용자가 많이 이용하거나 가장 짧은 거리를 지나는, 적은 수의 경로만을 지정한다. 경로-기반 기법으로는 다음 연구들이 존재한다. 첫째, Park et al. [4, 5] 연구는 인공 신경망(Artificial Neural Network)을 이용한 고속 도로에서의 주행 시간 예측 알고리즘을 제안하였다. 제안된 알고리즘은 시스템에 저장된 특정 경로의 교통 상황 패턴을 각각 SOFM (Self Organizing Feature Map) 기법 및 fuzzy c-means 클러스터링 (clustering) 기법을 이용하여 분류하였으며, 분류된 교통 상황 패턴을 통해 주행 시간을 예측한다. 이는 우수한 성능을 나타내지만, 알고리즘이 복잡하고, 많은 계산 시간을 요구하기 때문에 실제 응용에는 적합하지 않다. 둘째, Kwon et al [6] 연구는 출발 시간 범위를 고려한 주행 시간 예측 알고리즘을 제안하였다. 제안한 알고리즘은 현재 시간과 출발하고자 하는 시간과의 시차에 따라, 시차가 20분 이내일 경우 현재 관측되고 있는 차량 주행 시간을, 시차가 20분 이상일 경우 과거 차량 주행 시간 기록을 사용하여 주행 시간을 예측한다. 셋째, Zhang et al [7] 연구는 출발 시간에 따라 주행 시간 측정 함수의 계수가 가변적으로 변하는 주행 시간 예측 알고리즘을 제안하였다. 즉, 출발 시간에 따라 평균 주행 시간이 변하기 때문에, 이를 주행 시간 예측을 위한 함수에 반영하여 보다 정확하게 주행 시간을 예측한다. 마지막으로, Rice et al [9] 연구는 유사 주행 시간 패턴을 고

려한 최근접(nearest neighbors) 주행 시간 예측 알고리즘을 제안하였다. 즉, 현재 관측되는 데이터를 토대로 과거 주행 시간 기록 중 현재 날짜와 가장 유사한 일자의 데이터를 탐색하여, 주행 시간을 예측한다.

경로-기반 기법들은 도시와 도시간의 이동과 같은 고속 도로에서의 장거리 주행 상황에서 주행 시간을 예측하는데 적합하다. 즉, 주행이 가능한 경로가 적어, 시스템에서 전체 경로를 정의하기 편리한 경우에 사용된다. 반면 시스템에 저장된 특정 경로에서만 주행 시간 예측을 지원하기 때문에, 도로 구간별 교통 상황을 반영할 수 없으며, 또한 도심과 같이 다양한 경로에 대한 사용자 질의를 지원할 수 없으므로 정확도가 저하된다. 보다 정확한 주행 시간 예측을 위하여 Chen et al [1] 연구는 링크-기반(link-based) 주행 시간 예측 알고리즘을 제안하였다. 링크-기반 기법이란, 전체 경로를 구성하는 각 도로 구간(road segment)에서의 주행 시간을 측정하고, 측정된 구간별 주행 시간을 합산하여 총 주행 시간을 예측하는 기법이다. 각 구간별 교통 상황을 고려하여 정확도는 향상되었으나, 앞서 살펴본 경로-기반 연구들과 마찬가지로, 시스템에 저장된 특정 경로에서만 주행 시간 예측이 가능한 문제점이 있다. 이러한 문제점을 해결하기 위해, Kwon et al [2] 연구는 사용자 지정 경로에 대해 주행 시간을 예측하는 Micro T\* 알고리즘을 제안하였다. 사용자 지정 경로란 사용자로부터 출발지점, 도착지점 및 두 지점 사이의 전체 경로가 주어지는 것이다. Micro T\* 알고리즘은 먼저 전체 도로 네트워크 상의 모든 구간에 대해 구간별 평균 주행 시간을 측정하며, 만약 질의 경로가 주어지면 이를 구성하는 모든 구간의 평균 주행 시간을 합산하여 전체 주행 시간을 예측한다. 그러나 Micro T\* 알고리즘은 도로의 정체와 같은 외부적인 요인에 의해 시간대별로 그 예측 정확도가 저하되는 문제점이 존재한다.

### 3. Naive Bayesian 분류화 기법을 이용한 시간대별 평균 구간 속도 기반 주행 시간 예측 알고리즘

최근 첨단 교통정보 시스템의 발달과 더불어 다수의 주행 시간 예측 알고리즘들이 제안되었다. 그러나 기존에 제안된 대부분의 알고리즘들은 사용자 지정 경로에 대한 주행 시간 예측을 지원하지 못한다. 이러한 알고리즘들은 시스템에 저장된 특정 경로에 대한 주행 시간만을 예측한다. 따라서 이들 알고리즘을 사용하는 교통 정보 시스템은 사용자들이 자주 운행하는 경로를 미리 정의하고, 정의된 경로에 대한 주행 시간 예측 서비스만을 지원한다. 이러한 제약은 서로 다른 경로에 대한 다양한 사용자 질의를 고려한다면 적합하지 않으며, 결국 사용자 만족도를 저하시킨다. 이러한 문제를 해결하기 위해 사용자 지정 경로를 고려하는 주행 시간 예측 알고리즘인 Micro T\*[2]가 제안되었다. Micro T\* 알고리즘에서는 경로(route)를 여러 개의 도로 구간(load segments)으로 분할하여, 각 도로 구간마다 TVC(Time-Varying Coefficient) 기법을 기반으로 주행 시간을 측정한다. 최종적으로 측정된 구간별 주행 시간을 더하여 주행 시간을 예측한다. 그러나 Micro T\* 알고리즘은 도로 구간의 평균 주행 시간만을 고려하며, 러시아워나 새벽 시간과 같이 서로 다른 시간대의 주행 속도를 고려하지 못하여 정확성이 저하된다.

