KIPS Trans. Softw. and Data Eng.
Vol.10, No.2 pp.57~64
ISSN: 2287-5905 (Print), ISSN: 2734-0503 (Online)

https://doi.org/10.3745/KTSDE.2021.10.2.57

A Modeling of Realtime Fuel Comsumption Prediction Using OBDII Data

Hee-Eun Yang† · Do-Hyun Kim†† · Hoseop Choe†††

ABSTRACT

This study presents a method for realtime fuel consumption prediction using real data collected from OBDII. With the advent of the era of self-driving cars, electronic control units(ECU) are getting more complex, and various studies are being attempted to extract and analyze more accurate data from vehicles. But since ECU is getting more complex, it is getting harder to get the data from ECU. To solve this problem, the firmware was developed for acquiring accurate vehicle data in this study, which extracted 53,580 actual driving data sets from vehicles from January to February 2019. Using these data, the ensemble stacking technique was used to increase the accuracy of the realtime fuel consumption prediction model. In this study, Ridge, Lasso, XGBoost, and LightGBM were used as base models, and Ridge was used for meta model, and the predicted performance was MAE 0.011, RMSE 0.017.

Keywords : Fuel Consumption, Prediction Model, Stacking Ensemble, Regression Model, OBDII

OBDII 데이터 기반의 실시간 연료 소비량 예측 모델 연구

양 희 은† · 김 도 현†† · 최 호 섭†††

요 약

자율주행차 시대가 도래하면서 ECU (Electronic Control Unit)는 점차 고도화되고 있고, 이에 따라 차량에서 정확한 데이터를 추출하고 분석하려는 연구가 다양하게 시도되어 왔다. 그러나 ECU는 차량 제조사별로 상이한 프로토콜을 가지고 있어 상용 단말기로는 정확한 데이터 추출과 분석이 어렵다. 본 연구에서는 정확한 차량 데이터를 추출하기 위하여 전용 펌웨어를 개발하여 차량의 2019년 1월부터 2월의 실제 주행데이터 53,580건의 데이터를 추출하였으며, 20회가 넘는 실제 도로 주행을 통해서 데이터의 정확도를 검증하였다. 이러한 데이터를 바탕으로 실시간 연료 소비량 예측 모델의 정확도를 높이기 위하여 스태킹 앙상블 기법을 이용하였다. 본 연구에서는 베이스 모델로 Ridge, Lasso, XGBoost, LightGBM이 사용되고 메타 모델은 Ridge가 사용되었으며, 예측 성능은 MAE 0.011, RMSE 0.017로 최적의 결과를 보였다.

키워드 : 연료 소비량, 예측모델, 스태킹 앙상블, 회귀모델, OBDII

1. 서 론

자율주행차 시대가 도래하면서 차량의 전자제어장치(Electronic Control Unit; ECU)는 점점 복잡화·고도화되고 있어, 차량에서 정확한 데이터를 획득하고 분석하여 활용하는 것에 대한 중요성이 높아지고 있다[1]. 그러나 차량 제조사와 모델별, ECU 제조사가 각각 상이한 프로토콜을 갖고 있어 정확한 데이터를 획득하기 위해선 차량별 단말기 및 펌웨어를 개발할 필요성이 있다.

만약 상용 단말기를 이용할 경우 데이터의 정확성에 대한 판별이 어려워 데이터의 신뢰성 문제가 있을 수 있다. 또한 시중에 판매되는 차량정보수집장치(On Board Diagnosis II; OBDII) 단말기는 별도의 주행 데이터를 저장할 수 있는 장치가 없어 스마트폰이 연결되어 있을 때만 데이터를 그래프나 게이지 형태로 보여주고 Raw 데이터를 따로 추출하기가 사실상 불가능하다[2]. 또한 실제 차량 주행 중 스마트폰이나 컴퓨터를 연결하기란 번거로운 작업이기 때문에, 정확한 데이터 획득과 데이터 손실을 방지하기 위해는 Raw 데이터 확인이 가능하고 데이터를 저장할 수 있는 OBDII 단말기가 필요하다.

이에 본 연구에서는 MAF (Massive Air Flow)를 포함한 OBDII의 표준항목 데이터들을 수집하여 실제 운행 데이터를 이용한 연료 소비량 예측 모델을 제안한다. 이를 위해서 차량

※ 이 논문은 2020년 한국정보처리학회 춘계학술발표대회의 우수논문으로 "OBDII 데이터 기반의 회귀 분석을 통한 실시간 연료 소비량 예측"의 제목으로 발표된 논문을 확장한 것임.
† 정 회 원 : 단국대학교 EduAI센터 연구원
†† 정 회 원 : ㈜한국축산데이터 연구원
††† 비 회 원 : 단국대학교 EduAI센터 센터장/교수
Manuscript Received : July 8, 2020
First Revision : August 21, 2020
Accepted : August 25, 2020
* Corresponding Author : Hoseop Choe(hschoe@dankook.ac.kr)

※ This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (http://creativecommons.org/ licenses/by-nc/3.0/) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

의 데이터를 추출할 수 있는 OBDII 단말기와 펌웨어를 직접 제작하였으며, 가공한 데이터를 기반으로 순간 연료 분사량 변수 및 엔진회전수 등 차량에서 획득할 수 있는 실시간(초당) 변수를 통해 높은 예측 정확도를 달성할 수 있는 방법을 제안하고자 한다. 또한 제안하는 예측 모델의 성능을 관련 연구 결과와 비교하여 연료 소비량 예측의 효과성과 연료 소비량 절감을 기대할 수 있을 것으로 판단하였다.

