



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



A novel framework to promote eco-driving through smartphone-vehicle integration

Author: Javier E. Meseguer Anastasio

Advisors: Carlos Tavares Calafate
Juan Carlos Cano Escribá

June 2017

Acknowledgements

This thesis owes its existence to the help, support and inspiration of several people. Firstly, I would like to express sincere appreciation and gratitude to Prof. Carlos Tavares Calafate and Prof. Juan Carlos Cano, my thesis advisors, for their guidance during my research. Their support and suggestions have been precious for the development of this thesis content. Thank you for the countless advices and ideas.

I am also indebted to Prof. Pietro Manzoni for the great help received in the various papers where we have collaborated. Also, my sincere thanks to Prof. C. K. Toh. Although I do not know him personally, the contributions suggested to the articles where we have collaborated have significantly improved their quality.

My greatest gratitude goes to my nephew Christian, who patiently revised and corrected my first english version of any document, paper or dissertation draft that I have done during these four long years. Conversations about the papers via chat, that began late in the evening and ended with the sun rising behind the International Space Station. Thanks Kryst.

Dedicated to

Nadia and Aleksander

If Tetris has taught them anything, it's that errors pile up and accomplishments disappear.

Raquel and Christian

For believing that in the end I would get it.

My parents

Abstract

It was not that long ago, just in the first half on the 1990s, when mobile phones were first introduced, being big and expensive. All you could do with them was to make phone calls. Since then mobile devices have experienced a great technological advance: we carry smartphones in our pockets that provide Internet access, having accelerometers that can measure acceleration, a gyroscope that can provide orientation information, different wireless interfaces such as Bluetooth connections, and above all, great computing power.

On the other hand, the automobile industry has evolved significantly during the last 10 years. One of the most exciting advances in vehicle development is vehicle-to-vehicle V2V communication, which allows cars to communicate with each other over a dedicated Wi-Fi band, and share information about vehicle speed, route direction, traffic flow, and road and weather conditions. An example of such a system is GM's (General Motors) OnStar, introduced in 1996, and that provides automatic response in case of an accident, stolen-vehicle recovery, remote door unlock, and vehicle diagnostics. Also, the standard On Board Diagnosis (OBD-II), available for several years, allows us to connect to the Electronic Control Unit (ECU) via a Bluetooth OBD-II connector. This connection interface allows connectivity between the smartphone and the vehicle, and can be purchased for just over 15 euros.

The spectrum of possibilities that arise when combining the car and the smartphone is unlimited, such as performing the diagnosis of the car by assuming the tasks performed by the car's On Board Unit (OBU), or sending the collected data to a platform where the diagnosis or maintenance of the system

can be realized in order to detect possible faults, help you to save gas and reduce environment pollution, and notify you of your car's problems, among other features.

The general objective pursued with this doctoral thesis is to help drivers to correct bad habits in their driving. To achieve this we promote the combination between smartphones and vehicular networks to design and develop a platform able to offer useful tips to achieve safer driving and greater fuel economy. It is well-known that intelligent driving can lead to lower fuel consumption, with the consequent positive impact on the environment.

The proposal that has been carried out in this doctoral thesis begins with the data capture from the vehicles' OBD-II port and data analysis through the use of graphs, maps, and statistics, both, on the server itself and in the smartphone's application developed. We applied data mining techniques and neural networks to analyze, study and generate a classification on driving styles based on the analysis of the characteristics of each specific route used for testing.

In a second phase, we demonstrate the relationship between fuel consumption and driving style. To achieve that goal, the first thing that we had to realize was how to apply different algorithms for the instantaneous consumption calculation (this parameter cannot be obtained directly from the vehicle ECU). Later, we studied and analyzed all data that was collected from the drivers who shared their monitored data with the server.

Although drivers do not recognize themselves as being in a state of anxiety while driving, they are more stressed than in any other daily activity, for example, when trying to stay in the right lane, keeping the car at a certain speed, and starting and stopping the vehicle. In general, drivers are more concentrated than they think, which causes an increase in the heart rate.

Many factors influence heart rate while at rest, e.g. stress, medications, medical conditions, even genes play a role. In our study we also investigate how stress and the driving behavior influence the heart rate. So, in the last phase, we demonstrate the correlation between heart rate and driving style, showing how the driving style can make the heart rate vary by 3 %.

Resumen

No hace mucho tiempo, tan sólo en la primera mitad en la década de los 90, cuando los teléfonos móviles aparecieron, eran grandes y caros, todo lo que se podía hacer con ellos era realizar llamadas telefónicas. Desde entonces los dispositivos móviles han experimentado un gran avance tecnológico, llevamos teléfonos inteligentes en el bolsillo con acceso a Internet, acelerómetros que calculan la aceleración instantánea, giroscopios que proporcionan información de orientación, diferentes conexiones inalámbricas como Bluetooth, y sobre todo, gran capacidad de computación.

Por otro lado, la industria del automóvil ha evolucionado mucho durante los últimos 10 años. Uno de los avances más interesantes en el desarrollo de vehículos ha sido la conectividad, V2V, o comunicación vehículo a vehículo, permite a los automóviles comunicarse mediante Wi-Fi y compartir información sobre la velocidad del vehículo, la dirección de la ruta actual, el tráfico, así como las condiciones de la carretera y las condiciones ambientales. Por ejemplo, el sistema OnStar de GM (General Motors) fue introducido en 1996, el sistema tiene aviso en caso de accidente, recuperación de vehículo robado, desbloqueo remoto de la puerta, diagnóstico de vehículos, etc. Por otra parte, el estándar On Board Diagnosis (OBD-II), disponible desde hace varios años, permite conectarnos de forma sencilla a la ECU (Electronic Control Unit) mediante un conector Bluetooth OBD-II. Este interfaz de conexión permite la conectividad entre el dispositivo móvil y el vehículo, se puede adquirir por poco más de 15 euros.

El espectro de posibilidades que surgen al combinar el automóvil y el smart-phone es amplísimo, como por ejemplo realizar el diagnóstico del coche a través del móvil asumiendo las tareas que hace la unidad On Board Unit (OBU) del coche, o bien enviar los datos recogidos a una plataforma donde se pueda realizar el diagnóstico o mantenimiento del sistema, detectando posibles fallos puede ayudar a ahorrar en el consumo de combustible, notificar los problemas del coche en tiempo real, entre otras características.

El objetivo general que se persigue con esta tesis doctoral es ayudar al conductor a corregir malos hábitos en su forma de conducción. Conseguimos esto mediante la combinación entre smartphones y las redes vehiculares, diseñamos y desarrollamos una plataforma capaz de ofrecer consejos útiles para conseguir una conducción más segura y un mayor ahorro de combustible. Es conocido que una conducción inteligente puede llevarnos a un menor consumo de combustible, con el consiguiente impacto positivo que ello conlleva sobre el medio ambiente.

La propuesta que se ha llevado a cabo en esta tesis doctoral comienza con la obtención de los datos desde el OBD-II del coche y su presentación y análisis mediante el uso de gráficas, mapas, estadísticas, tanto en el propio servidor como en la aplicación móvil desarrollada para la obtención de datos recibidos desde la ECU. Se aplicaron técnicas de minería de datos y redes neuronales para analizar, estudiar y generar una clasificación sobre los estilos de conducción en base al análisis de las características de la vía sobre la que ha realizado la ruta.

En una segunda fase se demostró la relación entre el consumo de combustible con el estilo de conducción, para ello lo primero que tuvimos que realizar fue aplicar diversos algoritmos para el cálculo del consumo instantáneo, este parámetro no es posible obtenerlo directamente de la ECU del vehículo. Posteriormente se realizó el estudio y el análisis de todos los datos que se recogieron de los conductores que se prestaron a la realización del estudio enviando los datos al servidor.

Aunque los conductores no se reconozcan estar en estado de ansiedad, al conducir sé está más alerta que en cualquier otra actividad diaria, por ejemplo, intentando permanecer en el carril de la derecha, manteniendo el coche a una cierta velocidad determinada, arrancando y deteniendo el vehículo. En general, los conductores están más concentrados que lo que ellos piensan, lo que generan un aumento del ritmo cardíaco.

Muchos factores influyen en la frecuencia cardíaca en reposo, por ejemplo, el estrés, los medicamentos, las condiciones médicas, incluso los genes tienen su

influencia, el envejecimiento tiende a acelerarlo, y el ejercicio regular tiende a ralentizarlo. En nuestro estudio también investigamos cómo el estrés y el comportamiento en la conducción influyen en la frecuencia cardíaca. En la última fase vemos la correlación existente entre el ritmo cardíaco y el estilo de conducción, demostramos como el estilo de conducción puede llegar a modificar el ritmo cardíaco hasta en un 3 %.

Resum

No fa molt de temps, tan sols en la primera mitat en la dècada dels 90, quan els telèfons mòbils van aparèixer, eren grans i cars, tot el que es podia fer amb ells era realitzar telefonades. Des de llavors els dispositius mòbils han experimentat un gran avanç tecnològic, portem telèfons intel·ligents en la butxaca amb accés a Internet, acceleròmetres que calculen l'acceleració instantània, giroscopis que proporcionen informació d'orientació, diferents connexions sense fils com Bluetooth, i sobretot gran capacitat de computació.

D'altra banda, la indústria de l'automòbil ha evolucionat molt durant els últims 10 anys. Un dels avanços més interessants en el desenrotllament de vehicles ha sigut la connectivitat, V2V, o comunicació vehicle a vehicle, permet als automòbils comunicar-se per mitjà de la banda de Wi-Fi i compartir informació sobre la velocitat del vehicle, la direcció de la ruta actual, les condicions del tràfic, així com l'estat de la carretera i les condicions ambientals. Per exemple, el sistema OnStar de GM (General Motors) va ser introduït en 1996, el sistema té avís en cas d'accident, recuperació de vehicle robat, desbloqueig remot de la porta, diagnòstic de vehicles, etc. D'altra banda l'estàndard On Board Diagnosi (OBD-II), disponible des de fa diversos anys, permet connectar-nos de forma senzilla a l'ECU (Electronic Control Unit) per mitjà d'un connector Bluetooth OBD-II. Esta interfície de connexió permet la connectivitat entre el dispositiu mòbil i el vehicle, es pot adquirir per poc més de 15 euros.

L'espectre de possibilitats que sorgixen al combinar l'automòbil i el smartphone és il·limitat, com per exemple realitzar el diagnòstic del cotxe a través del mòbil assumint les tasques que fa la unitat On Board Unit (OBU) del cotxe, o bé

enviar les dades arreplegades a una plataforma on es puga realitzar el diagnòstic o manteniment del sistema, detectant possibles fallades, ajuda a estalviar en el consum de combustible, notificar els problemes del cotxe en temps real, entre altres característiques.

L'objectiu general que es persegueix amb esta tesi doctoral és ajudar al conductor a corregir mals hàbits en la seua forma de conducció. Aconseguim açò mitjançant de la combinació entre smartphones i les xarxes vehiculars, dissenyem i desenrotllem una plataforma capaç d'oferir consells útils per a aconseguir una conducció més segura i un major estalvi de combustible. És conegut que una conducció intel·ligent pot emportar-nos a un menor consum de combustible, amb el consegüent impacte positiu que això comporta sobre el medi ambient.

La proposta que s'ha dut a terme en esta tesi doctoral comença amb l'obtenció de les dades des de l'OBD-II del cotxe i la seua presentació i anàlisi per mitjà de l'ús de gràfiques, mapes, estadístiques, tant en el propi servidor, com en l'aplicació mòbil desenrotllada per a l'obtenció de dades rebudes des de l'ECU. S'apliquen tècniques de mineria de dades i xarxes neuronals per a analitzar, estudiar i generar una classificació sobre els estils de conducció basant-se en l'anàlisi de les característiques de la via sobre la qual ha realitzat la ruta.

En una segona fase es va a demostrar la relació entre el consum de combustible amb l'estil de conducció, per a això la primera cosa que vam haver de realitzar va ser aplicar diversos algorismes per al càlcul del consum instantani, este paràmetre no és possible obtindre-ho directament de l'ECU del vehicle. Posteriorment es va realitzar l'estudi i l'anàlisi de totes les dades que es van arreplegar dels conductors que es van prestar a la realització de l'estudi enviant les dades al servidor.

Encara que els conductors no es reconeguen estar en estat d'ansietat, al conduir s'està més alerta que en qualsevol altra activitat diària, per exemple, intentant romandre en el carril de la dreta, mantenint el cotxe a una certa velocitat determinada, arrancant i detenint el vehicle. En general, els conductors estan més concentrats que el que ells pensen, la qual cosa generen un augment del ritme cardíac.

Molts factors influeixen en la freqüència cardíaca en repòs, per exemple, l'estrès, els medicaments, les condicions mèdiques, fins i tot els gens tenen la seua influència, l'envelliment tendeix a accelerar-ho, i l'exercici regular tendeix a ralentir-ho. En el nostre estudi només estem interessats en com l'estrès i el comportament en la conducció influeixen en la freqüència cardíaca. En l'última

fase vam veure la correlació existent entre el ritme cardíac i l'estil de conducció, i vam demostrar com l'estil de conducció pot arribar a modificar el ritme cardíac un 3 %.

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Objectives and Methodology	2
1.3	Organization of the thesis	3
2	Background	5
2.1	Overview of the OBD-II standard	5
2.1.1	OBD-II Hardware and Protocols	7
2.1.2	ELM327 Bluetooth	8
2.2	Heart rate monitor: Bluetooth Low Energy (LE) & Chest Straps	9
2.3	Introduction to Neural Networks	10
2.3.1	Neural Network Training	11
2.4	Android Application development	13
2.5	Data Center: Multi-tier Architecture	17
3	DrivingStyles: A Smartphone Application to Assess Driver Behavior	19
3.1	Introduction	20

3.2 Related Work	21
3.3 DrivingStyles Architecture	22
3.4 Android application	24
3.4.1 Configuration options	24
3.4.2 Application Modules	25
3.5 DrivingStyles Web interface	27
3.6 Neural Networks based data analysis	28
3.6.1 Neural Network description	29
3.6.2 Training the Neural Network	31
3.6.3 Obtained results	32
3.7 Conclusions and future work	35
3.8 Acknowledgments	35
4 Assessing the Impact of Driving Behavior on Instantaneous Fuel Consumption	37
4.1 Introduction	38
4.2 Related Work	39
4.3 Overview of the DrivingStyles Architecture	40
4.3.1 Android Application and Web Interface	41
4.4 Fuel Consumption / Instantaneous Fuel Consumption Calculation	44
4.5 Greenhouse Gas Emissions Calculation	47
4.6 Experimental Results and Evaluation	48
4.7 Conclusions and future work	51
4.8 Acknowledgments	51
5 DrivingStyles: A Mobile Platform for Driving Styles and Fuel Consumption Characterization	53
5.1 Introduction	54
5.2 Related Work	56
5.3 DrivingStyles Architecture	57
5.3.1 DrivingStyles Android Interface	58
5.3.2 DrivingStyles Server Interface	59

5.4 Fuel consumption and greenhouse gas emissions calculation.	61
5.4.1 Fuel consumption	61
5.4.2 Greenhouse gas emissions calculation	64
5.5 Neural Networks-based data analysis.	64
5.6 Experimental results and evaluation	67
5.7 Conclusions and future work	68
5.8 Acknowledgments.	69

6 DrivingStyles: Assessing the Correlation of Driving Behavior with Heart Rate Changes 71

6.1 Introduction	72
6.2 General Overview of the DrivingStyles Architecture.	73
6.2.1 Android Application	73
6.2.2 Data Center	75
6.3 Research Strategy and Methodology	75
6.3.1 Participant	75
6.3.2 OBD-II Instrument	75
6.3.3 Heart Rate Monitor (HRM)	76
6.3.4 Measurement Result.	77
6.4 Experimental Results and Evaluation	78
6.4.1 On-road Tests (all routes)..	78
6.4.2 On-road Tests (single route).	79
6.5 Conclusions and Future Work	82
6.6 Acknowledgments.	84

7 On the Correlation Between Heart Rate and Driving Style in Real Driving Scenarios 85

7.1 Introduction	86
7.2 General Overview of the DrivingStyles Architecture.	87
7.2.1 Android Application	88
7.2.2 Data Center	89
7.3 Research Strategy and Methodology	89
7.3.1 Participant	89

- 7.3.2 OBD-II Car Instrument 90
- 7.3.3 Heart Rate Monitor (HRM) 90
- 7.3.4 Measurement Results 91
- 7.4 Experimental Results and Evaluation 92
 - 7.4.1 Driving tests in Urban Areas 93
 - 7.4.2 Driving Tests in Suburban Areas 94
 - 7.4.3 Driving Tests in Highways 95
- 7.5 Conclusions and Future Work 99
- 7.6 Acknowledgments 99

- 8 Summary of Achievements 101
 - 8.1 Neural Network Tunning 102
 - 8.2 Data Center 105
 - 8.3 Android Application 107
 - 8.4 Fuel Consumption and Greenhouse Gas Emissions 112
 - 8.5 Heart Rate Analysis 113

- 9 Conclusions, Publications and Future Work 115
 - 9.1 Conclusions 115
 - 9.2 Publications 117
 - 9.3 Products 118
 - 9.4 Open Reseach Issues 119

- Bibliography 125

List of Figures

2.1	16 pin J1962 OBD-II car proprietary connector.	7
2.2	Operational flow diagrams for the Heart Rate sensors.	9
2.3	Schematic representation of a Three-layer Neural Networks. . .	11
2.4	Android operating system stack of software components.	16
2.5	Overview of a three-tier application.	18
3.1	DrivingStyles: architecture overview.	23
3.2	DrivingStyles: architecture overviewSnapshots of the acceleration, speed, rpm parameters and map module.	26
3.3	Snapshot of a route map.	27
3.4	Trained neural network and associated weights.	30
3.5	Neural networks convergence behavior	33
3.6	Snapshots of route type and behavior.	34
4.1	System architecture of DrivingStyles.	40
4.2	Snapshots of route type and behavior.	42

4.3	Snapshots of the main screen and the data sending module. . .	43
4.4	Scheme of the different possibilities of MAF calculation.	45
4.5	Outline of the calculation of the Fuel Flow average.	46
4.6	Chart of consumption and CO_2 in relation to the driving behavior.	48
4.7	Comparative chart between two drivers and their behavior.	49
4.8	Box and whisker plot of Fuel Consumption / Driving Behavior.	50
5.1	System Architecture of DrivingStyles.	55
5.2	Snapshots of the fuel consumption, speed, acceleration parameters, and map module for the Android application.	59
5.3	Snapshots of driver behavior and route type corresponding to the data center web.	61
5.4	Scheme of the different MAF calculation possibilities regarding fuel consumption calculation.	63
5.5	Schematic representation of the Three-layer Neural Networks used in the DrivingStyles Architecture.	65
5.6	Chart of consumption and CO_2 in relation to the driving behavior.	67
5.7	Box and whisker plot of Fuel Consumption/Driving Behavior.	68
6.1	System architecture of DrivingStyles: Overview and block diagram architecture.	74
6.2	Snapshots of the main screen and the heart rate module.	76
6.3	Correlation between heart rate and driving behavior.	79
6.4	Box and whisker plot of heart rate vs driving behavior.	80
6.5	Correlation between heart rate and driving behavior (quiet-normal).	81
6.6	Correlation between heart rate and driving behavior (normal-aggressive).	82

6.7	Correlation between heart rate and driving behavior.	83
6.8	Box and whisker plot of heart rate vs driving behavior.	83
7.1	System architecture of DrivingStyles: Overview and block diagram.	88
7.2	On Board schema components of DrivingStyles instrumentation.	90
7.3	Snapshots of the main screen and the heart rate module for the Android application.	91
7.4	Box and whisker plot for heart rate vs. driving behavior.	97
7.5	Heart rate evolution along the three highway routes	98
8.1	Overall hardware and software system architecture.	101
8.2	Neural network training file.	104
8.3	Data Center Plots.	106
8.4	Android application's modules flow diagram.	109
8.5	Android application's flow diagram.	110
8.6	Android application's screen.	111
8.7	Methods used in Android app to calculate the instantaneous fuel consumption.	112

Chapter 1

Introduction

1.1 Motivation

Driving is a complex activity that requires a skilled behavior at different levels. Factors contributing to road traffic crashes generally fall into three categories: environmental, vehicle, and human, with the human factor being by far the most determinant. Obtaining an in-depth exploration of driving behavior and factors underpinning risky driving could therefore be useful to facilitate the establishment of effective policies to mitigate them.

In emergency situations our automatic reactions may not be appropriate, and shifting our behavior to more consciously control such rare events is unlikely to succeed because we do not have effective emergency training at our disposal. Behavior will reflect the psychological profile of each individual, and the situations the individual is exposed to.

Behavior may be aggressive, passive, distracted, alert, ignorant, confident, timid, skilled, arrogant, tolerant, angry or fall under many other descriptions. Of all the possible behaviors, we are only concerned with those behaviors that may be dangerous to driving. In particular, our study will only deal with three of these possible behaviors, and we will classify them as aggressive, normal and quiet.

Much of our driving behavior is automatic, developed from experiences and our perception of driving situations, typically requiring minimal attention. Behavior can change as a result of learning, provided we take up new strategies and practice them until they become new behaviors. Thus, it becomes necessary to develop an architecture that helps drivers notice their own behavior while driving, and help them to change it in case that this behavior is dangerous.

1.2 Objectives and Methodology

The main objective of this thesis is the creation and development of an architecture for monitoring drivers. We analyze the driving behavior, and how this behavior affects fuel consumption and heart rate, through the use of neural networks and data mining.

In this thesis we adopted a mostly pragmatic approach. First, we reviewed the state of the art in order to find the most relevant contributions in the area of knowledge associated to driving behavior. At first, there was not much literature in the field due to the novelty of the subject. More recently, we have seen a great interest in the research topics addressed in this thesis.

Second, and contrarily to other previous studies, we focused our efforts on the implementation of a new and complete architecture in real time in which we analyze thousands of data sent by drivers. Thanks to this architecture, and its implementation in real systems, we were able to obtain information about the car and the driver, which have allowed us to perform the different studies about behavior, and how it affects consumption and heart rate.

As the third and final point of this thesis, we performed the analysis of data mining on the detection of the driving behavior, and route type, as well as how driving behavior affects fuel consumption and the driver's heart rate.

In order to address this general objective, it is decomposed into several problems that will be tackled and developed in this thesis:

1. Development of an interface capable of interacting with the vehicle from the smartphone.
2. Training, setting, and creation of a Neural Network capable of determining the type of route and the driving style.
3. Creation of an application for the Android system able to monitor the driving style, the driver, and the vehicle in real time.

4. Creation of a Data Center, able to receive, store, and process large amounts of data to find out the individual driving style.
5. Development of a module capable of estimating the vehicle's consumption in order to study the correlation between the driving style and consumption.
6. Development of a module capable of integrating heart rate sensors in the developed application to allow studying the correlation between driving style and heart rate.

1.3 Organization of the thesis

This thesis is presented as a compilation of articles (adapted to the required format), and so, the rest of the thesis is organized as follows:

Chapter 2 provides an overview of the basic concepts and technologies used in the thesis. In Chapter 3 we present the Neural Networks developed and the proposed DrivingStyles architecture. Chapters 4 and 5 include a discussion of the results obtained regarding the correlation between fuel consumption and driving style. Chapters 6 and 7 include the different relevant publications in terms of the correlation between the driving behavior and heart rate changes. In Chapter 8 we make a summary of the milestones reached in the thesis. Finally, Chapter 9 presents the conclusions of this work, as well as ideas for future work. It also includes a summary of the different publications presented in this thesis.

Chapter 2

Background

The central chapters of this thesis are academic papers that have been published in different journals and international conferences. So, each of them already includes a part dedicated to revise the state of the art in the particular problem addressed by the paper. Nevertheless, we are including here a summary of the main technological topics associated to the thesis in order to help the reader to differentiate each of the topics that have been tackled.

2.1 Overview of the OBD-II standard

OBD-II¹ [1] is the abbreviation of On Board Diagnostics II, the second generation of the regulation that required all cars to have this type of device. The first generation was OBD-I², and a tightening of emission limits in 1996 led to the creation of the OBD-II. This system incorporates two oxygen sensors (Lambda probe), one located before the catalyst and another one after it, thus being able to verify its correct operation. If a problem or fault is detected, the OBD-II system lights a warning lamp on the dashboard called MIL (Malfunction Indicator Lamp) that warns the driver of a fault. The warning lamp usually carries the inscription «Check Engine» or «Service Engine Soon». The

¹<http://www.obdii.com/>

²<https://law.resource.org/pub/us/cfr/ibr/005/sae.j1979.2002.pdf>

system also keeps important information about the detected faults so that a mechanic can find and solve the problem. In the United States, all gasoline vehicles since 1996 must have OBD-II systems, similarly to all passenger cars and diesel trucks since 1997. In Europe, according to the directive 98/69EG [2], gasoline cars from the year 2000 onwards, diesel cars from 2003, and trucks since 2005 have to support OBD-II. In more detail:

1. OBD-I: Designed to detect electrical faults in the system and components. The so-called MIL (Malfunction Indicator Lamp) light turns off if the emissions problem is corrected by itself. It stays on until 3 consecutive driving cycles have passed, without the problem recurring. The memory is cleared after 40 cold starts. However, in the case of the fuel scan, 80 cold starts are required.
2. OBD-II: The standard specifies the type of diagnostic connector and its pin-out (see Table 2.1), the electrical signaling protocols and the messaging format. It monitors the emission systems behavior and components, as well as electrical faults, and stores information for later use. The OBD-II standard implements various working modes. This means that, depending on the information to be accessed, a different mode is needed. Once inside this mode of work, an extensive number of parameters is offered to access this information (see Table 2.2).

Table 2.1: Standard OBD-II pin lay-out

Pin	Description	Pin	Description
Pin 1	<i>VendorOption</i>	Pin 9	<i>VendorOption</i>
Pin 2	<i>SAEJ1850 Bus+ [3]</i>	Pin 10	<i>J1850 Bus[3]</i>
Pin 3	<i>VendorOption</i>	Pin 11	<i>Vendor Option</i>
Pin 4	<i>Chassis Ground</i>	Pin 12	<i>Vendor Option</i>
Pin 5	<i>Signal Ground</i>	Pin 13	<i>Vendor Option</i>
Pin 6	<i>CAN High (J – 2284)</i>	Pin 14	<i>CAN Low (J – 2284)</i>
Pin 7	<i>ISO9141 – 2KLine [4]</i>	Pin 15	<i>ISO 9141 – 2 Low [4]</i>
Pin 8	<i>Vendor Option</i>	Pin 16	<i>Battery Power</i>

Table 2.2: OBD-II Standard working modes.

Mode (hex)	Description
Mode 01	Show current data
Mode 02	Show freeze frame data
Mode 03	Show stored Diagnostic Trouble Codes
Mode 04	Clear Diagnostic Trouble Codes and stored values
Mode 05	Test results, oxygen sensor monitoring (non CAN only)
Mode 06	Test results, other component/system monitoring (Test results, oxygen sensor monitoring for CAN only)
Mode 07	Show pending Diagnostic Trouble Codes (detected during current or last driving cycle)
Mode 08	Control operation of onboard component/system
Mode 09	Request vehicle information
Mode 0A	Permanent Diagnostic Trouble Codes (DTCs) (Cleared DTCs)

2.1.1 OBD-II Hardware and Protocols

OBD-II defines a 16-pin standard hardware interface (female J1962 connector [5]), see Table 2.1 and Figure 2.1 for more details. The connector complies with ISO 15031-3: 2004 [6], which even specifies where the connector should be located, although it is usually found in any position, always under the dashboard.

The OBD-II offers five different protocols, although most vehicles implement only one of these protocols. All OBD-II connection pins use the same connector, but according to the protocol they differ in which pins, except pin 4 (Battery negative) and pin 16 (Battery positive) [6].

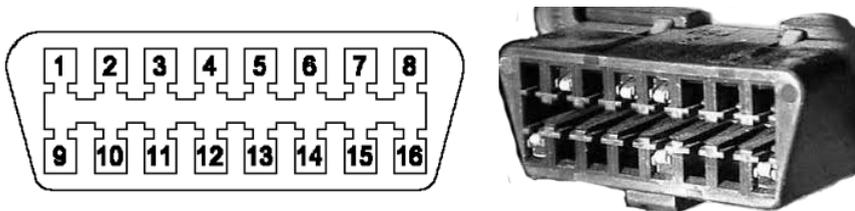
**Figure 2.1:** 16 pin J1962 OBD-II car proprietary connector.

Table 2.3: OBD-II standard protocols.

Mode	Description
SAE J1850 PWM [3]	<i>Pulse Width Modulation, used by Ford USA.</i>
SAE J1850 VPW [3]	<i>Variable Pulse Width, used by GM USA</i>
ISO 9141-2 [4]	<i>In European, Asian, and Chrysler vehicles with variants.</i>
ISO 14230 KWP200 [7]	<i>Keyword Protocol 2000 used by the VAG group.</i>
ISO 15765 CAN [8]	<i>ISO9141 – 2 KLine</i>

In general, European products, many Asian manufacturers and Chrysler adopt the ISO 9141 [4] protocol. General Motors uses SAE J1850 VPW, and Ford adopts the SAE J1850 PWM communication standards [3].

2.1.2 ELM327 Bluetooth

The ELM327 [9–11] is originally an 18F2480 PIC³ (Peripheral Interface Controller), and is programmed by ELM Electronics [12]. It acts as an interface between RS232 and OBDII for systems that use the ISO 15765-4 CAN, SAE J1850 PWM, SAE J1850 VPW, ISO 9141-2, ISO 14230-4, and SAE J1939 protocols, which are adopted by different vehicle models. However, PICs are in general a family of microcontrollers made by Microchip Technology®⁴.

The vehicle’s data transfer to the OBD-II follows several protocols, but none of them can be directly used in our smartphones. We need a device like the ELM327 designed to act as an interface between the OBD ports and the standard RS232 [9].