이러한 문제점을 해결하기 위해, 본 논문에서는 Naive Bayesian 분류화 기법을 이용한 새로운 주행 시간 예측 알고리즘을 제안한다. 제안된 알고리즘은 Micro T\* 알고리즘과 마찬가지로 사용자 지정 경로에 대해 주행 시간 예측이 가능할 뿐만 아니라, 경로를 구성하는 각 도로 구간에 대해 시간대별 평균 구간 속도를 고려함으로써 보다 정확한 주행 시간 예측이 가능하다. 시간대별 평균 구간 속도를 위해 구간별 속도 등급(velocity class)을 고려한다. 예를 들어 교통량이 많은 아침 러시아워의 시간대와 반대로 교통량이 적은 늦은 저녁의 시간대는 서로 구별되며, 이러한 시간대별 교통 상황을 고려하기 위해 하루를 다수개의 시간 그룹으로 나눈다. 도로 구간별 속도 등급이란 각 도로 구간마다 시간 그룹별 교통량의 측정값을 의미하며, 이를 측정하기 위해 본 논문에서는 Naive Bayesian 분류

화 기법을 사용한다. 사용된 Naïve Bayesian 분류화 기법 [10]은 결정 트리(decision tree)나 신경망(neural network)과 같은 다른 분류화 기법에 비해 그 정확도가 높으며, 계산 비용이 낮고, 또한 적은 양의 훈련 데이터(training data)를 요구한다. Naïve Bayesian 분류화 기법을 사용하여 도로 구간별 속도 등급을 측정하면, 이를 통해 쉽게 각 도로 구간별 주행 시간을 계산할 수 있으며, 질의 경로를 구성하는 도로 구간별 주행 시간을 합산하여 총 주행 시간을 예측할 수 있다.

### 3.1 용어 정의

본 절에서는 도로 네트워크에서의 주행 시간 예측에 필요한 용어를 정의한다.

*정의 1: 고속 도로란 수백에서 수천 미터의 도로가 연결되어 있는 것을 의미한다. 또한 이는 몇 개의 차선으로 이루어져 있으며, 하나 이상의 출구를 가질 수 있다.*

*정의 2: 고속 도로의 경로는 유한한 숫자의 연속적인 구간(segment) 혹은 도로(road)로 이루어져 있다.*

하나의 도로 구간(road segment)은  $s$  로, 하나의 경로(route)는  $r$ 로 표기한다. 한편, 다수의 도로 구간을 통한 경로의 구성 방법, 각 도로 구간별 주행 속도 계산 방법, 그리고 계산된 도로 구간별 주행 속도의 합산 방법에 대해 설명한다.

*정의 3: 출발 시간  $t$ 에서 경로  $r$ 에 대한 개별 주행 시간(individual travel time)이란, 여행자 개인이 특정 경로를 주행하는데 필요한 시간을 의미한다.*

*정의 4: 일정 시간 주기(period)  $p$ 에 출발한 차량의 경로  $r$ 에 대한 평균 주행 시간(mean travel time)이란, 시간 주기  $p$  동안 경로  $r$ 을 따라 차량이 이동하였을*

*때의 평균 주행 시간을 의미한다.*

개별 주행 시간 및 평균 주행 시간은 과거에 기록된 주행 시간 데이터를 통해 정의된다. 이는 여행자 그룹이 관심있는 경로를 운행하여 해당 경로에 대한 주행 데이터가 존재할 때, 평균 주행 시간이 측정 가능함을 의미한다.

### 3.2 속도 등급 측정 알고리즘

본 절에서는, 각 도로 구간별 속도 등급을 측정하는 알고리즘을 제안한다. 이를 위해 속도 등급 및 출발 시간 그룹을 제시하며, 이에 기반한 속도 등급 사전 측정 알고리즘 및 속도 등급 사후 확률 측정 알고리즘을 제시한다.

#### 3.2.1 속도 등급과 출발 시간 그룹

도로 네트워크에서 속도 등급(velocity class)은 하루의 시간대에 따라 달라질 수 있다. 이를 반영하기 위해, 각 도로 구간별 속도를 세 등급으로 분류한다. “속도 등급”의 각 속성들은 {VB, B, F}로 써, 각각 Very Busy, Busy 그리고 Free 를 의미한다. 만약 차량의 운행 속도가 0 ~ 35 km/hour의 범위 내에 존재한다면, 속도 등급은 VB (Very Busy)이다. 도로 구간에서의 속도 등급은 <표 1>과 같이 분류한다.

