Human activity recognition: new approaches based on machine learning and deep learning

Autor: Daniel García González

Tesis doctoral UDC / 2023

Directores:
Enrique Fernández Blanco
Miguel Ángel Rodríguez Luaces

Tutor:
Daniel Rivero Cebrián

Human activity recognition: new approaches based on machine learning and deep learning

Autor: Daniel García González

Tesis doctoral UDC / 2023

Directores:
Enrique Fernández Blanco
Miguel Ángel Rodríguez Luaces

Tutor:
Daniel Rivero Cebrián

Programa Oficial de Doctorado en Tecnologías de la Información y las Comunicaciones

PhD Thesis supervised by
Tesis doctoral dirigida por

Enrique Fernández Blanco
Departamento de Computación y Tecnologías de la Información
Facultad de Informática
Universidade da Coruña
15071 A Coruña (España)
Tel: +34 881016014
enrique.fernandez@udc.es

Miguel Ángel Rodríguez Luaces
Departamento de Computación y Tecnologías de la Información
Facultad de Informática
Universidade da Coruña
15071 A Coruña (España)
Tel: +34 881011254
miguel.luaces@udc.es

Tutored by
Tutorizada por

Daniel Rivero Cebrián
Departamento de Computación y Tecnologías de la Información
Facultad de Informática
Universidade da Coruña
15071 A Coruña (España)
Tel: +34 881011266
daniel.rivero@udc.es

Enrique Fernández Blanco y Miguel Ángel Rodríguez Luaces, como directores, y Daniel Rivero Cebrián, como tutor, acreditamos que esta Tesis cumple los requisitos para optar al título de doctor internacional y autorizamos su depósito y defensa por parte de Daniel García González, cuya firma también se incluye.

Enrique Fernández Blanco and Miguel Ángel Rodríguez Luaces, as directors, and Daniel Rivero Cebrián, as tutor, certify that this Thesis meets the requirements to qualify for the title of international doctor and we authorise its deposit and defence by Daniel García González, whose signature is also included.

A todas aquellas personas que me acompañaron y me apoyaron durante este largo viaje llamado doctorado.

To all those individuals who accompanied me and supported me during this long journey called PhD.

Agradecimientos

Es curioso cómo esta parte acaba siendo una de las más difíciles de escribir, sobre todo teniendo en cuenta que es el único hueco en el que se puede ser más informal.

Antes de empezar a mencionar a todas las personas que me ayudaron a llegar hasta aquí, me gustaría comentar unas cosillas. Lo primero de todo, quiero que conste que nunca en mi vida me había planteado empezar un doctorado. Me parecía algo que me quedaba muy lejos, la verdad. Además, pensaba que la vida en empresa sería lo mío, ya que era prácticamente de lo único que se hablaba durante mis años de carrera y era lo que nos vendían normalmente como "estable y cómodo". La investigación, al menos en España, tiene ese aura de inestabilidad y desdicha que suele echar para atrás a la gente (y no los culpo, vaya). Sin embargo, tras probar el trabajo en empresa me di cuenta de que ahí no pintaba nada. Mis inquietudes y mis intereses personales distan mucho de lo que ofrece la vida en empresa habitual. Sí que es cierto que, a la larga, acabas acumulando mucho más dinero y, normalmente, con una estabilidad mucho mayor. También es cierto que los trabajos pueden ser todos muy diversos, con diferentes condiciones y ambientes, pudiendo haber algo de ese estilo que entre dentro de mis "requisitos", por decirlo de alguna manera. Aún así, no podría estar más satisfecho del camino que he tomado. Soy consciente de que he tenido mucha suerte y me he juntado con personas que me han hecho el camino más fácil de lo normal. No obstante, el orgullo de contribuir a mí manera a la ciencia y participar mínimamente en la formación de los estudiantes que van pasando por la facultad no me lo quita nadie. Con esto no quiero decir que sea todo un camino de rosas, porque no lo es, pero estoy muy contento actualmente. Así que, quiero agradecerme a mí mismo el haberme atrevido a meterme aquí, e invito a cualquier otra persona que se pueda sentir de manera similar a hacer lo mismo. Salir de la burbuja cuesta, pero suele dar pie a las mejores decisiones de la vida.

En cualquier caso, sin duda las primeras personas que debo mencionar en estos agradecimientos son aquellas tres que participaron directamente en todo el proyecto, formalmente denominados mis directores o tutores:

- A **Quique** y **Dani**, por un ser un apoyo capital durante toda la tesis, teniendo que leer y revisar mil veces mis "tochos"; por hacerme caer de la burra más de una vez y, en especial, por ser tan cercanos y comprensivos desde que los tuve

como profesores en la carrera. Es una verdadera maravilla poder contar con personas con tanta vocación y tan comprometidas como ellos.

- A **Miguel**, por confiar en mí y en el proyecto, a pesar de no pertenecer a su dominio habitual ni habernos cruzado previamente. También por ser un pilar muy importante sobre todo durante los primeros años de desarrollo.

Dicho esto, también hay una serie de personas a las que me gustaría dedicarles algunas líneas, ya que, de alguna manera u otra, también me han ayudado a sacar esta Tesis adelante:

- A **Ariana**, por tener que aguantar todos y cada uno de los años, meses, semanas y días de esta aventura, tanto en los días buenos como en los malos (no sé en cuáles era más pesado, la verdad). Su punto de vista externo a este mundillo ha sido muy importante varias veces. Al fin y al cabo, más de una vez me ha tenido que ayudar a tomar alguna decisión de implementación o de escritura cuando estaba bloqueado. Además, ha tenido que soportar también mis desahogos cuando las cosas no estaban saliendo como deberían (o cuando estaban saliendo sospechosamente bien). Aunque su personalidad le impida mostrar tanto cariño como me gustaría, sé que es una de las personas que más se enorgullecen de mis logros y es un verdadero lujo poder tenerla a mi lado todos los días. ¡Gracias Aripé! ¡MUÁ!

- A **mis padres**, Víctor y Paz, por darme todos los recursos posibles y más para llegar hasta aquí y por todo su apoyo durante toda mi vida, aunque no acaben de entender muy bien qué es lo que hago. Aunque sé que en cierto modo no les gusta mucho que haya optado por esta vida investigadora (ya sabéis, la inestabilidad y esas cosas), sé que mientras esté yo a gusto en el fondo les parece bien. De igual manera, me gustaría mencionar también a mi hermano, **Sergio** (o KoL para los más amiguetes), que aún teniendo en cuenta la distancia y los cambios en nuestras vidas, siempre lo tengo presente.

- A ese grupito de personas a los que "no nos compila la vida". Compañeros de profesión (cada uno en lo suyo) y cada vez más amigos, aunque cada uno tengamos nuestras "movidas especiales". Con permiso de **Dylan** y **Darío**, me gustaría destacar un poco más a otras dos personas. Tranquilos compañeros, sois los mejores. A **Pablo**, por compartir vida y casa conmigo hasta hace poco y ayudarme a reafirmarme en que había tomado la mejor decisión cuando cambié mi orientación laboral. Quitando su obsesión por las plantas, fue el mejor compañero de piso que tuve durante todos esos años. A **Borja**, aunque más de una vez me saque de quicio (es un poco bocachancla), pero siempre me ayudó mucho durante toda mi vida académica, incluso en esta última etapa, siendo el nexo mediante el cual pude entrar en el laboratorio e iniciar mi doctorado de verdad.

- También me gustaría agradecer a **Isma** (o Kuismi, si te deja llamarlo así) y a mi círculo habitual de Vigo el celebrar cada uno de los logros que fui consiguiendo y mantenerse a mi lado a lo largo de estos años. Para continuar dando nombres propios como hasta ahora y no dejar a nadie de lado, dentro de ese círculo se encuentran, por un lado, mi grupo de amigos de toda la vida: **Andros**, **Enoc**, **Isaías**, **Laran**, **Regue**, **Rebo**, **Chris**, **Pabel**, **Sergio**, **Jaco**, **Marcos** y **Txori**. Algunos más cercanos que otros, por supuesto, pero siempre han estado todos ahí. Por otro lado, me gustaría también mencionar a mi "segunda familia" de Vigo: **Nieves**, **Davinia**, **Pablo** y **Rico**. Aún habiéndonos conocido hace escasos años, me han animado más de una vez durante todo el doctorado.

De igual manera, también me gustaría agradecer al **LBD** (Laboratorio de Bases de Datos) el darme la oportunidad de formar parte de este proyecto, aún tratando una temática fuera de sus competencias habituales. Al fin y al cabo, he podido hacer algo que me gusta y más o menos de la manera que me gustaría, con todos los recursos y facilidades que me han podido ofrecer. No creo que todo el mundo pueda decir lo mismo. En cualquier caso, dentro de este grupo, además de todas las personas que han ido pasando por el mismo y que lo conforman a día de hoy, me gustaría destacar a **Víctor**. Entramos el mismo día en el laboratorio y, desde entonces, es una de las mejores amistades que me llevo del laboratorio.

Por supuesto, también agradecer a todas las personas que me ayudaron a conformar la base de datos sobre la que se asentó todo este proyecto. Muchos de ellos son personas que ya he mencionado previamente. Sin su ayuda desinteresada no podría haber llevado a cabo ninguno de los desarrollos de esta Tesis.

Por otro lado, me gustaría mencionar también a la Universidad Aristóteles de Tesalónica, en especial al profesor **Apostolos N. Papadopoulos**, por darme tan buena acogida y ayudarme a descubrir nuevos conocimientos durante mi estancia en Grecia.

Por último, me gustaría agradecer también a las entidades que han financiado este proyecto, incluyendo la estancia realizada en Grecia, y que me han permitido llevar a cabo mi doctorado con total desempeño. Quiero agradecer al **CITIC** (Centro de Investigación en Tecnologías de la Información y las Comunicaciones), financiado por la Xunta de Galicia y la Unión Europea (European Regional Development Fund-Galicia 2014-2020 Program), con la beca ED431G 2019/01. También agradecer a la **Xunta de Galicia**/FEDER-UE, que ha financiado la mayor parte de esta tesis a través de la beca ED481A 2020/003.

Y con esto finalizo el capítulo de agradecimientos más largo que probablemente hayáis visto en este tipo de trabajos. Toda mi vida siendo el que escribe poco siempre, para acabar siendo el pesado de los "tochoposts"... Vaya tela.