Once the ELM327’s connection is established using one of the previously listed protocols, the device will send a message with the version of the ELM327 device, and later the character '>', which means that the device is ready to receive characters through the RS232 port.

The characters sent to the device can be interpreted by the ELM327, determining if it is a configuration command of the device itself, or a message for the vehicle, in which case the message will be reformatted and sent to the OBD interface. The device can determine who the message is addressed to since messages that start with 'AT' are used for the configuration of the ELM327

³<http://ww1.microchip.com/downloads/en/DeviceDoc/39564c.pdf>

⁴<http://www.microchip.com/>

device, while the OBD commands intended for the vehicle are only allowed to contain specific ASCII codes.

For either command type, the message must end with the carriage return character (*Hex '0D'*), so that it can be evaluated. If this is not the case, a timer is activated automatically, and after 20 seconds, it will abort the message.

2.2 Heart rate monitor: Bluetooth Low Energy (LE) & Chest Straps

Bluetooth low energy (LE), sometimes referred to as "Bluetooth Smart", is the low power-version of Bluetooth that was built for the Internet of Things (IoT). It is a light weight subset of Bluetooth, and was introduced as part of the Bluetooth 4.0 core specification.

One of those devices that take advantage of this technology are the chest straps. A heart rate monitor consists of two parts, a transmitter attached to a belt worn around the chest, and a receiver in the mobile device. As the heart beats, an electrical signal is transmitted through the heart muscle in order for it to contract. This electrical activity can be detected through the skin. The transmitter then sends an electromagnetic signal containing heart rate data to the mobile device. Heart rate sensors based on chest straps and Bluetooth Low Energy operate according to the following state chart (show in Figure 2.2):

1. When the electrode areas on the reverse side of the strap detect skin contact, Bluetooth Low Energy starts Peripheral role (as a Sensor) and advertises to be connectable with a Collector (mobile).
2. When the Collector sends a Connection Request after a detected advertising message, the Sensor accepts it and a short pairing phase will start if needed.

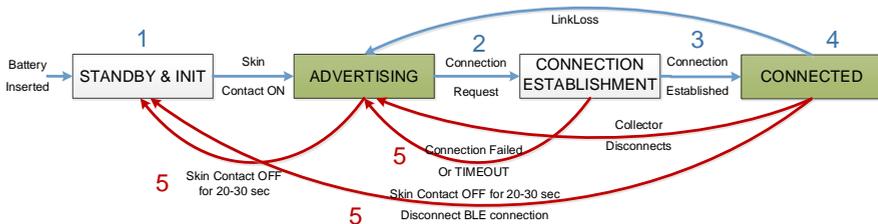


Figure 2.2: Operational flow diagrams for the Heart Rate sensors.

3. After pairing is finished, the Collector enables heart rate notifications from the Sensor by writing value 0x01 to the Client Characteristic Configuration descriptor.
4. The Sensor will measure heart rate once in a second and sends it as a Heart Rate Measurement Notification.
5. After the strap is removed, absence of skin contact is detected by the Sensor Contact. The Sensor terminate the BLE connection if contact is absent for 20-30 seconds. When the Sensor is in standby mode, its current consumption is very low ($< 1\mu A$).

2.3 Introduction to Neural Networks

Artificial Neural Networks (ANNs) [13] are inspired by biology, in nature neurons receive signals (inputs) from other neurons through synaptic connections that can be exciting or Inhibited. Depending on the signals received, a neuron sends a signal to other neurons in turn by means of the axon. A neuron does nothing, unless the collective influence of all its inputs reaches a threshold level. Whenever such a threshold is reached, the neuron produces an output, consisting of a pulse that moves from the cell body, through the axon, to the branches of this, and in this case it is said that triggers the neuron. A neuron with this behavior is said to be an *all or nothing* device.

Artificial Neural Networks consist of neurons such as shown in Figure 2.3, which represents a three-layer neural networks. The simulated neuron is seen as a node connected to others via bonds corresponding to axon-synapsis-dendrite connections. Each cell (process unit) supplies a value to its output, and this value propagates through the network of unidirectional connections to other cells in the network. Associated to each connection there is a weight W_{ij} , which determines the effect of the *jth cell* on the *ith cell*, which determines the influence of one node on another, influences by weight of the link that connects them. A large positive weight corresponds to a strong excitation, and a small negative weight corresponds to a weak inhibition.

The activation function is one in which each node combines the separate influences it receives on its input links to determine a global influence. Each cell (process unit) supplies a value to its output, and this value then propagates through the network of unidirectional connections to other cells of the network.

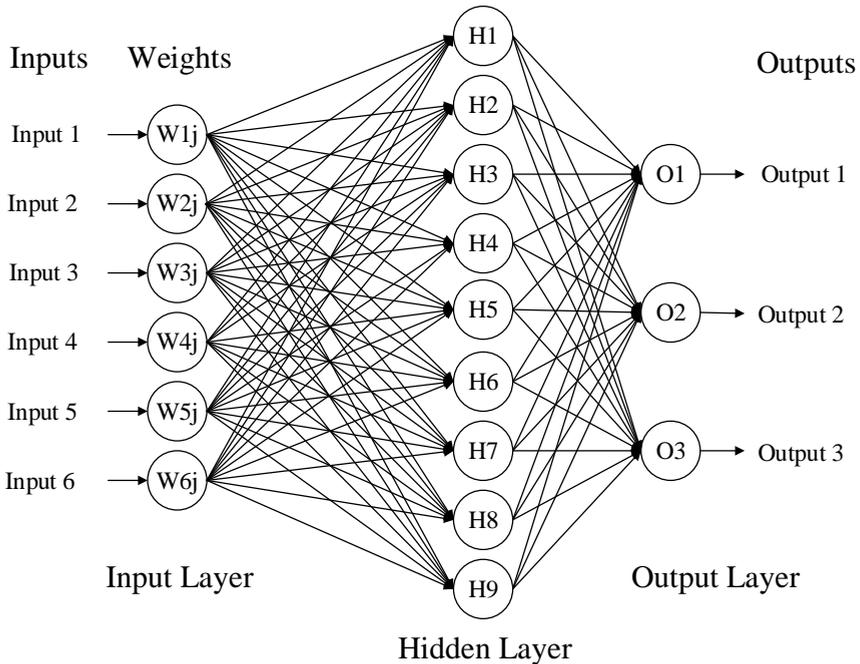


Figure 2.3: Schematic representation of a Three-layer Neural Networks.

2.3.1 Neural Network Training

The main characteristic of Neural Networks is their ability to learn from their inputs, and improve their success rate through learning. A Neural Network learns through an interactive process of adjustments of its synaptic weights and bias levels. The sequence of events that occur during learning is as follows:

1. Stimulate the Neural Network for its inputs.
2. The Neural Network changes in its parameters as a result of such stimulation.
3. The Neural Network responds differently to new inputs because of the changes that occurred in its internal structure.

The term learning algorithm refers to the set of well-defined rules for the solution of a learning problem. There is a great variety of learning algorithms (backpropagation for feedforward networks, Elman, Real-Time, Learning Algo-

rithm, etc.), each having its own advantages. Learning algorithms differ from each other in the way in which changes in synaptic weights are formulated.

The algorithm that has been chosen in this thesis is backpropagation because, examining the type of problem that we have to solve, we observe that it is a classification problem where, starting from an input, that in our case are the speed, acceleration, and the engine revolutions, we obtain as output the type of road, and the driving style.

Backpropagation [14] is a type of supervised learning network that gives very good results in classification problems, as in the present research thesis. It uses a two-phase propagation cycle. Once a pattern has been applied to the input of the network as a stimulus, it propagates from the first layer through the next layers of the network, until an output is generated. The output signal is compared to the desired output, and an error signal is calculated for each of the outputs. The error values are propagated backwards, starting from the output layer, to all the hidden neurons layer that contribute directly to the output.

The hidden neurons layer receive only a fraction of the total error signal, based roughly on the relative contribution that each neuron has contributed to the original output. This process is repeated, layer by layer, until all the neurons in the network receive an error signal that gives us their contribution to the total error. From here the connection weights of each neuron are updated, to make the network converge to a state that allows to correctly classify all the training patterns.

As the network trains, the neurons in the intermediate layers are organized and learn to recognize different characteristics of the input data. After training, neurons will respond with an active output if the new input contains a pattern that resembles that characteristic that individual neurons have learned to recognize during their training. Conversely, the units of the hidden layers have a tendency to inhibit their output if the input pattern does not contain the characteristic to be recognized, and for which they have been trained.

The application that has been used for the creation and training our proposed Neural Networks has been JavaNNS [15], which is a java version of the SNNS program [16] from TÜBINGEN University⁵. It has a friendly graphical interface in addition to an on-line manual⁶ to start working effectively. The data to train and validate the neural network are passed through a *.pat file. In our

⁵<http://www.ra.cs.uni-tuebingen.de/>

⁶<http://www.ra.cs.uni-tuebingen.de/software/JavaNNS/manual/JavaNNS-manual.html>

system the problem arises at the time of exporting the generated model to real code in the data center, since JavaNNS has not implemented the generation of code of any type. However, since it allows exporting files with the configuration of the extracted neural network, we were able to generate C code by using the `snns2.exe` [16] utility.

2.4 Android Application development

Android is an open source and Linux-based Operating System for mobile devices such as smartphones and tablet computers. Android was developed by the Open Handset Alliance⁷, led by Google, and other companies.

Android offers a unified approach to application development for mobile devices, which means developers only need to develop for the Android platform, and their applications should be able to run on different devices powered by Android.

The Android operating system is a stack of software components which is roughly divided into five sections and four main layers, as shown below in the architecture diagram in Figure 2.4. Next we discuss the most important blocks of the Android architecture.

1. **Linux kernel:** At the bottom of the layers is Linux - Linux 3.6 with approximately 115 patches. This provides a level of abstraction between the device hardware, and it contains all the essential hardware drivers like camera, keypad, display etc. Also, the kernel handles all the things that Linux is really good at, such as networking and a vast array of device drivers, which take the pain out of interfacing to peripheral hardware.
2. **Libraries:** On top of the Linux kernel there is a set of libraries including the open-source Web browser engine WebKit, the well known library `libc`, the SQLite database, which is a useful repository for storage and sharing of application data, libraries to play and record audio and video, and SSL libraries responsible for Internet security, among others.
3. **Android Libraries:** This category encompasses those Java-based libraries that are specific to Android development. Examples of libraries in this category include the application framework libraries, in addition to those that facilitate user interface building, graphics drawing, and database access.

⁷<http://www.openhandsetalliance.com/>

4. **Android Runtime:** This is the third section of the architecture, and it is available on the second layer from the bottom. This section provides a key component called Dalvik Virtual Machine, which is a kind of Java Virtual Machine specially designed and optimized for Android.
5. **The Dalvik VM:** Makes use of Linux core features like memory management and multi-threading, which is intrinsic in the Java language. The Dalvik VM enables every Android application to run in its own process, with its own instance of the Dalvik virtual machine.

The Android runtime also provides a set of core libraries which enable Android application developers to write Android applications using standard Java programming language.

6. **Application Framework:** The Application Framework layer provides many higher-level services to applications in the form of Java classes. Application developers are allowed to make use of these services in their applications.
7. **Applications:** You will find all the Android application at the top layer. Examples of such applications include Contacts, Books, Browsers, Games, etc.

Android is a powerful operating system competing with Apple iOS 10 and it supports great features. Some of them are listed in Table 2.4.

Table 2.4: Android Application Features.

Feature	<i>Description</i>
Connectivity	GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE, NFC and WiMAX
Storage	SQLite, a lightweight relational database, is used for data storage purposes
Media support	H.263, H.264, MPEG-4 SP, AMR, AMR-WB, AAC, HE-AAC, AAC 5.1, MP3, MIDI, Ogg Vorbis, WAV, JPEG, PNG, GIF, and BMP
Messaging	SMS and MMS
Web browser	Based on the open-source WebKit layout engine, coupled with Chrome's V8 JavaScript engine supporting HTML5 and CSS3
Multi-touch	Android has native support for multi-touch which was initially made available in handsets such as the HTC Hero
Multi-tasking	User can jump from one task to another and same time various application can run simultaneously
Resizable widgets	Widgets are resizable, so users can expand them to show more content or shrink them to save space
Multi-Language	Supports single direction and bi-directional text
GCM	Google Cloud Messaging (GCM) is a service that lets developers send short message data to their users on Android devices, without needing a proprietary sync solution
Wi-Fi Direct	A technology that lets apps discover and pair directly, over a high-bandwidth peer-to-peer connection
Android Beam	A popular NFC-based technology that lets users instantly share, just by touching two NFC-enabled phones together

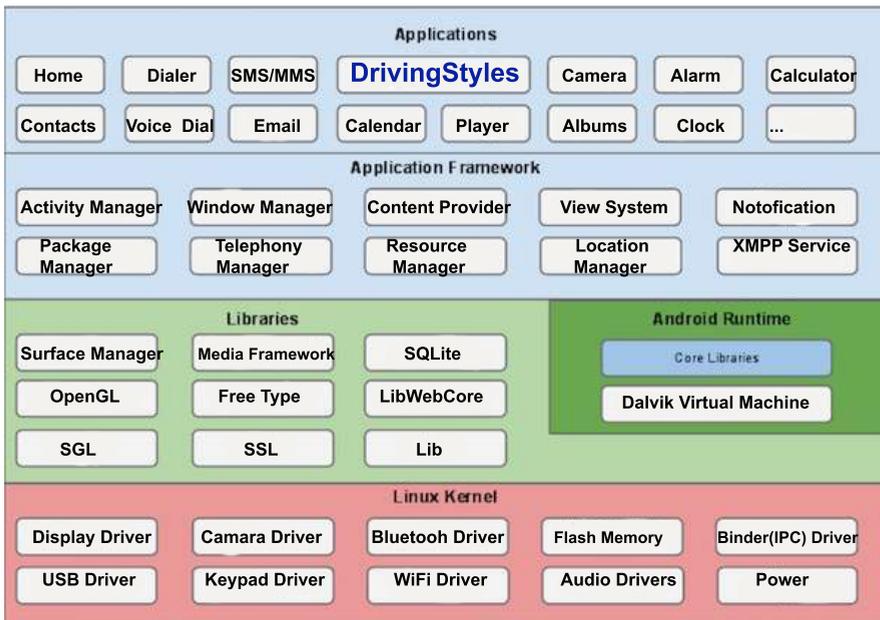


Figure 2.4: Android operating system stack of software components.

2.5 Data Center: Multi-tier Architecture

A Multi-tier Architecture is a software architecture in which different software components, organized in tiers (layers), provide dedicated functionality. The most common occurrence of a multi-tier architecture is a three-tier system consisting of a data tier (mostly encompassing one or several database servers), an logic tier (business logic) and a presentation tier (interface functionality). Web information systems, for instance, encompass a dedicated tier (web tier) between client and logic layer. Figure 2.5 shows the overview of a three-tier application.

Conceptually, a multi-tier architecture results from a repeated application of the client/server paradigm. A component in one of the middle tiers is client to the next lower tier and at the same time acts as server to the next higher tier.

Three-tier architecture:

1. **Presentation tier** This is the topmost level of the application. The presentation tier displays information related to such services as browsing. It communicates with other tiers by which it puts out the results to the browser/client tier and all other tiers in the network. In simple terms, it is a layer which users can access directly (such as a web page, or an operating system's GUI).
2. **Logic tier** The logical tier is pulled out from the presentation tier and, as its own layer, it controls an application's functionality by performing detailed processing.
3. **Data tier** The data tier includes the data persistence mechanisms (database servers, file shares, etc.) and the data access layer that encapsulates the persistence mechanisms and exposes the data. The data access layer should provide an API to the application tier that exposes methods of managing the stored data without exposing or creating dependencies on the data storage mechanisms.

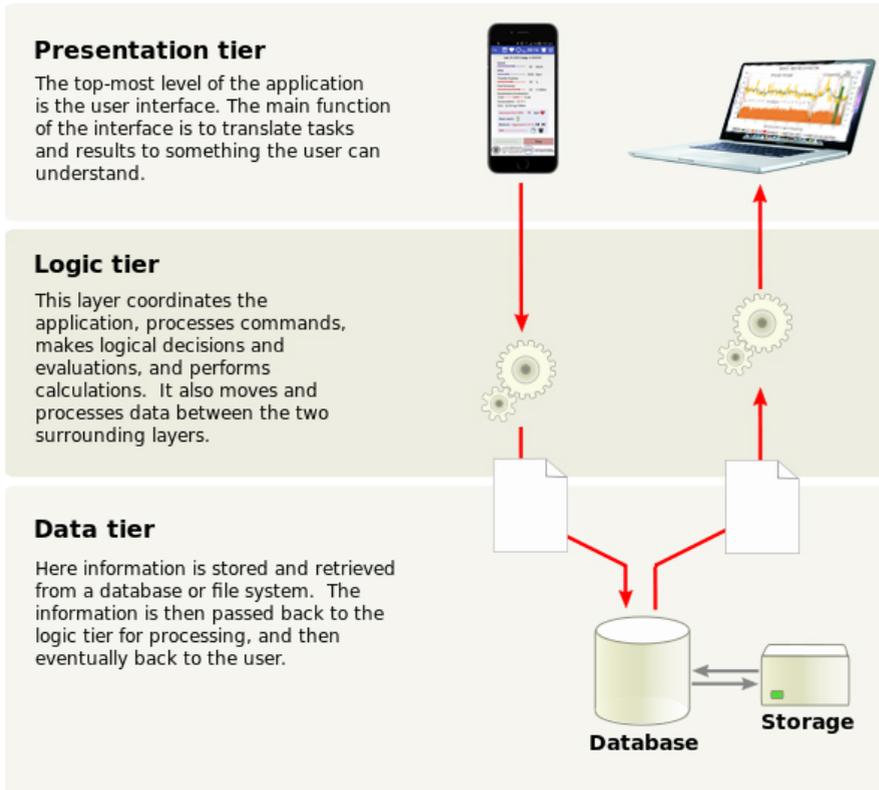


Figure 2.5: Overview of a three-tier application.

Chapter 3

DrivingStyles: A Smartphone Application to Assess Driver Behavior

Javier E. Meseguer, Carlos T. Calafate, Juan Carlos Cano,
Pietro Manzoni

DrivingStyles: A Smartphone Application to Assess Driver Behavior
18th IEEE Symposium on Computers and Communications
ISCC 2013 July 7-10, Split, Croatia.

Pages 535-540, ISBN: 978-1-4799-3755-4 CORE: B

Abstract

The DrivingStyles architecture integrates both *data mining* techniques and *neural networks* to generate a classification of driving styles by analyzing the driver behavior along each route. In particular, based on parameters such as speed, acceleration, and revolutions per minute of the engine (rpm), we have implemented a neural network based algorithm that is able to characterize the type of road on which the vehicle is moving, as well as the degree of aggressiveness of each driver. The final goal is to assist drivers at correcting the bad habits in their driving behavior, while offering helpful tips to improve fuel economy.

In this work we take advantage of two key-points: the evolution of mobile terminals and the availability of a standard interface to access car data. Our DrivingStyles platform to achieve a symbiosis between smartphones and vehicles able to make the former operate as an onboard unit. Results show that neural networks were able to achieve a high degree of exactitude at classifying both road and driver types based on user traces. DrivingStyles is currently available on the Google Play Store platform for free download, and has achieved more than 1550 downloads from different countries in just a few months.

3.1 Introduction

Mobile devices have experienced a technological breakthrough in recent years, evolving towards high performance terminals with multi-core microprocessors, being smartphones a clear representative exponent of this trend. In addition, the On Board Diagnostics (OBD-II) [1] standard, available since 1994, has recently become an enabling technology for in-vehicle applications due to the available of Bluetooth OBD-II connectors [9]. These connectors enable a transparent connectivity between the mobile device and the vehicle's Electronic Control Unit (ECU).

The range of possibilities that arise when combining cars and smartphones is endless, allowing, for example, diagnosing the car via mobile devices which assume the tasks that are typically performed by the On Board Unit (OBU) of the vehicle, or sending the collected data to a platform where diagnosis and vehicle maintenance can be done, detecting possible failures automatically.

Based on obtained data about the speed, acceleration, and revolutions per minute of the engine, we have implemented our DrivingStyles platform, which

is based on neural networks and permits characterizing the driving style of each user, as well as the type of road on which the vehicle is circulating. Currently, this information is being collected and used in applications aimed at improving road safety and promoting eco-driving [17], thus reducing fuel consumption and greenhouse gas emissions. The given recommendations addressing the driving style of each user can save up to 20 % of fuel while improving driving safety.

The rest of this paper is organized as follow: Section 3.2 briefly reviews previous related works. Then, Section 3.3 provides an overall overview of the DrivingStyles architecture. Both client and server side applications are described in more detail in sections 3.4 and 3.5, respectively. The neural network we have developed, along with the achieved accuracy results, are presented in section 3.6. Finally, section 3.7 presents the conclusions of our work.

3.2 Related Work

Technological achievements in the field of mobile are making smartphones very powerful devices. In the research world, this high computing power opens new and attractive research areas where many of them are able to generate economic profit and significant advantages for our society.

One of the main problems of eco-driving systems [17] is identifying the factors that affect energy consumption. Ericsson [18] suggests that, in order to save fuel, sudden changes in acceleration and high speed driving should be avoided. Johansson et al. [19] suggest maintaining low levels of deceleration, minimizing the use of the first and second gears, and putting every effort into using the 5th and 6th gears, while avoiding continuous gear changes.

There are several proposals that analyze which variables affect fuel consumption. Kuhler [20] presented a set of ten variables. These variables are used in laboratories that work with the consumption of fuel and gas emissions. Other authors, like André [21] and Fomunung [22], increased the number of parameters, or replaced some of them in order to improve the obtained results. The problem of all the previous proposals is that the environment in which the vehicle is circulating is not considered, despite it has a very significant influence on the energy consumption.

Regarding other works related to the proposal discussed in this chapter, it is worth highlighting the work by Chen et al. [23], who proposed an Android/OSGi vehicular platform able to diagnose and manage the status of a vehicle remotely, using visual intelligence to continuously update their appli-

cation services based on context, and without user intervention. Experiments conducted on a test vehicle showed that Android/OSGi applications are able to achieve higher performance compared to a pure Android development, especially when complex operations must be performed.

Focusing on road safety, systems such as e-NOTIFY [24] allow a fast detection of traffic accidents, improving the assistance to injured passengers by reducing the response time of emergency services. The proposed system requires installing OBUs on board vehicles that are in charge of detecting accidents, as well as notifying them to external control units (CU); the latter estimates the severity of the accident and automatically informs the emergency services. Zaldivar et al. [25] propose an Android-based application that monitors the vehicle through the On Board Diagnostics (OBD-II) interface, being able to detect accidents and sending details about the accident to pre-defined destinations through either e-mail or control units (SMS; these tasks are immediately followed by an automatic phone call to the emergency services.

Our solution differs from the former ones by providing an analysis of the driving style of each user in the scope of urban, suburban, or highway scenarios, which is based on neural network techniques, and promotes a more responsible and eco-driving behavior.

3.3 DrivingStyles Architecture

The proposed architecture applies data mining techniques to generate a classification of the driving styles of users based on the analysis of their mobility traces. Such classification is generated taking into consideration the characteristics of each route, such as whether it is urban, suburban or highway.

To achieve the overall objective, the system is structured around the following four elements, see Figure 3.1:

1. An application for Android based smartphones. Using an OBD-II Bluetooth interface, the application collects information such as speed, acceleration, engine revolutions per minute, throttle position, and the vehicle's geographic position. After gathering this information in real time, the user uploads the route data to the remote data center for analysis.
2. A data center with a web interface able to collect large data sets sent by different users concurrently, and to graphically display a summary of the

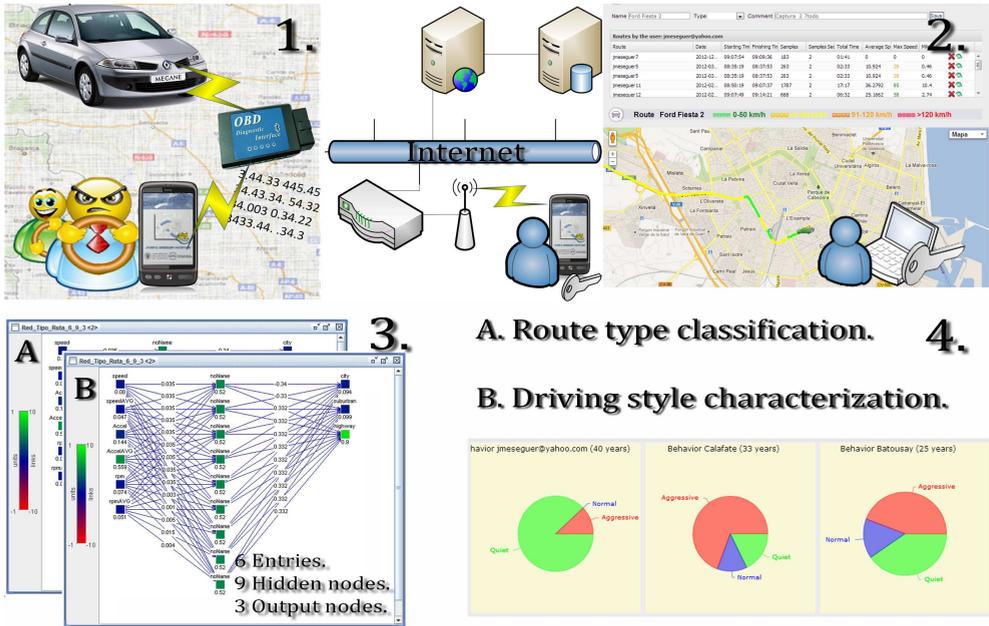


Figure 3.1: DrivingStyles: architecture overview.

most relevant results. Our solution is based on open source software tools such as Apache, PHP and Joomla®.

3. A neural network, which must be designed and trained using the most representative route traces in order to correctly identify, for each path segment, the driving style of the user, as well as identify the segment profile: urban, suburban or highway. To implement our neural network we adopted the backpropagation algorithm [14], which has been proven to provide good results in classification problems such as the one associated to this project.
4. Integration of the neural networks in the data center platform. The goal is to dynamically and automatically analyze user data, allowing users to find out their profiles as a driver, thus promoting a less aggressive and more ecological driving.

Figure 3.1 graphically shows the steps followed by users of the DrivingStyles platform to assess their driving styles. The first step is registering the user at

<http://www.drivingstyles.info>, and downloading the free Android application.

After installing the Android application in the smartphone, and after connecting to the bluetooth ELM327 [9] interface inside the car (this connector is mandatory on all vehicles since 2001), the data acquisition process can start. Once the route is completed and stored, the user can send it to the server, which checks the username and password before accepting the incoming XML file. Through the web interface, the user can view all the routes sent to the server, and have access to the different statistics, the map of the route, the characterization of the driving style, and the recommendations on eco-driving.

3.4 Android application

The Android application is a key element of the system, proving connectivity to the vehicle and to the DrivingStyles web platform. Currently, it can be downloaded for free from the DrivingStyles website ¹, or from Google Play ².

3.4.1 Configuration options

In order to adjust the functionality of our Android application to the user requirements, several configuration options must be defined related to user creation, connection options, GPS activation, and sensor sampling.

The available functionalities are the following:

1. **User creation:** A username and password matching those registered in the website must be introduced, allowing the server to validate the user credentials before storing any data in the database.
2. **OBD Connection:** Before starting the sampling process, the device must be paired with the OBD-II interface. Once the device is paired with the smartphone, it can be selected from the list of available devices within the application.
3. **GPS Activation:** although GPS data is not mandatory for data capturing purposes, it is necessary in the broader scope where the study is being done, since the variables retrieved from the vehicle are recorded along with the car's geographical position for later analysis.

¹<http://www.drivingstyles.info>

²<https://play.google.com/store/apps/details?id=com.driving.styles>

4. **Sensor sampling:** The user can individually select the vehicle sensors used by the application for monitoring and storage among the following ones: acceleration, engine revolutions per minute (rpm), throttle position, and instantaneous speed. Additionally, the user can define the sampling rate (number of samples per sensor, and per second).

3.4.2 Application Modules

The main module of our application launches the background processes responsible for capturing data sent by the OBD-II and the GPS interfaces, as well as the phone's accelerometer.

Besides showing the sensors we are monitoring, we can perform several parallel actions without affecting the data capture. Possible options are:

- Start and stop data capture of the Electronic Control Unit (ECU), the mobile's accelerometer, and the GPS.
- List the routes captured by the app.
- Driving style display.
- Show the current position of the vehicle on the map, as well as the detailed route followed whenever an Internet connection is available.
- Real time visualization of the speed, rpm, and acceleration in a time window of 10 seconds.

The route upload module is in charge of sending the users' traces to the website data center for further analysis. This module can be accessed either from the historic stored routes, or immediately after stopping the data capture. The information screen displays the header information of the selected route such as: date of the captured data, start time, finish time, and maximum speed.

This module includes a graphical interface in charge of showing routes on a map, as well as the collected statistics. Additionally, it also includes communication facilities for uploading the collected routes to the data center.

Finally, the file received in the DrivingStyles data center is stored in the corresponding user directory, and also generates a record in the database for every sample submitted for analysis.