<표 1> 속도 등급(Velocity Class)

속도 범위(km/hour)	속도 등급
0 ~ 35	VB
36 ~ 60	B
> 60	F

트래픽 데이터 튜플(tuple)에서  $g_1, g_2, \dots, g_n$  (where  $1 \leq i \leq n$ )를 출발 시간 그룹이라 하면, 하나의 튜플은 임의의 출발 시간 그룹  $g_i$ 으로 나타낼 수 있다. 예를 들어, 차량이 임의의 도로 구간에서 오전 10:00~11:00 사이에 출발하였다면, 출발 시

간 그룹은 1이 된다. <표2>는 9개의 출발 시간 그룹을 나타낸다.

<표 2> 출발 시간 그룹

출발 시간 범위	출발 시간 그룹
10:01 ~ 11:00	1
11:01 ~ 12:00	2
12:01 ~ 14:00	3
14:01 ~ 16:00	4
16:01 ~ 18:00	5
18:01 ~ 22:00	6
22:01 ~ 00:00	7
00:01 ~ 06:00	8
06:01 ~ 10:00	9

### 3.2.2 속도 등급 사전 확률 측정 알고리즘

임의의 도로 구간에 대한 정량적인 주행 시간을 계산하기 위해, 속도 등급 사전 확률 측정 알고리즘 (Velocity Class Prior Probability Measure Algorithm)을 제안한다. 이 알고리즘은 각각의 속도 등급을 위한 사전 확률을 계산하며 알고리즘은 다음과 같다. 먼저 임의의 도로 구간에 대한 튜플의 총 개수를  $tt$ 값으로 설정한다. 다음 프로시저 *Contain* ( $t, c$ ) 를 통해 튜플  $t$  가 속도 등급  $c$ 를 포함하는지의 여부를 결정한다. 각 속도 등급의 총 개수는 변수  $c.count$ 에 저장되며, 최종적으로  $PMc[c]$ 를 통해 각 속도 등급에 대한 사전 확률을 측정한다. <그림 1>은 속도 등급 사전 확률 측정 알고리즘을 나타낸다.

---

#### **Velocity Class Prior Probability Measure (C)**

*Input:* Velocity Class attributes  $c \in C$

*Output:* Velocity Class Prior Probability ( $PMc$ ) and Velocity Class Count( $c.count$ )

```

{
1) Total Tuple =  $tt$ ;
2) for each class attribute  $c \in C$  do
3)   for each tuple  $t$  in database do
4)     if Contain( $t, c$ ) then
5)        $c.count++$ ;
6)     end if
7)   end for each tuple
8) end for each  $c$ 
9) for each class attribute  $c \in C$  do
10)  $PMc[c] = c.count / tt$ ;
11) end for each class
12) return  $PMc$ ;
}
    
```

#### **Procedure Contain (t, c)**

```

{
13) Let  $S = c \mid c \in C$ 
14) If  $S \leq t$  then return 1 else return 0;
15) end if
}
    
```

---

<그림 1> 속도 등급 사전 확률 측정 알고리즘

### 3.2.3 속도 등급 사후 확률 측정 알고리즘

사후 확률을 계산하기 위해 각 속도 등급별 조건부 확률(conditional probability)을 측정한다. 제안한 속도 등급 사후 확률 측정 알고리즘(Velocity Class Posterior Probability Measure Algorithm)은 속도 등급의 사후 확률 및 조건부 확률, 그리고 최대 속도 등급에 대한 계산 과정을 모두 포함하며 그 과정은 다음과 같다. 먼저 알고리즘에서,  $M_{xc}$ 는 속도 등급의 조건부 확률을 계산하는데 이용된다. 다음  $CPMX_c$ 을 통해 분류하고자 하는 모든 속성값들에

대해, 등급별 조건부 확률을 측정된 후,  $PMX_c$ 를 이용하여 각 속도 등급별 사후 확률을 계산한다. 마지막으로 Maximize\_Velocity\_Class프로시저를 사용하여, 속도 등급 레이블(label)과 속도 등급을 계산한다. <그림 2>는 출발 시간이 주어졌을 때, 임의의 도로 구간에 대한 속도 등급 사후 확률 측정 알고리즘을 나타낸다.