2. 관련 연구

2.1 OBDII 규격

1970년대와 1980년대 초부터 자동차 제조사들은 전기/전자적으로 엔진을 제어하고 엔진 문제를 진단하기 시작하였다[3]. OBD 규격은 환경 규제에 부합하기 위한 목적이 컸으며, 진단 시스템이 점점 복잡해짐에 따라서 OBDII가 1990년대에 도입되기 시작하였다.

LA 지역의 스모그 문제를 해결하기 위해서 1966년식 모델의 차량부터 OBDII의 탑재가 의무화되었으며, 1968년부터 미국 전역의 차량들의 OBDII 탑재가 의무화되었다[4].

Fig. 1은 OBDII에 대한 이미지로써 1번~16번까지의 핀에 대한 단자의 물리적인 구성 및 각 핀이 지원하는 프로토콜 및 역할에 대한 설명이다.

2.2 OBDII Protocols + Data

본 연구에서 사용되는 OBDII 데이터는 디지털운행기록장치(Digital Taco Graph; DTG)에서 수집하는 데이터를 전부 포함한(단 상태코드 제외) 총 20개의 항목이 있다. 초 단위로 수집되는 이 데이터 항목들은 측정시각, 엔진회전수, 차량 속도, 레버의 위치, 가속페달량, 순간 연료소모량, 엔진경고등, GPS 위/경도, 기어 단 수, 페달 및 사이드 브레이크 유무, X/Y/Z축 기울기, 에어컨 유무, 순간 토크, 핸들 조향각, 안전벨트 착용 유무, 방향지시등 상태 등이다.

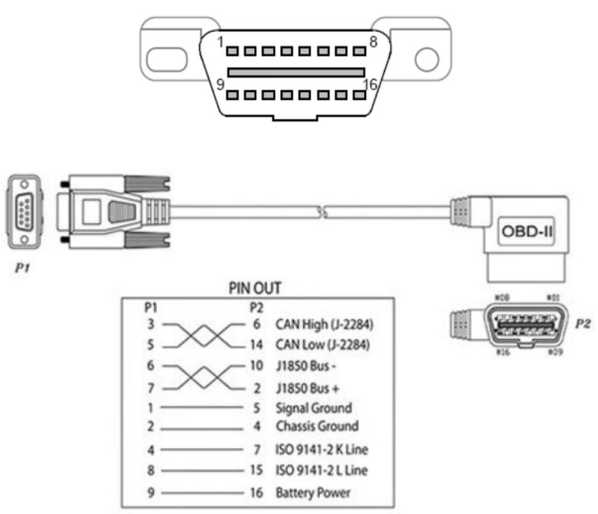

Fig. 1. Shape and PIN OUT Protocol of OBDII

2.3 연비 예측

OBDII에서 연비 정보를 제공하지 않아 정확한 연비를 예측하기 위해서 MAF 센서의 데이터를 통한 연비 계산부터 ANN을 통한 연비 예측까지 다양한 시도들이 연구되고 있다[1,6,7]. 특히 MAF 센서 데이터를 통한 연료 소모량을 계산하기 위해서는 공기/연료 비율(Air/fuel ratio)을 계산해야 하며, 해당 비율은 각 엔진/제조사의 ECU 세팅에 따라서 차이가 있을 수 있어 정확도가 낮다[8].

또한 기존의 관련 연구에서는 너무 적은 양의 데이터와 실제 도로 주행 데이터가 아닌 차량의 마력과 토크를 측정하는 장비인 다이노(Dyno) 등에서의 데이터 세트를 이용함에 따라 실제 주행 환경에서의 특성을 반영하기 어렵다. 본 연구에서는 이러한 부분들에 대해서 보완을 하여 더 정확도가 높은 모델을 제안하고자 한다.

2.4 스태킹 모델

본 연구에서는 모델의 성능을 높이기 위하여 앙상블 기법 중에 스태킹을 사용하였다. 여기서 앙상블이란 여러 개의 예측모형들을 결합하여 하나의 최종 예측모형을 만드는 방법이다. 스태킹 앙상블(Stacking ensemble)이라고도 불리는 이 방법은 개별 모델들에서 추출한 예측값들을 다시 학습 데이터 세트로 쌓는 모습에서 스태킹이라고 정의한다[10].