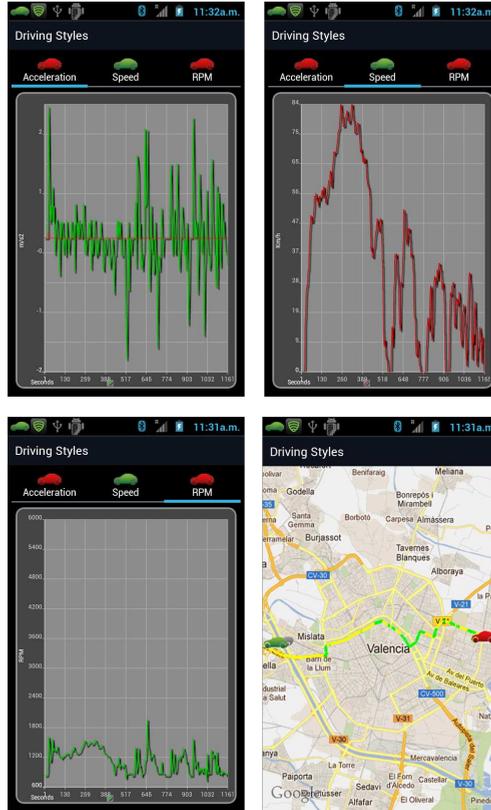


Figure 3.2: DrivingStyles: architecture overview Snapshots of the acceleration, speed, rpm parameters and map module.

The Map Module and Graphical Information are in charge of displaying the relevant information to the user in the most convenient manner. The graphs can be displayed in real-time or by selecting data from previously stored paths. Depending on the device model, the user can also zoom in and out to display all or part of the graph using the device’s touch screen.

The charts that appear on the screen are the acceleration, the speed, and the revolutions per minute (rpm). We have chosen these three parameters since they are the most relevant ones, and because they are the ones we selected for training our neural network.

The map module allows displaying the GPS position on the map. GPS coordinates are drawn using the Google Maps APIs. A green car icon indicates

the beginning of the route, and a red icon shows the current position of the vehicle. The path is shown by using different colors depending on the vehicle's speed (see Figure 3.2).

3.5 DrivingStyles Web interface

The second main component of our architecture corresponds to the data center and its web interface. For this endeavor we have selected open source software such as Apache HTTP, and Joomla!® as the content management system (CMS). The URL of this module is <http://www.drivingstyles.info>.

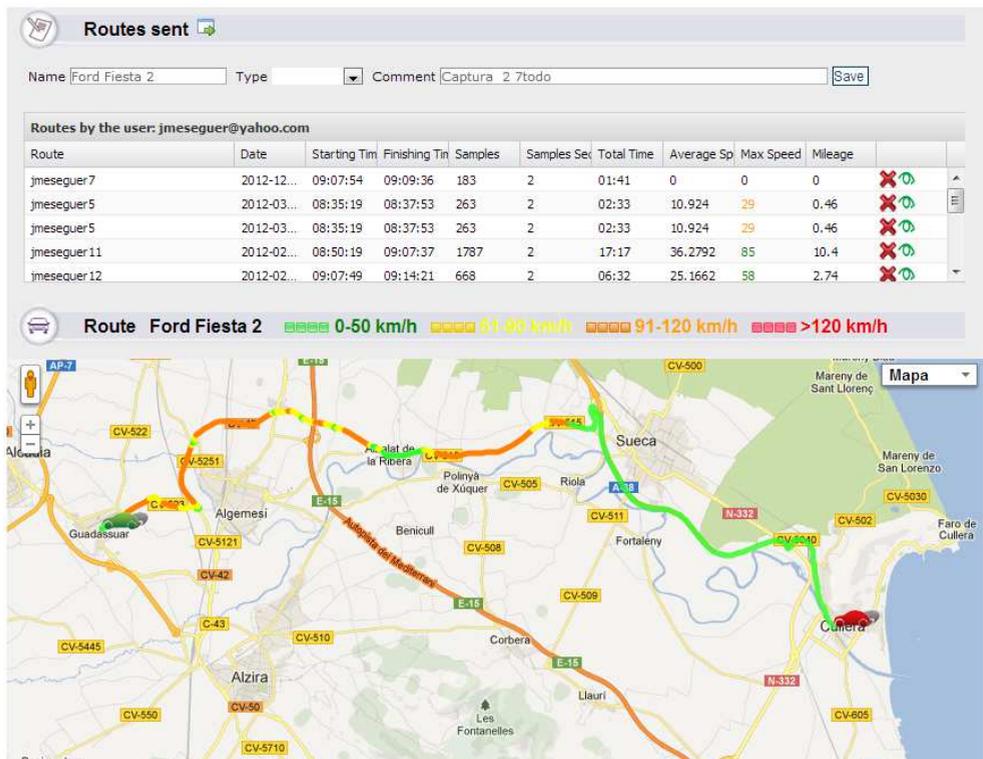


Figure 3.3: Snapshot of a route map.

Once the user is logged in, he is asked to record a number of important data, especially for future data mining studies. The most relevant items are sex, age, and other details concerning the vehicle used: car manufacturer, model, fuel type, and the theoretical 0-100 acceleration (important to normalize the

user behavior in our study). Finally, a third block allows drivers to indicate what they feel about their own behavior behind the wheel, i.e., whether they perceive themselves as aggressive, moderate, or quiet drivers.

In the Routes' section, the users can access all the routes they have uploaded. The first grid shows routes that are in the database, including the name, date, starting time, ending time, samples sent, total time, average speed, and kilometers travelled. Below the grid, the selected route is shown in a map. The path varies its color depending on the speed of the car (see Figure 3.3).

In addition, the last two graphs show the results returned by the neural network, including the driving style and the route characteristics (see Figures 3.6(b), and 3.6(a)).

Next, in section 3.6 we provide detailed information about the neural network we proposed for characterizing routes and drivers.

3.6 Neural Networks based data analysis

In this project we face a classification problem: starting from some input data, which in our case are the speed, the acceleration, and the revolutions per minute of the engine (RPM), we obtain as output the type of road and the driving style.

We train Artificial Neural Networks to classify the driving style of each user and the type of route based on a well-defined set of rules and the ECU input variables. There are many different learning algorithms, such as back-prop_momentum, Hebbian, or delta-rule, each one having its own advantages and disadvantages depending on the type of problem to solve [26]. In the scope of this work, we decided to choose backpropagation [14] since this kind of algorithm provides very good results in classification problems.

After considering the many variables that can be obtained from the Electronic Control Unit (ECU), we have chosen to train the neural network using the mean and the standard deviation values of: a) the speed, b) the vehicle acceleration, and c) the rpm. In all vehicles used for testing, these variables were easily obtained, while other variables, such as the position of the throttle, and despite being able to provide important information for the neural network training, finally had to be rejected because not all ECU manufacturers provide such information. The data input of each parameter is normalized between 0 and 1;

this normalization should take into consideration the whole range of possible values. The equation used to normalize the input parameters is the following:

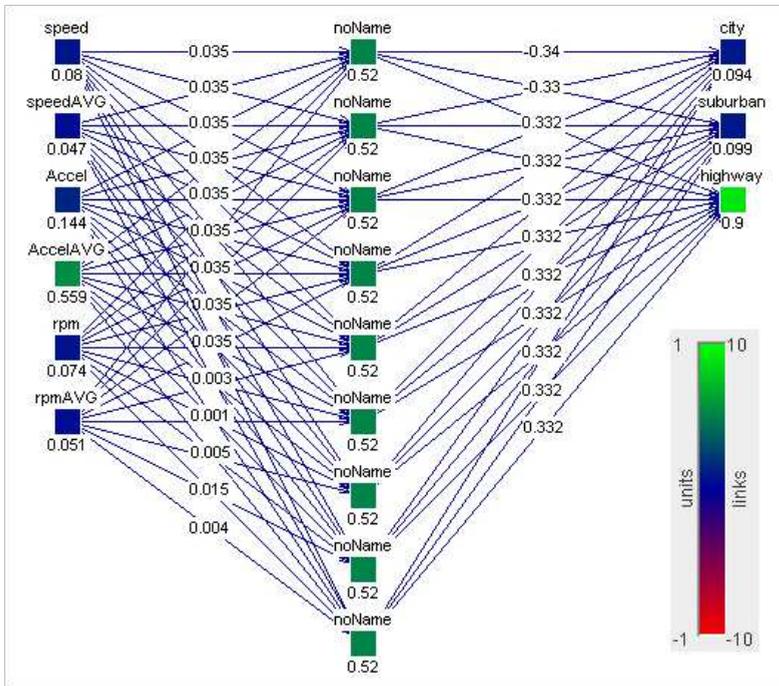
$$x^{p'} = \frac{x^p - \min(x)}{\max(x) - \min(x)} \quad (3.1)$$

Notice that $x^{p'}$ is the normalized value of input variable x for pattern p , x^p is the original value of variable x for that pattern, and $\min(x)$ and $\max(x)$ represent the minimum and maximum values for the input variable, i.e., the minimum and maximum values of the column corresponding to variable x .

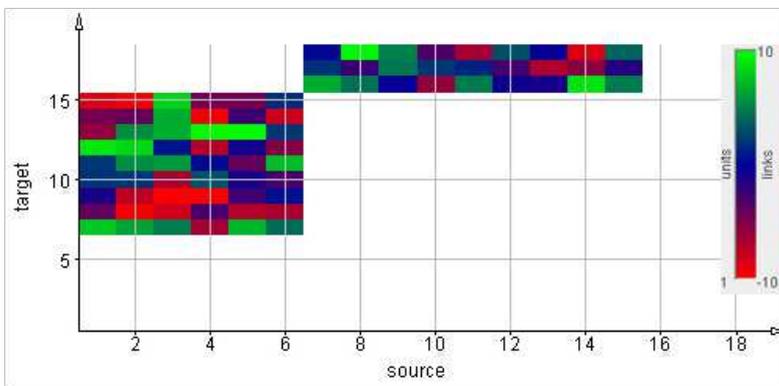
The application used for the creation and training of the neural networks required by this project was JavaNNS [15], which is a Java version of the SNNS software developed at the University of Tübingen.

3.6.1 Neural Network description

An empty neural network was created by defining the number of entries mentioned previously, along with the number of hidden nodes (nine in our case). A larger number of hidden nodes can improve the success rate, but it has the negative effect of increasing the response time. Two similar neural networks were created, each one with three output nodes; the first neural network (see Figure 3.4(a)) allows to classify the type of road (urban, suburban or highway), and the second permits to characterize the user's driving style (quiet, normal or aggressive). Random weights between -1 and 1 are assigned to the links of both neural networks during initialization. Figure 3.4(b) shows the equivalence color map with their weight values for the neural network identifying the road type (first one).



(a) Trained neural network.



(b) Weights of the links of the neural network.

Figure 3.4: Trained neural network and associated weights.

3.6.2 Training the Neural Network

After normalizing the input variables, the training set was generated. This data set is used to adjust the weights of the neural network. After the neural network is trained, we generate validation files. This data set is used to minimize overfitting, verifying that any increase in accuracy over the training data set actually yields an increase in accuracy over a broader data set that has not been used for training.

In order to do so, we rely on the average error for the test set: $E = \frac{\sum_{p=1}^m |d^p - y^p|}{m}$ where $|d^p - y^p|$ is the absolute error for pattern p , i.e., the absolute value of the difference between the desired output and the output obtained, m is the number of patterns, and the sum corresponds to the sum of absolute errors for all patterns.

We started training the neural network by adjusting the learning rate to 0.2 and observing how the error affects our neural network. The higher the learning rate, the higher the modification of the weights for each iteration, so the learning process becomes fast. However, this can cause unwanted oscillations in the network. The convergence results of the two neural networks are the following:

1. Road type characterization: By iterating with a learning rate of 0.2, we see how the learning process slows down for a local minimum MSE with an error of 0.021. The network starts converging quickly again beyond 300 cycles, as shown is the mean square error shown in Figure 3.5(a).
2. Driving style characterization: This characterization is more complex than the road type, and no local minimum is reached as quickly as in the previous case; also, the error of the neural network is much higher (see Figure 3.5(b)). At the end of these tests we chose to use the trained neural network with a learning rate of 0.4, which achieves the lowest error among all the tested values, with a mean square error of 0.43 and a sum of squared errors of 6.11 (SSE).

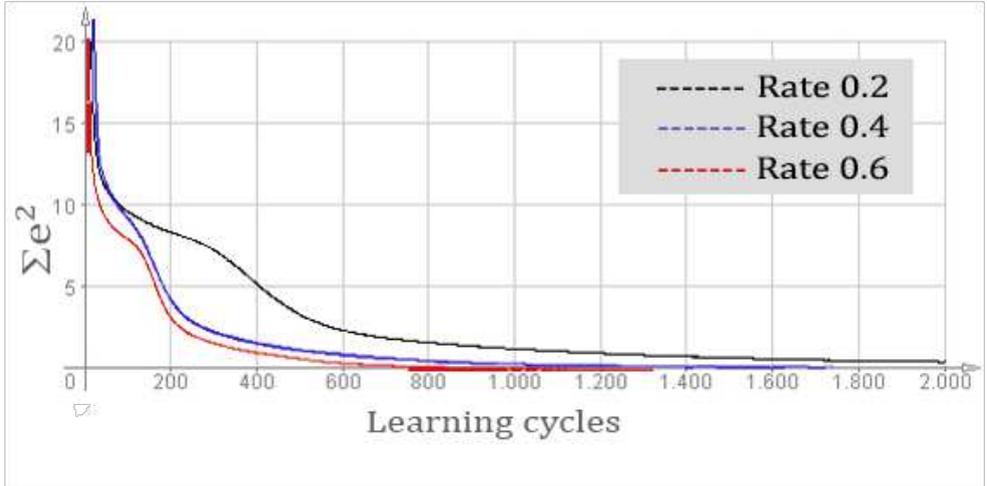
Once the neural network is successfully trained (see Figure 3.4(a)), the knowledge obtained must be converted into usable code. We use the `snns2.exe` application [16], to convert the trained neural network into C code, and this code was then integrated in to the web platform for data classification purposes.

3.6.3 *Obtained results*

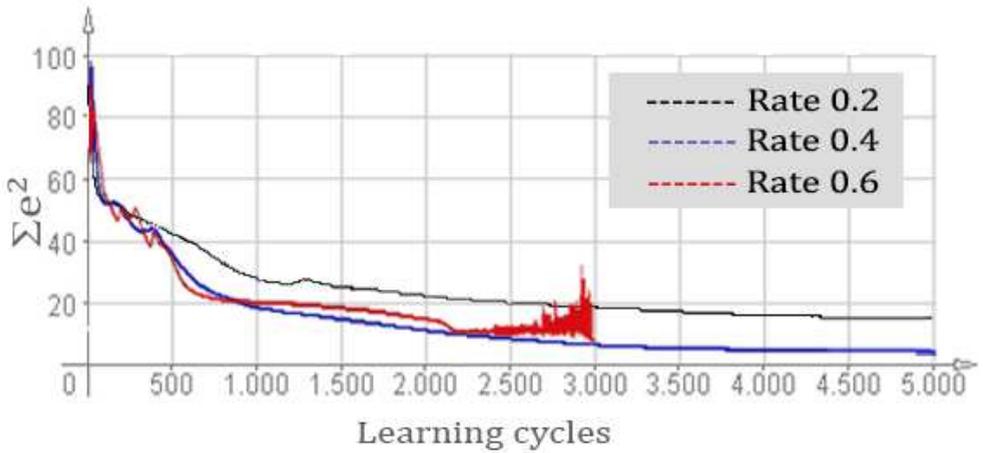
With the neural network already implemented, every time a route or route segment is selected, the system automatically returns the type of road, and the associated driving style. The overall behavior of each user can also be obtained by evaluating all the routes sent by the user.

The results obtained from a real set of data can be seen in Figure 3.6. Figure 3.6(a) graphically shows the result returned by our neural network when determining the route type. The abscissa axis corresponds to the timeline, and the vertical axis to the degree of matching for the dominant route type. We can see that, in this example, the user starts and ends in an urban environment, going through a suburban route in between; such results represent an accurate classification of the actual route followed by the driver. Figure 3.6(b) refers to the neural network results that determine the behavior on that route; as in the previous graph, the abscissa axis corresponds to the temporal line, and the ordinate axis is a behavior correlation percentage, where each color identifies a different type of driver behavior. We can observe that most of the time the driver is showing an aggressive behavior, which alternates with periods of quiet behavior. When analyzing the actual route with the driver, we find that periods of quiet behavior were associated with slow transit areas where vehicle queueing and semaphores impeded a more aggressive driving style.

Overall, and after analyzing a wide set of routes with different drivers, the system has shown to provide a correct classification of the different route traces registered in the database, both in terms of route types and driving styles, validating the proposed solution.

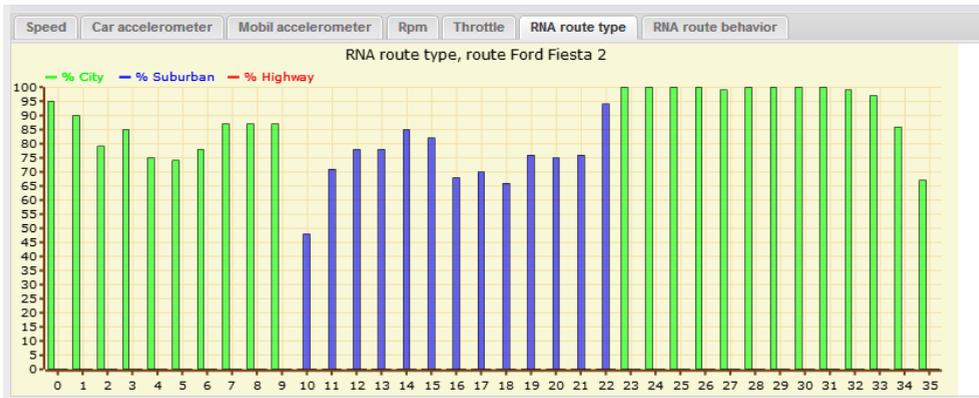


(a) Convergence when training the road type with 2000 cycles.

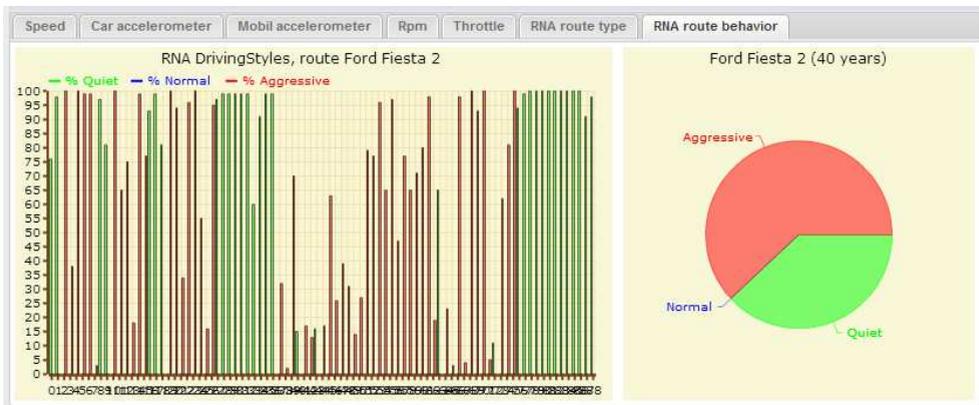


(b) Convergence when training the driving style with 5000 cycles.

Figure 3.5: Neural networks convergence behavior



(a) Route type.



(b) Route behavior.

Figure 3.6: Snapshots of route type and behavior.

3.7 Conclusions and future work

In this chapter we presented the DrivingStyles platform, which integrates mobile devices with data obtained from the Electronic Control Unit (ECU) to determine the type of road where the driver is circulating, as well as his driving habits. Its main goal is to help promoting a safer and more ecological driving style by making drivers more conscious about their behavior on the road. Our platform offers helpful tips to reduce fuel consumption with the consequent impact on the environment. We implemented this platform using real devices, and the results we obtained based on real user traces are quite encouraging, showing that the classification of both routes and driving styles using neural networks presents a high correlation with the actual routes and driver behavior.

The application, which is available for free download in the DrivingStyle's website³ and in the Google Play Store⁴, has achieved more than 1550 downloads from different countries in just a few months. This shows the great interest for applications integrating smartphones with vehicles.

As future work we plan to monitor the fuel consumption in order to correlate it with the driver aggressiveness so making the driver more aware of their bad habits while driving. We are also extending the platform to provide traffic recommendations based on real-time feedback about the congestion of different routes.

3.8 Acknowledgments

This work was partially supported by the Ministerio de Ciencia e Innovación, Spain, under Grant TIN2011-27543- C03-01.

³<http://www.drivingstyles.info>

⁴<https://play.google.com/store/apps/details?id=com.driving.styles>

Chapter 4

Assessing the Impact of Driving Behavior on Instantaneous Fuel Consumption

Javier E. Meseguer, Carlos T. Calafate, Juan Carlos Cano,
Pietro Manzoni

Assessing the Impact of Driving Behavior on Instantaneous Fuel
Consumption

12th IEEE Consumer Communications and Networking Conference

CCNC 2015 January 9-12, Las Vegas, Nevada USA.

Pages 443-448, ISBN: 978-1-4799-6390-4 CORE: B

Abstract

Despite the recent technological improvements in vehicles and engines, and the introduction of better fuels, road transportation is still responsible for air pollution in urban areas due to the increasing number of circulating vehicles, and their relative travelled distances. We develop a methodology to calculate, in real-time, the consumption and environmental impact of spark ignition and diesel vehicles from a set of variables such as Engine Fuel Rate, Speed, Mass Air Flow, Absolute Load, and Manifold Absolute Pressure, all of them obtained from the vehicle's Electronic Control Unit (ECU). Our platform is able to assist drivers in correcting their bad driving habits, while offering helpful recommendations to improve fuel economy. In this chapter we will demonstrate through data mining, to what extent does the driving style really affect (negatively or positively) the fuel consumption, as well as the increase or reduction of greenhouse gas emissions generated by vehicles.

4.1 Introduction

The increasing cost of fuel and the environmental pollution caused by greenhouse gas emissions has encouraged the research of more energy efficient vehicles. At the same time, mobile platforms have evolved according to user requirements, and have been applied in different fields of industry such as the automotive, where smartphones are starting to be integrated in vehicles; this clearly opens a new and exciting area of research.

The On Board Diagnostics (OBD-II) [7] standard has recently become an enabling technology for in-vehicle applications due to the availability of Bluetooth OBD-II connectors [9], which enables transparent connectivity between mobile devices and the vehicle's Electronic Control Unit (ECU).

Among the different strategies to reduce fuel consumption and greenhouse gas emissions, the DrivingStyles' platform [27] aims to sensitize drivers about their driving style, making them aware that the driving style is directly related to fuel consumption and gas emissions.

DrivingStyles implements a solution based on neural networks which is capable of characterizing the type of road on which the vehicle is circulating, as well as the driving style of each user [27]. In order to achieve this, the data is obtained from the ECU via the OBD-II Bluetooth interface, including the speed, acceleration, and revolutions per minute of the engine.

In this chapter we improve the DrivingStyles' platform by calculating the instantaneous fuel consumption; in this platform, the following variables are obtained in real time: Mass Flow sensor (MAF), manifold absolute pressure (MAP), and Intake Air Temperature (AIT). Currently, this information is being collected and used in applications aimed at improving road safety and to promote eco-driving [17], thus reducing fuel consumption and greenhouse gas emissions. Specifically we find that, by shifting towards a more efficient driving style, users can save up to 20 % of fuel while improving driving safety [28, 29].

This paper is organized as follows: in the next section we present some related works. Section 4.3 introduces the DrivingStyles architecture. Both client and server side applications are described in more detail in section 4.3. The calculations consumption and CO_2 emissions are presented in sections 4.4 and 4.5, respectively. Section 4.6 presents the performance results we obtained. Finally, section 4.7 concludes our work.

4.2 Related Work

One of the main problems of eco-driving systems [17] is identifying the factors that affect energy consumption. Ericsson [18] suggests that, in order to save fuel, sudden changes in acceleration and high speed driving should be avoided. Johansson et al. [19] suggest maintaining low levels of deceleration, minimizing the use of the first and second gears, and using, when ever possible, the 5th and 6th gears, while avoiding continuous gear shifts. There are several proposals that analyze which variables affect fuel consumption. Kuhler [20] introduced a set of ten variables that are used in laboratories for fuel consumption and vehicle emissions. Other authors such as André [21] and Fomunung [22] improve these results by increasing and replacing some of the parameters.

In previous works such as D.Y.C. Leung [30] and COPERT III [31], different tools were developed to collect in real time the engine and vehicle parameters from the OBD connector. Moreover, in conjunction with an exhaust analyzer Horiba OBS [32], a set of consumption and emission models were developed for vehicles equipped with spark ignition engines. Several commercial OBD-II scanner tools are available, that can read and record these sensor values. Apart from such scanners, remote diagnostic systems such as GM's OnStar, BMW's ConnectedDrive, and Lexus Link [33, 34] are capable of monitoring engine parameters from a remote location.

Our solution differs from all the previous ones by providing an analysis of the driving style for each user in the scope of eco-driving behavior based on

neural network techniques. By calculating the consumption and greenhouse gas emissions generated by both types of engines (spark ignition and diesel vehicles), we are able to closely relate both results, detailing the fuel savings achieved by soft driving patterns compared to aggressive ones.

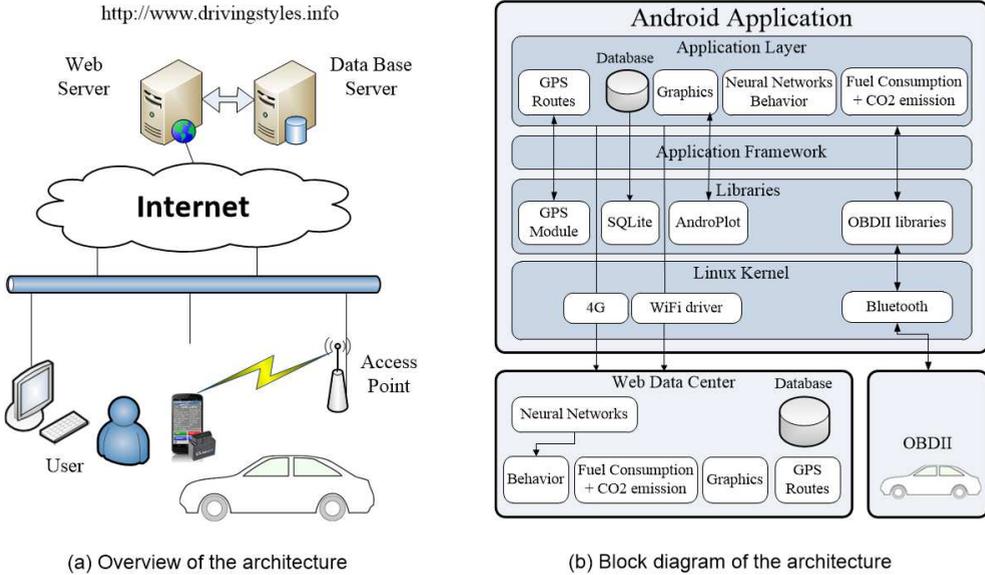


Figure 4.1: System architecture of DrivingStyles.

4.3 Overview of the DrivingStyles Architecture

The DrivingStyles architecture applies data mining techniques (process of discovering patterns in large data sets involving, methods of artificial intelligence, machine learning, statistics, and database systems) to generate a classification of the driving styles of users based on the analysis of their mobility traces. Such classification is generated taking into consideration the characteristics of each route, such as whether it is urban, suburban, or highway.

Figure 4.1 shows the system architecture which comprises four elements:

1. An application for Android based smartphones (see Figure 4.1.b). (i) Using an OBD-II Bluetooth interface, the application collects control information such as speed, acceleration, engine revolutions per minute, throttle position, and vehicle’s geographic position (obtained from GPS

mobile). (ii) In addition, we also obtain via OBD-II the Mass Flow sensor (MAF), Manifold Absolute Pressure (MAP), and the Intake Air Temperature (AIT) used in the calculation of fuel consumption. (iii) After gathering the information, the user uploads the route data to the remote data center for analysis.

2. A data center with a web interface to collect large data sets sent by different users concurrently, and to graphically display a summary of the most relevant results including the fuel efficiency. Our solution is based on open source software tools such as Apache, PHP and Joomla®.
3. A neural network, which must be trained using the most representative route traces in order to correctly identify, for each path segment, the driver's style, as well as the segment profile: urban, suburban or highway. We use the backpropagation algorithm [14], which has proven to provide good results in classification problems such as the one associated to this project.
4. Integration of the tuned neural networks in the data center platform. The goal is to use neural networks to dynamically and automatically analyze user data, allowing users to find out their profiles as a driver, as well as their fuel consumption (which is related to their driving behavior), thus promoting a less aggressive and more ecological driving (see Figure 4.2).

4.3.1 Android Application and Web Interface

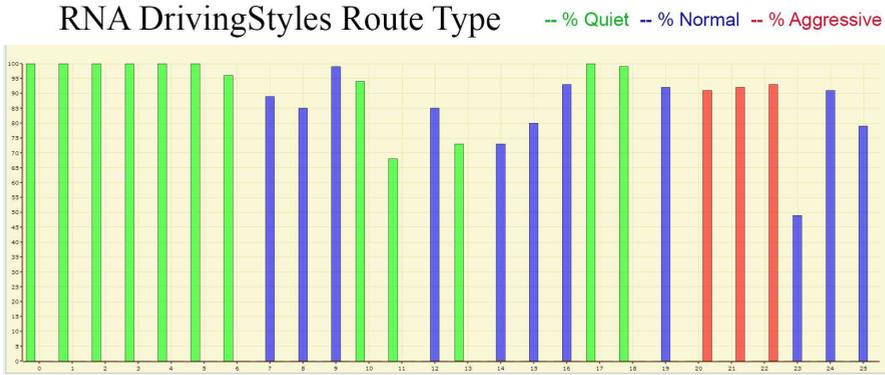
The Android application is a key element of our system, proving connectivity to the vehicle and to the DrivingStyles web platform. Currently, it can be downloaded for free from the DrivingStyles website <http://www.drivingstyles.info>, or from Google Play¹ (more than 4500 downloads).

The available functionalities are: (i) User creation, (ii) Connection options, (iii) GPS Activation, (iv) Sensor sampling.