### 3.2.4 속도 등급 측정 예제

트래픽 데이터 투플은 일곱 개의 속성값을 지닌

---

#### *Velocity Class Posterior Probability Measure(X, C)*

*Input: Road Segment Ids with Start Time Group and Velocity Class attribute*

*Output: Maximized Posterior Probability of Velocity Class attribute*

```

{
1)   $X = \{x_1, x_2 \mid x_1 \in \text{Road\_Segment\_List} \text{ and } x_2 \in \text{Group}\}$ 
2)  for each  $c \in C$  do
3)    for each  $x \in X$  do
4)       $M_{xc} = \emptyset$ ;
5)      for each  $t$  in Database do
6)        if  $(c \leq t \ \&\& \ x \leq t)$  then
7)           $xc.\text{count} ++$ ;
8)           $M_{xc} = xc.\text{count}/c.\text{count}$ ;
9)        end if
10)     end for each  $t$ 
11)   end for each  $x$ 
12) end for each  $c$ 
13) for each  $c \in C$  do
14)    $PM_c = \text{Velocity Class Prior Probability Measure}(c)$ ;
15)    $CPMX_c = 1$ ;
16)   for each  $x \in X$  do
17)      $CPMX_c *= M_{xc}$ ;
18)   end for each  $x$ 
19)    $PMX_c = CPMX_c * PM_c$ ;
20) end for each  $c$ 
21)  $\text{High\_Velocity\_Class} = \text{Maximize Velocity Class}(PMX_c)$ ;
22) return  $\text{High\_Velocity\_Class}$ ;
}

```

---

<그림 2> 속도 등급 사후 확률 측정 알고리즘

다. 출발 시간과 도착 시간은 차량이 특정 도로 구간을 지나는 시간 주기를 의미한다. 각 레코드는 출발시간과 도착시간뿐만 아니라, 차량ID, 구간ID, 출발 시간 그룹, 속도, 그리고 속도 등급을 포함한다. 출발 시간 그룹은 차량의 출발 시간에 의해 결정되며, 속도는 도로 구간의 길이를 출발 시간과 도착 시간의 차이로 나누어 계산하며, 이를 통해 속도 등급(VB, B, F)을 구한다. 예제에서 각 구간별 도로의 길이는 10 km이며, 트래픽 데이터 예제는 <표 3>과 같다.

<표 3> 트래픽 데이터 예제

차량 ID	구간 ID	출발 시간 그룹	출발 시간	도착 시간	속도 (km/hour)	속도 등급
1	1	1	10:00	10:23	26	VB
2	1	1	10:05	10:27	27	VB
3	1	1	10:05	10:25	30	VB
4	1	3	13:00	13:07	86	F
5	1	1	10:01	10:23	27	VB
6	1	1	10:00	10:09	67	F
7	1	3	13:05	13:13	75	F
8	1	3	13:06	13:15	67	F
9	1	9	6:01	6:13	50	B
10	1	1	10:30	10:43	46	B

<표 3>에서 속도 등급 VB, B, F의 총 개수는 각각 VB = 4, B = 2, F = 4 이며, 튜플의 총 개수 = 10개임을 알 수 있다. 따라서 속도 등급 별 사전 확률은 속도 등급 사전 확률 측정 알고리즘의  $PMc(C) = \text{속도 등급별 개수} / \text{총 튜플 수}$  에 따라 각각  $P(VB) = 0.4$ ,  $P(B) = 0.2$ ,  $P(F) = 0.4$  이다.

다음 도로 구간 및 출발 시간 그룹에 대한 속도 등급별 조건부 확률을 계산한다. <표 3>에서 속도 등급 = VB 일 때 항상 구간 = 1을 지나므로,  $P(\text{구간ID} = 1 | \text{속도 등급} = VB) = 1$ 이다. 도로 구간 및 출발 시간 그룹에 대한 속도 등급별 조건부 확률은 다음과 같다.

$$P(\text{구간ID} = 1 | \text{속도 등급} = VB) = 1$$

$$P(\text{구간ID} = 1 | \text{속도 등급} = B) = 1$$

$$P(\text{구간ID} = 1 | \text{속도 등급} = F) = 1$$

$$P(\text{출발 시간 그룹} = 1 | \text{속도 등급} = VB) = 1$$

$$P(\text{출발 시간 그룹} = 1 | \text{속도 등급} = B) = 0.50$$

$$P(\text{출발 시간 그룹} = 1 | \text{속도 등급} = F) = 0.25$$

이를 이용하여 최종적인 속도 등급 조건부 확률을 측정하며, 이는 다음과 같다. (X=입력의 튜플)

$$P(X | \text{속도등급} = VB)$$

$$= P(\text{구간ID} = 1 | \text{속도 등급} = VB) \times P(\text{출발 시간 그룹} = 1 | \text{속도 등급} = VB) = 1$$

$$P(X | \text{속도등급} = B)$$

$$= P(\text{구간ID} = 1 | \text{속도 등급} = B) \times P(\text{출발 시간 그룹} = 1 | \text{속도 등급} = B) = 0.50$$

$$P(X | \text{속도등급} = F)$$

$$= P(\text{구간ID} = 1 | \text{속도 등급} = F) \times P(\text{출발 시간 그룹} = 1 | \text{속도 등급} = F) = 0.25$$

앞서 구한 속도 등급의 사전 확률 및 조건부 확률을 이용하여, 속도 등급 사후 확률을 측정하며 이는 다음과 같다.

$$P(X | \text{속도 등급} = VB) \times P(\text{속도등급} = VB) = 0.40$$

$$P(X | \text{속도 등급} = B) \times P(\text{속도등급} = B) = 0.10$$

$$P(X | \text{속도 등급} = F) \times P(\text{속도등급} = F) = 0.10$$

마지막으로 High\_Velocity\_Class 프로시저와 속도 등급 사후 확률을 통해 구하고자 하는 속도 등급을 측정할 수 있다.