이 모델의 기본 구조는 학습 데이터와 테스트 데이터 세트를 사용하며 여러 개의 베이스 모델들을 이용하여 학습한다. 학습이 완료된 후, 예측을 수행하여 나온 개별 모델의 예측값을 다시 학습 데이터 세트로 사용하여 학습한다. 마지막으로 최종 예측 모델을 선정하여 학습하고, 최종적으로 예측값에 대한 평가를 한다.

3. 실시간 연료 소비량 예측 모델

3.1 데이터 수집을 위한 하드웨어 제작

본 연구에서는 기존에 구입할 수 있는 OBDII 젠더(단말기)들의 문제점을 해결하고자 별도의 하드웨어 단말기와 이에 맞춘 펌웨어를 개발하여 연구를 진행하였다.

1) 차량 ECU 제조사별 상이한 프로토콜

Fig. 2는 기존 연구에서 OBDII 단말기를 활용할 때 Task별 소요시간에 대한 설명이다. 상기 연구에서는 시중에 판매되고 있는 ELM327을 기반으로 연구를 진행하였으며, Task별 작업 시간을 보면 실제 사고 예측에 소요되는 시간보다 지원되는 프로토콜을 찾고 ELM327을 설정하는 데 많은 시간이 소요되었다[11].

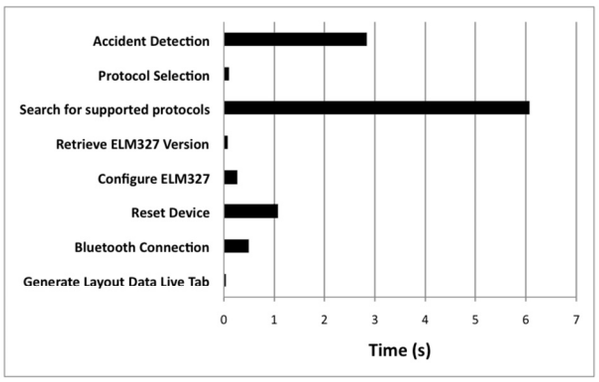

Fig. 2. Duration Per Task from Precedent Study

또한 차량 제조사와 모델별, 정확히는 ECU 제조사별로 상이한 프로토콜을 가지고 있기 때문에[12], 다양한 프로토콜을 지원하기 위해서는 자체의 단말기 및 펌웨어 개발 역량이 필수적이다.

2) 데이터의 신뢰성 문제

만약 상용 단말기를 이용할 경우 데이터의 신뢰성 문제가 있을 수 있다. ELM327의 경우 지원하는 차량과 프로토콜이 매우 한정적이며, 그 이외 차량들의 경우 데이터의 신뢰성을 보장하기 어렵다[9,13].

3.2 단말기 제작

위와 같은 이유로 본 연구에서는 직접 OBDII 하드웨어 및 펌웨어를 제작하여 연구하였으며, OBDII 하드웨어는 ARM 기반의 32비트 CPU와 데이터 저장을 위한 4GB의 롬 메모리가 있으며, 실시간 데이터를 기록 및 저장할 수 있도록 3G/LTE 모뎀과 블루투스 칩을 탑재하였다.

Fig. 3은 직접 제작한 OBDII 단말기 디스플레이 및 실제 포트에 연결되어 데이터를 수집/처리하는 하드웨어이며 SAEJ1850, ISO15765 및 K-Line(ISO9141-2) 프로토콜을 지원하여 국산 차량과 대부분의 수입 차량에서 사용할 수 있다. 단말기의 상세 스펙은 Table 1과 같다.

Fig. 3. OBDII Device Display and PCB Hardware

Table 1. OBDII Device Technical Specs

	Item	Content
1	Voltage	DC 8V~32V
2	Operating Temp.	-20℃~70℃
3	CPU	ARM7 120MHz 32bit
4	Internal Storage	128KB RAM / 1G ROM
5	CAN Port	CAN 2.0B 2Port
6	Communication	UART 1Port, USB 1Port
7	Electric Current	max 500mA
8	Data Columns	14 Items
9	Physical Size	75×52×22mm
10	Vibration / Impulse	66hz / 500g@10cm

3.3 펌웨어 개발

본 연구에서는 자체 제작한 디바이스를 제어하고 실행하기 위한 펌웨어를 제작하였으며, ECU로부터 CAN 통신을 통해서 차량의 데이터 수집, 수집된 데이터를 내부의 ROM에 저장 및 관리, 실시간 데이터를 모뎀 및 블루투스를 통해서 전송 등의 기본 기능을 수행한다. 또한 Fig. 4와 같이 실제 주행 데이터와 최대한 일치하도록 다양한 오차보정 알고리즘이 적용되어 있다. Table 2는 오차보정 알고리즘에 대한 추가 설명으로 Vehicle speed와 같은 CAN DATA, 그리고 외부 온도 및 타이어 스펙에 따라 오차보정 알고리즘을 통해 차속과 주행거리를 산출하고 있음을 나타내고 있다.