Acknowledgements

It's funny how this part ends up being one of the trickiest to write, especially considering it's the only place where you can be a tad more informal.

Before I start mentioning all the folks who helped me get to this point, I'd like to share a few thoughts. First and foremost, I want it to be known that I'd never in my life considered embarking on a PhD. It seemed like something far beyond my reach, to be honest. Moreover, I thought a career in the corporate world was my calling, as it was practically the only thing discussed during my years in college and was often presented as "stable and comfortable". Research, at least in Spain, has that aura of instability and unhappiness that tends to deter people (and I don't blame them, really). However, after trying my hand in the corporate world, I realised it wasn't for me. My interests and personal inclinations were far from what corporate life had to offer. Yes, it's true that, in the long run, you accumulate more wealth and usually enjoy greater stability. It's also true that jobs can vary greatly, with different conditions and environments, and there might be something in that realm that aligns with my "requirements", so to speak. Nevertheless, I couldn't be happier with the path I've chosen. I'm aware that I've been very fortunate and have surrounded myself with people who have made my journey smoother than usual. However, the pride of contributing to science in my own way and playing a small role in the education of the students passing through the university is something no one can take away from me. By saying this, I don't mean to imply that it's all plain sailing, because it's not, but I'm genuinely content with where I am now. So, I want to thank myself for having the courage to embark on this journey, and I invite anyone else who may feel similarly inclined to do the same. Breaking out of your comfort zone is tough, but it often leads to the best decisions in life.

In any case, undoubtedly, the first people I should mention in these acknowledgements are the three who were directly involved in the entire project, formally known as my supervisors or mentors:

- To **Quique** and **Dani**, for being vital support throughout the Thesis, having to read and review my "tomes" a thousand times; for talking some sense into me more than once, and especially for being so approachable and understanding since I had them as professors in college. It's truly a blessing to have people

with such dedication and commitment as they have.

- To **Miguel**, for placing trust in me and the project, despite it not falling within his usual domain, and for being a crucial pillar, especially during the early years of development.

That said, there are also a series of people I'd like to dedicate a few words to since, in one way or another, they've also helped me push through this Thesis:

- To **Ariana**, for enduring each and every year, month, week, and day of this adventure, both on good days and bad ones (I'm not sure which were more challenging for her, to be honest). Her outsider's perspective on this world has been invaluable on several occasions. After all, more than once, she had to help me make implementation or writing decisions when I was stuck. She's also had to tolerate my venting when things weren't going as planned (or when they were suspiciously smooth). Although her personality might not allow her to express as much affection as I'd like, I know she's one of the people most proud of my achievements, and I'm truly fortunate to have her by my side every day. Thank you Aripé! MUÁ!

- To **my parents**, Víctor and Paz, for providing me with all possible resources and more to get to this point and for their unwavering support throughout my life, even if they still don't quite grasp what I do. While I know they're not entirely thrilled with my choice of an academic career (you know, the instability and all), I also know that deep down, as long as I'm happy, they're okay with it. Similarly, I'd like to mention my brother, **Sergio** (or KoL for his close friends), who, despite the distance and the changes in our lives, is always present.

- To that group of people who "don't quite compile with life". Colleagues in the profession (each in their own way) and increasingly friends, even though we all have our quirks. With **Dylan**'s and **Darío**'s permission, I'd like to highlight two other individuals a bit more. Don't worry, comrades; you're the best. To **Pablo**, for sharing life and a home with me until recently and helping me reaffirm that I made the right choice when I changed my career path. Aside from his plant obsession, he was the best flatmate I had during all those years. To **Borja**, even though he has driven me crazy more than once (he can be quite outspoken), he always supported me throughout my academic life, even in this last stage, serving as the bridge that allowed me to enter the laboratory and truly begin my PhD.

- I'd also like to thank **Isma** (or Kuismi, if he lets you call him that) and my usual circle of friends in Vigo for celebrating each of the milestones I achieved and standing by my side over the years. To keep naming names as I have been and not leave anyone out, within that circle are, on one hand, my lifelong group

of friends: **Andros**, **Enoc**, **Isaías**, **Laran**, **Regue**, **Rebo**, **Chris**, **Pabel**, **Sergio**, **Jaco**, **Marcos**, and **Txori**. Some are closer than others, of course, but they've all been there. On the other hand, I'd also like to mention my "second family" in Vigo: **Nieves**, **Davinia**, **Pablo**, and **Rico**. Despite having only known them for a few years, they've encouraged me more than once throughout my PhD.

Likewise, I'd like to thank the **LBD** (Database Laboratory) for giving me the opportunity to be part of this project, even though it falls outside their usual scope. In the end, I've been able to do something I love, more or less the way I wanted, with all the resources and support they could provide. I don't think everyone can say the same. Within this group, in addition to all the people who have passed through and made up the lab today, I'd like to highlight **Víctor**. We both joined the lab on the same day and since then, he's been one of the best friendships I've made in the lab.

Of course, I also want to thank all the people who helped me create the database on which this entire project is based. Many of them are individuals I've already mentioned. Without their selfless assistance, none of the developments in this Thesis would have been possible.

Furthermore, I'd like to mention the Aristotle University of Thessaloniki, especially Professor **Apostolos N. Papadopoulos**, for welcoming me so warmly and helping me discover new knowledge during my stay in Greece.

Finally, I want to express my gratitude to the organisations that funded this project, including the stay in Greece, allowing me to pursue my PhD with full dedication. I want to thank the **CITIC** (Centre for Information and Communication Technologies Research), funded by the Xunta de Galicia and the European Union (European Regional Development Fund- Galicia 2014-2020 Programme), for the ED431G 2019/01 scholarship. I also want to thank the **Xunta de Galicia**/FEDER-EU, which funded a significant portion of this Thesis through the ED481A 2020/003 scholarship.

And with this, I conclude the longest acknowledgements chapter you've probably seen in this type of work. I spent my whole life being the one who wrote very little, only to become the one who writes these lengthy posts... How things change.

Abstract

Currently, the scientific community is giving significant attention to the field of human activity recognition (HAR), which has gained remarkable prominence as a topic of discussion. Since the irruption of smartphones and wearable devices in daily life, the costs and ease of conducting studies in this field have improved significantly. Moreover, its applicability in various research areas such as medicine, fitness, or home automation makes this topic even more attractive for researchers in the field. However, despite the remarkable advances made in the last decade, it is not possible to transfer that acquired knowledge to a real-life environment. That is because most of the related work has been carried out under laboratory conditions. In other words, with pretty specific indications, placing the measuring devices and performing the actions in an explicit way that does not represent at all the variability present in the real world. For those reasons, this Thesis has focused on orienting the research in this field towards a real-life environment. To that end, a dedicated dataset has been constructed to carry out the main research, based on the personal smartphone sensors of 19 different individuals. The main difference between that dataset and those already existing in the scientific community is that those individuals have been given as much freedom as possible to use their smartphones during data collection. Thus, even when performing the same action conceptually, the resulting data may vary, as each individual may use the smartphone differently, as is the case in everyday life. Hence, once the data was obtained, an in-depth study was carried out, in search of the best machine learning and deep learning models to classify the data, according to the actions studied. The results confirm the possibility of transferring the acquired knowledge to a real-life environment. In terms of their performance, it is worth mentioning tree-based models like Random Forest and other deep learning models such as Convolutional Neural Networks (CNN) or recurrent neural networks based on the Long Short-Term Memory (LSTM) technique, among the various methods used.

Resumen

En la actualidad, la comunidad científica está prestando gran atención al campo del reconocimiento de las actividades humanas (HAR), el cual ha cobrado notable protagonismo como tema de debate. Desde la irrupción de los smartphones y los dispositivos wearables en la vida cotidiana, los costes y la facilidad de realizar estudios en este campo han experimentado una mejora significativa. Además, su aplicabilidad en diversos campos de estudio como la medicina, el fitness o la domótica hacen que esta temática sea aún más atractiva para los investigadores del ámbito. Sin embargo, a pesar de los grandes avances realizados en la última década, no es posible transferir este conocimiento adquirido hacia un entorno de la vida real. Esto se debe a que la grandísima mayoría de los trabajos relacionados fueron llevados a cabo en condiciones de laboratorio. En otras palabras, con indicaciones muy específicas, colocando los dispositivos de medición y realizando las acciones de una forma muy concreta que no representa para nada la variabilidad presente en el mundo real. Por ello, esta Tesis se ha centrado en orientar la investigación en este campo hacia un entorno de la vida real. Para ello, se ha construido un conjunto de datos propio con el que poder llevar a cabo la investigación principal, a partir de los sensores de los smartphones personales de 19 individuos diferentes. La diferencia principal de dicho conjunto de datos con respecto a los ya existentes en la comunidad científica es que se les ha dado a dichas personas la mayor libertad posible para utilizar su smartphone durante las recolecciones de datos. De este modo, aún realizando la misma acción conceptualmente, los datos resultantes pueden variar, ya que cada individuo puede utilizar el smartphone de forma diferente, tal y como ocurre en la vida diaria. Así, una vez obtenidos los datos, se llevó a cabo un estudio exhaustivo sobre los mismos, en búsqueda de los mejores modelos de machine learning y deep learning para clasificar los datos según las acciones estudiadas. Los resultados confirman la posibilidad de transferir el conocimiento adquirido hacia un entorno de la vida real. Entre los métodos utilizados, conviene destacar, en relación a sus rendimientos, a los modelos basados en árboles, como Random Forest, y otros de deep learning como las redes de neuronas convolucionales (CNN) o las redes neuronales recurrentes basadas en la técnica de Long Short-Term Memory (LSTM).