### 3.3 주행 시간 예측 알고리즘

앞에서 설명한 속도 등급 측정 알고리즘들을 바탕으로 주행 시간을 예측하는 주행 시간 예측 알고리즘을 제안한다. 제안하는 알고리즘은 앞서 제시한 속도 등급 사후 확률 측정 알고리즘을 이용하며, 다음과 같이 수행한다. 먼저, 질의 경로와 출발 시간을 입력 받은 후, 이에 맞는 출발 시간 그룹을 결정한다. 속도 등급 사후 확률 측정 알고리즘을 통해 도로 구간의 속도 등급을 측정 후, 이를 이용하여 해당 도로 구간에서의 도착 시간을 예측한다. 예측된 도착 시간을, 연속된 다음 도로 구간에서의 출발

**Travel Time Prediction (Start\_ Time, Road\_Segment \_List, C)**

*Input:* Start\_ Time gives start time for initial road segment.

*Road\_Segment \_List* contains all road segments from origin to destination. *C* contains all Velocity Class

*Output:* Display predicted travel time for every road segment from origin to destination

```

{
1)  Group= Convert Start_ Time to Group;
2)  for (i=0; Road_Segment _List!=NULL ; i++) do
3)    X.road_segment_id = Road_Segment _List .ID;
4)    X.group = Group;
5)    High_Velocity_Class = Velocity Class Posterior Probability
        Measure (X,C);
6)    End_Time = End Time Measure (X, High_Velocity_Class);
7)    Group= Convert End_Time to Group;
8)    Answer =  $\cup$  End_Time;
9)  end for
}

```

〈그림 3〉 주행 시간 예측 알고리즘

시간으로 설정한다. 질의 경로를 구성하는 모든 도로 구간에 대해, 위의 과정을 반복하여 임의의 도로 구간에 대한 주행 시간을 예측한다. 〈그림 3〉은 주행 시간 예측 알고리즘(Travel Time Prediction Algorithm)을 나타낸다.

한편, 주행 시간 예측 알고리즘에서 도착 시간 측

정 알고리즘(End Time Measure Algorithm)은 다음과 같다(6번째 줄). 먼저 구간ID와 출발 시간을 통해, 해당 구간에서의 출발 시간 그룹, 속도 등급을 구한다. 다음, 속도 등급을 이용하여 해당 구간의 평균 속도 및 도착 시간을 계산한다. 예를 들어 모든 도로 구간의 길이가 10km라고 하자. 구간

**End Time Measure (x, c)**

*Input:* Road Segment Length and Highest Probable Velocity Class

*Output:* End time of a Road Segment

```

{
1)  Ck = {c1, c2, ..., ck} ;
2)  for (i =1; i <= k; i++) do
3)    if (c == Ci) then
4)      Velocity = Ci.average_velocity ;
5)      End_Time = x.length / Velocity ;
6)    end if
7)  end for
8)  return End_Time
}

```

〈그림 4〉 도착 시간 측정 알고리즘



ID = 1, 출발 시간은 10시 05분이라면, 평균 구간 속도 및 주행 시간은 각각 28km/hour, 21 분이 된다. 따라서 사용자는 10시 26분에 도로 구간의 끝에 도착한다. <그림 4>는 도착 시간 측정 알고리즘을 나타낸다.

#### 4 성능 평가

##### 4.1 실험 환경

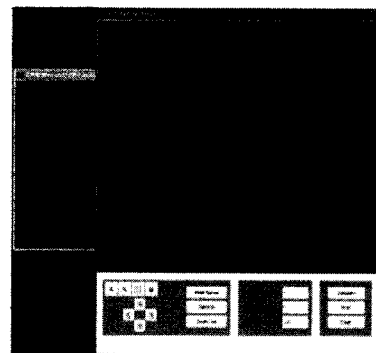
주행 시간 예측 기법간에 성능을 측정하기 위해, mean absolute relative error (MARE) [1]를 사용한다. MARE는 가장 간단하고, 또한 잘 알려진 에러 측정 방법이다. MARE 는 원하는 시간 범위 내에서 연관된 에러의 등급을 측정하며, 식 (1)와 같이 정의된다.

$$MARE = \frac{1}{N} \sum_t \frac{|x(t) - x^*(t)|}{x(t)} \quad (1)$$

$x(t)$  는 관측값,  $x^*(t)$  는 예상값, 그리고 N 은 샘플의 수를 나타낸다. 실험에 사용된 데이터는 synthetic 데이터 집합과 real 데이터 집합이며, 각각의 특성은 다음과 같다. 먼저 synthetic 데이터 집합을 위하여 170,000개의 노드(node)와 220,000개의 에지(edge)로 구성된 임의의 도로 네트워크를 생성하였으며, 생성된 네트워크를 기반으로 1년간의 트래픽(traffic) 데이터를 생성하였다. 실험을 위해 일년 중 1~361일의 트래픽 데이터는, 훈련 데이터(training data)로 사용하였으며, 남은 4 일간의 트래픽 데이터는 테스트 데이터(test data)로 사용하였다. 훈련 데이터란, 분류화 기법에서 사용되는 속도 등급을 구하기 위한 기록 데이터이며, 테스트 데이터는 성능평가에서 사용되는 질의 데이터를 의미한다.  $x(t)$ 는 현재 시간으로서, 그 범위는 오전 8시(morning)부터 오후 2시(noon)까지 이다. 또한 질의시 차량의 출발 시간은 현재 시간  $x(t)$ 로부터 30분, 60분의 두 가지 시차( $l$ )를 적용하였으며, 이는