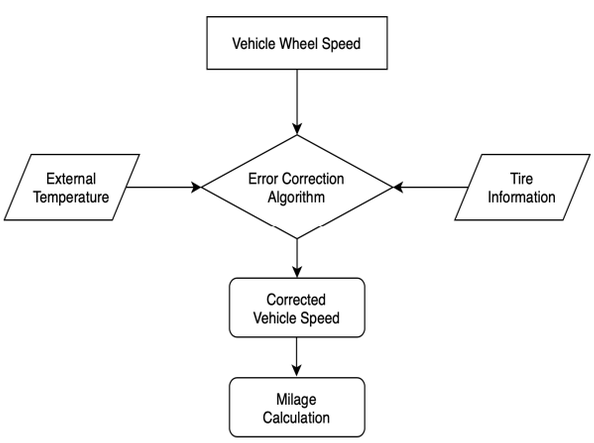

Fig. 4. Flow Chart for Error Correction Algorithm

Table 2. Error Correction Algorithm Description

Vehicle ECU	Firmware Input	Vehicle Speed
vehicle speed CAN DATA	ext. temp.: 1~5 tire info: tire spec	error corrected result

3.4 리버스 엔지니어링

OBDII 단말기로부터 얻은 데이터를 모델링에 사용하기 위해서 RAW 바이너리 데이터를 분석하여 각 데이터가 의미하는 수치를 파악하는 리버스 엔지니어링 과정이 필요하다. 제작한 하드웨어에서 추출한 데이터를 csv파일로 기록하였을 때 Fig. 5와 같이 각 데이터 인덱스의 바이너리 값을 얻을 수 있다.

OBDII에서 얻은 바이너리 데이터는 차량별 프로토콜이 상이하므로 리버스 엔지니어링 작업을 통해 각 항목이 의미하는 변수를 역으로 알아내는 과정이 필요하다. 그래서 각 인덱스가 의미하는 데이터는 차량을 직접 주행 및 차량 기능 동작을 통해 바이너리 값이 의미하는 데이터 세트를 구축하였다. Table 3은 Fig. 5에 대한 바이너리 데이터를 변수에 맞춰 변환한 값의 일부이다.

또한 ECU를 통해서 얻기 어려운 GPS값, 각 축의 기울기를 획득하기 위하여 별도의 GPS 수신기와 자이로센서를 추가하였다.

ID	A	B	C	D	E	F	G	H	a	b	c	d	e	f	g	h
113	FF	20	10	00	FF	00	00	98	255	32	16	0	255	0	0	152
113	FF	20	10	00	FF	00	00	5C	255	32	16	0	255	0	0	92
113	FF	20	10	00	FF	00	00	10	255	32	16	0	255	0	0	16
113	FF	20	10	00	FF	00	00	D4	255	32	16	0	255	0	0	212
113	FF	20	10	00	FF	00	00	98	255	32	16	0	255	0	0	152
113	FF	20	10	00	FF	00	00	5C	255	32	16	0	255	0	0	92
113	FF	20	10	00	FF	00	00	10	255	32	16	0	255	0	0	16
113	FF	20	10	00	FF	00	00	D4	255	32	16	0	255	0	0	212
113	FF	20	10	00	FF	00	00	98	255	32	16	0	255	0	0	152
113	FF	20	10	00	FF	00	00	5C	255	32	16	0	255	0	0	92
113	FF	20	10	00	FF	00	00	10	255	32	16	0	255	0	0	16
113	FF	20	10	00	FF	00	00	D4	255	32	16	0	255	0	0	212
113	FF	20	10	00	FF	00	00	98	255	32	16	0	255	0	0	152
113	FF	20	10	00	FF	00	00	5C	255	32	16	0	255	0	0	92
113	FF	20	10	00	FF	00	00	10	255	32	16	0	255	0	0	16
113	FF	20	10	00	FF	00	00	D4	255	32	16	0	255	0	0	212

Fig. 5. Part of Binary Data from Data Index (Captured from Excel)

Table 3. Part of Binary Data Transferred to Features

1. TIMS	2. NE (RPM)	3. Vss (Km/h)	4. TRIP (km)	5. TPS (%)	6. FCO (uL)	7. Trip FCO (L)
20190111093800	1692	14	0.2	24	1473	0.043
20190111093801	1746	18	0.2	24	1460	0.044
20190111093802	1793	22	0.2	23	1518	0.046
20190111093803	1450	26	0.2	19	1050	0.047
20190111093804	1467	28	0.2	19	661	0.048
20190111093805	1619	29	0.2	26	1358	0.049
20190111093806	1798	32	0.2	31	2060	0.051

4. 평가실험

4.1 실험데이터

본 연구에서는 자체 제작한 펌웨어 단말기를 이용하여 아반떼AD 모델의 2019년 1월부터 2019년 2월의 주행 데이터를 수집하였다. 전체 데이터 53,580건(초당 Data Set)에서 80%에 해당하는 데이터를 학습 데이터, 20%에 해당하는 데이터를 테스트 데이터(Validation/Test Data)로 사용하였다. 데이터 구성은 Table 4와 같다.