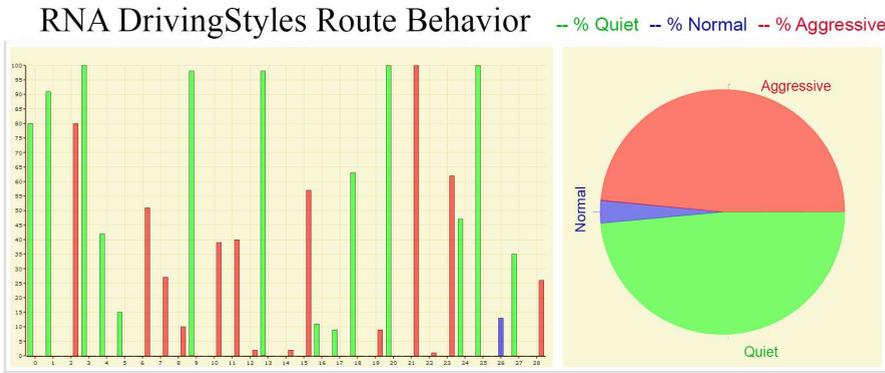
Our application captures data sent by the OBD-II and the GPS interfaces, as well as the phone's accelerometer (see Figure 4.3).

Besides showing the sensors that we are monitoring, we can perform several parallel actions without affecting the data captured.

¹<https://play.google.com/store/apps/details?id=com.driving.styles>



(a) Urban routes.



(b) Suburban routes.

Figure 4.2: Snapshots of route type and behavior.

The route upload module is in charge of sending the users’ traces to the website data center for further analysis. This module can be accessed either from the historic stored routes, or immediately after stopping the data capture. The information screen displays the header information of the selected route, such as: (i) date of the captured data, (ii) start time, (iii) finish time, (iv) maximum speed and (v) fuel consumption.

Our application includes a graphical interface for showing the routes on a map, as well as the collected statistics. Additionally, it also includes communication facilities for uploading the collected routes to the data center.



Figure 4.3: Snapshots of the main screen and the data sending module.

The second main component of our architecture corresponds to the data center and its web interface. For this, we have selected open source software such as Apache HTTP, and Joomla® as the content management system (CMS). We have used a CMS, combined with the use of a resource wrapper, which detaches our system from the presentation layer, thus focusing on the problem of driving styles characterization and the influence of the latter fuel consumption. The URL of this module is <http://www.drivingstyles.info>.

Next, we present our fuel consumption estimation approach relating it with the driver style as captured by the DrivingStyles platform.

4.4 Fuel Consumption / Instantaneous Fuel Consumption Calculation

Fuel consumption is usually represented as the ratio of fuel consumed per distance travelled, being measured in terms of litres per 100 kilometres (or alternatively as MPG-Miles Per Gallon). In this work we focus on petrol and diesel engines. Although the basic designs of gasoline and diesel engines are similar, the mechanics are different. A gasoline engine compresses its fuel and air charge, and then initiates combustion by the use of a spark plug. A diesel engine just compresses air until the combustion chamber reaches a temperature for self-ignition to occur. So, at a given speed in kilometres per hour, fuel consumption can be calculated as follows:

$$\text{Fuel Consump. [l/100km]} = \frac{\text{Fuel Flow [l/h]}}{\text{Speed [km/h]}} \cdot 100 \quad (4.1)$$

Instantaneous fuel economy/consumption is calculated from the current fuel flow and the current vehicle speed. It can be only be calculated when the vehicle is moving and the engine is operating.

$$\text{Instanta. Fuel Consump. [l/km]} = \frac{\text{Fuel Flow [l]}}{\text{Speed [km]}} \quad (4.2)$$

Again, not all vehicles support all the OBD PIDs, and there are usually many manufacturer-defined custom PIDs that are not defined in the OBD-II standard. So, the OBD standard does not provide a fuel consumption parameter; instead, it provides other values that enable its calculation. Depending on the variables that the ECU can supply, the mathematical procedure to determine fuel consumption is different (see Figure 4.4).

1. By combining the Engine Fuel Rate (PID 015E), also known as Fuel Flow (litres/hour), and Speed (PID 010D), it is easy to calculate fuel consumption. However, while speed is mandatorily available, fuel rate is not. In fact, it was unavailable in all the vehicles we used to carry out our tests. This can be due to two reasons: (i) the manufacturer chooses not to make it available, or (ii) there is no sensor inserted in the fuel line between the fuel tank and the engine carburetor to measure litres per hour.

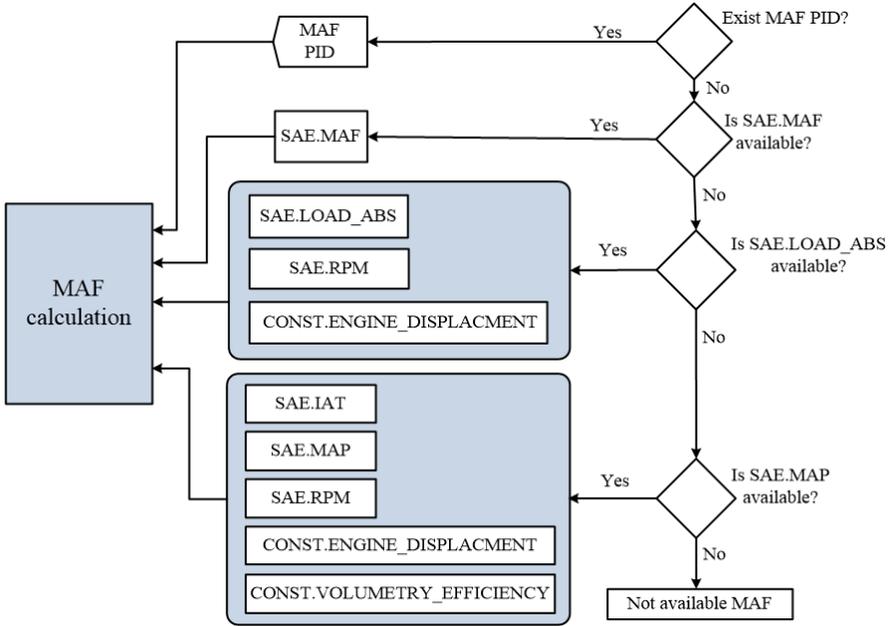


Figure 4.4: Scheme of the different possibilities of MAF calculation.

- If the MAF PID is available, but the Engine Fuel Rate is not, we can calculate fuel rate as Fuel Flow (litres/hour) by dividing the Mass Air Flow (PID 0110) · 3600 sec. by the product of air-to-fuel ratio and Fuel Density (see Table 4.1):

$$Fuel\ Flow\ [l/h] = (MAF \cdot 3600) / AFR_A \cdot FD \quad (4.3)$$

where MAF refers to Mass Air Flow (g/s), AFR_A to the actual Air-to-Fuel Ratio (see Table 4.1), and FD - Fuel Density (g/l, see Table 4.1), allowing us to directly calculate fuel consumption.

- If MAF is not available there are two ways to calculate it (see Figure 4.4).
 - As a function of absolute load (PID 0143), RPM (PID 010C) and Engine Displacement (EngDisp, volume of an engine's cylinders in cm^3 , intake stroke is the fluid admission phase of a reciprocating cylinder). The equation is:

$$\begin{aligned}
 MAF [g/s] = & 1.184 [g/l] \cdot EngDisp [l/intakestroke] \\
 & \cdot load\ abs/100 \cdot enginespeed [rpm] \\
 & /2 \cdot [rpm/intakestroke] /60 [sec/min]
 \end{aligned}
 \tag{4.4}$$

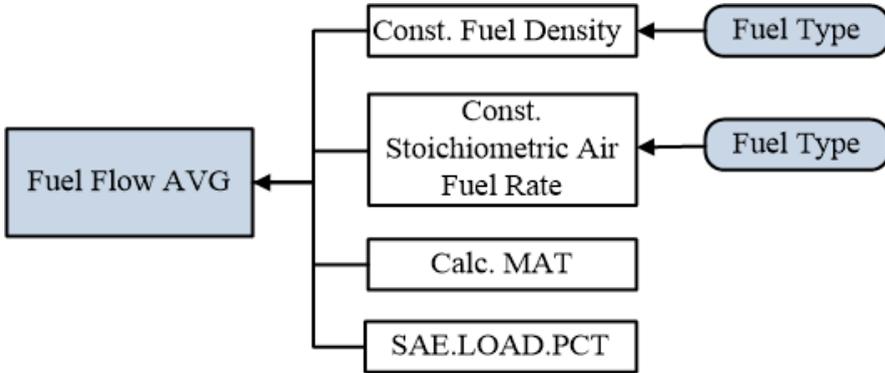


Figure 4.5: Outline of the calculation of the Fuel Flow average.

- As a function of the Intake Manifold Pressure (PID 010B), RPM (PID 010C), Intake Air Temperature (PID 010F) and Engine Displacement. A synthetic variable called IMAP can be used to estimate the Mass Air Flow (MAF) of an internal combustion vehicle, much like a MAF sensor. In order to make this calculation, the engine displacement and volumetric efficiency of the engine must be provided.

$$IMAP = RPM \cdot MAP/IAT/2
 \tag{4.5}$$

Where RPM (PID 010C) is the engine speed in RPM (PID 010C), MAP (PID 010B) is the Manifold Absolute Pressure measured in kilopascal kPa, and IAT (PID 010F) is the Intake Air Temperature measured in degrees Kelvin. This integrated value can be converted into total air flow (grams) using the following formula:

$$\begin{aligned}
 MAF [g/s] = & (IMAP/60) \cdot (VolEff/100) \\
 & \cdot EngDisp \cdot MMAir/R
 \end{aligned}
 \tag{4.6}$$

Where VolEff is Volume Efficiency (which relates the actual and the theoretical volumetric flow rate in %), EngDisp is the Engine Displacement

Table 4.1: Ideal air/fuel ratio (*grams of air to 1 gram of fuel*) – *Density(g/dm³)*.

Fuel Type	Ratio by mass	Density g/dm^3
Gasoline	14.7 : 1	820
Diesel	14.5 : 1	750

(the volume of an engine's cylinders in cm^3) and R is $8.314 J/^\circ K/mole$. These parameters are used to formulate the equation in order to obtain reliable OBD data, which is then compared to the data given by the vehicle manufacturer to ensure its accuracy.

The Air Fuel Flow can then be calculated as follows: (see Figure 4.5).

$$Fuel\ Flow\ [l/h] = (MAF \cdot 3600)/AFR_A \cdot FD \quad (4.7)$$

4.5 Greenhouse Gas Emissions Calculation

The most significant greenhouse gases are generated from direct combustion carbon dioxide CO_2 , Methane (CH_4), and Nitrous oxide (N_2O), among others. CO_2 is always generated when burning fuel that contains carbon. Since the carbon in the fuel is combined with the oxygen in the air: $C + O_2 \rightarrow CO_2$, the amount of CO_2 can be calculated by the atomic masses of carbon and oxygen and the carbon content of the fuel. The atomic mass of carbon is 12_U and oxygen is 16_U , meaning that $CO_2 = 12_U + 2 \cdot 16_U = 44_U$. Burning 1 kg of carbon produces $44/12 \approx 3,67kg$ of CO_2 in complete combustion, and so the CO_2 emission of combustion is $3,67 \cdot C_c \cdot m_{fuel}$ where $C_c = fuel\ carbon\ content\ (mass\ basis)$. Considering that the carbon content of diesel fuel is 85,7 % the CO_2 emission when burning 1 kg ($m_{fuel} = 1kg$) of diesel fuel is:

$$\begin{aligned}
 m_{CO_2} &= 3.67 \cdot C_c \cdot m_{fuel} \\
 m_{CO_2} &= 3.67 \cdot 0,857 \cdot 1 [kg] = 3.15 [kg/1kg\ fuel] \\
 &\quad \text{Density of diesel fuel is } 0.84 [kg/l] \\
 m_{CO_2} &= 3.15 [kg] \cdot 0.84 = 2.64 [kg/1l\ fuel] \quad (4.8)
 \end{aligned}$$

Driving in a fuel-efficient manner can save fuel, money, and reduce greenhouse gas emissions. Among the factors that can affect fuel consumption, such as:

vehicle age and condition, outside temperature, weather, and traffic conditions, we consider that driver behavior can be the most relevant parameter.

4.6 Experimental Results and Evaluation

In our project, we focus on measuring fuel consumption variations associated with different driver behaviors. In order to achieve this objective, we rely on the collaboration of 264 drivers from around the world using our platform, including countries like India, Brazil, Central America, and Europe [35]. In this particular study, we analyzed the behavior of 34 representative routes (each divided into 20 second periods) using the neural network described earlier. For each section, the neural network returns the corresponding driver behavior, and we combine this data with the fuel consumption data corresponding to that route.

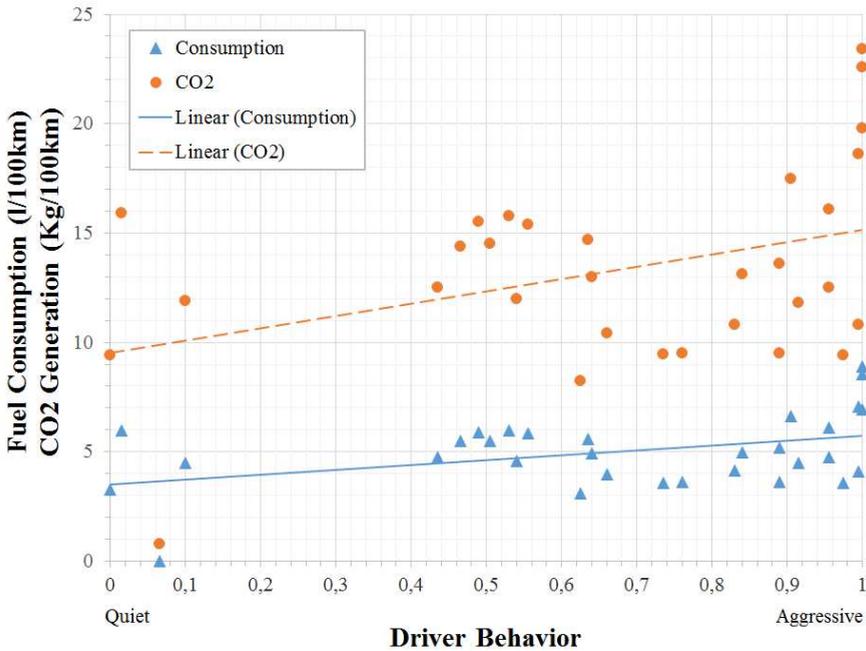


Figure 4.6: Chart of consumption and CO_2 in relation to the driving behavior.

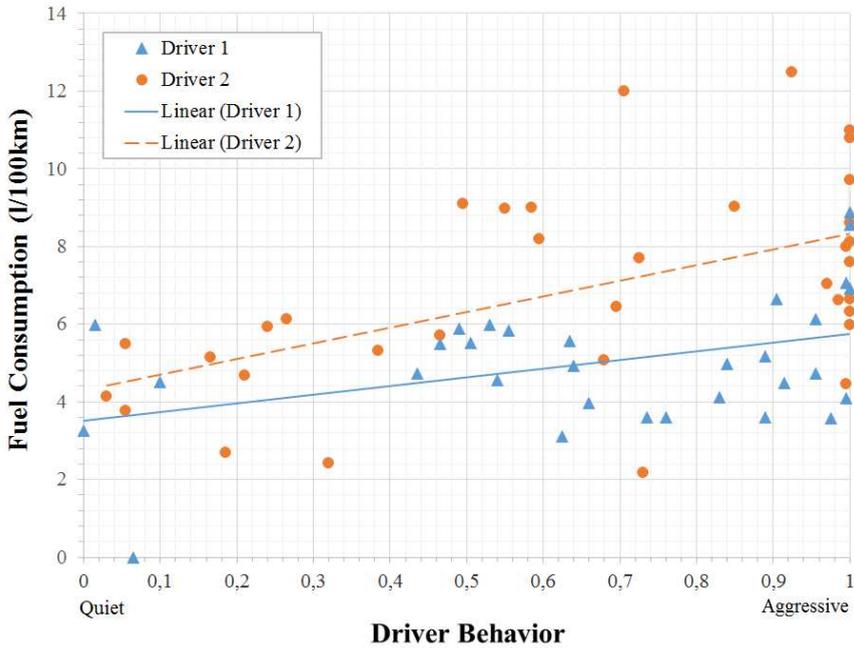


Figure 4.7: Comparative chart between two drivers and their behavior.

We carried out several types of tests to validate our proposals. The first requires a same person to drive a single vehicle model, and then analyzing all the behaviors recorded during driving, as well as the instantaneous consumption. The results of this test have been very significant, as shown in Figure 4.6. Notice that more aggressive driving behaviors cause consumption to increase significantly, liters of fuel, likewise increasing the generation of CO_2 .

Figure 4.8 displays the differences between quiet, normal, and aggressive driving behavior in terms of fuel consumption; aggressive drivers provoke fast starts and quick accelerations, driving at high engine revolutions, and causing sudden speed changes. Conversely, a quiet driving behavior would be smooth, without sudden speed changes or continuous gear shifts. It is clear that fuel consumption increases when the driver behavior becomes more aggressive, with average differences of up to 1,5 liters per 100 km. Regarding CO_2 emissions, they increase by 50%, going from 10 to 15 Kg/100km, depending on whether you are a quiet or an aggressive driver, as shown in Figure 4.6.

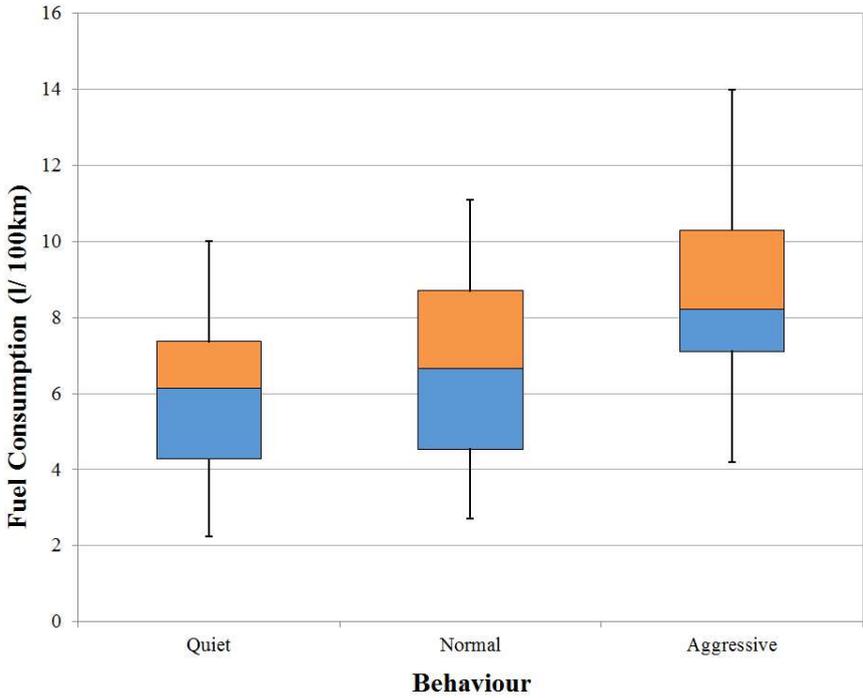


Figure 4.8: Box and whisker plot of Fuel Consumption / Driving Behavior.

The second battery of tests was performed using the same vehicle but with different drivers. Figure 4.7 shows the behavior of two drivers driving the same vehicle. We can see that the behavior of both drivers is clearly different, being that the second driver clearly has a more aggressive behavior, and so consumption is noticeably higher compared to the other driver.

As a result, we conclude that aggressive driving behaviors, besides being dangerous and not recommended, can be economically expensive and ecologically harmful. A trend towards aggressive driving behavior may actually increase fuel costs by more than 20 %, not to mention that additional CO_2 emissions to the environment could be avoided.

4.7 Conclusions and future work

Reducing energy consumption and greenhouse gas emissions requires drivers to adopt an efficient driving style. This paper presents our DrivingStyles platform, which integrates mobile devices with data obtained from the Electronic Control Unit (ECU) to determine the type of road where the driver is traveling, as well as his driving habits. Using only an Android Smartphone and an OBD-II adapter, it is possible for a driver to improve his driving behavior and fuel efficiency, motivating him to improve his driving behavior.

In this chapter, it has been demonstrated that the driving style is directly related to fuel consumption. Specifically, adopting an efficient driving style allows achieving fuel savings ranging from 15 % to 20 %. An aggressive driving style always results in a greater energy consumption and CO_2 emissions, whereas smooth driving ends up providing a greater energy efficiency as well as reduced gas emissions.

The application, which is available for free download in the DrivingStyle's website² and in the Google Play Store³, has achieved more than 4500 downloads from different countries in just a few months.

As future work, we intend to extend this platform to provide driving recommendations based on real-time feedback about the driver's health conditions and the instant driving pattern.

4.8 Acknowledgments

This work was partially supported by the Ministerio de Ciencia e Innovación, Spain, under Grant TIN2011-27543- C03-01.

²<http://www.drivingstyles.info>

³<https://play.google.com/store/apps/details?id=com.driving.styles>

DrivingStyles: A Mobile Platform for Driving Styles and Fuel Consumption Characterization

Javier E. Meseguer, C. K. Toh, Carlos T. Calafate, Juan Carlos Cano,
Pietro Manzoni

DrivingStyles: A Mobile Platform for Driving Styles and Fuel
Consumption Characterization

Journal of Communications and Networks

arXiv:1611.09065v1 [cs.CY] 28 Nov 2016

Accepted 28 November 2016. JCR Impact Factor 2015 = 1,462 Q2

Abstract

Intelligent Transportation Systems (ITS) rely on connected vehicle applications to address real-world problems. Research is currently being conducted to support safety, mobility and environmental applications. This paper presents the *DrivingStyles* architecture, which adopts data mining techniques and neural networks to analyze and generate a classification of driving styles and fuel consumption based on driver characterization. In particular, we have implemented an algorithm that is able to characterize the degree of aggressiveness of each driver. We have also developed a methodology to calculate, in real-time, the consumption and environmental impact of spark ignition and diesel vehicles from a set of variables obtained from the vehicle's Electronic Control Unit (ECU). In this chapter, we demonstrate the impact of the driving style on fuel consumption, as well as its correlation with the greenhouse gas emissions generated by each vehicle. Overall, our platform is able to assist drivers in correcting their bad driving habits, while offering helpful tips to improve fuel economy and driving safety.

5.1 Introduction

Intelligent Transportation Systems (ITS) introduce advanced applications aimed at providing innovative services, offering traffic management and enabling users to be better informed, including support for safety, mobility, and environmental applications. In parallel to ITS, mobile devices have experienced technological breakthroughs in recent years, evolving towards high performance terminals with multi-core microprocessors. The smartphone is a clear representative outcome of this trend.

In addition, the On Board Diagnostics (OBD-II) [36] standard, available since 1994, has recently become an enabling technology for in-vehicle applications due to the availability of Bluetooth OBD-II connectors. These connectors enable a transparent connectivity between the mobile device and the vehicle's Electronic Control Unit (ECU).

When combining high performance smartphones with OBD-II connectivity, new and exciting research challenges emerge, promoting the symbiosis between vehicles and mobile devices, and thereby achieving novel intelligent systems. *DrivingStyles* implements a solution based on neural networks, which is capable of characterizing the driving style of each user [27], as well as the fuel

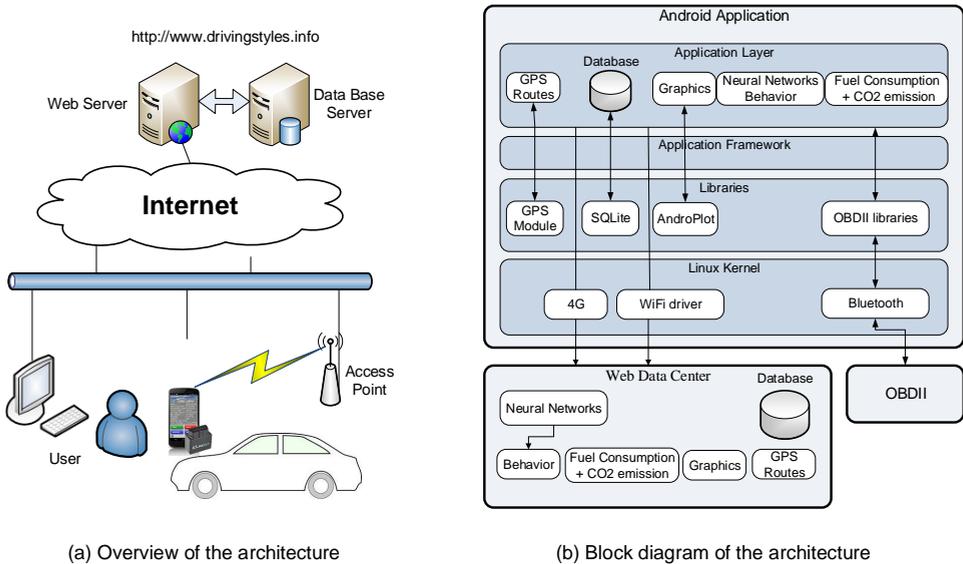


Figure 5.1: System Architecture of DrivingStyles.

consumption [37]. In order to achieve this functionality, the data is obtained from the ECU via the OBD-II Bluetooth interface, including the speed, acceleration, revolutions per minute of the engine, Mass Flow sensor (MAF), manifold Absolute Pressure (MAP), and Intake Air Temperature (AIT). Currently, this information can be collected and used in applications aimed at improving road safety and promoting eco-driving, thus reducing fuel consumption and greenhouse gas emissions. Specifically we find that, by shifting towards a more efficient driving style, users can save up to 20% of fuel while improving driving safety, thereby reducing greenhouse gases as we detail later on.

This paper is organized as follows: in the next section we present some related works. Section 5.3 introduces the DrivingStyles architecture (both the Android and the server interface). Models for fuel consumption and CO_2 emissions, are described in more detail in section 5.4. The tuning of the neural network, along with the obtained results, are presented in sections 5.5 and 5.6, respectively. Finally, section 5.7 presents the conclusion of our work.

5.2 Related Work

Technological advancements in the field of mobile telephony are making smartphones very powerful. This high computing power opens new and attractive opportunities for research. When coupled with the eco-driving concept, it has gained great significance in recent years [38]. An example is the prototype of an onboard unit developed by Hernandez et al. [39]. These driving techniques save fuel consumption, regardless of the technology used inside the vehicle. One of the main problems of eco-driving systems is identifying the factors that affect energy consumption. Ericsson [18] suggests that, in order to save fuel, sudden changes in acceleration and high speed driving should be avoided. Johansson et al. [19] suggest maintaining low levels of deceleration, minimizing the use of the first and second gears, and putting every effort into using the 5th and 6th gears, while avoiding continuous gear changes.

There are several proposals that analyze which variables affect fuel consumption. Kuhler [20] introduced a set of ten variables that are used in laboratories for fuel consumption and vehicle emissions analysis. Other authors such as André [21] improve these results by increasing and replacing some of the parameters. In previous works such as Leung [30] and COPERT III [31], different tools were developed to enable real-time collection of engine and vehicle parameters from the OBD connector.

Several commercial OBD-II scanner tools are available that can read and record these sensor values. Apart from such scanners, remote diagnostic systems such as GM's OnStar, BMW's Connected Drive, and Lexus Link are capable of monitoring engine parameters from a remote location. Car manufacturers used eco-monitoring to reflect the instant, historical, and time-elapsd fuel economy, and is used in the car through on-board trip computers [40]. Our solution differs from all the previous ones by providing a real-time analysis of the driving style of each user in the scope of eco-driving behavior, and based on neural network techniques. By calculating the consumption and greenhouse gas emissions generated by both types of engines (spark ignition, and diesel vehicles), we are able to closely relate both results, detailing the fuel savings achieved by soft driving patterns when compared to aggressive ones.

5.3 DrivingStyles Architecture

Our proposed architecture applies data mining techniques to generate a classification of the driving styles of users based on the analysis of their mobility traces. Such classification is generated taking into consideration the characteristics of each route, such as whether it is urban, suburban, or a highway, and it is then correlated with the fuel consumption and emissions of each driver.

To achieve the overall objective, our system comprises four elements:

1. An application for Android, based smartphones. Using an OBD-II Bluetooth interface, the application collects control information (by default every second, but it is configurable by the user) such as speed, acceleration, engine revolutions per minute, throttle position, and the vehicle's geographic position. In addition, we also obtain via OBD-II the mass flow sensor (MAF), the manifold Absolute Pressure (MAP), and the Intake Air Temperature (AIT) that are used in the calculation of fuel consumption. After gathering the information, the user can upload the collected data to the remote data center for analysis.
2. A data center offering a web interface to collect large data sets sent by different users concurrently, and to graphically display a summary of the most relevant results, like driving styles and route characterization of each route sent. Our solution is based on open source software tools such as Apache, PHP and Joomla®.
3. A neural network, which has been trained using the most representative route traces in order to correctly identify, for each path segment, the driving style of the driver, as well as the segment profile: urban, suburban or highway. We use the backpropagation algorithm [14], which has proven to provide good results in classification problems such as the one associated to this project.
4. Integration of the tuned neural networks both within the mobile device itself, and in the data-center platform. The goal is to use neural networks to dynamically and automatically analyze user data, reporting to the drivers in real time and allowing them to find out their driver profile, thus promoting a less aggressive and more ecological driving.

The block diagram of the DrivingStyles architecture is shown in Figure 5.1b. This consists of three blocks: the mobile application on a Android device, the data center platform, and an On Board Diagnostics (OBD-II) device.

The basic layer in the Android device is the Linux kernel, which contains all the essential hardware drivers to interact with the OBD-II device via Bluetooth. The top layer includes both Android's native libraries and our own libraries. Specifically, we developed the OBD-II communications module, along with the libraries for graphical data representation, at this layer. The next level up is the Application Framework; this layer manages the basic functions of the mobile device, and the communications with the developed libraries.