각각 Kwon et al[2]과 Rice at al[9]에서 사용된 것이다. 관련 분야의 연구에서는 시차의 감소에 따라 예측 정확도가 향상되는 것을 고려하여, 사용된 시차보다 작은 시차값에 대해서는 높은 예측 정확도를 나타냄을 보장하였다. 이에 따라 본 논문에서는 기존 관련 연구에서 쓰인 최대 시차값을 이용한다. 만약  $x(t) =$  오전 8시이고 시차  $l = 30$ 분 일 경우, 질의 경로에서 차량의 출발 시간은 각각 8:30이다. 또한 1시간 단위로 시간을 나누어 출발 시간 그룹을 설정하였다. 예를 들어 오전 10시~ 오전 11시는 출발 시간 그룹 1이 된다. real 데이터 집합은 부산대학교의 궤적 데이터 생성기를 사용하였으며, 인터페이스는 <그림 5>와 같다. 부산대학교에서는 GPS 센서가 부착된 차량을 운행하여, 부산시의 도로 구간별 차량 주행 속도 및 시간 등 실제 교통 상황 정보를 수집하였다. 수집된 데이터는 시간대별, 요일별로 분류되었다. 분류된 데이터를 기반으로 실제 차량 운행 정보와 매우 유사한 이동 객체 궤적을 생성한다.[11] 본 논문에서는 해당 생성기를 사용하여 3000개의 궤적을 생성하였다. 실험을 위해, 3일간의 트래픽 데이터를 훈련 데이터로 사용하였으며, 2일간의 트래픽 데이터는 테스트 데이터로 사용하였다. 현재 시간과 출발 시간에는 30분과 60분의 시차( $l$ )을 고려하였다. 또한, 출발 시간 그룹은 1시간 단위로 분류되었다.

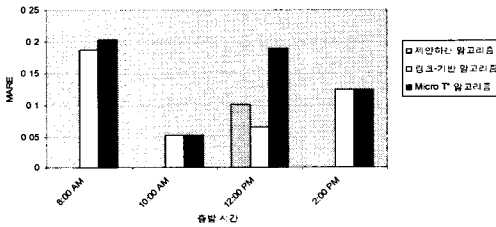
한편 성능평가 대상으로는 Chen at al[1]에서 제안한 링크-기반(link-based) 알고리즘과, Micro T\* [2] 알고리즘을 사용하였다.



<그림 5> 부산대학교 궤적 데이터 생성기의 사용자 인터페이스

4.2 Synthetic데이터 집합을 통한 성능 평가

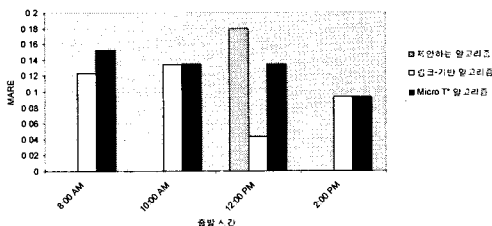
<그림 6>과 <그림 7> 및 <표 4>와 <표 5>는 현재 시간으로부터 각각 30분과 60분의 시차를 고려하였을 때, 각 주행 시간 예측 알고리즘들의 대한 성능 결과를 나타낸다. 오전 8시부터 오후 2시까지, 제안하는 알고리즘은 링크-기반 알고리즘과 Micro T\* 알고리즘에 비해 우수한 성능을 나타낸다. 제안한 알고리즘은 현재 시간과의 시차가 30분일 경우 오전과 러시아워 동안 만족스러운 성능을 보여주며, 시차가 60분일 경우에도 다른 두 예측 알고리즘들에 비해 좋은 성능을 보인다. 한편 링크-기반 알고리즘의 경우, Micro T\* 알고리즘에 비해 약간 향상된 성능을 보인다.