OBDII에서 추출한 데이터 컬럼과 타입은 Table 5와 같다. 데이터 세트의 총 컬럼은 20개로 구성되어 있으며, 데이터 타입은 연속형, 논리형, 범주형으로 구성되어 있다. 논리형 데이터 컬럼으로는 0과 1로 구성된 페달 및 사이드 브레이크, 에어컨 on/off, 안전벨트 착용 여부, 엔진경고 컬럼이 있다. 범주형은 P, R, N, D로 구성된 레버 위치 컬럼과 L, R, ER 로 구성된 방향지시등, 기어 단 컬럼이 있다. 나머지 데이터 컬럼은 연속형 타입으로 구성되어 있다. 측정시각의 경우, YYYYMMSShhmmss 형태로 시간을 나타내고 있는 연속형 변수이다.

Table 4. Data Classification for Modeling

	Start Date	End Date	Count
Training data	2019.01.09	2019.01.29	42,864
Validation/Test data	2019.01.30	2019.02.12	10,716

Table 5. Description of Each Items in Feature Set

Column	Description	Type	Column	Description	Type
TIMS (iso8601)	Timestamp	Num.	Break_foot (0, 1)	Foot Break	Bool.
RPM (rpm)	Engine Revolution	Num.	Break_side (0, 1)	Side Break	Bool.
Vss (km/h)	Vehicle Speed	Num.	Slop_x (°)	X Axis Inclination	Num.
Lever (P,R,N,D)	Lever	Cat.	Slop_y (°)	Y Axis Inclination	Num.
TPS (%)	Acc. Pedal	Num.	Slop_z (°)	Z Axis Inclination	Num.
FCO (uL)	Fuel Consumption	Num.	ACON (0, 1)	Air Conditioner	Bool.
Mil (0, 1)	Engine Checklight	Bool.	Torque (N.m)	Torque	Num.
Latitude (°)	GPS Lat	Num.	Handle (°)	Handle Angle	Num.
Longtitude (°)	GPS Lng	Num.	Belt (0, 1)	Seatbelt Active	Bool.
Gear (1,2,3,4,5, 6,14)	Gear Pos.	Cat.	Light (L,R,ER)	Turn Indicator	Cat.

4.2 실험데이터 가공

학습에 사용할 변수들을 확인한 결과, 변수 간의 단위가 달라서 정규분포 형태를 적용하는 데 문제가 있었다. 예를 들어서 Fig. 6과 같이 'Gear(기어 단)' 데이터는 값이 1~14인 반면, 'Break' 데이터는 0~1 사이의 값을 포함하고 있다. 변수들의 숫자 편차가 커서 이러한 편차를 줄이기 위해 Scaling 과정을 통하여 데이터를 가공하였으며, Scaling 방법 중 MinMax Scaler를 이용하여 모든 변수의 값이 0~1 사이가 되도록 데이터를 재조정하였다[14].

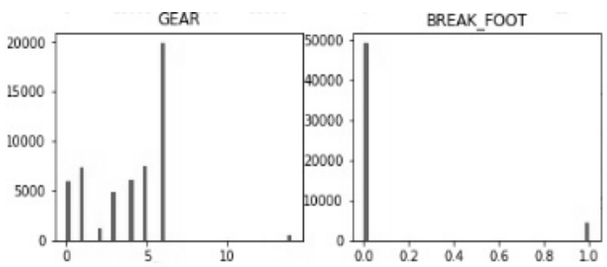

Fig. 6. Data Histogram of Gear and Foot Break

4.3 예측 모델 비교

본 연구에서는 스태킹 기법을 사용하기에 앞서 Linear Regression, Ridge, Lasso, XGBRegressor, LightGBM 예측모델의 성능을 비교하였다. 이 과정에서 성능을 더 잘 평가하기 위해 cross-validation (k=5) 을 사용하여 모델별로 어떠한 매개변수가 선택되는지를 확인하였다.

또한 Gridsearch CV를 활용하여 최적의 매개변수를 찾고, 이를 이용하여 모델의 성능을 향상시키고자 했다. 예시로 Lasso 모델에서 GridSearchCV의 변수인 grid_ridge의 cv_result를 통한 alpha값의 변화에 따라 평균값의 변화를 파악한 결과, param 리스트는 0.001, 0.005, 0.008, 0.05, 0.03, 0.1, 0.5, 1, 5를 사용하였으며 그 중 최적의 파라미터 값은 5로 확인되었다.

최적의 파라미터를 도입하였을 때, 각 모델의 상관계수는 Table 6과 같다. Linear Regression과 Ridge에서는 TRIP(L), TRIP(KM), Vss, BELT, SLOP_Y, HANDLE, FCO(uL) 변수가 모델에 많은 영향을 끼치며, Lasso에서는 TRIP(KM), BELT, Vss, ACON, GEAR가 중요 변수로 나타났다. XGBoost에서는 TRIP(L), TRIP(KM), SLOP_Z, LATITUDE, SLOP_X 순서로 상관계수가 높았고, LightGBM에서는 TRIP(KM), LATITUDE, SLOP_Y 순서로 상관계수가 높은 결과를 보였다.