Finally, at the application layer, we developed the different modules of the *DrivingStyles* architecture, such as the fuel consumption and CO_2 emissions estimators, the neural networks behavior, GPS routes, and graphics. Also, the application provides real-time feedback from the device to the user such that, when it detects high levels of aggressiveness (above a certain threshold), the device automatically generates an acoustic signal to alert the driver.

5.3.1 *DrivingStyles Android Interface*

The first step for a user is to register at <http://www.drivingstyles.info>, and to download the free Android application. After installing the Android application in the mobile device, and after connecting to the Bluetooth ELM327 interface inside the car (this connector is mandatory on all vehicles since 2001), the data acquisition process will start (see Figure 5.1a).

The Android application is a key element of our system, proving connectivity to the vehicle and to the *DrivingStyles* web platform. Currently, it can be downloaded for free from the *DrivingStyles* website¹, or from Google Play ².

Once the mobile application is installed and configured, the user must pair the mobile device with the ELM327 (OBD-II Bluetooth device) to start getting data. The data obtained from the different variables such as acceleration, engine revolutions per minute (RPM), speed, mass flow sensor (MAF), manifold absolute pressure (MAP), and intake air temperature (IAT) are analyzed by the application, showing users the characteristics related to their driving, fuel consumption, and CO_2 emissions.

In order to adjust the application functionality, it offers several configuration options, i.e., User creation, Connection options, GPS Activation, Sensor sampling, and type of fuel definition. Once configured, our application captures data sent by the OBD-II and the GPS interfaces, as well as the phone's accelerometer showing the monitored sensors, and performing several monitoring

¹<http://www.drivingstyles.info>

²<https://play.google.com/store/apps/details?id=com.driving.styles>

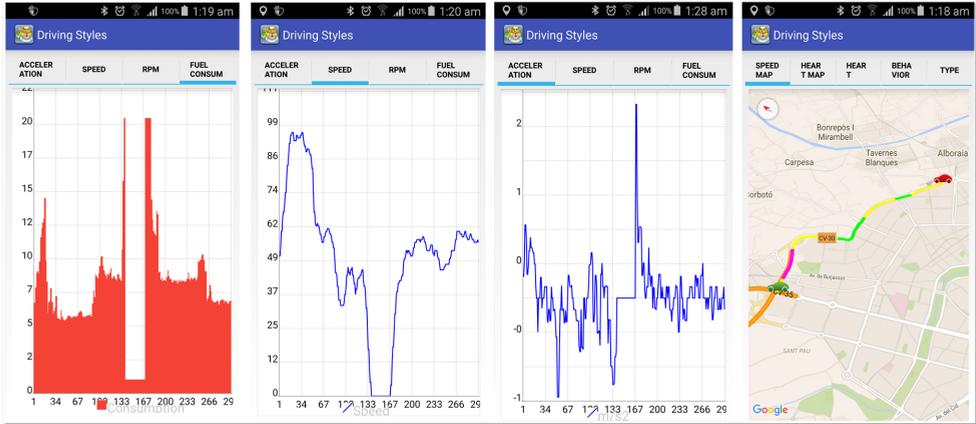


Figure 5.2: Snapshots of the fuel consumption, speed, acceleration parameters, and map module for the Android application.

actions in real time without affecting the data captured. Figure 2, shows some snapshots of our DrivingStyles application.

In addition, routes traces can also be sent to the website data center for further analysis. This module can be accessed either from the historic stored routes, or immediately after stopping the data capture. The information screen displays the header information of the selected route, such as: (i) date of the captured data, (ii) start time, (iii) finish time, (iv) maximum speed, and (v) fuel consumption. The URL of the DrivingStyles web interface is <http://www.drivingstyles.info>.

5.3.2 *DrivingStyles Server Interface*

The second main component of our architecture corresponds to the data center and its web interface. To implement this component, we have selected open source software such as Apache HTTP, and Joomla® as the Content Management System (CMS). We have used a CMS, combined with the use of a resource wrapper, which detaches our system from the presentation layer, thus focusing on the driving styles characterization problem. This module can be found in <http://www.drivingstyles.info>.

Basically the server receives data sent from the Android application of each user, and it provides functionality to work with User, Routes, and Statistics.

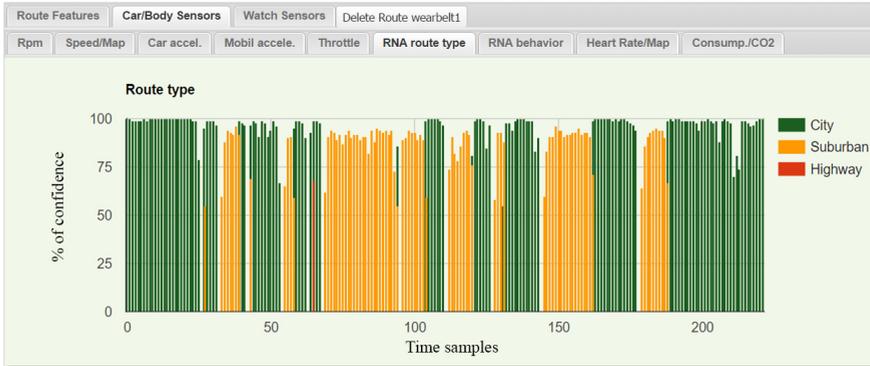
Characterization

Once the user is logged in, he is asked to record a number of important data, especially for future data mining studies. The most relevant items are sex, age, and other details concerning the vehicle used: car manufacturer, model, fuel type, and the theoretical 0-100 acceleration level (important to normalize the user behavior in our study).

In the Routes' section, users can access all the routes they have uploaded. When selecting car/body sensors, the system displays nine graphs for the different sensors obtained from the OBD-II (direct and indirect variables), as well as the route and driver behavior (see Figure 5.3). Next, we present our fuel consumption estimation approach relating it with the driver style as captured by the DrivingStyles platform.



(a) Confidence interval of the Driver Behavior neural network corresponding to the route analyzed.



(b) Confidence interval of the Route Type neural network corresponding to the route analyzed.

Figure 5.3: Snapshots of driver behavior and route type corresponding to the data center web.

5.4 Fuel consumption and greenhouse gas emissions calculation

5.4.1 Fuel consumption

Fuel consumption is usually represented as the ratio of fuel consumed per distance travelled, being measured in terms of litres per 100 kilometres (or alternatively as MPG-miles per gallon).

In this work, we focus on gasoline and diesel engines. Although the basic designs of gasoline and diesel engines are similar, the mechanics are different.

A gasoline engine compresses its fuel and air charge and then initiates combustion by the use of a spark plug. A diesel engine just compresses air until the combustion chamber reaches a temperature for self-ignition to occur. So, at a given speed in kilometres per hour, instantaneous fuel consumption can be calculated as follows:

$$\text{Inst. Fuel Consump. [l/km]} = \frac{\text{Fuel Flow [l]}}{\text{Speed [km]}} \quad (5.1)$$

Notice that it can only be calculated when the vehicle is moving and the engine is operating.

In addition, the Fuel Flow PID must be available, which often does not occur since most vehicles fail to support all the standard OBD PIDs. In fact, although there are many manufacturer-defined custom PIDs (not part of the OBD-II standard), the OBD standard itself does not provide a fuel consumption parameter. Instead, it provides other values that enable its calculation. Depending on the variables that the ECU can supply, the mathematical procedure to derive fuel consumption is different, as described below (see Figure 5.4) :

1. By combining the Engine Fuel Rate (PID 015E), also known as Fuel Flow (litres/hour), and Speed (PID 010D), it is easy to calculate instantaneous fuel consumption. However, while speed is mandatorily available, fuel rate is not. In fact, it was unavailable in all vehicles we used to carry out our tests. This can be due to two reasons: (i) the manufacturer chooses not to make it available, or (ii) there is no sensor inserted in the fuel line between the fuel tank and the engine carburetor to measure litres per hour.
2. If the MAF PID is available, but the Engine Fuel Rate is not, we can calculate fuel rate as Fuel Flow (litres/hour) by dividing the Mass Air Flow (PID 0110) · 3600 sec. by the product of air-to-fuel ratio and Fuel Density (using a fuel density equal to 820 g/dm³ for gasoline and 720 g/dm³ for diesel):

$$\text{Fuel Flow [l/h]} = (\text{MAF} \cdot 3600) / \text{AFR}_A \cdot \text{FD} \quad (5.2)$$

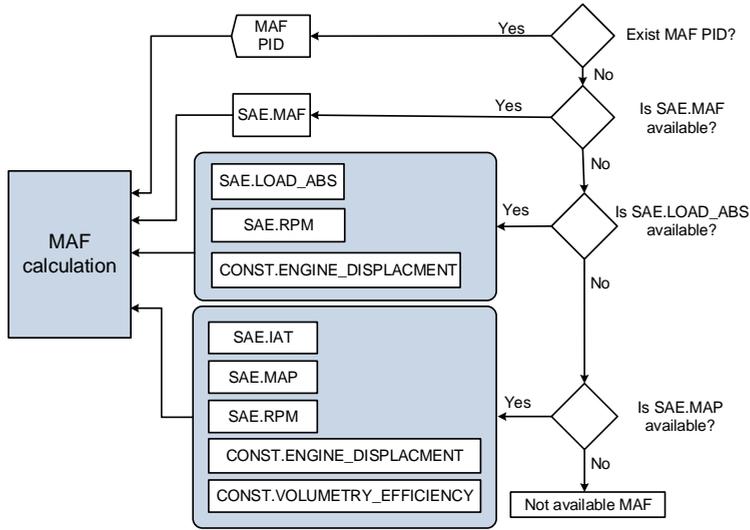


Figure 5.4: Scheme of the different MAF calculation possibilities regarding fuel consumption calculation.

where MAF refers to Mass Air Flow (g/s), AFR_A to the actual Air-to-Fuel Ratio (being 14.7 and 14.5 grams of air to 1 gram of fuel for gasoline and diesel respectively), and FD to the Fuel Density. The ratio between Fuel Flow and Speed, allows us to directly calculate fuel consumption.

3. Finally, If MAF is not available, there are two additional ways to calculate it (See [41] for more details):
 - As a function of the Absolute Load (PID 0143), the RPM (PID 010C) and the Engine Displacement (EngDisp, volume of an engine's cylinders in cm^3), Intake stroke is the fluid admission phase of a reciprocating cylinder.
 - As a function of the Intake Manifold Pressure (PID 010B), RPM (PID 010C), Intake Air Temperature (PID 010F), and Engine Displacement.

5.4.2 Greenhouse gas emissions calculation

The most significant greenhouse gases are generated from direct combustion of carbon dioxide (CO_2), Methane (CH_4), and Nitrous oxide (N_2O), among others. CO_2 is always generated when burning fuel that contains carbon. Since the carbon in the fuel is combined with the oxygen in the air: $C + O_2 \rightarrow CO_2$, the amount of CO_2 can be calculated by the atomic masses of carbon and oxygen, and the carbon content of the fuel. The atomic mass of carbon is 12_U and oxygen is 16_U , meaning that $CO_2 = 12_U + 2 \cdot 16_U = 44_U$. Burning 1kg of carbon produces $44/12 \approx 3.67kg$ of CO_2 in complete combustion, and so the CO_2 emission of combustion is $3.67 \cdot C_c \cdot m_{fuel}$ where $C_c = \text{fuel carbon content (mass basis)}$. Considering that the carbon content of diesel fuel is 85.7 % the CO_2 emission when burning 1kg ($m_{fuel} = 1kg$) of diesel fuel is:

$$\begin{aligned}
 m_{CO_2} &= 3.67 \cdot C_c \cdot m_{fuel} \\
 m_{CO_2} &= 3.67 \cdot 0.857 \cdot 1 [kg] = 3.15 [kg/1kg \text{ fuel}] \\
 &\quad \text{Density of diesel fuel is } 0.84 [kg/l] \\
 m_{CO_2} &= 3.15 \cdot [kg] \cdot 0.84 = 2.64 [kg/1l \text{ fuel}] \tag{5.3}
 \end{aligned}$$

Driving in a fuel-efficient manner can save fuel, money, and reduce greenhouse gas emissions. Among the factors that can affect fuel consumption, such as: vehicle age and condition, outside temperature, weather, and traffic conditions, we consider that driver behavior can be one of the most relevant parameter. Next, we provide detailed information about the neural network we proposed for characterizing driver styles.

5.5 Neural Networks-based data analysis

Neural networks [13] use artificial intelligence and automatic processing techniques to learn how to find patterns in data, thereby improving their success rate at making decisions or predictions. A learning algorithm is used to generate the neural network. For example, the driving style of each user and the type of route can be characterized from a well-defined set of rules and the ECU input variables.

There are many different learning algorithms such as `backprop_momentum`, `Hebbian`, or `delta-rule`, each one having its own advantages and disadvantages depending on the type of problems. In our project, we face a classification

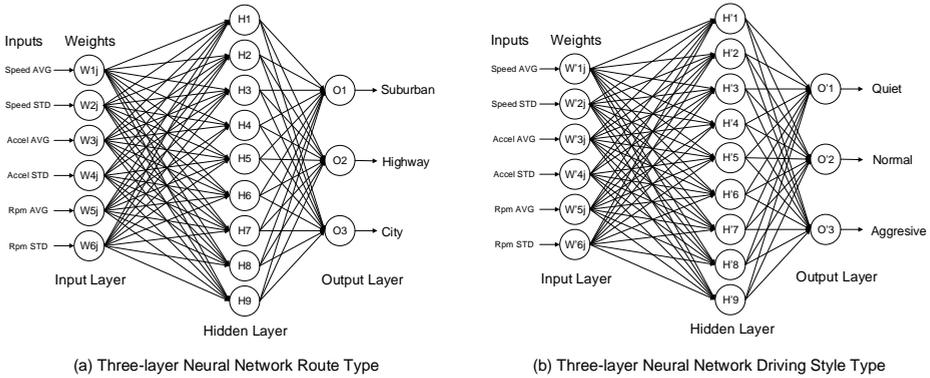


Figure 5.5: Schematic representation of the Three-layer Neural Networks used in the DrivingStyles Architecture.

problem: starting from some input data, which in our case are the speed, acceleration, and revolutions per minute (rpm) of the engine, we intend to obtain as outputs the type of road and the driving style. The problem of classifying the driver behavior and the route type with a supervised learning is to find a function that best maps a set of inputs to its correct output. We tried several types of algorithms in this direction, including backpropagation, backprop-momentum, and batch backpropagation, and the results evidenced that backpropagation [14, 26] was the best algorithm for our study since it achieved the lowest sum of squared errors (SSE) in terms of prediction.

A data preprocessing stage is selected from all the possible input variables of the neural network. From all the possible data, we keep a subset of these variables. In practice, this subset is not the minimum; instead, it is a compromise between a manageable number (not too large) of variables and an acceptable network performance. In this work, after considering the many variables that can be obtained from the Electronic Control Unit (ECU), we have chosen to train the neural network using: the mean and standard deviation of speed, the vehicle acceleration, and the rpm value.

In all the vehicles used for testing, these variables are easily obtained. Other variables, such as the position of the throttle, which would provide important information for the neural network training, have to be rejected because not all manufacturers provide such information. The data input of each parameter is normalized between 0 and 1; this normalization should take into consideration

Characterization

the range of possible values. The schematic representation of our three-layer neural network can be seen in Figures 5.5a, and 5.5b.

The application used for the creation and training of the neural networks required by this project is JavaNNS [15], a java version of the SNNS program from the University of Tübingen³.

First, an empty neural network was created, defining the number of entries mentioned previously, and the number of hidden nodes. A larger number of hidden nodes can improve the success rate, but has the negative effect of increasing the response time. On the contrary, with a large number of nodes, the network becomes a memory bank that can recall the training set to perfection, but does not perform well on samples that were not part of the training set. There are three output nodes for each neural network, one that characterizes the type of road (urban, suburban, or highway), and another one that characterizes the user's driving style (quiet, normal, or aggressive), see Figures 5.1a and 5.1b respectively.

Subsequently, we train the network with a total of 16038 samples, each representing a 3-second drive period (13.3 hours in total belonging to 7 drivers of different ages and sex). We initially adjust the learning rate to learning intervals of 0.2, and then modify this value to observe how the error affects the neural network (JavaNNS application [15] computes the mean square error in each learning iteration). The higher the learning rate, the greater the weight updating following each iteration; therefore, learning becomes faster, but it is prone to cause unwanted oscillations in the network. As the network training progresses, the number of learning cycles that take place in the tests is adjusted until the final trained network is obtained.

Once the neural network was successfully trained, the knowledge obtained was converted into C code, and this code was then integrated into our DrivingStyles platform. With the neural network already implemented, every time a route or route segment is selected, the system automatically returns the type of road, as well as the associated driving style. Figure 5.3 shows the results obtained by our neural network including the driving styles and route characterization of a particular route of one of our users.

Overall, with different traces analyzed, along with the drivers using our application, we have shown a correct classification of the different routes registered, both in terms of route types and driving styles, thus validating our proposed solution.

³<https://www.uni-tuebingen.de/en/university.html>

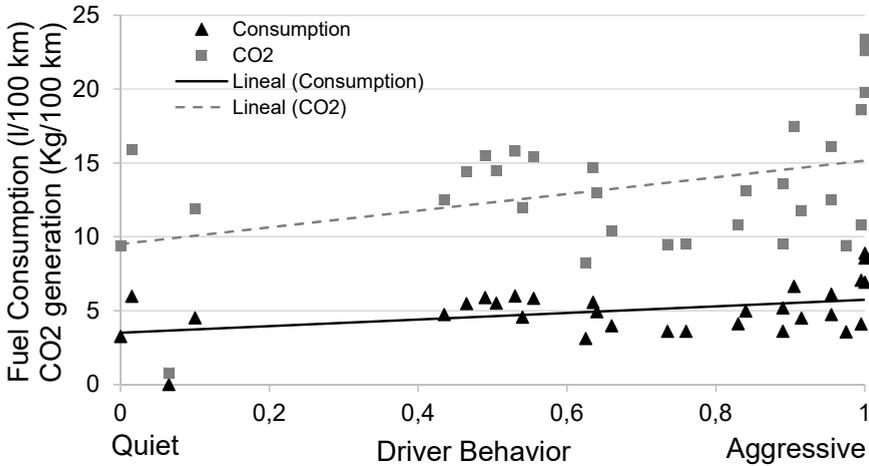


Figure 5.6: Chart of consumption and CO_2 in relation to the driving behavior.

5.6 Experimental results and evaluation

In our project, we focus on characterizing the driving style of different drivers, and then measuring the associated fuel consumption variations. In order to achieve this objective, we rely on the collaboration of 534 drivers from around the world using our platform, including countries like India, Brazil, Central America, and Europe. In this particular study, we analyzed the behavior of 75 representative routes (each divided into 10 second periods) using the neural network described earlier. For each section, the neural network returns the corresponding driver behavior, and we combine this data with the fuel consumption data corresponding to that route.

We carried out several types of tests to validate our proposals. Figure 5.6 shows the fuel consumption and CO_2 emissions reported by different drivers classified according to their driving style. The results of this test show that a more aggressive driving behavior causes fuel consumption to increase significantly, while also increasing the generation of CO_2 . To gain further insight into these correlations, Figure 5.7 displays the differences between quiet, normal, and aggressive driving behavior in terms of fuel consumption; aggressive drivers provoke fast starts and quick accelerations, driving at high engine revolutions, and causing sudden speed changes. Conversely, a quiet driving behavior would be smooth, without sudden speed changes or continuous gear shifts. It is

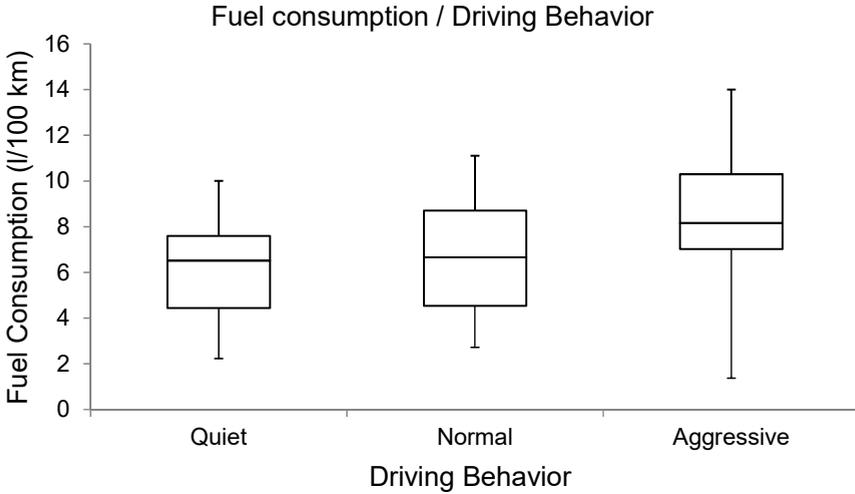


Figure 5.7: Box and whisker plot of Fuel Consumption/Driving Behavior.

clear that fuel consumption increases when the driver behavior becomes more aggressive, with average differences of up to 1.5 liters per 100km. In our experiments, an aggressive driver uses an average of 8 liters per 100km, and a quiet driver only 6.6 liters per 100km, meaning that the difference in terms of fuel consumption is not negligible, as the former may consume up to 20 % more fuel depending on the driving style. Regarding CO_2 emissions, they may increase by 50%, going from 10 to 15Kg/100km, depending on whether drivers are quiet or aggressive.

5.7 Conclusions and future work

This paper presents our DrivingStyles platform, which integrates mobile devices with data obtained from the vehicle's engine Electronic Control Unit (ECU) to characterize driver habits, as well as the associated fuel consumption and emissions. Our platform helps to promote a more ecological driving style by emphasizing on the relationship between driving style and fuel consumption, which has a clear and direct impact on the environment. It has been also demonstrated that the driving style is directly related to fuel consumption. Specifically, adopting an efficient driving style allows achieving fuel savings ranging from 15% to 20%. An aggressive driving style always results in a greater energy consumption and more CO_2 emissions, whereas smooth driv-

ing ends up providing a greater energy efficiency and reduced gas emissions. The application, which is available for free download in the DrivingStyle's website and Google Play Store, achieved more than 5800 downloads from different countries in just a few months. This emphasizes the great interest about research on this topic. As future work, we intend to extend this platform by providing route recommendations based on real-time feedback about the congestion state of different alternative routes, as well as providing estimated greenhouse emissions for different routes.

5.8 Acknowledgments

This work was partially supported by the Ministerio de Economía y Competitividad, Programa Estatal de Investigación, Desarrollo e Innovación Orientada a los Retos de la Sociedad, Proyectos I+D+I 2014, Spain, under Grant TEC2014-52690-R.

Chapter 6

DrivingStyles: Assessing the Correlation of Driving Behavior with Heart Rate Changes

Javier E. Meseguer, Carlos T. Calafate, Juan Carlos Cano
DrivingStyles: Assessing the Correlation of Driving Behavior with
Heart Rate Changes
2nd EAI International Conference on Smart Objects and Technologies
for Social Good
GOODTECHS 2016 November 30 – December 1, Venice, Italy.
Pages:, ISBN: To be published
Best Paper Award

Abstract

Driving safety is of utmost importance in our society. The number of fatalities due to car accidents is still very high, and reducing this trend requires as much attention as possible. There are situations where the emotional conditions of drivers vary due to either reasons beyond their control, or because they decide to change their driving style. Hence, we consider that such frequent situations deserve more scrutiny.

In this chapter we designed an Android application able to monitor in real-time both physiological data from the driver and diagnostic data from the vehicle to study their correlation. More specifically, we study the connection between driving aggressiveness and heart rate. The vehicle diagnostic data is obtained using an OBD-II connector. Among the various non-invasive biomedical sensors available nowadays, in this work we focus on heart rate sensors, either packaged in belts or in smartwatches.

6.1 Introduction

It is known that prolonged or repeated stress, such as long traffic jams or driving on severely congested roads, is related to increased aggressiveness. In fact, authors like Gibson [42], Cohen [43], and Gravina et al. [44, 45] have identified a potentially dangerous aggressiveness level as a result of driver stress. Thus, any method or system that can help at lowering the levels of aggressiveness when driving is welcome.

Our proposal just needs basic devices, such as a mobile phone, and an On Board Diagnostics (OBD-II) device [1, 4, 10], available for less than 20 dollars, along with a heart rate band or a smartwatch. Our novel DrivingStyles architecture adopts data mining techniques and neural networks to analyze and generate a classification of the driving styles based on an analysis of the characteristics of the driver along the route followed. It ensures that the driver can be constantly aware of its level of aggressiveness and driving stress, and how this affects to his heart rate. In a previous study [37], we developed a methodology to calculate, in real-time, the impact that the driving style will have on the consumption and environmental impact of spark ignition and diesel vehicles. We demonstrated that an aggressive driving style increases the fuel consumption, as well as the emission of greenhouse gases.

driving behavior also leads to a heart rate increase of at least three beats per minute with respect to a quiet behavior. The analysis has been carried out with 460 minutes of driving, taking 27663 direct samples (obtained from the vehicle's ECU and the driver's heart rate band), which corresponds to 5532 time windows where the driver behavior and road types are analyzed. Our platform is able to assist drivers in correcting their bad driving habits, while offering helpful recommendations to improve fuel economy, and driving safety.

This paper is organized as follows: in the next section we present the DrivingStyles architecture. The method used for the analysis of the variables is presented in section 6.3. We present experimental results in section 6.4. Finally, in section 6.5, we review the main conclusions and discuss future work.

6.2 General Overview of the DrivingStyles Architecture

Our proposed architecture applies data mining techniques to generate a classification of the driving styles of users based on the analysis of their mobility traces using neural networks. Such classification is generated taking into consideration the characteristics of each route, such as whether it is urban, suburban, or highway. To achieve the overall objective, the system is structured around the following two elements:

1. An application for Android-based smartphones which is responsible for collecting data from the car and the driver's heart rate band or smartwatch, which also analyzes routes and driver behavior using neural networks.
2. A cloud-based data center to collect large data sets sent by different users concurrently. Subsequently, these data are analyzed using data mining and expert systems, in order to generate useful information.

6.2.1 Android Application

Using an OBD-II Bluetooth interface, the Android application (see Figure 7.1b) collects information such as speed, acceleration, engine revolutions per minute, throttle position, and the vehicle's geographic position. It also obtains information from a wearable heart rate monitor, chest belts and smartwatches. This information is analyzed on the device itself, performing the analysis of driver behavior and road type (using neural networks), instantaneous fuel consumption, greenhouse gas emissions, and heart rate measurement.

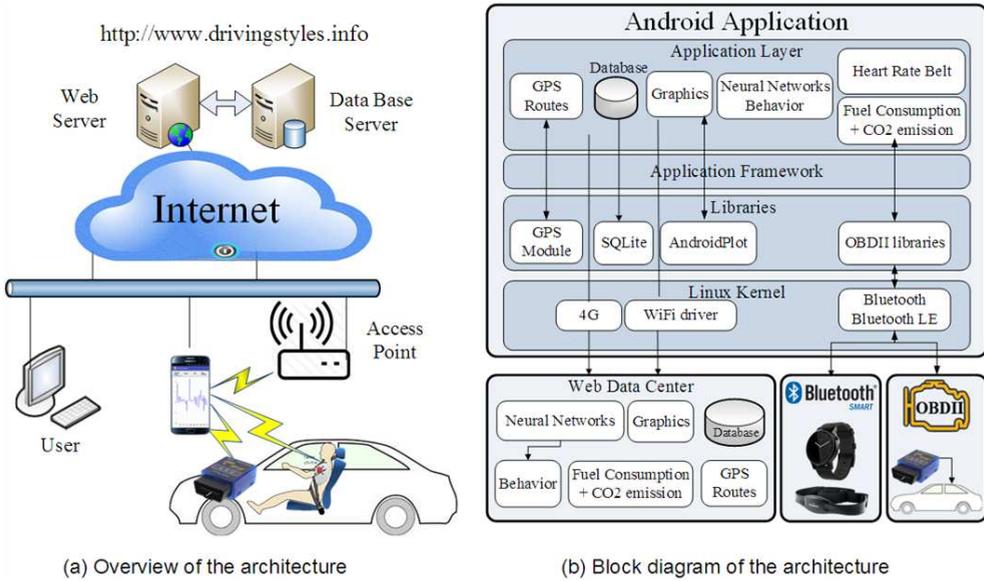


Figure 6.1: System architecture of DrivingStyles: Overview and block diagram architecture.

We then provide feedback from the device to the user in a way that, when the application detects high levels of aggressiveness, (above a certain threshold), the device generates an acoustic signal to alert the driver. Furthermore, if the user has a wearable device, such as a smartwatch, it is notified by a vibration event as well.

Eventually, the user uploads the route data to the remote data center for a more comprehensive analysis. The Android application is a key element of our system, proving connectivity to the vehicle and to the DrivingStyles web platform. The application, which is available as a free download in the DrivingStyle’s website¹ and from Google Play Store², has achieved nearly 6000 downloads from different countries in just one year. This indicates the great interest of this type of user-level applications. For more detailed information please refer to [27, 37].

¹<http://www.drivingstyles.info>

²<https://play.google.com/store/apps/details?id=com.driving.styles>

6.2.2 Data Center

Our data center³ is able to collect large data sets sent by different users (see Figure 7.1a). There are currently 485 registered users and 411 routes sent for study. The data center allows users and administrators to access data about routes and per-user statistics. In particular, users can access all the routes they have uploaded.

6.3 Research Strategy and Methodology

We now present the methodology we have followed in order to correlate driving aggressiveness and driver heart rate by using the data provided by our DrivingStyles architecture.