Resumo

Na actualidade, a comunidade científica está a prestar moita atención ao campo do recoñecemento das actividades humanas (HAR), o cal cobrou considerable protagonismo como tema de debate. Dende a irrupción dos smartphones e os dispositivos wearables na vida cotiá, os custos e a facilidade de realizar estudos neste eido experimentaron unha mellora significativa. Ademais, a súa aplicabilidade en diversos campos de estudo como a medicina, o fitness ou a domótica fan que este tema sexa aínda máis atractivo para os investigadores da materia. Porén, a pesar dos grandes avances acadados na última década, non é posible trasladar estes coñecementos adquiridos a un entorno da vida real. Isto débese a que a gran maioría dos traballos relacionados realizáronse en condicións de laboratorio. Noutras palabras, con indicacións moi específicas, colocando os aparellos de medida e realizando as accións dun xeito moi concreto que non representa para nada a variabilidade presente no mundo real. Por iso, esta Tese centrouse en dirixir a investigación neste campo cara a un entorno da vida real. Para iso, construíuse un conxunto de datos propio co que realizar a investigación principal, baseado nos sensores dos smartphones persoais de 19 individuos diferentes. A principal diferenza deste conxunto de datos con respecto aos xa existentes na comunidade científica é que estas persoas tiveron a maior liberdade posible para usar o seu smartphone durante a recollida de datos. Deste xeito, aínda realizando conceptualmente a mesma acción, os datos resultantes poden variar, xa que cada individuo pode utilizar o smartphone de forma diferente, tal e como ocorre na vida diaria. Así, unha vez obtidos os datos, realizouse un estudo exhaustivo sobre eles, na procura dos mellores modelos de machine learning e deep learning para clasificar os datos segundo as accións estudadas. Os resultados confirman a posibilidade de transferir os coñecementos adquiridos a un entorno da vida real. Entre os métodos utilizados, convén destacar, en relación aos seus rendementos, aos modelos baseados en árbores, como Random Forest, e outros de deep learning como as redes de neuronas convolucionais (CNN) ou redes neuronais recorrentes baseadas na técnica de Long Short-Term Memory (LSTM).

Contents

List of Figures

List of Tables

List of Acronyms

ADAM ADAptive Moment estimation

ADL Activity of Daily Living

AI Artificial Intelligence

ANOVA ANalysis Of VAriance

Bi-LSTM BIdirectional Long Short-Term Memory

CITIC Centro de Investigación en Tecnologías de la Información y las Comunicaciones

CNN Convolutional Neural Network

DT Decision Tree

DS-CNN Depth-wise Separable Convolutional Neural Network

ECG ElectroCardioGram

FN False Negatives

FP False Positives

GBM Gradient Boosting Machine

GEMA GEstión de la MovilidAd

GPS Global Positioning System

HAR Human Activity Recognition

HHAR Household Human Activity Recognition

HSD Honestly Significant Difference

IMU Inertial Measurement Unit

KNN K-Nearest Neighbour

LSTM Long Short-Term Memory

MLP MultiLayer Perceptron

mHealth Mobile Health

MSE Mean Squared Error

NB Naïve Bayes

NFL No Free Lunch

PAMAP Physical Activity Monitoring and Assessment System

RBF Radial Basis Function

ReLU REctified Linear Unit

RF Random Forest

RNN Recurrent Neural Networks

SHAR Smart Home Activity Recognition

SME Small and Medium-sized Enterprise

SVM Support Vector Machine

TN True Negatives

TP True Positives

UCI University of California, Irvine

UniMiB UNIversity of MIlano-Bicocca

WISDM WIreless Sensor Data Mining

XGB eXtreme Gradient Boosting

Preface

This Thesis is divided into six main chapters representing the work's general development. Afterwards, the shared bibliography is listed. Finally, a couple of appendixes are included. The first one shows the publications resulting from the research, while the latter contains a summary of the work done, in Spanish. That structure is detailed below:

- **Chapter 1. Introduction.** It is divided into two parts. The initial section provides an overview of the human activity recognition field, including the fundamental motivations driving this Thesis. Then, the second part lists the main objectives of the Thesis.
- **Chapter 2. Core concepts.** This chapter provides a thorough overview of the essential technical concepts that serve as the building blocks for understanding the whole Thesis.
- **Chapter 3. State of the art.** This chapter describes the evolution of the human activity recognition field at the scientific level, with references to some outstanding works and especially highlighting some benchmark datasets.
- **Chapter 4. Methodology and results.** It shows a summary of the contributions made in this Thesis. Regarding each of the published articles, an in-depth examination of their execution, along with the attained results and encountered challenges throughout their development, is presented.
- **Chapter 5. Conclusions and future work.** Two different sections are dedicated to defining conclusions and exploring potential avenues for future work.
- **Chapter 6. Research results.** It highlights the merits achieved during this Thesis' development.
- **Bibliography.** This chapter lists the main bibliographical references on which the Thesis was based.

- **Appendix A. Articles.** It contains the scientific articles resulting from the work carried out. Those publications are:
  - Garcia-Gonzalez, D., Rivero, D., Fernandez-Blanco, E., and Luaces, M. R. (2020). **A public domain dataset for real-life human activity recognition using smartphone sensors.** *Sensors*, 20(8):2200. DOI: 10.3390/s20082200.
  - Garcia-Gonzalez, D., Rivero, D., Fernandez-Blanco, E., and Luaces, M. R. (2023). **New machine learning approaches for real-life human activity recognition using smartphone sensor-based data.** *Knowledge-Based Systems*, 262:110260. DOI: 10.1016/j.knosys.2023.110260.
  - Garcia-Gonzalez, D., Rivero, D., Fernandez-Blanco, E., and Luaces, M. R. (2023). **Deep learning models for real-life human activity recognition from smartphone sensor data.** *Internet of Things*, page 100925. DOI: 10.1016/j.iot.2023.100925.
- **Appendix B. Resumen extendido en castellano.** It provides a comprehensive summary of all the work conducted, presented in Spanish.

Chapter 1

Introduction

This chapter is dedicated to introducing all the necessary elements to understand the context of this Thesis. Thus, in the first place, Section 1.1 outlines the current issues in human activity recognition (HAR) research and the reasons behind undertaking this Thesis. Then, Section 1.2 presents the main goals that will shape the core of the work's development.

1.1 Motivation

The ability to reliably and automatically identify the movements made by a human being is a challenge that has been the subject of considerable research over the last decades. The main interest lies in the multiple applications that systems that detect such actions could have. For example, within the world of healthcare [Subasi et al., 2018, Demrozi et al., 2020, Liu et al., 2021] and fitness [Attal et al., 2015, Zainudin et al., 2017], it would be possible to know the movements made by an individual to be able to make a more appropriate diagnosis. In addition, it would also be possible to carry out a treatment with an in-depth and more comfortable control for both parties. Furthermore, advances in this field could also be applied to home automation [Raeiszadeh and Tahayori, 2018, Du et al., 2019] or leisure [Ma, 2021], as it would be possible to automate and trigger actions based on the movements made by the individual. In order to detect those actions, it is possible to use both video cameras [Ke et al., 2013, Beddiar et al., 2020] and motion sensors that the individuals may wear [Aggarwal and Xia, 2014, Wang et al., 2019, Soleimani and Nazerfard, 2021]. Concerning the latter, the most common ones are the accelerometer and the gyroscope. The former is used to detect vibrations or slight movements in the individual. As for the gyroscope, its task is to measure the different oscillations or turns that may occur. In the past, these sensors were much more expensive and less accessible. However, since the advent of wearable

devices, and especially smartphones, the human activity recognition field has seen a significant increase in research. That is mainly because those devices are already embedded with such sensors. In addition, over a decade ago, those devices have become enormously popular in the developed world, to the extent that many people carry an activity bracelet on their wrist and, above all, a smartphone in their pockets, which they use on a daily basis. That makes the research costs in this field more affordable, with easier access to sensors with high-grade accuracy. Thus, researchers find HAR a pretty attractive option in which to do their bit [Lara and Labrador, 2012, Hassan et al., 2018, Tang et al., 2022].

Nevertheless, there are some issues to be taken into account. Firstly, it is necessary to keep control of the temporality of the data, which is particularly difficult when working with large amounts of information such as those produced by the sensors discussed above. Although remarkable progress has already been made in this respect [Shoaib et al., 2016, Qi et al., 2019, Xia et al., 2020], there are still activities where the relationship between the action and its data has not yet been fully resolved. That is mainly due to the conditions under which those studies were carried out. In general, in those works, the individual performing the action does so in a very controlled way, with pretty specific indications on how to carry it out [Xu et al., 2019]. In addition, the measuring device used is placed in a precise spot and in a definite way, such as on the wrist [Lawal and Bano, 2019] or waist [Jeong and Oh, 2021]. While it is true that, in those cases, the vast majority of the actions studied have been solved with high accuracy, such results would not be entirely reliable if applied to a real-life environment. In everyday life, such specific conditions do not usually occur, so the acquired knowledge cannot be directly transferred to a more realistic environment. For example, the same action would not necessarily yield the same data in different individuals [Lago et al., 2019]. In the case of smartphones, their use and the way they are carried differ for each person. Such variations would notably affect the measurements reported by the device's sensors. Even distinct smartphone models could yield slightly different data [Stisen et al., 2015]. That point would not be as critical using activity wristbands, as they would always be worn on the wrist. However, their use is much lower than that of smartphones. Today, the vast majority of people have a smartphone that they carry with them everywhere they go. Given that, using any other kind of wearable device, such as activity wristbands, is more of a personal choice. On the other hand, the performance of the action does not have to be strictly the same for everyone. While, theoretically, it would be the same, there might be slight variations that could lead to confusion in classification. For example, when walking, not everyone bends in or moves their legs equally. In fact, some of these differences could also be due to the body diversity of each individual, even when using the device in the same way [Sansano et al., 2020]. Without going any further, using the same example as above, the length or width of each person's leg when walking could lead to different measurements if the device was worn, for example, in a trouser pocket. Moreover, the

same placement could lead to different results when considering, for example, tight leggings versus baggy cargo trousers. In fact, the issue of personalising classification models for large numbers of people has also been studied extensively in recent years [Lane et al., 2011, Solis Castilla et al., 2020, Ferrari et al., 2020].

For all those reasons, the motivation for this Thesis lies in the current difficulty of applying, outside a laboratory environment, all the advances achieved so far in this field. Prior to the beginning of this Thesis, no realistic dataset existed. Therefore, new ones need to be published. Consequently, a comprehensive study on how to address them could be undertaken. For that purpose, it would be essential to analyse the suitability of all artificial intelligence (AI) techniques previously employed in HAR. The goal would be to identify the most appropriate options and adapt them accordingly to align with the specific context. In this regard, it is worth noting that obtaining an optimal algorithm for all situations is not feasible. By focusing on specific domains, such as the new data orientation, it becomes easier to enhance the results for that particular case. However, such improvement may come at the expense of lower performance in other scenarios, as stated by the No Free Lunch (NFL) theorems [Wolpert and Macready, 1997]. In addition, it would be imperative to consider that those data could exhibit unique characteristics that have not been observed yet in previous HAR developments. Thus, it would also be crucial to investigate the most suitable methodology to process such data and to study the best way to prepare them to feed the relevant models. As a result, all the progress achieved in them could be directly applied to real-life environments, according to the activities studied in them.