<그림 6> MARE (시차 =30분, Synthetic 데이터)

<표 4> MARE (시차 =30분, Synthetic 데이터)

Time	NBC	Link Based	Linear Regression
8:00 AM	0	0.18693	0.20281
10:00 AM	0	0.05315	0.05315
12:00 PM	0.10143	0.0658	0.18986
2:00 PM	0	0.125	0.125

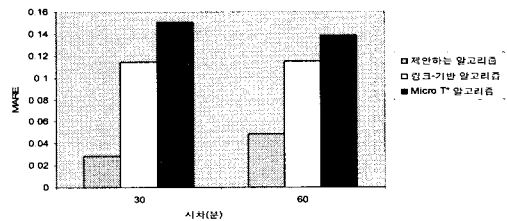


<그림 7> MARE (시차 =60분, Synthetic 데이터)

<표 5> MARE (시차 =60분, Synthetic 데이터)

Time	NBC	Link Based	Linear Regression
8:00 AM	0	0.12268	0.1528
10:00 AM	0	0.13413	0.13474
12:00 PM	0.17989	0.04318	0.13486
2:00 PM	0	0.09286	0.09286

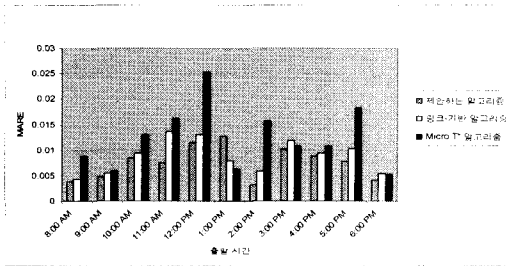
<그림 8>은 각 주행 시간 예측 알고리즘의 평균 MARE를 나타낸다. 시차가 30분일 경우, 시차가 60분일 때보다 MARE가 42%나 감소하였다. 현재 시간으로부터 시차가 30분일 경우, 링크-기반 알고리즘과 Micro T\* 알고리즘에 비해 각각 75%와 81% MARE 감소를 나타낸다. 또한 현재 시간으로부터 시차가 60분일 경우에도 유사한 성능 평가 결과를 보여주며, 링크-기반 알고리즘과 Micro T\* 알고리즘에 비해 각각 58%와 65%의 MARE 감소를 나타낸다.



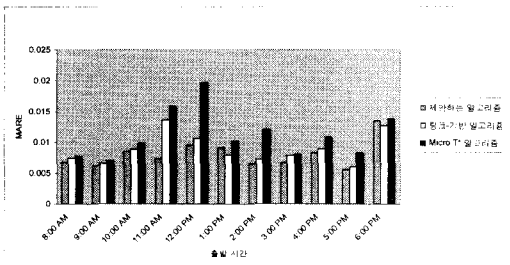
<그림 8> 평균 MARE (8AM~2PM)

4.3 Real 데이터 집합을 통한 성능 평가

현재 시간은 오전 8시~ 오후 6시 사이의 범위를 지니며, 이를 1시간 단위로 증가시켜가며 MARE를 측정하였다. <그림 9>와 <그림 10>은 각각 시차를 30분과 60분으로 하였을 때, 세가지 예측 알고리즘들에 대한 성능 평가 결과를 나타낸다.

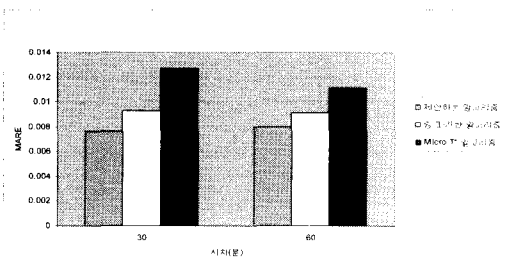


<그림 9> MARE (차차 =30분, Real 데이터)



<그림 10> MARE (차차 =60분, Real 데이터)

제안하는 알고리즘이 링크기반 알고리즘과 Micro T\* 알고리즘에 비해 전체적으로 우수한 성능을 보인다. 제안하는 알고리즘의 경우, 평균 구간 속도가 감소하면 주행 시간 예측에 대한 정확도가 향상된다. 따라서 제안하는 알고리즘은, 평균 구간 속도가 가장 낮은, 러시아워의 시간대에서 가장 우수한 성능 결과를 나타낸다.



<그림 11> 평균 MARE (8AM~2PM)

<그림 11>은 세가지 예측 알고리즘의 평균 MARE를 나타낸다. 시차가 30분 일 때, 제안하는 알고리즘, 링크기반 알고리즘, Micro T\* 알고리즘의 MARE는 각각 0.00764, 0.00927, 0.01269

이다. 따라서 제안하는 알고리즘은 링크기반 알고리즘에 비해 17%, Micro T\* 알고리즘에 비해 39% 성능이 향상되었다. 시차가 60분 일 경우에도 유사한 성능을 나타내며, 제안하는 알고리즘의 MARE는 링크기반 알고리즘에 비해 16%, Micro T\* 알고리즘에 비해 31% 감소되었다. 따라서 제안하는 알고리즘이 다른 두 주행 시간 예측 알고리즘에 비해 성능이 향상되었다.

### 5. 결론 및 향후 연구

본 논문에서는 도로 네트워크 상에서 사용자 지정 경로를 위한 효율적이고 확장 가능한 주행 시간 예측 기법을 제안하였다. 제안하는 방법은 사용자 지정 경로에 대해 주행 시간을 예측할 수 있을 뿐만 아니라, Naïve Bayesian 분류화 기법을 사용하여, 도로 구간별 평균 구간 속도를 측정함으로써 보다 정확한 주행 시간을 예측할 수 있다. 제안한 기법을 syntactic 및 real 데이터 집합을 사용하여, 기존에 제안된 주행 시간 예측 알고리즘인 링크기반(link-based) 알고리즘 및 Micro T\* 알고리즘과 성능평가를 수행하였다. 성능 평가 결과, 제안하는 알고리즘이 syntactic 데이터 집합의 경우 42~81%, real 데이터 집합의 경우 17~31% MARE측면에서 성능이 우수함을 보였다. 향후 연구로는, 데이터 분류시 일별 시간뿐만 아니라 주별 요일까지 고려하여 주행 시간을 예측하는 것이다.