6.3.1 Participant

The data reported in the present study were collected from a 35 years old male driver, without heart diseases, and whose heart rate while at rest lies between 70 bpm (beats per minute) and 75 BMP. We have analyzed twenty-one routes of varying durations, and under completely different environments (urban, suburban or highway), and also at different weather conditions (rainy, sunny, cloudy, etc.) and road conditions. This diversity allowed us to analyze the system reliability under different environmental conditions [46].

The driver was equipped with an Android device with our DrivingStyles application, and a heart rate band (brand Geonaute, although any other compatible band could be used as well) attached to the driver's chest.

6.3.2 OBD-II Instrument

The vehicle used for testing is a gasoline model of the KIA brand with manual transmission. It was instrumented with an interface compatible with the On Board Diagnostics (OBD-II) standard [1, 4], available since 1994 [10], and that has recently become an enabling technology for in-vehicle applications due to the appearance of Bluetooth OBD-II connectors [9, 10]. These connectors enable a transparent connectivity between the mobile device and the vehicle's Electronic Control Unit (ECU).

³<http://www.drivingstyles.info>

6.3.3 Heart Rate Monitor (HRM)

Regarding heart rate monitor (HRM) devices, there are mainly two types on the market: the smartwatch (or other wrist band) and the chest strap. Smartwatch models tend to be less accurate than chest-strap HRMs. Tests were conducted with different models of both types. The first devices used were wrist devices, including the Motorola 360 smartwatch. In this model the back of the watch hosts the heart rate sensor. Despite using oximeter technology pulse measurement, in our tests, the sampling frequency of the smartwatch was too low and, in combination with the high battery consumption when the heart rate measurement is activated, made us disregard this device from the beginning. So, we opted for the heart rate belt device instead.

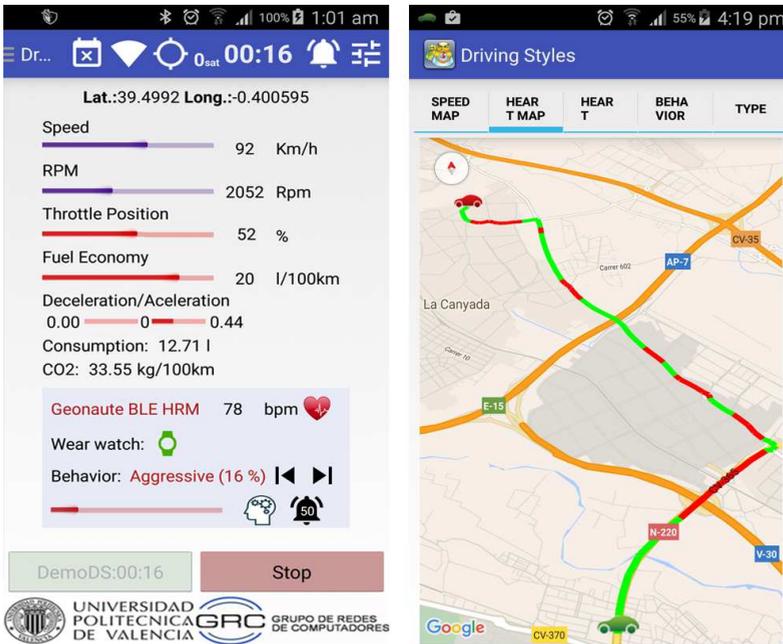


Figure 6.2: Snapshots of the main screen and the heart rate module.

Heart rate belt operation is simple, an electrical signal is transmitted through the heart muscle in order for it to contract. This electrical activity can be detected through the skin. The transmitter part of the heart rate monitor is placed on the skin around the area that the heart is beating, and picks up this signal. The transmitter then sends an electromagnetic signal containing heart rate data to the wrist receiver which displays the heart rate.

Table 6.1: Mean, standard deviation, and range of route time, speed, and heart rate.

	<i>Mean</i>	<i>SD</i>	<i>Range</i>
Route time (<i>minutes</i>)	24.63	26.37	6.32 – 81.44
Speed (<i>km/h</i>)	67.44	41.30	0 – 135
Heart rate (<i>bpm</i>)	79.73	10.87	55 – 115

As we can see in Figure 7.3, the Android app displays the heart rate in real-time, as well as a map representation of the heart rate compared with the average of the route undertaken so far, being red if it is higher than average, and green otherwise. It is mandatory that the mobile device used supports Bluetooth Low Energy (Bluetooth LE, BLE) to connect with the heart rate monitor.

6.3.4 Measurement Result.

The total time of the twenty-one routes considered for this study has been 7 hours and 40 minutes (460 minutes). Regarding the heart rate, 27663 direct samples (one sample every second) have been obtained. Also 5520 driving behavior measures calculated by the neural network have been used in the test, reflecting the behavior of the driver at measurement time (behavior analysis is performed with data from 5 seconds before performing the calculation). See Table 6.1 for further information.

Before performing the statistical calculations, the samples were normalized between 0 and 1. The neural network developed returns a value between 0 and 100, as a result of analyzing each type of behavior. These values must also be normalized before the statistical study.

In Section 6.4 we proceed to analyze the correlation between driving behavior and the driver's heart rate.

Table 6.2: Slope-intercept form equation of single route and all routes.

Behavior single route	y	m	R^2	R
Quiet-Normal	$8.692x + 72.72$	8.692	0.172	0.414
Normal-Aggressive	$5.667x + 74.049$	5.667	0.041	0.203
Quiet-Normal-Aggressive	$6.937x + 73.362$	6.937	0.173	0.416
Behavior all routes	y	m	R^2	R
Quiet-Normal-Aggressive	$2.959x + 76.557$	2.959	0.005	0.071

6.4 Experimental Results and Evaluation

We can assume that drivers can be exposed to higher levels of stress during rush-hours in a city [47]. Similarly, the sparsest traffic conditions can be found on country side roads, driving on highways or in sparsely populated areas. Hence, these two conditions should represent the far ends that we should find in the routes under analysis.

Our study is based on a set of twenty-one different routes made by the same driver on a same vehicle in an attempt to eliminate these factors, and focuses solely on the relation between driving aggressiveness and heart rate. Then, we focus on a particular route to have a more in-depth perspective of the results obtained and the overall findings. In both cases, we obtain through linear regression the line that better describes the correlation between both data sets. This way, a positive gradient shall validate our assumption of a positive correlation between driver behavior and heart rate, as intended.

6.4.1 On-road Tests (all routes).

First, we analyzed the twenty-one routes mentioned previously in section 6.3.4. The total time of all these routes is 460 minutes of driving, which corresponds to a total of 27663 direct samples and 5532 behavior samples calculated by the system. Notice that the developed neural network evaluated the three types of driver behavior at once, generating an output score for each that allows generating an output in the range from 0 to 1.

Figures 6.3 and 6.4 show the correlation plots between heart rate and behavior of the 27663 samples obtained for the study. Table 6.2 shows the equation of the corresponding slope-intercept form, where the slope given by m , which

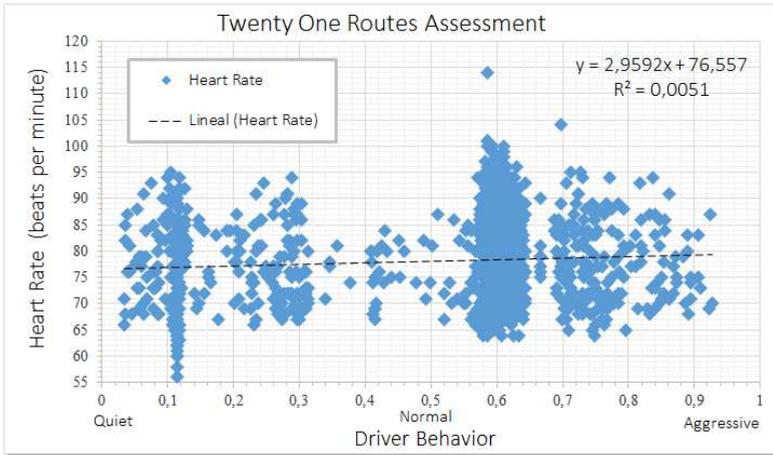


Figure 6.3: Correlation between heart rate and driving behavior.

equals to 2.959, and the y-intercept by b , which is equal to 76.557 (see Figure 6.3). As it can be observed, the intended correlation between driving styles and heart rate R is 0.071. The correlation value obtained is not as significant when compared to result for a particular route, as shown in the following section.

These results were mostly expected since the driver remains seated in all cases, and so the additional physical burden requiring a higher heart beat is not comparable to more demanding situations. It is noteworthy mentioning, though, that in this section we are studying routes of many types, some from urban scenarios and other from highway scenarios, being the behavior less aggressive for the latter. So, overall, we find that the difference between a quiet behavior and an aggressive behavior for a specific driver is a heart rate increase of 3,72 %. Figure 6.4 shows the box and whisker plot of heart rate vs driving behavior. We find that the difference in heart rate between quiet and aggressive behavior is 3,25 % (about three beats per minute).

6.4.2 On-road Tests (single route).

In this second part of the analysis, we study a specific route chosen from the set of twenty-one routes analyzed in this chapter. The DrivingStyles platform, in addition to analyzing the behavior, is also able to compute the route type.

In particular, this route has a duration of 6 minutes and 33 seconds, circulating at an average speed of 24 km/h and a maximum speed of 57 km/h, covering a

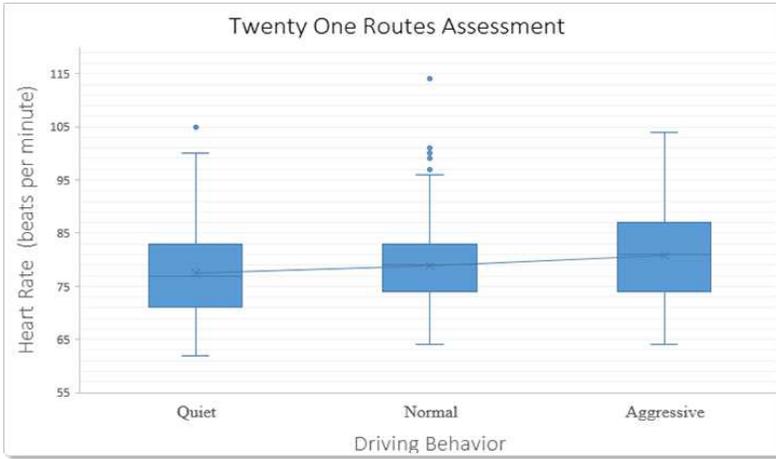


Figure 6.4: Box and whisker plot of heart rate vs driving behavior.

total of 2.17 km; this corresponds to 381 behavior type samples, also including data about the driver’s heart rate that we use in the statistical analysis.

After normalizing the data as explained in section 6.3.4, we split our analysis into three parts: the difference in terms of heart rate between quiet and normal behavior (see Figure 6.5), the same difference between normal and aggressive behavior (see Figure 6.6), and a full comparison between quiet, normal, and aggressive driving (see Figures 6.7 and 6.8):

1. Concerning the first case study i.e., when comparing the heart rate between a quiet behavior and normal behavior (see Figure 6.5), we find that the linear trendline has a positive slope (8.69) and the *R – squared* value or coefficient of determination is 0.17 (how close the data are to the fitted regression line). As shown in this figure, there is a clear correlation between heart rate and driver behavior, being the heart rate when the system detects normal behavior about 10.67% higher compared to a quiet behavior.
2. In Figure 6.6 we compare the normal driver behavior against an aggressive behavior for the same route. We find that the slope of the regression line is lower than for the plot previously discussed (quiet behavior vs normal behavior), having a value of 5.66, being the coefficient of determination significantly lower, with a value of 0.04. Observing both plots (see Figures 6.5 and 6.6), we find that, regarding the outputs of the neural network

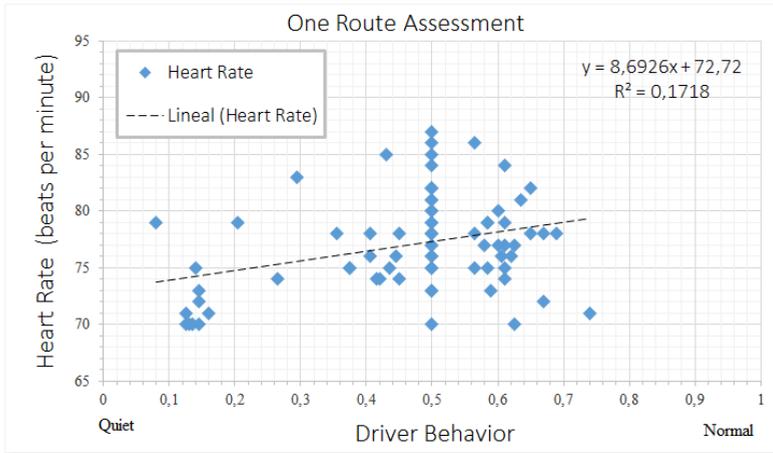


Figure 6.5: Correlation between heart rate and driving behavior (quiet-normal).

implemented, the computation of the driver’s behavior tends to provide as outcome that is either a quiet behavior or an aggressive behavior in most cases, being intermediate values more scarce. In this case, the heart rate difference is 7.20 % higher between aggressive behavior and the normal behavior.

3. In the third scenario, all the system’s outputs were jointly analyzed (see Figures 6.7 and 6.8). We find that the linear trendline remains positive, being the slope value of 6.93, and the coefficient of determination is 0.17. In this last analyzed case the difference between a quiet behavior and an aggressive behavior is 8.61 %. The results obtained are very similar to the first plot (see Figure 6.5), which leads us to consider whether it would be interesting, in future studies, to train the neural network to have only two outputs instead: quiet behavior and aggressive behavior.

Finally, the box and whisker plot (see Figure 6.8) displays the differences between quiet, normal, and aggressive driving behavior vs heart rate; for this test subject, an aggressive driving provoked an increased heart rate. If we look at the value of the median in the three types of behavior (quiet, normal and aggressive) we see that the difference in heart rate between a quiet and normal behavior is 2.78 % (about two beats per minute); similarly, between a normal and an aggressive behavior, this difference is 2.41 % (about two beats per minute as well). Summarizing, according to our findings, the driver’s pulse

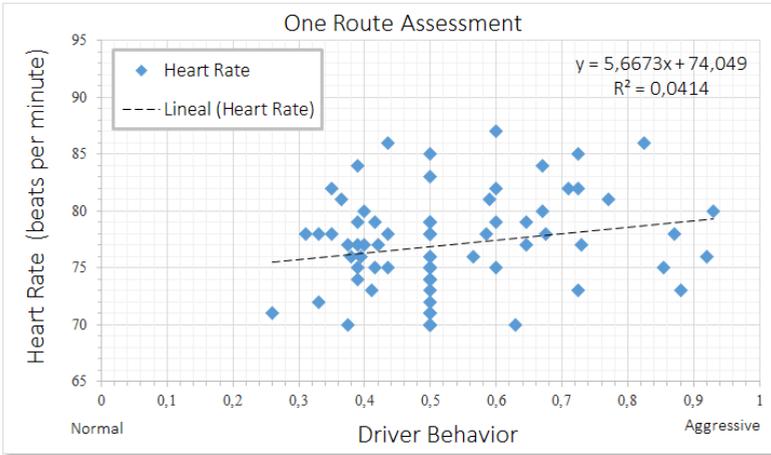


Figure 6.6: Correlation between heart rate and driving behavior (normal-aggressive).

increased by 5.18 % (slightly more four beats per minute) when the driver switched to a more aggressive driving compared to a quite driving style.

6.5 Conclusions and Future Work

In this chapter we studied the correlation of the driver heart rate with respect to his driver behavior. We based our study on the use of our novel DrivingStyles architecture, that combines new technologies such as smartphones and wearable body sensors with the modern software implementations of artificial neural networks.

The results of the present study indicated that aggressive driving causes an increase in the heart rate, being able to rise it by up to three beats per minute on average. Based on our experimental results, we have reached the conclusion that the difference in terms of heart rate between a quiet and aggressive behavior can become very noticeable. In statistical terms, we also found that, as the number of samples increases, the correlation between the driver behavior and heart rate becomes lower. This was expected since increasing the number of routes whose behavior is largely quiet, makes the percentage of values with an aggressive behavior to decreases, i.e., an urban route that causes stress and aggressiveness will get closer to our results that a long highway route where the stress is non-existent or scarce. Since this study has been conducted taking a middle-aged male subject as reference, in future works we will expand the

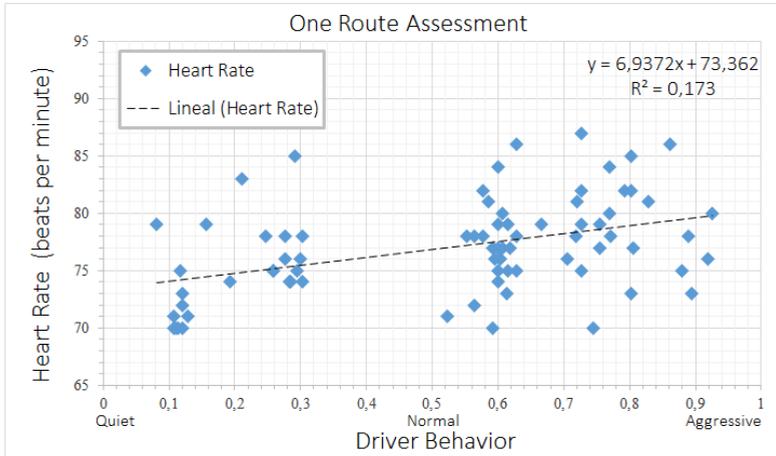


Figure 6.7: Correlation between heart rate and driving behavior.



Figure 6.8: Box and whisker plot of heart rate vs driving behavior.

scope of our study to women, and also to drivers of different ages, which will help at covering a wider range of possible cases. This will allow us to study the differences between various age segments, as well as to differentiate driving aggressiveness and heart rate based on the driver's gender.

6.6 Acknowledgments

This work was partially supported by the Ministerio de Economía y Competitividad, Programa Estatal de Investigación, Desarrollo e Innovación Orientada a los Retos de la Sociedad, Proyectos I+D+I 2014, Spain, under Grant TEC2014-52690-R.

Chapter 7

On the Correlation Between Heart Rate and Driving Style in Real Driving Scenarios

Javier E. Meseguer, Carlos T. Calafate, Juan Carlos Cano
On the Correlation Between Heart Rate and Driving Style
in Real Driving Scenarios
Mobile Networks and Applications
Article 833, 16 February 2017
DOI: 10.1007/s11036-017-0833-x JCR Impact Factor 1,538 Q2.

Abstract

Driving safety is of utmost importance in our society. The number of fatalities due to car accidents is still very high, and reducing this trend requires as much attention as possible. There are situations where the emotional conditions of drivers vary due to reasons beyond their control, or because they decide to change their driving style. Hence, we consider that such frequent situations deserve more scrutiny. In this work we addressed such issues by designing an Android application able to monitor in real-time both physiological data from the driver and diagnostic data from the vehicle (this data is obtained using an OBD-II connector) to study their correlation.

Among the various non-invasive biomedical sensors available nowadays, we have adopted heart rate sensors, either packaged in belts or in smartwatches. This allows studying the relationship between driving aggressiveness and heart rate. For our analysis we focused on fourteen different routes accounting a total driving time of 6 hours and 2 minutes, which we have split into three separate categories: urban, suburban, and highway routes. We analyzed the correlation between the heart rate and the driving style for each of the three groups. Our experiments show that the differences in terms of heart rate between quiet and aggressive behavior range between 2.5 % and 3 % beats per minute higher for the latter behaviour compared to the former.

7.1 Introduction

It is known that prolonged or repeated stress, such as long traffic jams or driving on severely congested roads, is related to increased aggressiveness. In fact, authors like Gibson [42], Cohen [43], and Gravina et al. [44, 45] have identified a potentially dangerous aggressiveness level as a result of driver stress [48, 49]. Thus, any method or system that can help at lowering the levels of aggressiveness when driving is welcome.

Recently, the smartphones have become ubiquitous, allowing to perform all types of tasks using both internal and external sensors connected wirelessly. In addition, access to the vehicles Electronic Control unit (ECU) was made possible through the availability of cheap On Board Diagnostics (OBD-II) devices [1, 4, 11], thus enabling a seamless vehicle-smartphone integration. Such integration paves the way for novel strategies in terms of driving style analysis, providing a detailed feedback about the driver actions upon the vehicle.

In this chapter we focus our analysis on the correlation between driving style and driving stress measured as variations in the driver’s heartbeats. To accomplish this goal we rely on our novel DrivingStyles architecture, which adopts data mining techniques and neural networks to analyze and generate a classification of the driving styles based on an analysis of the characteristics of the driver along the route followed. It ensures that drivers can be constantly aware of their level of aggressiveness and driving stress, and how this affects to their heart rate [13, 14]. In a previous study [37], we developed a methodology to calculate, in real-time, the impact that the driving style will have on fuel consumption, and the environmental impact of spark ignition and diesel vehicles [20]. We demonstrated that an aggressive driving style increases the fuel consumption, as well as the emission of greenhouse gases [19, 50].

In this chapter we go one step forward and demonstrate that a more aggressive driving behavior also leads to a heart rate increase between 2.5% and 3% beats per minute with respect to a quiet behavior. The analysis has been carried out based on traces that correspond to 371 minutes of driving, taking 22290 direct samples (obtained from the vehicle’s ECU and the driver’s heart rate band), where the driver behavior and road types are analyzed. Our platform is able to assist drivers in correcting their bad driving habits, while offering helpful recommendations to improve fuel economy and driving safety. In terms of requirements, only off-the-shelf devices such as a mobile phone and an OBD-II device, along with a heart rate band or a smartwatch, are required.

This paper is organized as follows: in the next section we present the DrivingStyles architecture. The method used for the analysis of the variables under study is presented in Section 7.3. Experimental results are then presented in Section 7.4. Finally, in Section 7.5, we review the main conclusions and discuss future work.

7.2 General Overview of the DrivingStyles Architecture

Our proposed architecture applies data mining techniques to generate a classification of the driving styles of users based on the analysis of their mobility traces using neural networks. Such classification is generated taking into consideration the characteristics of each route, such as whether it is urban, suburban, or highway. To achieve the overall objective, the system is structured around the two following elements:

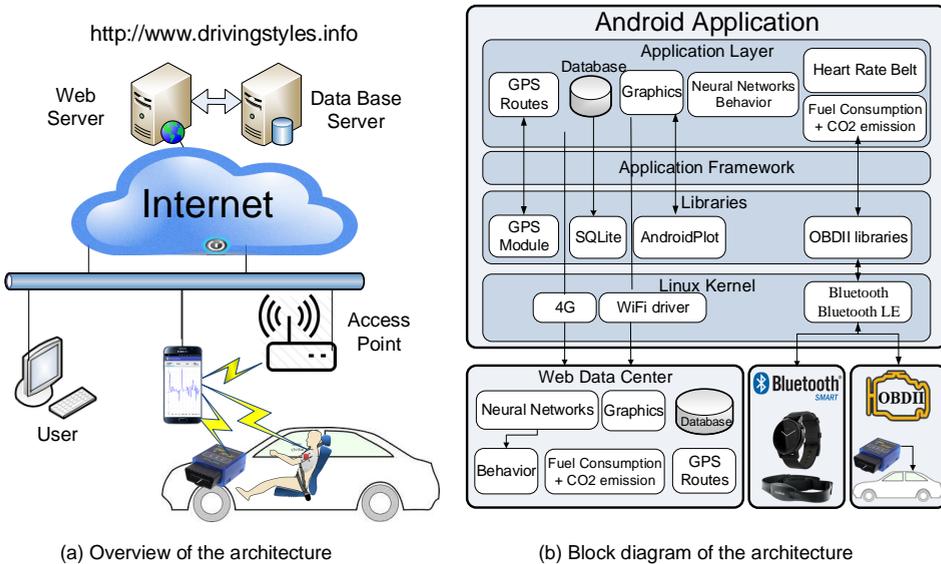


Figure 7.1: System architecture of DrivingStyles: Overview and block diagram.

1. An application for Android-based smartphones, which is responsible for collecting data from the car and the driver’s heart rate band or smart-watch, and that also analyzes routes and driver behavior using neural networks.
2. A cloud-based datacenter to collect large data sets sent by different users concurrently. Subsequently, these data are analyzed using data mining and expert systems, in order to generate useful information.

7.2.1 Android Application

Using an OBD-II Bluetooth interface, the Android application (see Figure 7.1.b) collects information such as speed, acceleration, engine revolutions per minute, throttle position, and the vehicle’s geographic position. It also obtains information from a wearable heart rate monitor, chest belts, and smartwatches. This information is analyzed on the device itself, performing the analysis of driver behavior, and road type (using neural networks), instantaneous fuel consumption, greenhouse gas emissions, and heart rate measurement.

We then provide feedback from the device to the user in a way that, when the application detects high levels of aggressiveness (above a certain threshold which can be configured from the application settings), the device generates an acoustic signal to alert the driver. Furthermore, if the user has a wearable device such as a smartwatch, it is notified through a similar vibration event.

Eventually, the user uploads the route data to the remote data center for a more comprehensive analysis. For more detailed information please refer to [27, 37].

7.2.2 Data Center

Our data center¹ is able to collect large data sets sent by different users (see Figure 7.1.a). It allows users and administrators to access data about routes and provides per-user statistics. To implement this component, we have selected open source software such as Apache HTTP, and Joomla® as the content management system (CMS).

Basically, the server receives data sent from the Android application of each user, and it provides functionality to work with User, Routes, and Statistics. Once the user is logged in, he is asked to record a number of important data, especially for future data mining studies. The most relevant items are sex, age, and other details concerning the vehicle used: car manufacturer, model, and fuel type, among others.

7.3 Research Strategy and Methodology

We now present the methodology we have followed in order to correlate driving aggressiveness and driver heart rate by using the data provided by our DrivingStyles architecture.

7.3.1 Participant

In this study we have relied on fourteen routes (total of 6 hours and 11 minutes) of varying duration carried out by a single subject in completely different environments (urban, suburban or highway), and for different road weather conditions (rainy, sunny, and cloudy). This scenario diversity allowed us to analyze the system reliability under in different situations [46], thus achieving a higher confidence on results.

¹<http://www.drivingstyles.info>

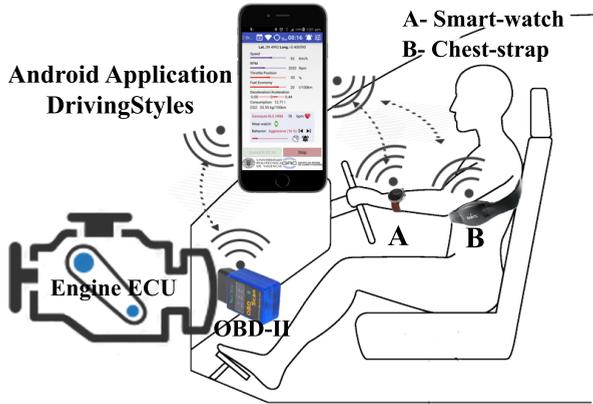


Figure 7.2: On Board schema components of DrivingStyles instrumentation.

The data reported in the present study were collected from a 36 year-old male driver, without heart diseases, and whose heart rate while at rest lies between 70 BMP (beats per minute) and 75 bpm. The driver was equipped with an Android device running our DrivingStyles application, and a heart rate band (brand Geonaute, although any other compatible band could be used as well) attached to the driver's chest.

7.3.2 OBD-II Car Instrument

The vehicle used for testing is a gasoline KIA model with manual transmission. It was instrumented with an interface compatible with the OBD-II standard [1, 4], available since 1994 [11], and that has recently become an enabling technology for in-vehicle applications due to the appearance of Bluetooth OBD-II connectors [9, 11]. These connectors enable a transparent connectivity between the mobile device and the vehicle's ECU (see Figure 7.2).

7.3.3 Heart Rate Monitor (HRM)

Regarding heart rate monitoring (HRM) devices, there are mainly two types on the market: the smartwatch (or other wrist band), and the chest strap. Notice that, due to technological restrictions, smartwatch models tend to be less accurate than chest-strap HRMs. Tests were conducted with different models of both types. The first devices used were wrist devices, including the Motorola 360 smartwatch. In this model, the back of the watch hosts the heart rate sensor. Despite using oximeter-based technology for pulse measurement, the

sampling frequency of the smartwatch was too low for our purposes, and, in combination with the high battery consumption when the heart rate measurement function is activated, made us disregard this device from the beginning. So, we opted for the heart rate belt device instead.



Figure 7.3: Snapshots of the main screen and the heart rate module for the Android application.

As we can see in Figure 7.3, the Android app displays the heart rate in real-time, as well as a map representation of the heart rate compared with the mean value for the part of the route completed so far, being red if it is higher than the mean, and green otherwise.

7.3.4 Measurement Results

This study was carried out with a total of fourteen routes (nine urban routes, two suburban routes, and three highway routes), and a total driving time of 6 hours and 11 minutes (371 minutes). Regarding the heart rate, 22290 direct samples (one sample every second) have been obtained. Also, 4458 driving behavior measures calculated by the neural network have been used in the test, reflecting the behavior of the driver at measurement time (behavior analysis is performed with data taken from a time window starting 5 seconds before performing the calculation).

Table 7.1: Number of samples, time, and mean heart rate value for the different routes under analysis.

Route Type	Number Routes	Samples	Total Time	Heart Rate Mean
Urban	9	4291	1h 03' 39"	72.95
Suburban	2	4353	1h 12' 33"	77.77
Highway	3	13664	3h 47' 18"	78.85

In Tables 7.1 and 7.3 we can see the parameters associated to the different routes taken, including the total number of samples and the total driving time for each route, as well as the mean heart rate. Since these data refer to real everyday life situations, urban routes are typically short-lived, while highway routes are much longer, having suburban routes an intermediate duration.

In Section 7.4 we proceed by analysing the correlation between driving behavior and the driver's heart rate.