1.2 Objectives

Taking into account all that has been discussed in the previous section, the main objective of this Thesis is to contribute to the advancement of research in the field of human activity recognition, specifically by promoting its application in real-life environments. To that end, it is essential to gather new data from more realistic situations, so that they can be manipulated later by the whole scientific community. All the knowledge acquired until now in the field could then be applied to this new approach, adapting it accordingly. Thus, the first objective of this Thesis is to conduct an extensive literature review encompassing the entire HAR domain. The goal is to identify crucial aspects that must be considered for contributing relevant advancements in the field. With that in mind, the next step will be to elaborate a new dataset, in which the individuals who contribute their measurements can do so in a much looser way, according to the peculiarities of each one. Then, from that dataset, it will be necessary to find the best way to approach it, from traditional machine learning techniques to the most recent architectures based on deep learning. Given that, a series of research challenges are inferred that will make up the core of this Thesis. They are summarised in the following four points:

- **Thoroughly reviewing the literature of the entire HAR field.** In order to carry out relevant developments in this field, it is imperative to acquire comprehensive knowledge of previous work conducted by the scientific community. Although their research was undertaken under different circumstances from those pursued in this Thesis, their findings can hold significant value. After all, gaining insights into the most effective approaches for processing smartphone sensor data, along with recognising common challenges and their respective solutions, becomes an essential requirement. Moreover, staying up to date on current trends and developments is crucial to avoid repeating past mistakes and identifying new research opportunities.

- **Establishing the essential guidelines for generating a dataset that accurately reflects reality.** As discussed in Section 1.1, the current orientation of all studies in HAR precludes their direct application to real-life environments in general. In order to try to initiate the reorientation of research towards that problem, a new dataset needs to be gathered that can be exploited by the entire scientific community. To that end, the personal smartphones of different individuals will be used, so that each of them can use it as they do regularly. As for the activities to be studied, the aim is to gather a group with enough diversity among them but without being as fine-grained as in the studies currently being carried out. In this way, a starting point can be established to study the potential of this new orientation, which can then be focused on more specific actions, as appropriate. Thus, a more realistic dataset than those produced so far will be achieved, with greater freedom and variability in the studied data.

- **Exploring the effectiveness of prevalent machine learning and deep learning techniques in HAR for real-world scenarios.** Once the data have been collected, it is necessary to study the evolution of machine learning and deep learning techniques applied to a HAR theme. Based on that study, a series of approaches will be selected that, according to the information obtained in the previous point, have the best potential to yield good results. At the same time, a comparison will be sought between them, with different configurations of hyperparameters and features, in order to obtain as much information as possible. Furthermore, the investigation will also aim to explore the application of alternative techniques that are less commonly used in HAR but have the potential to contribute positively to the new direction pursued in this Thesis.

- **Pursuing the optimal strategies to address the unique challenges posed by HAR data in real-life environments.** Gathering data in a free and flexible way differs notably from collecting data in laboratory conditions. The contingencies that could arise when constructing a dataset that follows the guidance proposed in this Thesis could be abundant. Additionally, due to

the absence of HAR datasets that follow such orientation within the scientific community, it will be essential to investigate and resolve any potential issues that could emerge during the data collection process. That calls for the identification and resolution of these challenges in real time, which may present entirely novel circumstances not encountered previously. For that purpose, a comprehensive study will be conducted to determine the optimal methods for processing the proposed dataset, along with exploring various configurations and architectures for the selected artificial intelligence models.

Chapter 2

Core concepts

This chapter provides a comprehensive overview of the essential concepts required to understand the research conducted for this Thesis. To begin with, Section 2.1 provides an explanation of the operational principles of the classification algorithms employed. Following that, in Section 2.2, the selected evaluation metrics are presented for comparing and assessing the obtained outcomes. Then, Section 2.3 highlights a couple of statistical tests used to compare the similarity between the results. Finally, Section 2.4 briefly describes the validation and optimisation techniques that have been extensively used throughout this Thesis.

2.1 Classification algorithms

This section incorporates all the classification algorithms employed to tackle the challenges mentioned in Chapter 1. Firstly, Section 2.1.1 encompasses some of the best and most classical machine learning algorithms used in HAR. Secondly, Section 2.1.2 focuses on the cases selected for the cutting-edge field of deep learning, which currently holds great significance within the scientific community.

2.1.1 Machine learning

In the HAR domain, a wide variety of machine-learning algorithms can be employed. In this particular case, the following algorithms were chosen: Support Vector Machine (SVM), Decision Tree (DT), Multilayer Perceptron (MLP), Naïve Bayes (NB), K-Nearest Neighbour (KNN), Random Forest (RF), and Extreme Gradient Boosting (XGB). The choice of these algorithms was primarily based on their extensive usage and favourable outcomes within the HAR field [Ronao and Cho, 2016, Chen et al., 2017, Ignatov, 2018]. In addition, it should be noted that XGB stands out as a relatively new addition to this area. However, its

inclusion was deemed appealing due to its increasing popularity in recent years and exceptional performance in various machine learning competitions [Nielsen, 2016].

2.1.1.1 Support Vector Machines

Support Vector Machines are machine learning models frequently employed in binary classification scenarios [Cortes and Vapnik, 1995]. These models seek to identify a hyperplane that maximises the margins between two predefined and labelled classes. To accommodate non-linear hyperplanes, SVMs use kernels, which are essential hyperparameters. These kernels transform non-linear spaces into linear spaces by altering the dimension in which they are represented, enabling the application of a linear approach. The specific hyperparameters required vary depending on the kernel employed (e.g., linear, polynomial, or radial basis function). However, the fundamental hyperparameter common to all kernels is C, which determines the permissible number of model errors while influencing the margin width of the resulting hyperplane. In addition, other fundamental hyperparameters significantly impact the hyperplane definition. For instance, *gamma* (not applicable to linear kernels, among others) determines the level of curvature that the hyperplane can exhibit, allowing for more pronounced or smoother curves depending on the data samples. Similarly, in polynomial kernels, the degree of the polynomial heavily influences the curvature of the hyperplane. Notably, when the degree is set to 1, the result is equivalent to a linear kernel's (a straight line).

Although SVMs are commonly employed for binary classification tasks, they can also be used for multiclass problems. Under such circumstances, a frequent approach is to select a *one-vs-one* or *one-vs-all* strategy. In the former, classes are modelled in pairs, with multiple binary classifications performed until a final outcome is obtained. Conversely, in the latter approach, individual classifiers are created by confronting each class against the rest, resulting in a specific classifier for each scenario.

2.1.1.2 Decision Trees

Decision Trees represent knowledge through tree structures that closely resemble human thought processes. To do that, they generate a series of rules or questions to predict and classify input data. Given that, several algorithms can be used to create decision trees, including *ID3* [Quinlan, 1986], *C4.5* [Quinlan, 2014], or *CART* [Breiman et al., 1984]. Nonetheless, the latter has a widely accepted version that is readily available and requires no modifications for comparison purposes. For that reason, only the process of creating such a decision tree algorithm is detailed below, based on the following steps:

1. Initially, the algorithm identifies the attribute that best distinguishes each class and assigns it as the tree root node. That attribute is often determined using statistical measures such as information gain, which quantifies the expected

reduction in uncertainty achieved by dividing the dataset based on a specific attribute.

2. Next, the algorithm establishes a criterion for partitioning the data, based on the probability distribution of each class within the tree.

3. Finally, the algorithm creates branches that divide the dataset into subsets, known as internal nodes. To evaluate the quality of those divisions, the algorithm utilises the Gini Index, which measures the effectiveness of the resulting subsets. A lower Gini Index indicates a better division.

After completing those steps, the algorithm repeats the first and second steps until it reaches a leaf node in each branch. A leaf node represents a subset of data that cannot be further divided.

2.1.1.3 Multilayer Perceptron

The Multilayer Perceptron is a widely employed neural model in modern times and one of the earliest machine learning techniques to appear [Bishop et al., 1995, Taud and Mas, 2018]. Unlike traditional neural networks, MLP can comprise multiple layers of neurons. In the simplest case, it consists of three main layers: the input layer, followed by one hidden layer, and concluding with the output layer. In this way, data is fed into the network through the input layer, with predictions generated by the output layer. In addition, hidden layers can be multiple, allowing the model to capture greater complexity for the specified problem. Given that, each layer can be represented as follows:

$$y = f(W \times x + b) \tag{2.1}$$

There, f denotes the activation function, which describes the non-linear input-output relationships. That enables the model to exhibit greater flexibility in representing arbitrary associations. Then, W corresponds to the layer weights, which are adjusted as errors are identified, with the addition of a learning rate that can either be constant or dynamic. Similarly, x represents the input data vector from the preceding layer, while b signifies the bias vector, which is an additional set of weights that facilitates the production of the layer output data. Given that, a loss function must be defined to train the network. This function yields a high value when the predicted classes do not align with the ground truth and a low one when they do. In light of that, the aim during model training is to minimise the given loss value by adjusting the layer weight values (W). For that purpose, optimisers are employed to seek suitable weight values that minimise that loss. To that end, these algorithms utilise an alpha parameter to mitigate overfitting by penalising abnormally large weight magnitudes.

2.1.1.4 Naïve Bayes

The Naïve Bayes classifiers refer to a collection of classification algorithms based on Bayes' Theorem [Rish et al., 2001]. This theorem expresses the conditional probability of an event A given event B, in terms of the conditional probability of B given A and the marginal probability of A. This definition is formalised through Bayes' Rule:

$$\Pr(A|B) = \frac{\Pr(B|A)\Pr(A)}{\Pr(B|A)\Pr(A) + \Pr(B|\neg A)\Pr(\neg A)} \tag{2.2}$$

Thus, these classifiers do not represent a single algorithm but rather a family of algorithms that share a common principle: the assumption that each pair of classified features is independent of one another. The variations among these algorithms primarily stem from the assumptions made about the distribution of $\Pr(B|A)$. For instance, continuous feature values may be taken for granted to follow a Gaussian distribution, a given multinomial distribution, or Bernoulli's multivariate event model, where the introduced features are independent binary variables (booleans) [Murphy et al., 2006]. Anyhow, despite the seemingly simplistic assumptions made by these methods, they have proven effective in various tasks. Furthermore, they offer exceptional speed compared to more sophisticated machine learning algorithms, making them worthwhile for exploration.