성능 측정 지표는 평균절대오차인 MAE(Mean Absolute Error)와 평균제곱근오차인 RMSE(Root Mean Square Error)를 이용하였다. MAE는 회귀모델의 오류 지표 중 하나로 모델의 예측값과 실제값의 차이를 모두 더한다는 개념으로써 절대값을 취하기 때문에 가장 직관적으로 알 수 있는 지표이며, MSE보다는 이상치로부터 영향을 크게 받지 않는다. 이에 본 연구에서는 변동성이 적은 편에 속하는 MAE를 성능 지표로 살펴보는 것이 적절하다고 판단하였다. RMSE는 오차의 제곱에 대하여 평균을 취하고 이를 제곱근한 것을 나타낸다. 오류의 제곱을 구하는 MSE에 root를 씌운 값으로 실제 오류 평균보다 커지는 MSE의 단점을 보완한 지표로 MSE, RMSE는 값이 작을수록 추정의 정확성이 높아진다.

Table 6. Feature Importance

	Linear regression	Ridge	Lasso	XGBoost	Light GBM
1	TRIP(L)	TRIP(L)	TRIP(KM)	TRIP(L)	TRIP(KM)
2	TRIP(KM)	TRIP(KM)	BELT	TRIP(KM)	LATITUDE
3	Vss	Vss	Vss	SLOP_Z	SLOP_Y
4	BELT	BELT	ACON	LATITUDE	BREAK_FOOT
5	SLOP_Y	SLOP_Y	GEAR	SLOP_X	LONGTITUDE
6	HANDLE	HANDLE	TIMS	LONGTITUDE	TRIP(L)
7	FCO(uL)	FCO(uL)	SLOP_Y	SLOP_Y	Vss
8	GEAR	GEAR	BREAK_FOOT	RPM	BELT
9	LATITUDE	LATITUDE	LATITUDE	FCO(uL)	ACON
10	TIMS	TIMS	TRIP(L)	Vss	HANDLE

Table 7과 같이 MAE 값을 측정한 결과, XGBoost의 성능이 제일 좋았으며 Lasso의 성능이 가장 저조하게 나온 것을 알 수 있다. RMSE 값을 측정하였을 때, XGBoost가 0.018 로 성능이 좋았으며, Lasso의 성능이 0.249 로 저조한 결과를 보였다.

Table 6과 같이 모델별 상관계수가 상이하게 나타났기 때문에, 본 연구에서는 모델들의 단점을 보완할 수 있도록 앙상블 학습 중 하나인 스태킹 모델을 사용하였다. 개별 모델이 예측한 데이터를 다시 학습 데이터로 사용하여 학습하는 스태킹 기법(k-fold=5, shuffle=false)에 사용할 개별 베이스 모델로 Ridge, Lasso, XGBoost, LGB, Linear Regression을 사용하였다.

그리하여 본 연구에서는 Fig. 7과 같이 먼저 train과 test 데이터를 5 fold로 나누었다. 즉 데이터를 5개의 fold(k-fold=5)로 나눈 후, 개별 모델별로 5 fold로 나누어진 데이터를 기반으로 학습을 진행하였다. 모델마다 검증 데이터를(X_test) 입력하여 예측 후, 예측 결과값을 저장하였다. 그리고 5개의 예측값들은 다시 메타 모델의 학습 데이터로 사용하였다. 이때,

Table 7. MAE and RMSE by Model

	Linear regression	Ridge	Lasso	XGBoost	Light GBM
MAE	0.121	0.121	0.125	0.012	0.024
RMSE	0.17	0.17	0.249	0.018	0.034

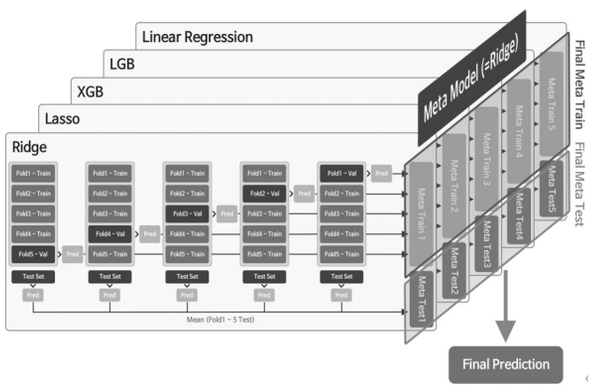

Fig. 7. Flow Chart of Stacking Ensemble

최종 메타 모델 선정은 회귀모델인 Ridge, Lasso, XGBoost, LGB, Linear Regression을 각각 메타 모델로 사용하였을 때 가장 좋은 성능을 보인 모델인 Ridge 모델을 선택하여 사용하였다.