7.4 Experimental Results and Evaluation

In general we can assume that urban drivers can be exposed to higher levels of stress during rush-hours [47, 49]. Similarly, the sparsest traffic conditions can be found in scenarios such as countryside roads, while driving on highways, or in sparsely populated areas [48]. Hence, these two conditions should represent the far ends that we should find in the routes under analysis.

Three case studies are analyzed: the driver moving in an urban scenario (9 routes), the same driver moving in a suburban scenario (2 routes), and the driver circulating in a highway (3 routes). Overall, total time of these case studies adds up to 6 hours and 11 minutes.

In all cases, we obtain through linear regression the approximation that better describes the correlation between both data sets. A positive gradient shall validate our assumption of a positive correlation between driver behavior and heart rate, as intended (see Table 7.2).

Table 7.2: Slope-intercept form equation of urban, suburban, and highway routes.

Behavior Quiet-Normal-Aggressive	y	R
Urban (9 routes)	$3.054x + 70.221$	0.103
Suburban (2 routes)	$1.727x + 71.603$	0.069
Highway (3 routes)	$8.474x + 73.552$	0.416

Table 7.3: Mean, standard deviation, and range of route time, speed, and heart rate of urban, suburban, and highway routes.

Urban routes	<i>Mean</i>	<i>SD</i>	<i>Range</i>
Routes time (<i>minutes</i>)	7.75	2.64	3.11 – 13.10
Speed (<i>km/h</i>)	31.14	24.85	0 – 84
Heart rate (<i>bpm</i>)	71.92	6.08	60 – 89
Suburban routes			
Routes time (<i>minutes</i>)	35.96	3.56	33.45 – 38.48
Speed (<i>km/h</i>)	34.26	26.51	0 – 112
Heart rate (<i>bpm</i>)	72.39	5.93	61 – 92
Highway routes			
Routes time (<i>minutes</i>)	75.33	3.05	77.0345 – 78.12
Speed (<i>km/h</i>)	100.18	22.53	0 – 134
Heart rate (<i>bpm</i>)	78.46	6.19	60 – 114

7.4.1 Driving tests in Urban Areas

The combined data from all urban routes adds up to 71 minutes and 22 seconds (4282 seconds) of urban driving, which corresponds to a total of 857 behavior samples calculated by the system (see Tables 7.1 and 7.3).

By performing linear regression we find that equation $3.054x + 70.221$ is the corresponding regression line in the slope-intercept form, where the slope is given by $m = 3.054$, and the y-intercept by $b = 70.221$ (See Table 7.2 for a slope comparison against the other two studies). As it can be observed, the intended correlation between driving styles and heart rate R is 0,103 in the urban scenario. The correlation value is relatively low when compared to the result for highway routes (where $R = 0.416$), but it is significantly greater than the correlation obtained in suburban driving (where $R = 0.069$), as shown in the following sections.

The slope of the line also follows this trend: the slope of the equation for highway driving is the largest ($m = 8.474$), followed by the urban cases where $m = 3.054$.

These results were mostly expected, since driving in urban areas is more stressful in most cases than suburban driving by requiring more driver attention and greater control due to the presence of other drivers. Surprisingly, the greater slope was achieved for the equation describing the highway scenario. *A priori*, highway driving is less stressful, but we should point out that the total time of the route is much higher, resulting in driver tiredness, as discussed in more detail in Section 7.4.3.

We find that the difference in heart rate between quiet and aggressive behavior is 3.05 %, about three beats per minute, as shown in Figure 7.4.a.

7.4.2 Driving Tests in Suburban Areas

In this second part of the analysis, we study two specific routes chosen from the set of fourteen routes analyzed, specifically those corresponding to suburban driving. Notice that such discrimination is possible since the DrivingStyles platform, in addition to analyzing the driver behavior, is also able to compute the route type.

The two analyzed routes have a total duration of 82 minutes and 33 seconds, circulating at a mean speed of 34.26 km/h and a maximum speed of 112 km/h , covering a total of 28.25 km ; this corresponds to 4353 driving behavior-type samples, also including data about the driver's heart rate that we use for statistical analysis (see Tables 7.1 and 7.3).

The results are not too noticeable compared to the previous case (urban driving, see Section 7.4.1) and the next one (highway driving, see Section 7.4.3). In this case we find that, when the driver has a normal behavior, the heart rate often becomes lower with respect to a quiet behavior, as can be observed clearly in the box and whisker plot (see Figure 7.4.b) which displays the differences in terms of heart rate between quiet, normal, and aggressive driving behavior. This means that differences between quiet and normal behaviour are not too significant, and that, at times, quiet driving is not a driver decision, but it is rather forced due to external conditions (e.g. traffic jam), which can cause stress. For this test subject, though, aggressive driving provoked a slightly increased heart rate.

7.4.3 Driving Tests in Highways

In this third part of the analysis, we evaluate three routes with a total duration of 3 hour and 46 minutes, and where the mean speed is of 100.18 km/h , the maximum speed is of 134 km/h , and the total distance covered is 394 km . In total, we have 13664 driving behavior-type samples, also including data about the driver's heart rate that we use in the statistical analysis (see Tables 7.1 and 7.3). The most relevant characteristics of the three highway routes can be seen in Table 7.4.

Table 7.4: General characteristics of highway routes.

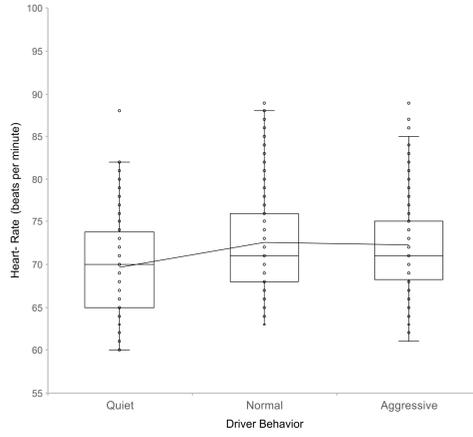
	<i>Time</i>	<i>Start Time</i>	<i>Arrival Time</i>	<i>Max Speed</i>	<i>Mean Speed</i>
Highway 1	81.44	09 : 45 : 33	11 : 07 : 18	134 <i>km/h</i>	96.99 <i>km/h</i>
Highway 2	79.32	18 : 35 : 25	19 : 54 : 58	120 <i>km/h</i>	100.67 <i>km/h</i>
Highway 3	74.42	16 : 16 : 11	17 : 30 : 55	119 <i>km/h</i>	103.16 <i>km/h</i>

Results show that the linear trendline remains clearly positive, being the slope value $m = 8.474$, and the coefficient of determination $R = 0.416$. This correlation coefficient indicates that there is a clear lineal relationship between aggressive driving behavior and heart rate. Highway driving is the one that most clearly shows the impact of driving aggressiveness on heart rate values. It should also be denoted that the routes studied in this section are all long routes, being able to produce an increased heart rate due to the fatigue caused by prolonged driving.

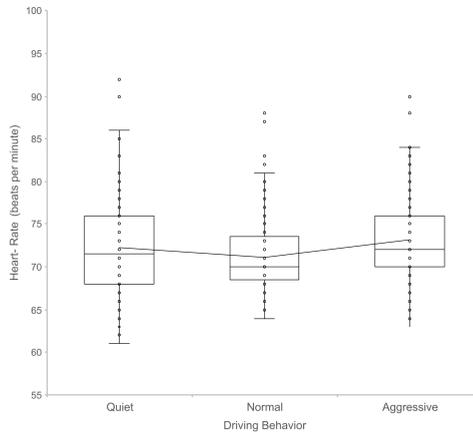
We now proceed to study these three routes in more detail, highlighting the particularities of each of them:

1. In one of the highway routes the driver had an unexpected stop requiring getting out of the vehicle, which affected his heart rate, as we can see in Figure 7.5.a. Specifically, the heart rate increased until reaching 117 bpm, taking several minutes to regain a normal heart rate. However, as we can observe in the same chart, the system does not detect any change in the driving behavior. Another thing that can be observed in the chart is that the system detects a quiet behavior at the end of the route because the driver is approaching the end of his trip. In fact, it can clearly be observed how the heart rate decreases slowly, being that the DrivingStyles system detects the behavior becoming *quiet*, and graphically represents these final points in green.

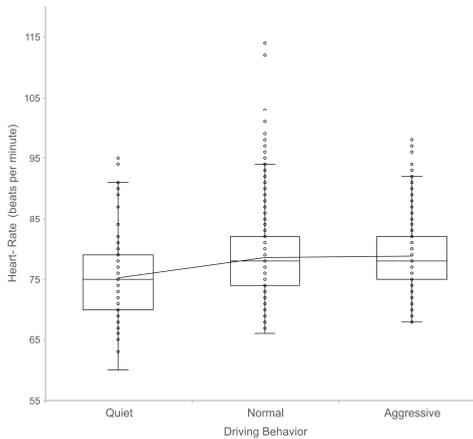
2. In the second long-duration route (see Figure 7.5.b) it can be seen how the heart rate decreases gradually due to the driver sleepiness, and at a certain moment the driver decides to turn on loud music to stay awake. The chart shows how the heart rate has a peak and reaches 91 bpm, but, as can be seen, the system does not detect a driver behavior change. Thus, this example highlights that indeed loud music can help at increasing the heart rate, preventing sleepiness and reducing the chances of having an accident, without necessarily causing changes in the driving behavior.
3. In the last studied route, we can see that there is a period of the route where the driver is clearly stressed and altered; the system detects that it has an aggressive behavior during 17 minutes, as can be seen in Figure 7.5.c. The driver then calms down and the heart rate normalizes for the rest of the trip.



(a) Urban routes.



(b) Suburban routes.



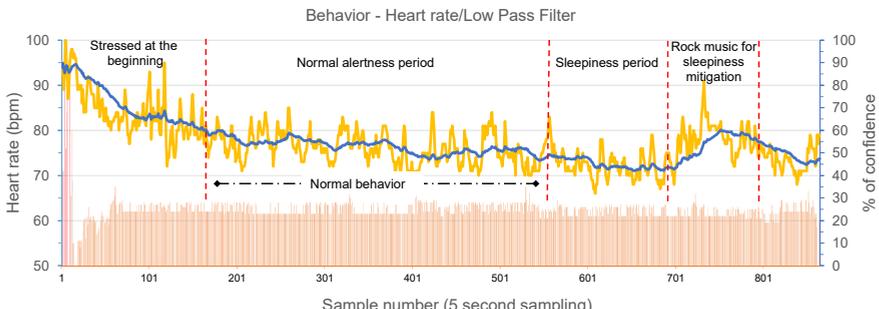
(c) Highway routes.

Figure 7.4: Box and whisker plot for heart rate vs. driving behavior.



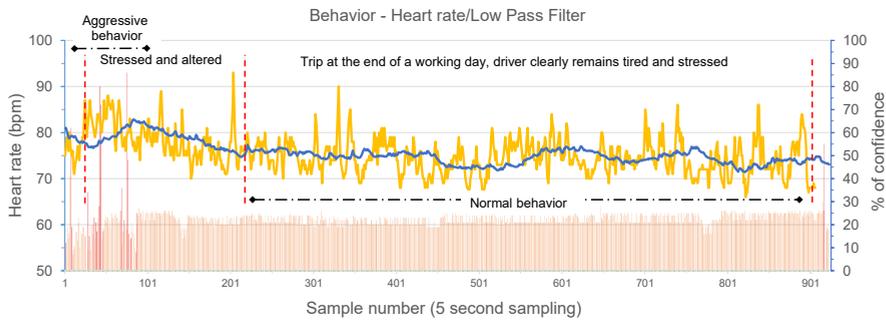
Behavior: Quiet Normal Aggressive Heart Rate (bpm) Heart rate low pass filter

(a) Highway 1: Heart rate increases to 117 bpm due to an unexpected stop.



Behavior: Quiet Aggressive Normal Heart Rate (bpm) Heart rate low pass filter

(b) Highway 2: Heart rate decreases gradually due to driver sleepiness.



Behavior: Quiet Normal Aggressive Heart Rate (bpm) Heart rate low pass filter

(c) Highway 3: Route where the driver is clearly stressed and altered.

Figure 7.5: Heart rate evolution along the three highway routes

7.5 Conclusions and Future Work

In this chapter we studied the correlation between driver heart rate and driver behavior. We based our study on the use of our novel DrivingStyles architecture, that combines new technologies such as smartphones and wearable body sensors with modern software implementations of artificial neural networks.

The results of the present study indicated that aggressive driving caused an increase in the heart rate, being able to rise it by up to 2.5 % and 3 % beats per minute. In statistical terms, we also found that, as the number of samples increases, the correlation between the driver behavior and heart rate becomes significant. In terms of highway driving, we find that in general fatigue can affect driving behavior, although in the initial part of the trip the driver tends to progressively relax. The same occurs near the end of the trip. Since this study has been conducted taking a middle-aged male subject as reference, in future works we will expand the scope to more drivers of different ages, which will help at covering a wider range of possible cases.

7.6 Acknowledgments

This work was partially supported by the Ministerio de Economía y Competitividad, Programa Estatal de Investigación, Desarrollo e Innovación Orientada a los Retos de la Sociedad, Proyectos I+D+I 2014, Spain, under Grant TEC2014-52690-R.

Summary of Achievements

In this chapter of the thesis we bring together all the achievements accomplished throughout this thesis. As a summary of this section, Figure 8.1 shows a reduced scheme with all the elements involved in the architecture that we have proposed in this thesis. We also use this section to develop those topics that were not fully developed in the different published papers.

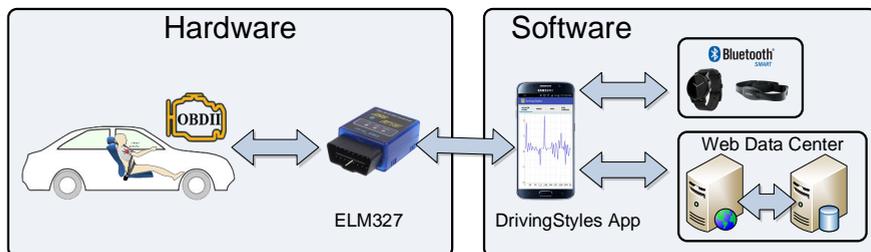


Figure 8.1: Overall hardware and software system architecture.

8.1 Neural Network Tuning

One of the stages of data preprocessing consists in selecting the possible input variables for the Neural Network (route type and driving style). If we reduce too much the number of input variables, we will have fast learning cycles, but at the cost of decreasing the network performance due to the lack of important variables for the classification. On the contrary, we can include too many input variables, some strongly correlated and therefore redundant, resulting in networks that provide all the information we have but which are overloaded with very long cycles, and sometimes low performance because it offers information indiscriminately.

Table 8.1: Neural Networks input variables.

Input	Description
Input 1	<i>Average speed.</i>
Input 2	<i>Standard deviation speed.</i>
Input 3	<i>Average acceleration.</i>
Input 4	<i>Standard deviation acceleration.</i>
Input 5	<i>Rpm Average.</i>
Input 6	<i>Rpm Standard deviation.</i>

Table 8.1 shows the input variable we selected to determine the DrivingStyle. The subset is not minimal, but it represents a compromise between a manageable (not too large) number of variables, and an acceptable network performance. Notice that the selected variables can be obtained from the Electronic Control Unit (ECU). Three of those possible variables that we have chosen to train the neural network are speed, vehicle acceleration, and engine revolutions (the mean and standard deviation of the three variables). While we were able to measure the value of this variables in all the vehicles used for testing, other variables such as the position of the accelerator’s pedal, which we also considered initially that could provide useful information for the neural network training, had to be discarded because not all the ECUs return this value (see Table 8.1).

The steps to generate the two neural networks that will be applied later to characterize the route and driver’s behavior are the following:

1. First, we create the empty neural network, defining the number of inputs that in our case are six (average speed and its standard deviation, mean acceleration and its deviation, and the mean rpm and its standard deviation). In both neural networks, a greater number of hidden nodes allows to improve the success rate, but has the negative effect of increasing the response time. The three outputs in the characterization of the road type are: city, suburban, and highway, and for the neural network that characterizes the style of driving of the user the possible outputs are: quiet, normal, and aggressive; We have to obtain two neural networks separately, but that share the same input variables.
2. Once we create the empty neuronal network with six input neurons, nine hidden neurons, and three output neurons, we choose the backpropagation learning function [14]. We obtained good results to characterize the routes and driving style. There are other learning functions such as backprop-momentum, hebbian, delta-rule, etc., but from the beginning we chose backpropagation because it gives very good results in classification problems like the one we are describing [14, 26].
3. The data of each input parameter are normalized between 0 and 1. This normalization is done with the whole range of possible values, i.e., the minimum and maximum value of each variable of the input domain used in the creation of the training and test files (see Figure 8.2). The normalized value of the input variable x of the pattern p , the original value of the variable x of that pattern p , and $\min(x)$, $\max(x)$, which refer to the minimum and maximum values of that variable. One of the reasons for this normalization is that the activation of the output neuron of the multilayer neuron network can only reach values between 0 and 1. By normalizing the data in this way all values will be within the same range, and the results obtained by the different models can be compared directly. Once we normalized the input variables we can already generate the validation and test files.
4. Then, the weights are generated in a totally random way for the neural network links. For this purpose, we use the `randomize weights` function to perform a random initialization of the neural network weights, by default between -1 and 1.

The neural network is saved in the initialized state to make sure that all the tests that are performed depart from the same initial weights, thus avoiding chance, and it is only necessary to load the standard validation

```
SNNS pattern definition file V3.2
generated at 09-07-2012 15:13:15
# Input pattern 1:
0.433333 0.0333333 0.029918 0.353793 0.434195 0.00764575
# Output pattern 1:
0 0 1
# Input pattern 2:
0.470833 0.0544862 0.117623 0.471716 0.449088 0.0472043
# Output pattern 2:
0 0 1
.....
```

Header

N° Training Patterns

N° Inputs

N° Outputs

Urban Suburban Highway

Quiet Normal Agressive

Figure 8.2: Neural network training file.

and test files generated previously from the web application to begin the training.

5. Training; once the JavaNNS application is configured¹, we start the training of the neural network, so that the application calculates the average quadratic error. In each learning iteration we need to have the error window visible and the weights window. We set the learning rate or learning ratio to 0.2, and we modify it to observe how it affects the error of the neural network. The higher the learning rate, the greater the change in the weights at each iteration, so learning becomes faster; on the other hand, it can cause unwanted oscillations in the network. Also the number of learning cycles that are carried out in the tests will be adjusted.

A more detailed explanation of this point can be seen in Chapter 3, where we present the paper published as "DrivingStyles: a smartphone application to assess driver behavior".

¹<http://www.ra.cs.uni-tuebingen.de/software/JavaNNS/manual/JavaNNS-manual.html>

8.2 Data Center

The implemented Data Center is another key element of our architecture. It allows visualizing to the drivers, routes or segments of their routes, and the system automatically returns the type of road, and the associated driving style. It was also used at the beginning of the project to select the most interesting driver's route based on, data received from the drivers who participated in our study, to train the two neural networks detecting driving behavior and route type.

As the architecture evolved, more elements were incorporated that gave more functionality to the Data Center, such as the number of plots that can be displayed (a total of 17 plots, see Figures 8.3(b) and 8.3(c)), and the data base was expanded, tripling the database fields (a total of 37 variables sent from the mobile and smartwatch).

In our implementation, we use the Apache server (v.2.0.48) from the Apache Software Foundation (ASF) ² [51], as the front-end Web server. For the middle-tier, we use the J2EE application server. It is integrated with Tomcat Web container v.8.5.11, which is compatible with the Servlet 3.1 and JavaServer Pages 2.1(JSP)[52] specifications.

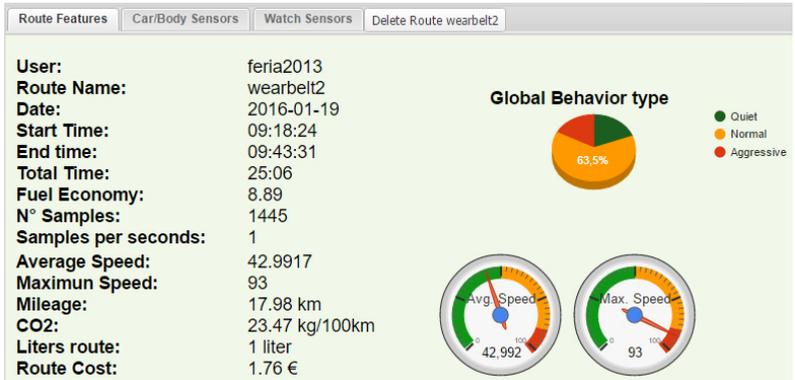
Tomcat also provides the communication mechanisms with the Apache Web servers in the front-end tier. We use Sun JVM from JDK v.1.5.0 04 for Linux. For the backend tier, we use a machines running MySQL v.5.6.35³ [53] database servers.

As a content management system (CMS) we used Joomla®⁴, which is an open source platform on which Web sites and applications can be created. Joomla connects the site to a MySQLite or MySQL database in order to make content management and delivery easier for both the site manager and web visitors.

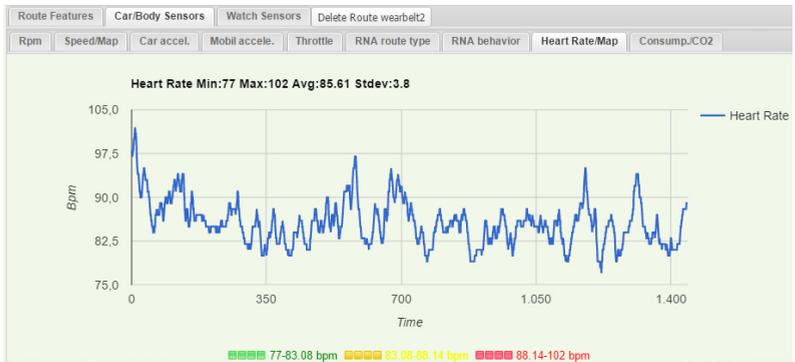
²<http://httpd.apache.org>

³<http://www.mysql.com>

⁴<https://www.joomla.org>



(a) Route Feature Summary.



(b) Route plots Car/Body Sensors.



(c) Route plots Watch Sensors.

Figure 8.3: Data Center Plots.

8.3 Android Application

In our DrivingStyles architecture we first moved all computational load to the server. Once the different estimation algorithms were debugged, especially those concerning the neural networks, we implemented the main logical element that we had developed on the server also in the Android app. This way, DrivingStyles handles the complexity of real-time applications by sharing it with the Data Center.

Figures 8.4 and 8.5 shows the Android application run-time flowchart. As we can see the application has several well defined blocks. On the one hand we have the data capture part, either from the vehicle's ECU, a heart rate band to get the heart rate, or a smartwatch to communicate with the driver. Moreover, the graphical visualization includes the plots and the main screen, the configuration part, and finally the data sending part.

The first step for a new user is to register at the DrivingStyles Data Center⁵. After connecting to the Bluetooth ELM327 interface we only have to match the phone with the car OBD-II; this way we enable a communications path between the vehicle and the DrivingStyles data center.

To end this section about the Android App, Figure 8.6 summarizes most of the application functionalities. The application was created from the outset with a double functionality, the first was to obtain the ECU data, to be able to perform the data mining that has been used in this thesis, and a second market oriented functionality to achieve the highest number of drivers. The application has now more than 6,000 installations downloaded from Google Play⁶.

Among other functionalities, we can highlight the set of plots of all parameters analyzed:

1. Obtained directly from the ECU, as acceleration, rpm, speed, and those the parameters indirectly calculated from variables obtained from the ECU, such as consumption.
2. Others plots obtained through the use of Neural Networks, behavior and route type.
3. Plot and map related to heart rate.

⁵<http://www.drivingstyles.info>

⁶<https://play.google.com/store/apps/details?id=com.driving.styles>

In addition, the driver can visualize, among other variables, instantaneous fuel consumption, heart rate, ECU variables, and driving style, all in real-time. It is also possible to configure the sensitivity levels, where the system will notify the user when a certain threshold of aggressiveness is exceeded; this is achieved either by vibrations and acoustic signals in the mobile device, or by messages and vibrations in the wear watch.

Currently, the application supports more than 4586 devices. The current version of DrivingStyles is the Android 5.0 Lollipop (API Level: 21) that supports more than 4586 different devices. The application evolution can be seen in the Table 8.2.

Table 8.2: DrivingStyles evolution version.

Ver.	Ver. name	Uploaded	Status	API levels	SDK	Size
5	2.0.1	5 Feb 2016	in Prod	21+	21	9.02 MB
4	2.0	21 Jan 2016	Unpublished	21+	21	8.97 MB
3	1.2	11 Jan 2015	Unpublished	14+	21	4.96 MB
2	1.1	12 Dec 2012	Unpublished	8+	–	1.58 MB
1	1.0	21 Sep 2012	Unpublished	8+	–	1.57 MB

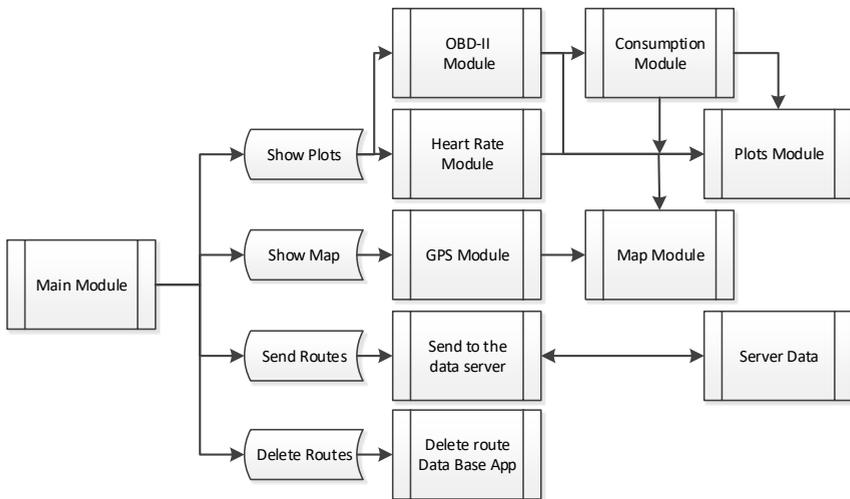


Figure 8.4: Android application's modules flow diagram.

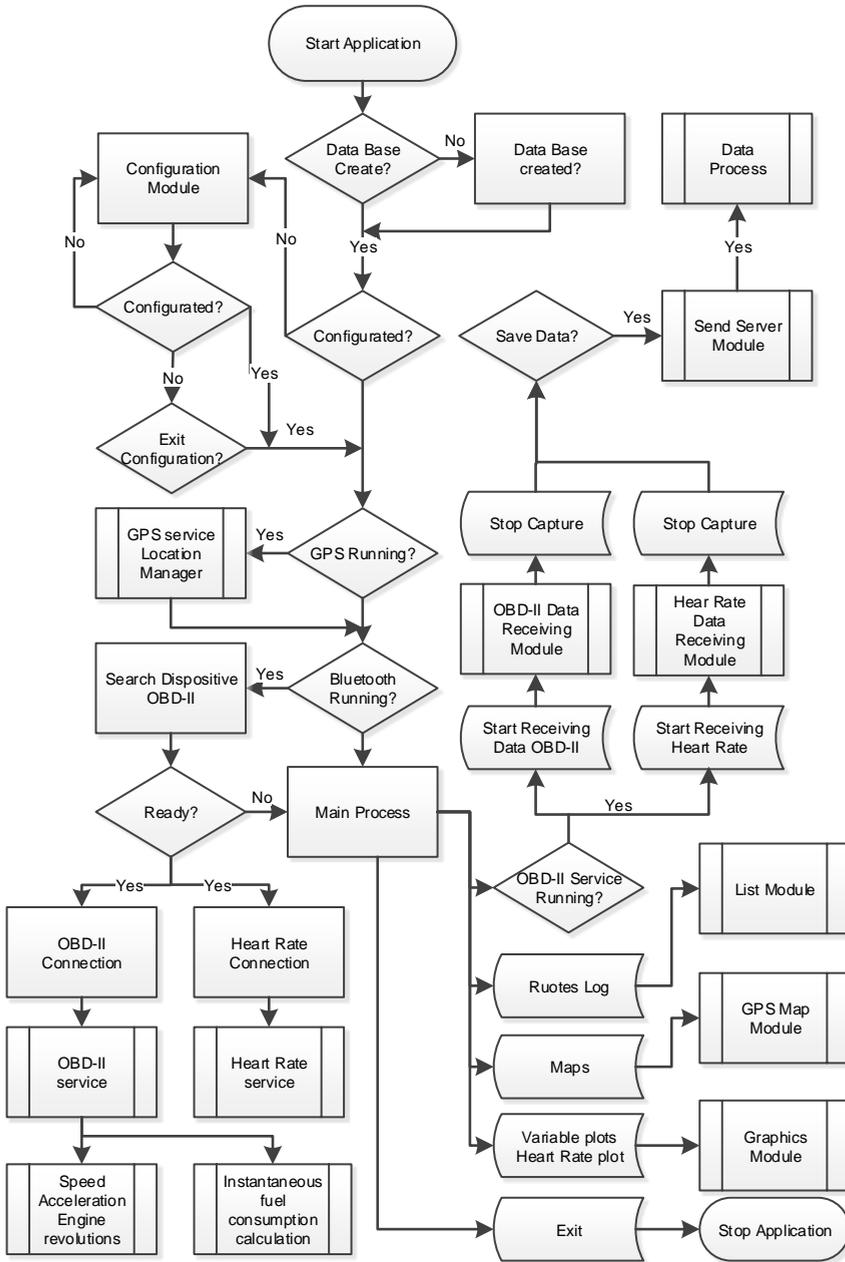


Figure 8.5: Android application's flow diagram.