2.1.1.5 K-Nearest Neighbours

The K-Nearest Neighbour algorithm is a supervised and instance-based approach, which needs prelabelled data, as well as not being able to explicitly create a model [Peterson, 2009, Cunningham and Delany, 2020]. Instead, it stores the training instances and uses them during the prediction stage. Accordingly, the choice of the k value plays a crucial role in this algorithm, as it determines the number of neighbours considered in the neighbourhood for classifying the specified groups. Given that, the algorithm follows a set of steps for each observation in the data:

1. First, the distances between the selected observation and all other observations in the dataset are calculated. These distances provide similarity measures between the elements and are computed using predefined functions such as the Euclidean or Manhattan distances.

2. The closest k-elements are then selected, and a majority vote is conducted among them. The dominant class determines the final classification, considering the weights assigned to each class.

One notable challenge with KNN is the substantial memory and time requirements as the dataset size increases. Since it evaluates every observation in the data, computational resources can become significant when the number of features and data points is large. Nonetheless, KNN is regarded as an algorithm capable of delivering excellent results while being relatively easy to comprehend and implement.

2.1.1.6 Random Forest

Models based on the Random Forest algorithm have gained significant popularity in recent times [Breiman, 2001, Athey et al., 2019]. That is because by creating multiple decision trees from labelled data, these models can generate highly robust solutions. That is due to its ability to select the best possible solution in a more general and flexible manner, as well as mitigating overfitting by utilising multiple decision trees, which contributes to the strength of these models. Thus, the algorithm can be summarised into the following steps:

1. Initially, the algorithm randomly selects various subsets from the provided dataset.

2. Subsequently, decision trees are constructed for each of those subsets by following the steps described in Section 2.1.1.2. The number of decision trees built is determined by the number of estimators hyperparameter.

3. Once the trees are created, predictions are obtained from each of them. At this point, a voting process is conducted based on the resulting values, where the dominant class determines the final outcome.

4. Finally, the class with the highest number of votes is selected as the final prediction.

When making predictions using the created model, this algorithm tends to be slower when measured against others. That is primarily due to the need to average the outcomes of each tree in the final model. Despite that, it continues to be extensively used in present times due to its ability to generate highly robust models with exceptional performance. In addition, it is faster to train than many other contemporary artificial intelligence algorithms.

2.1.1.7 Extreme Gradient Boosting

Extreme Gradient Boosting is not an independent algorithm but rather a refined implementation of the Gradient Boosting one [Chen and Guestrin, 2016]. However, it is worth considering, as it has achieved notable success in various competitions and consistently demonstrated excellent results in the relevant literature [Nielsen, 2016]. Regarding its implementation, it offers enhanced efficiency and flexibility by parallelising the tree-boosting process.

Concerning the Gradient Boosting Machine (GBM) algorithm, it aims to construct a model by iteratively creating multiple "weak" prediction models, typically decision trees. That process involves the sequential creation of those trees in a stage-wise manner, following the same procedure outlined in Section 2.1.1.2. Also, like Random Forest, the number of estimators hyperparameter determines the number of trees to be generated. The objective is to progressively enhance the final model's performance.

That is achieved by defining a loss function that evaluates the performance of the most recent tree, with the assumption that the classification of all observations in the built trees will continually improve. Consequently, the resulting model is more robust, easier to fine-tune, and yields excellent outcomes. Nonetheless, it is essential to exercise caution during training, as GBM can be sensitive to overfitting and noise.

2.1.2 Deep learning

In the domain of human activity recognition, a wide range of artificial intelligence algorithms are employed. Among these, algorithms focused on the deep learning branch often yield the best results. Unlike traditional machine learning, these algorithms can automatically extract patterns from data without requiring manual feature engineering. This fact has led to an increasing number of researchers opting for this approach due to its ease of application and favourable outcomes. In any case, in the HAR field, two deep learning models, namely Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM), stand out among the rest. Hence, both techniques were employed in developing the proposed models for this Thesis.

2.1.2.1 Convolutional Neural Networks

Convolutional Neural Networks [Fukushima and Miyake, 1982, LeCun et al., 1999] are highly prevalent models nowadays. Since the gradient modification introduced in [Krizhevsky et al., 2017], they have become state-of-the-art in information extraction across various domains. These networks consist of multiple layers comprising neurons or filters that receive distinct pieces of information as input. Each filter is supplied with data from a sliding window or kernel applied over the initial signal or image. Unlike traditional neural networks, the weights of those filters remain the same [Fernandez-Blanco et al., 2020a]. As a result, the output ($X^{(l)}$) is obtained by convolving the input features ($X^{(l-1)}$) with a set of learnable filters ($W^{(l)}$), followed by the addition of biases ($b^{(l)}$). Then, an activation function ($g^{(l)}$) is applied. In the context of HAR research, and specifically in this Thesis, the most commonly used activation function is the Rectified Linear Unit (ReLU), which outputs 0 for negative values, while preserving the positive values. Thus, that process can be represented by the following equation (note that the symbol "$*$" denotes convolution):

$$X^{(l)} = g^{(l)}(X^{(l-1)} * W^{(l)} + b^{(l)}) \tag{2.3}$$

That scheme can be repeated iteratively, with each layer extracting additional features from the information accumulated in preceding layers.

After extracting the features from the input matrix and propagating them through each layer, they are fed into a fully connected perceptron, also known as a Dense layer. For the final prediction and the probability vector $p_t = [p_{t_1}, p_{t_2}, ..., p_{t_k}] \in \mathbb{R}^k$, the softmax function is normally employed. This function transforms the input values into a probability distribution, with values ranging from 0 to 1. Specifically,

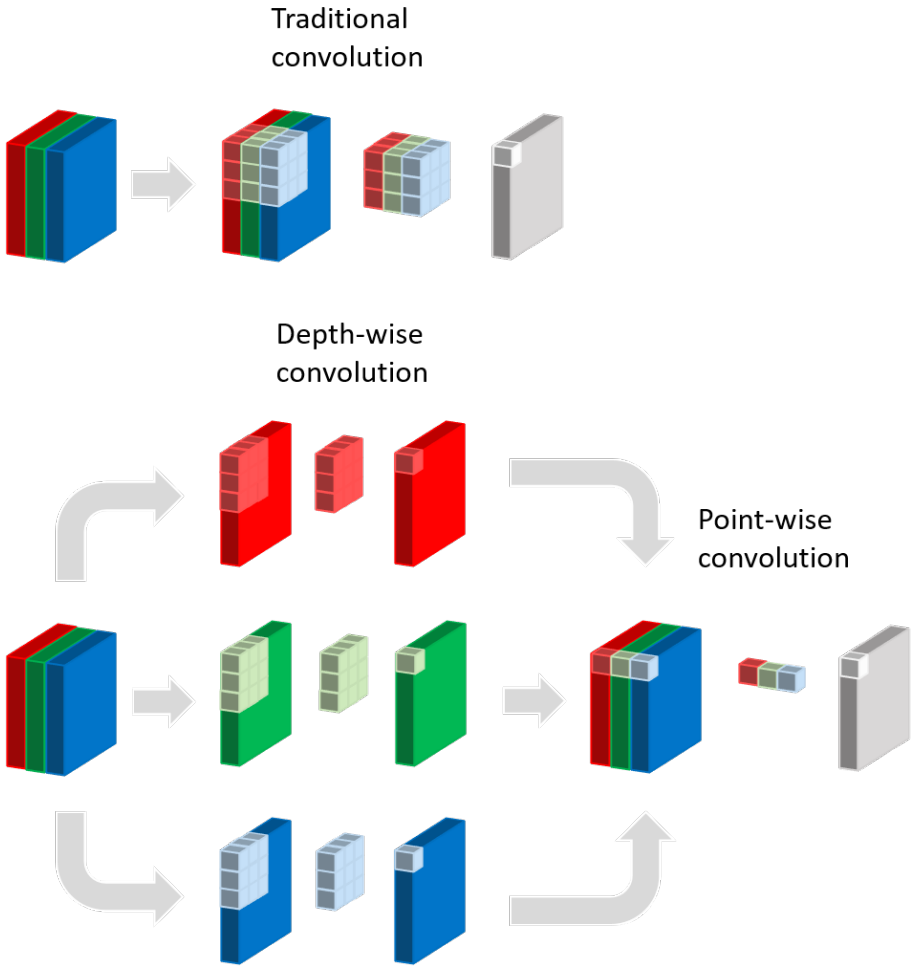


Figure 2.1: Comparison of a traditional convolution and its equivalent Depth-wise Separable convolution.

the input values are obtained from the output of the previously mentioned perceptron (z), giving rise to the following operation:

$$p_{t_i} = \frac{e^{z_i}}{\sum_{j=1}^{k} e^{z_j}} \tag{2.4}$$

The results are subsequently returned, by selecting the label with the highest probability following the softmax operation.

Anyhow, when using a large number of samples, there is a variant of this technique worth mentioning: the Depth-wise Separable Convolutional Neural Networks (DS-CNN) [Chollet, 2017]. This modification is recognised for significantly reducing the parameter requirements by applying the kernel separately to each available input signal's channel, rather than utilising it on all of them simultaneously [Fernandez-Blanco et al., 2020b]. That convolution operates similarly to the traditional approach but with fewer features for each channel. After that, the information obtained from each channel is combined through another convolution, projecting the resulting data onto a new feature map. However, the distinction lies in the fact that this convolution is performed as a point-wise convolution (i.e., 1x1 convolution). As depicted in Figure 2.1, that results in fewer operations by integrating the data from different channels. Consequently, computations are carried out using less data, achieving an equivalent outcome as traditional CNNs.