4.4 최종 결과

본 연구에서는 2019년 1월부터 2019년 2월까지의 차량 주행 데이터를 이용해 연비 예측 모델의 효용성을 확인하였다. Table 8은 기존 연구에서 제시하는 예측 모델과 본 논문의 모델을 비교한다. 기존 연구의 경우, 연비 예측에 대한 성능지표 및 예측에 사용된 데이터가 본 연구와 상이하여 직접적인 성능 비교에는 한계가 있다. 따라서 본 연구에서 제시하는 스태킹 모델을 이용해 총 2가지 경우에 대해 시뮬레이션을 수행하였다. 기존 연구와 유사하게 변수를 적게 적용(feature = 4개)하였을 경우, MAE는 1.310, RMSE는 2.132의 결과를 보였으나, 모든 변수를 적용(feature = 20개)할 때 MAE는 0.011, RMSE는 0.017의 결과를 보였다.

Table 8에서 인용된 3가지 연구 결과들은 다음과 같다. Cho의 모델과 Wawrzyniec의 모델 모두 7개의 변수만을 적용하였으며, Cho의 모델[14]은 오류율이 약 7.68%의 성능을 나타내었고, Wawrzyniec의 모델[15]은 MAPE 13.3%, RMSE 1.42의 성능을 나타내었다. Lee의 모델[16]은 최대 정확도가 88%로 보고되었고, surface regression 모델을 적용하였으나, 2가지(RPM, TPS)의 변수만을 활용하여 다양한 환경변수를 고려하지 않았다.

반면, 본 연구에서의 연비 예측 모델의 성능은 변수 4개(시각, 엔진회전수, 차량속도, 가속페달량)를 적용하였을 때, MAE는 1.310, RMSE는 2.132가 나왔으며, 변수 20개를 모두 사용하였을 때는 MAE 0.011, RMSE 0.017의 결과를 보였다. 즉, 다양한 환경변수를 고려할 경우, 최소 변수를 적용했을 때보다 MAE는 약 18%, RMSE는 약 25% 정도 좋은 예측 결과를 보여주었다.

Table 8. MPG Prediction Results

	Accuracy	Model
Cho[14] (feature=7)	Error rate 7.68%	Multi-linear regression
Wawrzyniec[15] (feature=7)	MAPE 13.3% RMSE 1.42	ANN
Lee[16] (feature=2)	max 88%	surface regression
Proposed model (feature=4)	MAE 1.310 RMSE 2.132	Stacking ensemble
Proposed model (feature=20)	MAE 0.011 RMSE 0.017	

5. 결 론

본 연구는 OBDII의 표준항목을 이용해 가장 효과적인 연비 예측 모델을 개발하고 스태킹 기법을 이용하여 주행 변수와 연비 간 관계를 예측하였다. 기존에도 차량 데이터 관련 연구가 다양하게 시도되어 왔으나, 차량 제조사별 상이한 프로토콜을 가지고 있어 데이터 추출과 분석이 어려웠다. 본 연구에서는 연료 소비량의 정확한 예측을 위해 별도의 하드웨어를 개발하고, 20회가 넘는 실제 도로주행을 통해서 데이터의 정확도를 검증하였다.

기존 연비 예측 논문들은 현장에서 데이터의 수집에 많은 시간과 비용 문제로 상관관계가 높은 최소의 변수만을 이용하여 주행 연비를 예측하였다. 따라서 대부분 속도, 엔진회전수, 가속도, 순간 연료분사량을 이용하였다.

그러나 본 논문에서는 GridSearchCV를 통하여 개별 회귀모델의 최적의 매개 변수를 확인한 결과, 도로 경사도, GPS 위도/경도와 같은 일부 환경변수들도 연비 예측에 영향을 끼치는 것으로 확인되어 20개의 주행 변수를 이용하여 예측 모델을 생성하였다. 스태킹 앙상블 모델을 사용하였으며, 베이스 모델로는 Ridge, Lasso, XGBoost, LightGBM, 메타모델로는 Ridge 모델을 사용하여 MAE 0.011, RMSE 0.017 성능을 갖는 연비 예측 모델을 제안하였다. 현재 예측 모델의 성능이 최소 변수(featrue=4)를 적용했을 때보다 MAE는 약 18%, RMSE는 약 25% 정도 좋은 성능을 나타냈다.

제안한 모델을 기반으로 운전자에게 보다 정확한 연료 소비량을 제공함으로써, 기업의 비용 절감 및 환경보존 등의 효과도 기대할 수 있다. 향후 더 나아가 신경망 모델을 적용하여 예측 모형의 성능을 고도화하는 연구를 진행할 것이며, 다

양한 차종의 데이터를 OBDII 단말기로부터 획득하여 모델에 적용한다면 범용적인 모델을 활용한 실제 응용에 활용될 수 있을 것으로 기대된다.

References

[1] T. Lee, J. Jung, J. Kang, H. Choi, and J. Ko, "System for analyzing big data collected while driving a car," *The Institute of Electronics and Information Engineers*, pp.1367-1370, 2018.