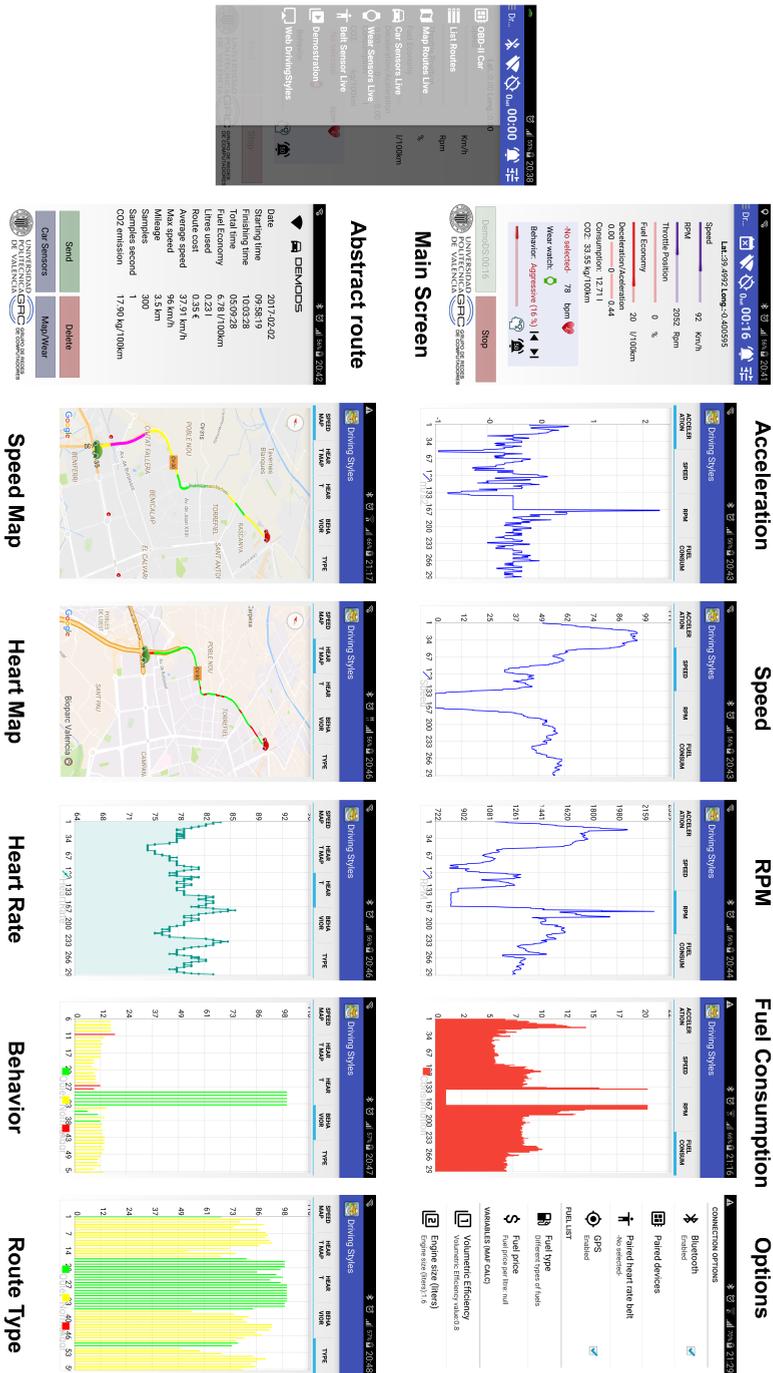


Figure 8.6: Android application's screen.

8.4 Fuel Consumption and Greenhouse Gas Emissions

We also investigate to find out how driving style affects fuel consumption and greenhouse gas emissions. Since our DrivingStyles architecture returns driver behavior in real-time, we need to know the vehicle’s instantaneous fuel consumption in order to carry out the study. The problem we run into is that the vehicle’s ECU does not directly provide the instantaneous fuel consumption, and so it must be calculated from other parameters returned by the ECU and parameters related to the design of the vehicle engine.

In Figure 8.7 we show the four methods that our DrivingStyles architecture used to calculate instantaneous fuel consumption, using the appropriate method.

The Android application is able to analyze which variables can be obtained from the ECU of our vehicle, and thus the calculation of instantaneous consumption.

Once the instantaneous fuel consumption and driving behavior were obtained in real time, we were able to collect thousands of samples and perform a set of tests to carry out the study.

As it has been presented in Chapters 4 and 5 of this thesis, adopting an efficient driving style allows achieving fuel savings ranging from 15% to 20%. As we see in Figure 5.6 of chapter 5, an increase in aggressiveness driving behavior causes fuel consumption and CO₂ to increase significantly. Likewise, Figure 5.7 shows how an aggressive behaviors causes a greater fuel consumption that normal or quiet behavior. An aggressive driving style always results in a greater energy consumption and CO₂ emissions.

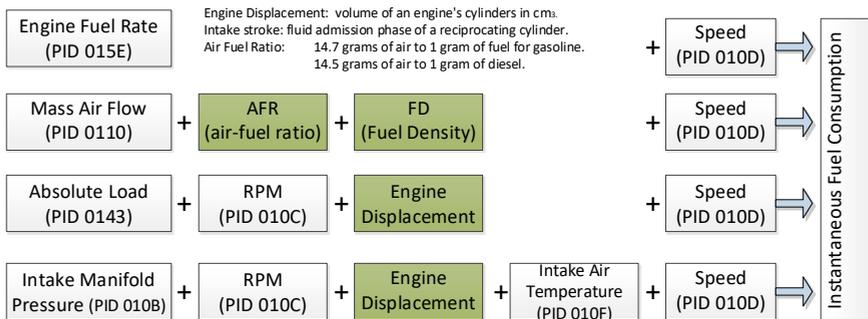


Figure 8.7: Methods used in Android app to calculate the instantaneous fuel consumption.

8.5 Heart Rate Analysis

After demonstrating that the driving style significantly affects fuel consumption and therefore CO_2 generation, we studied how the driving style affects the driver's heart rate.

The first devices used to capture the heart rate were wrist devices, specifically the Motorola 360 smartwatch. In this model the back of the watch hosts the heart rate sensor. Its heart rate monitor periodically checks the pulse in the background at regular intervals, but these intervals are too long for our study requirements, and also cause a high battery consumption when the heart rate measurement is activated. Therefore, we opted for the heart rate belt device to get the pulse of the heart.

Despite the aforementioned drawback, we continued with the development oriented to the smartwatch, capturing of all the sensors that it can offer. The sensors can be seen in Figure 8.3(c) and Figure 8.6. Also, we use the smartwatch for sending warnings and alerts to the driver at the moment that he/she exceeds certain driving aggressivity thresholds.

Subsequently, we use a new type of device to obtain the driver's heart rate with a sampling rate of at least 1 sample per second. This sampling frequency, appropriate for our study, was obtained by using a heart rate chest strap, which uses the Bluetooth 4.0 Low Energy (BLE) technology.

We proceeded, as in the case of fuel consumption, to obtain thousands of samples with the driver's heart rate and driving style. The results obtained can be seen in Chapters 6 and 7. Figure 7.4, shows how the heart rate changes according to the driving behavior and how in all three groups of routes studied (urban, suburban, and highways), the heart rate is significantly higher when the behavior is aggressive. In Figure 7.5, the analysis is performed on three specific routes, each with its own characteristics, but in all of them we observe that when the driving behavior is quiet the driver's heart rate decreases, and on the contrary, when the driving behavior is more aggressive, or there is a change in driving, the heart rate increases. The results of the study indicated that aggressive driving causes an increase in the heart rate, being able to rise it by up to three beats per minute on average.

Conclusions, Publications and Future Work

9.1 Conclusions

The results of the works of this thesis have been detailed in the papers that have been published in different journals and international conferences which address the related topics. These papers have been included as the central chapters of this document, as each one of them addresses one part of the key problems in which the main thesis objective has been decomposed. We did not want to restrict our research to a set of theoretical results, since they can be applied to real deployments. This is why we have created open-source developments that can be used by others.

The main objective of this thesis was to create the DrivingStyles architecture, that allows analysing the driver's behavior, and how this behavior affects fuel consumption and can cause an increase or decrease in the driver's heart rate.

The proposed architecture is able to contribute at making driving safer, and current technologies allow us to make a more decisive contribution to safe driving by allowing to do a real-time analysis of driving behaviour, which also enables the generation of real-time alerts if necessary. Overall, we consider

that this thesis, as well as other studies that are currently being carried out, will help at providing safer driving, and thus save lives.

The conclusion of this thesis are the following:

1. Development of a client-side application supporting vehicle interaction. The first thing we work out was an approach to determine the driving style through the development of an application that uses artificial intelligence techniques to classify driving styles. To that purpose it relies on values such as speed, acceleration, and engine revolutions per minute, all obtained from the vehicle's ECU.

To meet the aforementioned goal we implemented a training algorithm based on neural networks, able to characterize the current road type and the driver behavior. The application has been developed for the Android platform¹ to promote free software, but it could have been done in iOS² as well, reminding that between both they encompass more than 90% of the market.

Using the OBD-II Bluetooth interface (ELM327) [9, 10] the application obtains information such as speed, acceleration, rpm, accelerator pedal position, along with the vehicle geographic position using GPS (Global Positioning System). Once the information is collected the user will send the route data to the data center for analysis, being it also analyzed in Real Time within the developed application.

2. Creation of a Data Center that collects all data sent by the drivers. We have advocated in this case for developing the server with free software tools like Apache, Java, PHP, and Joomla. After collecting the data of the drivers we use the most representative ones to train the Neural Networks, that constitutes the third objective of this thesis.
3. Analysis of stored data on the server using Neural Networks. The obtained data are used to train a Neural Network through the backpropagation algorithm. This technique provides good results in classification problems such as the one addressed in this thesis. At the same time, a second version of the Android app was implemented with the same Neural Network.
4. We developed modules able to compute the instantaneous fuel consumption as well as the greenhouse gases emitted. From these two parameters

¹<http://www.android.com>

²<http://www.apple.com/ios>

we were able to analyze how the different behaviors in the driving style result in a greater or smaller fuel consumption and, consequently, the environmental impact that this entails. We demonstrate that, with a quiet driving style, we can reduce fuel consumption between 15 % and 20 %.

5. The last contribution of this thesis includes a discussion of the results obtained regarding the heart rate, where we find that it is also affected by the driving style. In fact, we demonstrate that an aggressive driving style can raise the heart rate between 2.5 % to 3 %. We also perform a comparative analysis between different routes and different driving styles, evaluating the changes produced in the drivers' heart rate.

9.2 Publications

In the framework of this thesis, we have written several papers, with the collaboration of other researchers. These papers describe the solutions proposed to the different problems that have been exposed during the research phase.

Journals

- Javier E. Meseguer, Carlos T. Calafate, Juan Carlos Cano "On the correlation between heart rate and driving style in real driving scenarios" *Mobile Networks & Applications*. Article 833, 16 February 2017. DOI: 10.1007/s11036-017-0833-x JCR Impact Factor 1,538 Q2.
- Javier E. Meseguer, C. K. Toh, Carlos T. Calafate, Juan Carlos Cano, Pietro Manzoni "DrivingStyles: A mobile platform for driving styles and fuel consumption characterization" *Journal of Communications and Networks*. Accepted 28 November 2016. JCR Impact Factor 2015 = 1,462 Q2.

International Conferences

- Javier E. Meseguer, Carlos T. Calafate, Juan Carlos Cano, Pietro Manzoni (2013) "DrivingStyles: A Smartphone Application to Assess Driver Behavior" 18th IEEE symposium on Computers and Communications ISCC 2013 July 7-10, Split, Croatia. Pages 535-540, ISBN: 978-1-4799-3755-4 CORE: B.
- Javier E. Meseguer, Carlos T. Calafate, Juan Carlos Cano, Pietro Manzoni "Characterizing the Driving Style Behavior using Artificial Intelli-

gence Techniques" 38th IEEE Conference on Local Computer Networks. LCN 2013 October 21-24, Sydney, Australia. ISBN/ISSN: 978-1-4799-0536-2 CORE: A.

- Martínez Tornell, Sergio, Javier E. Meseguer, Carlos T. Calafate, Juan Carlos Cano, Pietro Manzoni "Smartphones as the keystone for leveraging the diffusion of ITS applications" 9th European Congress and Exhibition on Intelligent Transport Systems and Services (ERTICO-ITS in Europe 2013).
- Javier E. Meseguer, Carlos T. Calafate, Juan Carlos Cano, Pietro Manzoni (2015) "Assessing the Impact of Driving Behavior on Instantaneous Fuel Consumption" 12th IEEE Consumer Communications and Networking Conference CCNC 2015 January 9-12, Las Vegas Nevada, USA. Pages 443-448, ISBN/ISSN: 978-1-4799-6390-4/2331-9860 CORE: B.
- Javier E. Meseguer, Carlos T. Calafate, Juan Carlos Cano "DrivingStyles: Assessing the Correlation of Driving Behavior with Heart Rate Changes" GOODTECHS, 2nd EAI International Conference on Smart Objects and Technologies for Social Good Best. November 30 – December 1 2016, Venice. Italy. To be published. **Best Paper Award**

9.3 Products

Apart from the scientific papers, we have created a set of products that gather the ideas and the implementations used for the validation of the research works. Apart from the implementations that have been made to validate the ideas and algorithms described in the papers, an extra effort has been invested for the creation of final products that can be exported to other deployments. All the software development that has been implemented as part of the DrivingStyles architecture, has been deposited in the service of patents of the Universitat Politècnica de València³ in the area of knowledge of Computer Science, Information Technology, Electronic Technology and Communications with the item KNS-R-16474-2013.

The Android application can be free downloaded from the DrivingStyles website⁴ or from Google Play⁵. Our URL is <http://www.drivingstyles.info>.

³<http://www.upv.es/entidades/SGI/indexc.html>

⁴<http://www.drivingstyles.info/index.php/en/>

⁵<https://play.google.com/store/apps/details?id=com.driving.styles>

9.4 Open Research Issues

With respect to future lines of work that have emerged under the framework of this thesis, we can distinguish several areas.

On the one hand, it would be really interesting to implement different Artificial Intelligence (AI) techniques such as expert systems, Bayesian networks, or artificial intelligence based on behaviors, and perform a comparative study with respect to the backpropagation technique we have used on the effectiveness of these techniques in obtaining solutions to the classification problems addressed in the thesis.

On the other hand, all the information gathered could be used by different applications oriented to improve road safety. This could be achieved through a Big Data analytics platform, and in this regard data mining plays an important role in the analytical discovery process, becoming a key element to predict future outcomes and help to save lives. We intend to extend this platform to provide traffic recommendations based on real-time feedback about the congestion of different routes.

Another very interesting area of research is preventing driver sleepiness. The cost of having a security system based on our architecture would be nearly zero, merely requiring adjusting the detection of driving styles algorithms. Some high-end manufacturers are adding sleep sensors to their cars right at the factory (e.g. Lexus brand), that placed a camera in the dashboard that tracks the driver's face, rather than the vehicle's behavior, and alerts the driver if his or her movements seem to indicate sleep; similarly, Saab uses two cameras in the cockpit to monitor the driver's eye movement, and alerts the driver with a text message in the dash, followed by a loud audio message. In general, we consider that vehicle/driver interactions are worthy of more scrutiny, and we are confident that this topic will receive much attention in a near future.

Glossary

AIT	Intake Air Temperature. 39, 55, 57
ANN	Artificial Neural Network. 10
API	Application Program Interface. 26
ASCII	American Standard Code for Information Interchange. 9
ASF	Apache Software Foundation. 105
BLE	Bluetooth Low Energy ,Bluetooth LE. 77, 113
BMP	Bitmap (file name extension). 15
BMW	Bayerische Motoren Werke. 56
BPM	Beats Per Minute. 75, 90
CAN	Controller Area Network. 6
CDMA	Code Division Multiple Access. 15
CMS	Content Management System. 27, 43, 59, 89, 105
CSS3	Cascading Style Sheet 3. 15
CU	Control Units. 22
DTC	Diagnostic Trouble Codes. 7
ECU	Electronic Control Unit. vii, viii, 20, 25, 28, 35, 38, 44, 54, 55, 90, 102
EDGE	Enhanced GPRS. 15
ELM327	PIC microcontroller ELM Electronics. 8, 24, 58
GCM	Google Cloud Messaging. 15

GIF	Stands for Graphics Interchange Format version 5. 15
GM	General Motors. 56
GPS	Global Positioning System. 24–26, 41, 58
GSM	Global System for Mobile Communications. 15
H.263	Video compression standard. 15
H.264	MPEG-4 Part 10, Advanced Video Coding (MPEG-4 AVC). 15
HRM	Heart Rate Monitor. 76, 90
HTML5	HyperText Markup Language 5 (and file extension). 15
HTTP	Hypertext Transfer Protocol (world wide web protocol). 27, 43, 59, 89
IDEN	Integrated Digital Enhanced Network. 15
iOS	iPhone Operating System. 14
IoT	Internet of Things. 9
ISO 14230	KWP200 Keyword Protocol 2000. 8
ISO 15031	Communication between vehicle and external equipment for emissions-related diagnostics. 7
ISO 15765	CAN ISO9141. 8
ISO 9141	In European; Asian; and Chrysler vehicles. 6, 8
ITS	Intelligent Transportation Systems. 54
J1962	Female Connector OBD-connector met Paars Core appropriate for the auto's Obdii Interface. 7
JDK	Java Development Kit. 105
JPEG	Stands for Joint Photographic Experts Group. 15
JVM	Java Virtual Machine. 105
LTE	Long Term Evolution is part of the GSM. 15
MAF	Mass Air Flow. 39, 45, 46, 57, 62
MAP	Manifold Absolute Pressure. 39, 46, 55, 58
MIL	Malfunction Indicator Lamp. 5, 6
MMS	Multimedia Messaging Service. 15
MPEG-4	Motion Picture Experts Group Layer-4 Video. 15
MPG	Miles Per Galon. 44, 61
MySQL	My Structured Query Language. 105
NFC	Near Field Communication. 15

OBD-I	On Board Diagnostics I. 6
OBD-II	On Board Diagnostics II. vii, 5, 6, 20, 22, 38, 55, 72, 73
OBU	On Board Unit. vii, 20
PHP	Hypertext Preprocessor. 23, 57
PIC	Peripheral Interface Controller. 8
PID	Parameter IDs. 45, 62
PID 010B	PID Intake Manifold Pressure. 46, 63
PID 010C	PID RPM. 45, 46, 63
PID 010D	PID Fuel. 44, 62
PID 010F	PID Intake Air Temperature. 46, 63
PID 0110	PID Mass Air Flow. 45, 62
PID 0143	PID Absolute Load. 45, 63
PID 015E	PID Engine Fuel Rate. 44, 62
PNG	Portable Network Graphics (graphic file standard/extension). 15
RPM	Revolutions Per Minute. 28, 46, 58
RS232	Recommended Standard 232. 8
SAE	Specifies requirements for a vehicle data communications net- work. Compliance committee recently formed (J1699). 6, 8
J1850	
SAE	Glossary of terms for vehicle networks. 8
J1939	
SMS	Short Service Message. 15, 22
SSE	Sum of Squared Errors. 31
UMTS	Universal Mobile Telecommunications System. 15
URL	Uniform Resource Locator. 27, 43, 59
V2V	vehicle-to-vehicle. vii
WAV	Windows Wave (audio format/file extension). 15
Wi-Fi	IEEE 802.11b wireless networking. 15
WiMAX	Worldwide Interoperability for Microwave Access, Inc. (IEEE 802.16 wireless broadband standard). 15

Bibliography

- [1] Keyword Protocol 2000. “International Organization for Standardization”. In: *International Organization for Standardization* (1999) (cit. on pp. 5, 20, 72, 75, 86, 90).
- [2] Directive 98/69/EC. *Passenger cars and light commercial vehicles*. European parliament and the council, 1998 (cit. on p. 6).
- [3] D. John Oliver. “Implementing the J1850 Protocol”. In: *Intel Corporation Obtainable from <http://developer.intel.com/design/intarch/papers/j1850wp.htm>*. (1988) (cit. on pp. 6, 8).
- [4] ISO. *ISO 9141-2 1994/Amd 1:1996-1196*. Tech. rep. International Organization for Standardization, 1996 (cit. on pp. 6, 8, 72, 75, 86, 90).
- [5] Pat. US 20140086242 A1 J1962 Device for transmitting signals from a vehicle. 2002 (cit. on p. 7).
- [6] ISO 15031-4:2014. “Communication between vehicle and external equipment for emissions-related diagnostics”. In: (2004) (cit. on p. 7).
- [7] ISO. “International Organization for Standardization”. In: *Road vehicles, Diagnostic systems, Keyword Protocol 2000*, (1999) (cit. on pp. 8, 38).

- [8] ISO/TC 22/SC 31 Data communication. “ISO 15765: Road vehicles, Diagnostics on Controller Area Networks (CAN)”. In: *International Organization for Standardization* (2004) (cit. on p. 8).
- [9] Elm Electronics Circuits. *ELM327DS. OBD to RS232 Interpreter*. Ed. by ELM327DS. Hobbyist., 2010 (cit. on pp. 8, 20, 24, 38, 75, 90, 116).
- [10] Awais Khan and Aii Raza. “Development of an on board diagnostic kit for troubleshooting of compliant vehicles”. In: *Emerging Technologies ICET IEEE 9th International Conference* (2013) (cit. on pp. 8, 72, 75, 116).
- [11] ELM327DS. “OBD to RS232 Interpreter”. In: *Elm Electronics – Circuits for the Hobbyist* (208) (cit. on pp. 8, 86, 90).
- [12] Datasheet. “ELM327 OBD to RS232 Interpreter”. In: *ELM Electronics* (2010) (cit. on p. 8).
- [13] S. Haykin. “Neural Networks: A Comprehensive Foundation”. In: *NY: Macmillan, p. 2* (1994) (cit. on pp. 10, 64, 87).
- [14] R. Hecht-Nielsen. “Theory of the backpropagation neural network”. In: *Neural Networks IJCNN, International Joint Conference* (1989) (cit. on pp. 12, 23, 28, 41, 57, 65, 87, 103).
- [15] Igor Fischer et al. “JavaNNS (Java Neural Network Simulator)”. In: *User Manual, Version 1.1* (2000) (cit. on pp. 12, 29, 66).
- [16] <http://www-ra.informatik.uni-tuebingen.de/SNNS> Stable release 4.3 / July 6, 2008 (cit. on pp. 12, 13, 31).
- [17] Kanok Boriboonsomsin, Alexander Vu, and Matthew Barth. “Co Eco-Driving: Pilot Evaluation of Driving Behavior Changes among U.S. Drivers”. In: *University of California Transportation Center* (2010) (cit. on pp. 21, 39).
- [18] E.Eriksson. “Independent driving pattern factors and their influence on fuel-use and exhaust emission factors”. In: *Transportation Research Part D: Transport, 2001 – Elsevier, 325-345J.* (2001) (cit. on pp. 21, 39, 56).

- [19] H. et al. Johansson. “Impact of EcoDriving on emissions”. In: *International Scientific Symposium on Transport and Air Pollution, Avignon, France*. (2003) (cit. on pp. 21, 39, 56, 87).
- [20] M. Kuhler and D. Kartens. “Improved driving cycle for testing automotive exhaust emissions”. In: *SAE Technical Paper Series 780650*. (1978) (cit. on pp. 21, 39, 56, 87).
- [21] M. André. “Driving cycles development: Characterization of the methods.” In: *SAE Technical Papers Series 961112*. (1996) (cit. on pp. 21, 39, 56).
- [22] I. Fomunung, S. Washington, and R. Guensler. “A statistical model for estimating oxides emissions from light duty motor vehicles.” In: *Transportation Research Part D, pp.333-352*. (1999) (cit. on pp. 21, 39).
- [23] Chen M.C., Chen J.L., and Chang T.W. “Android/OSGi-based vehicular network management system”. In: *Elsevier Computer Communications, vol. 34, no. 2, pp. 169 – 183*, (2011) (cit. on p. 21).
- [24] M. Fogue et al. “Automatic Accident Detection: Assistance Through Communication Technologies and Vehicles.” In: *Vehicular Technology Magazine, IEEE*, (2012) (cit. on p. 22).
- [25] J. Zaldivar et al. “Providing accident detection in vehicular networks through OBD-II devices and Androidbased smartphones”. In: *Local Computer Networks (LCN), IEEE 36th Conference* (2011) (cit. on p. 22).
- [26] Rangachari An et al. “Efficient Classification for Multiclass Problems Using Modular Neural Networks”. In: *IEEE transactions on neural networks* (1995) (cit. on pp. 28, 65, 103).
- [27] J.E. Meseguer et al. “DrivingStyles: a smartphone application to assess driver behavior”. In: *18th IEEE symposium on Computers and Communications (ISCC)* (2013) (cit. on pp. 38, 54, 74, 89).
- [28] “Manual of eco-driving for cars”. In: *IDEA (Institute for Energy Diversification and Saving of Energy)*. (2007) (cit. on p. 39).

- [29] IDES. “Traffic: civilization or barbarism. The risk Observatory”. In: *Institute for security studies (IDES)*. URL: <http://www.seguretat.org> (2006) (cit. on p. 39).
- [30] D.Y.C. Leung and D.J. Williams. “Modelling of Motor Vehicle Fuel Consumption and Emissions Using a Power-Based Model”. In: *Environmental Monitoring and Assessment*, 65:21–29 (2000) (cit. on pp. 39, 56).
- [31] COPERT III. “Computer programme to calculate emissions from road transport Methodology and emission factors (Version 2.1)”. In: *Technical report No 49* (2000) (cit. on pp. 39, 56).
- [32] HORIBA. “On Board PM Mass Measurement for US and EU In Use Compliance”. In: *Cambridge Particle Meeting 16th May 2008* (2008) (cit. on p. 39).
- [33] BMW Press Release. “Drive a BMW and enjoy personalised radio”. In: [*available at <http://www.zercustoms.com/news/BMW-Connected-Drive-Personalised-Radio.html>*] (2007) (cit. on p. 39).
- [34] V Barabba et al. “A multimethod approach for creating new business models”. In: *The General Motors OnStar project* (2002) (cit. on p. 39).
- [35] <http://drivingstyles.info/index.php/en/users-world> (cit. on p. 48).
- [36] Standardization. *Road vehicles, Diagnostic systems, Keyword Protocol 2000*. Tech. rep. International Organization Standardization, 1999 (cit. on p. 54).
- [37] J.E. Meseguer et al. “Assessing the Impact of Driving Behavior on Instantaneous Fuel Consumption”. In: *IEEE 12th Consumer Communications and Networking Conference (CCNC)* (2015) (cit. on pp. 55, 72, 74, 87, 89).
- [38] A. Meintz and M. Ferdowski. “Designing efficient hybrid electric vehicles”. In: *IEEE Vehicular Technology Magazine, IEEE, Volume:4, Issue 2* (2009) (cit. on p. 56).

- [39] U. Hernandez et al. "Vehicle on board platform: Communications test and prototyping". In: *Intelligent Vehicles Symposium (IV), 2010 IEEE*, pp. 967-972 (2010) (cit. on p. 56).
- [40] S. Matsumoto and H. Kawashima. "Fundamental study on effect of preceding vehicle information on fuel consumption reduction of a vehicle group". In: *Journal Communications Networks*. 15, 173-178 (2013) (cit. on p. 56).
- [41] Circuit Cellar magazine. "VR-Based Fuel Consumption Gauge". In: *Circuit Cellar magazine, Issue 183*, pp. 59 - 67, October 2005. (2005) (cit. on p. 63).
- [42] Gibson PM and Wiesenthal DL. "The Driving Vengeance Questionnaire (DVQ) the development of a scale to measure deviant drivers". In: *LaMarsh Research Programme Report Series, No. 55* (1996) (cit. on pp. 72, 86).
- [43] Cohen S. "After effects of stress on human performance and social behavior, a review of research and theory". In: *Psychol Bull* 88:82-108. (1980) (cit. on pp. 72, 86).
- [44] Raffaele Gravina and Giancarlo Fortino. "Automatic Methods for the Detection of Accelerative Cardiac Defense Response". In: *IEEE Transactions On Affective Computing, Vol. 7, NO. 3* (2016) (cit. on pp. 72, 86).
- [45] Alessandro Andreoli et al. "SPINE-HRV: A BSN-Based Toolkit for Heart Rate Variability Analysis in the Time-Domain". In: *Wearable and Autonomous Systems, LNEE 75*, pp. 369-389 (2010) (cit. on pp. 72, 86).
- [46] A. Riener. "Subliminal Persuasion and Its Potential for Driver Behavior Adaptation". In: *IEEE Transactions On Intelligent Transportation*, pp. 1-10 . (2012) (cit. on pp. 75, 89).
- [47] Timo Lajunen, Dianne Parker, and Heikki Summala. "Does traffic congestion increase driver aggression?" In: *Transportation Research Part F* 2 225-236 (1999) (cit. on pp. 78, 92).

- [48] Karel A, Cornelia J.G., and Tineke H. “Driving with a congestion assistant; mental workload and acceptance”. In: *Psychophysiology in Ergonomics Volume 40, Issue 6, Pages 1019–1025* (2009) (cit. on pp. 86, 92).
- [49] Shuui S, Kazushi T, and Masanori K. “Heart rate variability during long truck driving work”. In: *Journal of Human Ergology Vol. 30 No. 1-2 P 235-240* (2001) (cit. on pp. 86, 92).
- [50] Rakha H. “Simple Comprehensive Fuel Consumption and CO2 Emission Model based on Instantaneous Vehicle Power”. In: *Transportation Research Board Annual Meeting. Washington D.C.* (2011) (cit. on p. 87).
- [51] The Apache HTTP Server Project. <http://httpd.apache.org>. “The Apache Software Foundation.” In: (2003) (cit. on p. 105).
- [52] Inc. J2EE: Java 2 Platform Sun Microsystems. “Enterprise Edition. Available from <http://java.sun.com/j2ee>”. In: (2014) (cit. on p. 105).
- [53] MySQL AB 5.7. “Available from <http://www.mysql.com>.” In: (2015) (cit. on p. 105).