2.1.2.2 Long Short-Term Memory

In contrast to their predecessor, Recurrent Neural Networks (RNN), LSTM networks [Hochreiter and Schmidhuber, 1997] possess the capability to retain or discard data selectively. To that end, they use a series of modifications applied to the data using specialised components known as cell states. For ease of understanding, an illustration of an LSTM unit is provided in Figure 2.2. As can be seen, a typical LSTM network comprises memory blocks called cells, which facilitate the transfer of two distinct states: the cell state (c) and the hidden state (h). In this way, a structure incorporating three different gates is implemented, allowing those blocks to retain data, as outlined below:

1. Forget Gate (represented by the red gate in Figure 2.2). It removes irrelevant information that is no longer useful for learning. To do that, the input data of the current time (x_t) and the hidden state of the previous cell (h_{t-1}) are multiplied by their respective weight matrices (W). Also, a bias term (b) is added to improve data fitting. The resulting regulatory filter, or sigmoidal function (σ), is defined as:

$$f_t = \sigma\left(W_{xf} \times x_t + W_{hf} \times h_{t-1} + b_f\right) \tag{2.5}$$

That would result in a value between 0 and 1. When multiplied by the cell state, it decides whether that information should be continued or not.

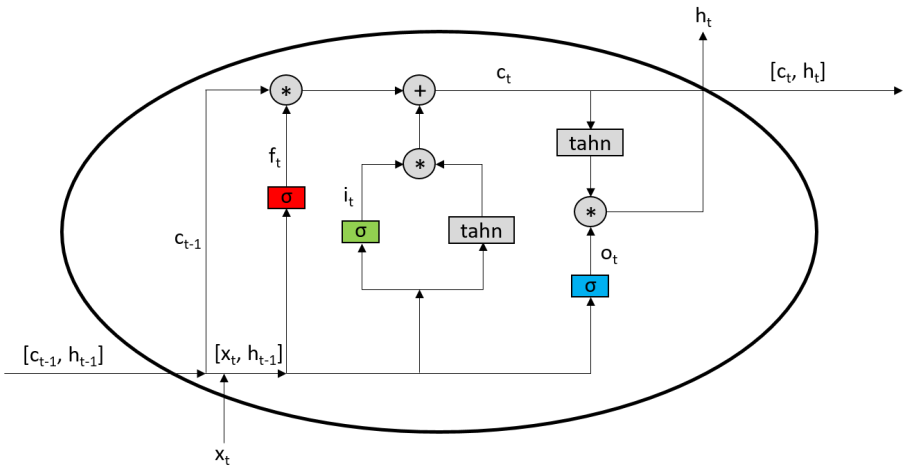


Figure 2.2: Example of a LSTM unit, as shown in [Guan and Plötz, 2017] (weight matrices and bias not displayed).

2. Input Gate (the green one in Figure 2.2). It is responsible for adding relevant information to the model and filtering out any that may be redundant. To that end, another sigmoidal function is constructed, multiplied by a hyperbolic tangent one (*tanh*) that outputs the data between -1 and 1. In this way, the tanh function decides which data can be added later to the model, using a sum operation with the information of the forget gate. These functions are represented as follows:

$$i_t = \sigma\left(W_{xi} \times x_t + W_{hi} \times h_{t-1} + b_i\right) \tag{2.6}$$

$$c'_t = tanh\left(W_{hc} \times h_{t-1} + W_{xc} \times x_t + b_c\right) \tag{2.7}$$

3. Output Gate (the blue one in Figure 2.2). This gate decides which outcome to keep, regarding that not all information flowing through the cell state may be adequate. In a similar way as before, sigmoidal and hyperbolic tangent functions are multiplied to filter those data. These functions are shown below:

$$o_t = \sigma\left(W_{xo} \times x_t + W_{ho} \times h_{t-1} + b_o\right) \tag{2.8}$$

$$c''_t = tanh(c_t) \tag{2.9}$$

Thus, new cell and hidden states are obtained. Then, they are transferred to the next unit, repeating the above process. Those states are calculated as follows:

$$c_t = f_t \times c_{t-1} + i_t \times c'_t \tag{2.10}$$

$$h_t = o_t \times c''_t \tag{2.11}$$

As for the prediction and the probability vector $p_t = [p_{t_1}, p_{t_2}, ..., p_{t_k}] \in \mathbb{R}^k$, these are calculated from the resulting hidden state (h_t). That forms a softmax function (s), already commented in Section 2.1.2.1, which results in the following equation:

$$p_t = s(W_{hk} \times h_t + b_k) \tag{2.12}$$

Finally, the class label k_t is assigned to the one with the highest value in the vector of probabilities.

Alongside traditional LSTMs, their bidirectional variant (Bi-LSTM) is also extensively used recently in the literature, with excellent results. This modification was initially introduced for preceding RNNs [Schuster and Paliwal, 1997], but it can be applied similarly in various networks. The distinctive feature of this variant is that it enables networks to store information in both directions, typically by incorporating future context (given that LSTMs conventionally store data unidirectionally from the past). To implement this modification, two distinct LSTM models are trained: one processes the input data (x) in the backwards direction, while the other operates forwards, as illustrated in the example Bi-LSTM network shown in Figure 2.3. During the training of each model, at each time step, a merging stage (f) is executed to combine the obtained outputs. That stage can be performed in many ways, but the most widely used approach is concatenation. Consequently, the outcomes (y) of the first model are concatenated with those of the second model, thereby enabling the latter to incorporate information from both directions in subsequent time steps.

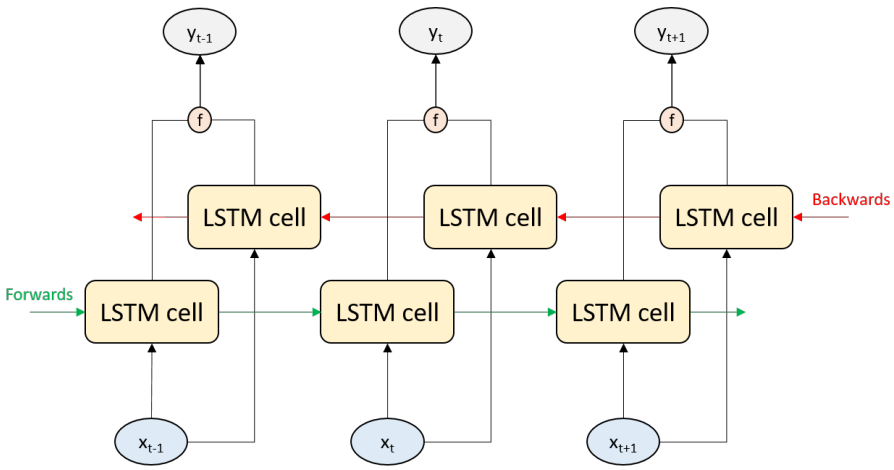


Figure 2.3: Example of a Bi-LSTM network.

2.2 Evaluation metrics

One of the most fundamental and easily interpretable metrics is the confusion matrix. A confusion matrix is a table that allows for the visualisation of a classification model's performance using a set of test data. To highlight that, Table 2.1 provides a simple example. There, note that TN, FN, TP and FP correspond to the number of true negatives, false negatives, true positives and false positives, respectively. In light of that, various commonly used terms can be derived from the confusion matrix to evaluate its performance, including precision, recall, accuracy, and F_1-score [Hossin and Sulaiman, 2015].

		Model output	
		False	**True**
Ground truth	**False**	TN	FP
	True	FN	TP

Table 2.1: Binary confusion matrix example.

Precision and recall are metrics used to measure the quality and quantity of the classifications made, respectively. Precision measures the number of true positives divided by the total number of positive results. Recall, on the other hand, measures the number of true positives divided by the total number of actual positive results that should have been returned. The formulas for these metrics are as follows:

$$Precision = \frac{TP}{TP + FP} \tag{2.13}$$

$$Recall = \frac{TP}{TP + FN} \tag{2.14}$$

Similarly, accuracy and F_1-score metrics are employed to evaluate the performance of a model in test. Accuracy represents the measure of correctly identified cases, while the F_1-score is calculated based on the harmonic mean of precision and recall. Given that, their formulas would be as follows:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{2.15}$$

$$F_1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{2.16}$$

Nonetheless, for multiclass problems like the one addressed in this Thesis, those metrics' computation changes. As a model example, Table 2.2 demonstrates how those values would be calculated, contrasting with the binary case illustrated in the previous example. Consequently, the overall precision and recall of the entire model

		Model output		
		Class 1	Class 2	Class 3
Ground truth	**Class 1**	TP	FP	
	Class 2	FN	TN	
	Class 3			

Table 2.2: TP, TN, FP and FN calculations for the "Class 1" class of a multiclass confusion matrix example.

are determined through various types of averaging, among which the following stand out: *micro* and *macro* [Grandini et al., 2020]. The micro approach considers the total of true positives, false negatives, and false positives to compute the metric, making it suitable for problems involving mutually exclusive classes. Conversely, the macro approach calculates the average metric for each label, regardless of the proportion of each one in the dataset. Concerning accuracy, it is typically computed in the same manner as in the macro case.

Furthermore, the F_1-score offers various weighting options to evaluate multiclass classification problems. In addition to the previously discussed approaches, there is also a variant of the macro strategy called *macro-weighted*. That approach considers the proportions of the data by averaging the precisions and recalls of each class involved.

Although accuracy is the widely used measure overall, the F_1-score is also closely related to the accurate classification of groups, while being less influenced by potential imbalances between classes in the datasets [Bekkar et al., 2013]. Given that, in situations where imbalances occur, accuracy may provide an inaccurate representation of the final results and the F_1-score should be calculated too.

2.3 Statistical significance methods

While the evaluation metrics discussed above give a measure of the quality of the results, the fact that one of those values is slightly higher than another does not necessarily mean that there is an actual difference between them. In order to be able to assess that fact, statistical significance techniques are used, which allow for comparing two or more results and concluding whether the models that produced them are statistically similar. In other words, they help researchers assess whether the observed differences or relationships between variables are statistically meaningful or merely the result of random variation.

Although there are numerous methods for such an evaluation, in this Thesis only two have been used: Student's t-test [Student, 1908] and Tukey test [Tukey, 1949]. The first technique compares the means of two groups to determine if they are

significantly different from each other. In addition, it assumes that the data follow a normal distribution and that the variances of the two groups are equal. Given that, the formula for the two-sample independent t-test is as follows:

$$t = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} \tag{2.17}$$

There, each $\bar{X}$ corresponds to the sample means of each specific group. Similarly, s and n are the sample standard deviation and sample sizes of the two groups, respectively. The resulting t-value is then compared to the critical one resulting from the t-distribution with degrees of freedom (df) that is calculated using the following given formula:

$$df = n_1 + n_2 - 2 \tag{2.18}$$

That critical value is usually determined using a fixed significance level of 0.05 or 0.1. If the calculated t-value is superior to that value, it means that the difference between the means is considered statistically significant.