### Acknowledgments

부산대학교의 궤적 데이터 생성기를 제공하여 주시고, 아울러 사용하도록 허락해주신 부산대학교의 이기준 교수님께 감사드립니다.

### References

1. Chen, M., Chien, S., "Dynamic freeway travel time prediction using probe vehicle data: Link-based vs. Path-based..", Journal

of Transportation Research Record, TRB Paper No. 01-2887, Washington, D.C., 2001.

2. Kwon, J., Petty, K., "A travel time prediction algorithm scalable to freeway networks with many nodes with arbitrary travel routes.," Transportation Research Board 84th Annual Meeting, Washington, D.C., 2005.
3. Chun-Hsin, W., Chia-Chen, W., Da-Chun, S., Ming-Hua, C., Jan-Ming, H., "Travel Time Prediction with Support Vector Regression.," In: Proceedings of IEEE Intelligent Transportation Systems Conference, 2003.
4. Park, D., Rilett, L., "Forecasting multiple-period freeway link travel times using modular neural networks.," Journal of Transportation Research Record., vol. 1617, 1998, pp.163-170.
5. Park, D., Rilett, L., "Spectral basis neural networks for real-time travel time forecasting.," Journal of Transport Engineering, vol. 125(6), 1999, pp.515-523.
6. Kwon, J., Coifman, B., Bickel, P. J., "Day-to-day travel time trends and travel time prediction from loop detector data.," Journal of Transportation Research Record, No. 1717, TRB, National Research Council, Washington, D.C., 2000, pp. 120-129.
7. Zhang, X., Rice, J., "Short-Term Travel Time Prediction.," Transportation Research Part C, vol. 11, 2003, pp. 187-210.
8. Van der Voort, M., Dougherty, M., Watson, S., "Combining KOHONEN maps with ARIMA time series models to forecast traffic flow.," Transportation Research Part C, vol. 4, 1996, pp. 307-318.
9. Rice, J., Van Zwet, E., "A simple and effective method for predicting travel times on freeways.," In: IEEE Trans.

Intelligent Transport Systems, vol. 5, no. 3, 2004, pp. 200-207.

10. Han, J., Kamber, M., "Data Mining: Concepts and techniques.," Second Edition, Morgan Kaufmann publishers, 2006.
11. 황정래, 강혜영, 이기준, "시공간 유사성을 이용한 도로 네트워크 상의 유사한 궤적 검색," 한국정보처리학회 논문지, vol. 13-D, no. 03, 2006, pp. 0337-0346

**엄정호**

2004년 전북대학교 컴퓨터 공학과(공학사)  
 2006년 전북대학교 컴퓨터 공학과(공학석사)  
 2006년~현재 전북대학교 컴퓨터 공학과 박사과정  
 관심분야: 공간 데이터베이스, 공간 색인 구조, GIS  
 E-mail : jhum@dblab.chonbuk.ac.kr

**니하드 카림 초우더리**

2004년 Chittagong University of Engineering & Technology (공학사)  
 2006년~현재 전북대학교 대학원 컴퓨터공학과 석사과정  
 관심분야 : 데이터 마이닝, 공간 데이터베이스, 질의 처리 알고리즘  
 E-mail : nihad@dblab.chonbuk.ac.kr

**이현조**

2006년 전북대학교 컴퓨터공학과 (공학사)  
 2008년 전북대학교 대학원 컴퓨터공학과 (공학석사)  
 2008년~현재 전북대학교 대학원 컴퓨터공학과 박사과정  
 관심분야 : 데이터 마이닝, 공간 데이터베이스, 고차원 색인 구조  
 E-mail : hjlee@dblab.chonbuk.ac.kr

**장재우**

1984년 서울대학교 전자계산기공학과 (공학사)  
 1986년 한국과학기술원 전산학과 (공학석사)  
 1991년 한국과학기술원 전산학과 (공학박사)  
 1996년~1997년 Univ. of Minnesota, Visiting Scholar  
 2003년~2004년 Penn State Univ., Visiting Scholar.  
 1991년~현재 전북대학교 컴퓨터공학과 교수  
 관심분야 : 공간 네트워크 데이터베이스, 상황인식, 하부저장구조  
 E-mail : jwchang@chonbuk.ac.kr

**김연중**

1995년 전북대학교 컴퓨터공학과 학사

1997년 전북대학교 대학원 컴퓨터공학과 (공학석사)

1999년 전북대학교 대학원 컴퓨터공학과 박사과정  
수료

1999년~현재 링크웨어(주) 근무

관심분야 : 정보 검색, 지리 정보 검색, 공간 데이터  
베이스

E-mail : yjkim@dbl原因.chonbuk.ac.kr