[2] W. J. Lee and D. S. Ko, "Classification of the safe threats using clustering of vehicle OBD2 Data by road section type," *Korean Institute of Information Technology*, Vol.18, No.4, pp.1-8, 2020.

[3] G. Geraldo, "Differences between on board diagnostic systems (EOBD, OBD-II, OBD-BR1 and OBD-BR2)," SAE Technical Paper 2006-01-2671, 2006.

[4] D. Rimpas, A. Papadakis, and M. Samarakou, "OBD-II sensor diagnostics for monitoring vehicle operation and consumption," *Energy Reports*, Vol.6, pp.55-63, 2020.

[5] T. H. DeFries, M. Sabisch, S. Kishan, F. Posada, J. German, and A. Bandivadekar, "In-use fuel economy and CO_2 emissions measurement using OBD data on US light-duty vehicles," *SAE International Journal of Engines*, Vol.7, No.3, pp.1382-1396, 2014.

[6] A. Aliyu and S. Adeshina, "Classifying auto-MPG data set using neural network," *2014 11th International Conference on Electronics, Computer and Computation (ICECCO)*, Abuja, pp.1-4, 2014.

[7] M. N. Jamala, and S. S. Abu-Naser, "Predicting MPG for automobile using artificial neural network analysis," *Information Systems Research*, Vol.2, No.10. pp.5-21, 2018.

[8] K. H. Ahn, A. G. Stefanopoulou, and M. Jankovic, "Tolerant ethanol estimation in flex-fuel vehicles during MAF sensor drifts," *Proceedings of the ASME Dynamic Systems and Control Conference 2009*, pp.581-588, 2009.

[9] S. Boverie, D. Dubois, X. Guerandel, O. de Mouzon, and H. Prade, "Online diagnosis of engine dyno test benches: A possibilistic approach," *Proceedings of the 15th European Conference on Artificial Intelligence (ECAI'02)*, pp.658-662, 2002.

[10] D. Opitz and R. Maclin, "Popular ensemble methods: An empirical study," *Journal of Artificial Intelligence Research*, Vol.11, No.1, pp.169-198, 1999.

[11] J. Zaldivar, C. T. Calafate, J. C. Cano, and P. Manzoni, "Providing accident detection in vehicular networks through OBD-II devices and Android-based smartphones," *2011 IEEE 36th Conference on Local Computer Networks*, Bonn, pp.813-819, 2011.

[12] FUNK, Tom. System and method for implementing added services for OBD2 smart vehicle connection. U.S. Patent, US10249103B2, 2019.

[13] De Schutter, B. and T. J. J. van den Boom, "Model predictive control for max-min-plus-scaling systems - efficient implementation," *Sixth International Workshop on Discrete Event Systems, 2002 Proceedings*, pp.343-348, 2002.

[14] W. Cho, "Big Data-Based Fuel Consumption Estimation Model using Actual On-road DTG Data and Spatial Data," Graduate School of Business IT, Kookmin University, 2016.

[15] B. Predic, M. Madic, M. Roganovic, M. Kovačević, and D. Stojanovic, "Prediction of passenger car fuel consumption using artificial neural network: A case study in the city of Niš," *Automatic Control and Robotics*, Vol.15, No.2, pp.105-116, 2016.

[16] M. Lee, Y. Park, K. Jung, and J. Yoo, "Estimation of fuel consumption using in-vehicle parameters," International Journal of U- and E- Service, Science and Technology, Vol.4, No.4, pp.37-46, 2011.

양 희 은

https://orcid.org/0000-0002-6303-7658
e-mail : yanghe@skku.edu
2012년 성신여자대학교 컴퓨터정보학부(학사)
2020년 성균관대학교 데이터사이언스융합학과(석사)
2019년 ~ 현 재 단국대학교 EduAI센터 연구원
관심분야 : 인공지능, 강화학습, 컴퓨터비전, 정보분석

김 도 현

https://orcid.org/0000-0002-8577-603X
e-mail : kimtj@skku.edu
2018년 중앙대학교 컴퓨터공학과(학사)
2020년 성균관대학교 데이터사이언스융합학과(석사)
2020년 ~ 현 재 ㈜한국축산데이터 연구원
관심분야 : 인공지능, 패턴분석, 추천시스템, 데이터마이닝

최 호 섭

https://orcid.org/0000-0002-1211-8263
e-mail : hschoe@dankook.ac.kr
1998년 경남대학교 국어국문학과(학사)
2000년 경남대학교 국어국문학과(석사)
2007년 울산대학교 컴퓨터정보통신공학과 (박사)
2001년 ~ 2002년 한국전자통신연구원 지식정보검색연구팀 위촉연구원
2006년 ~ 2012년 한국과학기술정보연구원 정보기술개발단 선임연구원
2012년 ~ 2013년 문화체육관광부 국립현대미술관 정보화담당관
2015년 ~ 2017년 한국EDS 기술이사
2018년 ~ 현 재 단국대학교 EduAI센터 센터장
관심분야 : 자연언어처리, 시맨틱기술, 정보검색, 정보추출