As for the Tukey test, also known as Tukey's Honestly Significant Difference (HSD) test, it is used to compare multiple means in a pairwise fashion. That means that, although the groups are all assessed at the same time, underneath they are actually being done pair by pair, until every possible duo has been checked. Furthermore, the Tukey test is typically applied after finding a significant result in an analysis of variance (ANOVA) test, from the mean squared error (MSE) calculation. To compute that MSE, the predicted value ($\hat{y_i}$) is subtracted from the observed one (y_i), squaring the resulting difference for each observation. Afterwards, all the squared differences are summed up and divided by the sample size (n). Its formula is displayed below:

$$MSE = \frac{\sum_{i=1}^{n}(y_i - \hat{y_i})^2}{n} \tag{2.19}$$

Given that, the formula for Tukey's HSD test is also shown below:

$$HSD = q \times \sqrt{\frac{MSE}{n}} \tag{2.20}$$

Note that q refers to the critical value from the studentised range distribution, which depends on the number of groups being compared and the total number of observations. In this way, the HSD value is compared to the differences between means for each pairwise comparison. If the difference between two means is greater than the HSD value, then those means are considered statistically significantly different.

2.4 Validation and optimisation techniques

One of the most commonly used techniques in machine learning for model validation is *cross-validation* [Kohavi et al., 1995]. Before training the model, the data is divided into training and test sets a certain number of times to ensure that the models are pushed against different data. This elementary division is also known as *hold-out*. The training set is used to train the model, while the test set is utilised to evaluate its performance on previously unseen data. One popular method for that division is *k-fold cross-validation*, where the original dataset is partitioned into k equally sized subsets. One of these subsets is selected as the validation set for testing the model, while the remaining subsets are used for training. Then, the process is repeated k times, with each portion serving as the test set once. The results obtained from the model are then averaged, and performance evaluation metrics are calculated. Anyhow, there is a variant of this technique called *stratified k-fold cross-validation* worth mentioning. This variant aims to ensure that the proportion of each class in the created subsets is nearly identical, thus mitigating the possible impact of dataset imbalance on model performance.

Nevertheless, employing such an approach can give rise to one of the most prevalent challenges in machine learning research. This challenge is commonly known as the overfitting problem [Vanneschi et al., 2010, Cogswell et al., 2015]. Overfitting occurs when a model excessively memorises the training dataset rather than extracting meaningful patterns. This problem is particularly pronounced in deep learning due to the large number of trainable parameters, significantly improving the network's memorisation capability. For this reason, various regularisation techniques are typically employed to mitigate this issue. For instance, a frequently used method is to implement an early-stopping mechanism in those models [Prechelt, 1998]. This mechanism aims to halt the training process before the model begins to overfit, ensuring that the best weights obtained by the model so far are preserved. While it cannot guarantee avoidance of stopping at a local minimum, that procedure can enhance the generalisation capability of the models. Furthermore, in addition to the previous technique, each proposed model can integrate a Dropout layer too. This layer randomly omits part of the outputs, compelling the model to rely on alternate connections. With that approach, the model's generalisation capacity could also experience a significant enhancement.

As for model optimisation, one of the most widely employed techniques in the field is *grid search* [Liashchynskyi and Liashchynskyi, 2019]. Grid search involves an exhaustive search for a given algorithm's best combination of hyperparameters. In this way, each possible combination of the specified hyperparameters is tested, resulting in a time-consuming but effective process to determine the optimal model performance. Finally, the best combination is determined by assessing various evaluation metrics, with accuracy or F_1-score being the most commonly utilised criteria for selection.

Chapter 3

State of the art

The initial research in the human activity recognition field dates back to the late 1990s [Gavrila, 1999, Aggarwal and Cai, 1999]. In those works, researchers sought to be able to classify the different poses or facial expressions that an individual was performing. In this way, they used cameras to extract various images of the individuals executing those "actions". That information was then processed and fed into a traditional machine learning model to produce a final result. In any case, it was still only a modest amount of work and very limited technologically.

With the arrival of the 2000s, the subject began to become more popular, and it did not take long for works to appear in which the studied actions already presented some kind of physical movement [Mantyjarvi et al., 2001, Bao and Intille, 2004]. The aim was no longer just the correct classification of the previously mentioned poses, but also to recognise whether the person was running, talking, or any other similar activity. For that, it became necessary to use movement sensors, specifically accelerometers, placing them on different body parts, depending on the action to be studied. As with the images from before, the data acquired from those sensors underwent processing to introduce them into a traditional machine learning model with which to obtain the final classification.

However, it was not until around 2010 that this field started to become very prominent. At that point, wearable devices and, above all, smartphones, were already a well-established reality in the developed world. Thus, during that time, different datasets were presented that would later be taken as a basis by the entire scientific community. The most noteworthy sensor-based ones are discussed in Section 3.1. Then, in Section 3.2, the evolution in HAR is described from that juncture, encompassing works that used those datasets or contributed their own.

3.1 Popular datasets

Within the sensor-based HAR field, several datasets have served as benchmarks for validating experiments and expanding knowledge in the field [Ramanujam et al., 2021]. The data included in those datasets come from various wearable devices, such as activity wristbands, heart rate monitors and, more recently, smartphones. Among the latter, the most widely exploited dataset by the scientific community is the UCI HAR one [Anguita et al., 2013a]. It explores activities such as walking, sitting or going upstairs, based on data from the accelerometer and gyroscope of a specific smartphone. To carry it out, 30 people took part in the study, placing the smartphone on the left side of their waist. Regarding the activities, each one was performed for a few seconds in order to collect at least one specific feature from them. In addition, the output data were sampled at a frequency of 50 Hz, and the entire procedure took place within a laboratory setting.

Alongside the UCI HAR dataset, the WISDM one [Kwapisz et al., 2011] is also extensively used on a global scale. The activities included therein are highly similar to those from the previously mentioned dataset. Additionally, both datasets included activities that were studied for a comparable duration of several seconds. The main difference lies in the placement of the smartphone, which, in this case, was positioned in one of the front trouser pockets of each of the 29 individuals who took part in the study. Moreover, only accelerometer data were used, with a fixed frequency of 20 Hz. Once again, the whole process was conducted under controlled laboratory conditions.

Following the same premise, the HHAR dataset [Stisen et al., 2015] gathered data from eight smartphones and four smartwatches. The smartphones encompassed four distinct models, while the smartwatches consisted of two different types. Each participant had the smartphones placed tightly in a pouch attached to their waist, while two smartwatches were worn on each of their wrists. In total, only nine individuals participated in the study. The activities performed were basic examples such as walking, cycling, or running, but they were recorded over an extended period of time of five minutes. Although the data collection did not take place in a laboratory environment, participants were instructed to follow specific routes within designated timeframes. Regarding the sampling rate, efforts were made to use the maximum value supported by Android. However, the actual sampling rates displayed some variability.

Also worth mentioning is the UniMiB SHAR dataset [Micucci et al., 2017]. Its data collection process involved a particular smartphone placed in the front trouser pocket of each of the 30 participants. In this case, only accelerometer data were used, with a fixed sampling frequency of 50 Hz. Concerning the activities to be studied, these included walking, standing up, running and jumping, among others. The entire process followed a specific flow controlled by the researchers who conducted the study, in laboratory conditions.

Additionally, a set of commonly used datasets did not rely on smartphones specifically. Instead, these used a variety of sensors distributed across the body of the individual in question. Among them, the PAMAP2 one [Reiss and Stricker, 2012] deserves the initial mention. One of the most compelling points of this dataset is the activities it includes. Apart from the activities already examined in previous datasets, this one explores actions such as ironing clothes, playing football, and cleaning the house. In order to gather the data, three inertial measurement units (IMUs), alongside a heart rate monitor, were utilised. Those units consisted of an accelerometer, gyroscope, and magnetometer. In addition, they were positioned on different body parts, namely the dominant wrist, ankle, and chest. Also, the sampling frequencies for each unit were set at 100 Hz. However, it is worth noting that only nine individuals participated in the study, which was done at a laboratory. Nevertheless, no specific information was provided regarding the duration of data collection for each activity.

Following the same logic, the Opportunity dataset [Chavarriaga et al., 2013] was gathered in an environment that simulated the kitchen of a house, with its usual accessories: table, fridge and coffee machine, among others. For that purpose, only four individuals were fitted with multiple measuring devices across their entire body, focusing mainly on the shoulder and wrist areas. Those devices included accelerometer, gyroscope and magnetometer sensors, as well as other types of ambient and room location sensors. As for the activities to be carried out, in addition to those already mentioned in the first datasets in this section, it also includes other more specific ones such as opening or closing the dishwasher or cleaning the table. To that end, a dedicated workflow was established to execute the activities sequentially. Moreover, no specific sampling rate was configured for the sensors during the process.

Finally, there is a health-oriented dataset called mHealth [Banos et al., 2014, Banos et al., 2015] which was collected from wearable devices equipped with sensors such as accelerometers, gyroscopes, magnetometers, and heart rate monitors. The sampling frequency for those devices was set at 50 Hz. A total of 10 individuals took part in the study, engaging in activities such as running, raising a knee, jumping, and going downstairs, among others. The devices were positioned on the participants' chest, right wrist, and left ankle. Each activity was performed either for one minute or for the time it took to perform 20 repetitions, depending on the nature of the action. In addition, while the participants had the freedom to initiate the data collection, those specific conditions were maintained.

As can be seen, although the datasets described hold significant differences between them, they all present a common problem. Specifically, the data gathering conditions remain fixed, with the measuring devices positioned on specific body parts and activities performed in a predefined manner and for a set duration. To highlight that issue, as well as their differences, a summary of essential information from each discussed dataset is presented in Table 3.1. Note that the abbreviations used in that table correspond to accelerometer (acc.), gyroscope (gyro.), magnetometer (magn.),

Dataset	Sensor(s) used	Activities recording time	Number of subjects	Sampling frequency	Device(s) used	Device placement	Environment
UCI HAR	Acc. and gyro.	Few seconds	30	50 Hz	1 smartphone	Left belt	Controlled
WISDM	Acc.	Few seconds	29	20 Hz	1 smartphone	Front pants leg pocket	Controlled
PAMAP2	Acc., gyro., magn. and ECG monitor	-	9	100 Hz	Multiple wearable devices	Wrist, chest and dominant ankle	Controlled
Opportunity	Acc., gyro., magn., and amb.	Fixed flow duration	4	-	Multiple wearable devices	Upper body, hip, leg and shoes	Controlled
HHAR	Acc. and gyro.	5 minutes	9	Variable	8 smartphones and 4 smartwatches	Waist and wrist	Semi-controlled
UniMiB SHAR	Acc.	Fixed flow duration	30	50 Hz	1 smartphone	Trouser front pockets	Controlled
mHealth	Acc., gyro., magn., and ECG monitor	1 minute or fixed flow duration	10	50 Hz	Multiple wearable devices	Right wrist, left ankle and chest	Semi-controlled

Table 3.1: Comparison of the main HAR datasets based on sensor data.

ambient sensor (amb.), and electrocardiogram (ECG). For that reason, during the development of this Thesis, a separate dataset was built in which the participants could execute the specified activities in a much more realistic way. In that dataset, data were collected from participants' personal smartphones, enabling them to carry out and measure the actions for as long as they desired, with their own smartphone positioned in their preferred habitual manner. In this way, a new dataset gathered in a free environment with no specific conditions was obtained, as intended in the objectives of the Thesis.

3.2 Latest approaches and current challenges

All the datasets that emerged after the advent of wearable devices and the establishment of smartphones globally led to a considerable increase in HAR development. From that point on, the variability, evolution and optimisation of artificial intelligence models using that type of data has been constantly increasing. In fact, during the 2010s, numerous papers were published comparing different machine learning algorithms, with different configurations and processing the data in different ways. Those first works were mainly based on the application of Support Vector Machines, as they were offering the best results in practice [Anguita et al., 2013b, Reyes-Ortiz et al., 2014]. However, other algorithms such as K-Nearest Neighbours, Multilayer Perceptrons, those based on decision trees such as Random Forest or even those based on Bayes' Theorem, among others, were also commonly applied. Some works, such as [Wu et al., 2015] and [Chen et al., 2017], compared some of those algorithms, with different parameters and features, with SVM being the one that yielded the best performance. Furthermore, the last mentioned study also investigated the impact of smartphone orientations on the collected data. In that regard, the findings revealed that variations could notably influence the final results. Similarly, efforts were made to identify the most effective features for training those models, as demonstrated in studies like [Seto et al., 2015] and [Sousa et al., 2017]. The results from those works indicated that frequency-based parameters appeared to be the most appropriate for HAR, yielding the highest accuracy rates among the trained classifiers.

Years later, approximately between 2015 and 2020, the irruption of deep learning did not go unnoticed, and many researchers began to apply it in their studies in HAR [Yang et al., 2015]. The fact of not having to do manual feature extraction and being able to implement the models more straightforwardly made this aspect very attractive to test in the field. As before, comparisons were also made between those new algorithms and the most commonly used ones to date. For example, in [Ronao and Cho, 2016] and [Ignatov, 2018], they proved the superiority of deep learning algorithms over those used so far, including SVM, where Convolutional Neural Networks were the best by far. In fact, the latter is one of the most widely used options by the scientific community nowadays, given their speed and ease of

use [Sikder et al., 2019]. Regardless, considering the severe temporal character of this kind of data, it is also pretty common to use models based on the Long Short-Term Memory technique, as well as its bidirectional variant [Hernández et al., 2019, Li and Wang, 2022], with very similar results. That is due to the very nature of this technique, which can store information from the past or even the future, depending on the variant applied. However, one drawback of these methods is that they need a significant amount of data and time to achieve adequate training, which sets them apart from CNNs. Anyhow, there were works that opted to directly compare those two techniques, with a slight inclination towards using CNN over LSTM [Badshah, 2019, Wan et al., 2020, Teng et al., 2020]. However, although both methods are well-suited for capturing many activities of daily living (ADL), there appears to be a slight preference for CNN over LSTM due to its faster processing speed and more manageable implementation.

In contrast, not all research in this field has exclusively focused on using accelerometer and gyroscope sensors. Studies such as [Figueiredo et al., 2019] and [Voicu et al., 2019] proved the potential of incorporating other sensors, such as the magnetometer or GPS, yielding excellent results in their respective investigations. Specifically, those sensors have shown efficacy in capturing data related to different long-themed activities, such as walking or running, as proved in those studies.

Nevertheless, despite all the progress mentioned earlier, there is still a common issue shared by all those works. The data were collected under highly controlled conditions, with specific instructions in place. Therefore, it is unrealistic to anticipate similar favourable outcomes when applying their proposed models in real-life settings. Although some studies like [Ustev et al., 2013] and [Janko et al., 2018] have tried to address that problem, they are not actually practical for everyday use. Those works achieved good results by adjusting the phone's coordinates to match the Earth's. In addition, they even used different models of smartphones without a significant decrease in accuracy. However, when they changed the smartphone's orientation, the performance did suffer. Moreover, those studies did not tackle the challenge of putting the smartphone in different places, rather than just a trouser pocket.

For those reasons, the work carried out during this Thesis span focused on orienting research in this field towards more realistic environments [Garcia-Gonzalez et al., 2020a, Garcia-Gonzalez et al., 2023b, Garcia-Gonzalez et al., 2023a]. With that specific purpose, a dedicated dataset was gathered to follow those guidelines [Garcia-Gonzalez et al., 2020b]. In this way, the data included therein were exploited in subsequent work, searching the most convenient approaches to address such orientation. In fact, some very recent research, such as [Hnoohom et al., 2020] and [Hu et al., 2023], have already taken into account that new dynamic in their studies. Thus, it is feasible to think that the current trend in HAR will focus on this new orientation. Hence, all the knowledge acquired since the 1990s could be implemented into the real world in the coming years, optimising it as new technologies and needs arise in daily life.

Chapter 4

Methodology and results

Based on the objectives mentioned in Section 1.2, it has been possible to advance in the field of human activity recognition, orienting all the findings towards a real-life environment, as this need was previously highlighted. Therefore, the contributions focused on creating a more realistic database and searching for the best models to classify such data, from the most traditional to the most current, following the objectives set at the beginning of the Thesis.

In summary, this chapter highlights the main achievements of this Thesis, divided into three sections closely aligned with each of the papers attached in Appendix A, respectively.

4.1 Real-life data gathering

This section is heavily based on the contents of the article presented in Appendix A.1.

As previously mentioned, there are multiple problems in the existing datasets in the current literature, making the transfer of the research carried out so far in human activity recognition to real life very difficult. Therefore, the first contribution of this Thesis focused on gathering a new dataset that would promote the orientation of research in this field towards everyday environments. In order to make the possible findings more far-reaching, smartphones were used as the collection mechanism, due to their global use compared to any other type of wearable device.

In terms of sensors, four different ones were used: accelerometer, gyroscope, magnetometer and GPS. The accelerometer is essential to detect any slightest movement on the device. Moreover, it is the most frequently seen in any HAR-based study. As for the gyroscope, it has been shown in numerous studies to help improve classifications of actions performed, so it should be a positive in this new orientation as well. Regarding the magnetometer and GPS, although they are less common

in this field, their own natures could be beneficial for the actions to be studied. Both help to know the current point where a person is, so any displacement in this direction could help to perceive the orientation and the speed at which these changes are taking place. That fact, although irrelevant for static movements such as raising the hand or sitting down, should be positive for identifying differences in activities with displacement such as walking, running or cycling.

Furthermore, that data collection should come from individuals with different peculiarities, from the physical diversity of each one to the use and model of the personal smartphones of each one. For that reason, a straightforward Android application was implemented from which the different users could start and end their collection sessions, as well as send all this information to a dedicated data collection server. Each session consisted of performing a specific action for the duration of the individual's session, from the moment the action was initiated until it ended, both by pressing the corresponding button. In the end, 19 people participated in the study, ranging in age from approximately 25 to 50 years old. However, there is little gender diversity, with only two women out of the 19 participants. Nevertheless, the physical peculiarities, alongside the individual's varied habits and preferences regarding the use and positioning of their smartphone, exhibit significant diversity. Therefore, variability, although improvable, is also present.

With regard to the actions studied, the following four were established:

- Inactive: any activity that involves not having a smartphone on you.
- Active: any action that involves movement, but without going to a particular place. Examples include teeth brushing, dancing at a concert or playing video games.
- Walking: any displacement of the individual between two points without using a motor vehicle of any kind. For example, the activity of jogging would be classified as "walking".
- Driving: any journey made in a motorised vehicle without the need to be its driver. For example, travelling by bus would be classified as "driving".

As can be seen, those activities are typically performed over long periods of time (several minutes or even hours). This is noteworthy because, normally, the other datasets focus on much shorter times (i.e. between 2 and 10 seconds), with more specific actions such as sitting or raising the hand. Studying that type of more diffuse and long-themed activities provides an opportunity to observe, in a preliminary way, if this new orientation is feasible in HAR. Then, depending on the progress made and the definite context, new and more specific activities could be studied based on previous knowledge.

However, data gathering of that nature is very different from those previously carried out in the field. As such, some unforeseen events arose that rendered the data processing and needed to be addressed:

- Lack of sensors. Not all smartphones that took part in the dataset had all four sensors available. While the accelerometer and GPS are already mandatory on at least all current Android systems, that was not the case for the gyroscope and magnetometer at the time of the study. Out of the 19 people who participated in the data collection, five people had that problem, with at least one of those sensors missing. More specifically, the contribution of those individuals corresponds to about 20% of all measurements taken for the dataset. Therefore, instead of seeing that as a problem, it was decided to make different datasets according to the available sensors among all participants. That resulted in three different datasets, depending on which sensors produced the data and the individuals who had them available on their smartphones: accelerometer + gyroscope + magnetometer + GPS (main), accelerometer + magnetometer + GPS and accelerometer + GPS. Although the number of total participants decreases in the main dataset, that distinction enables further exploration of the impact of those missing sensors on the final ranking with the new orientation.

- Differences in the sampling rate. Even by trying to set the frequency of each sensor to the maximum allowed by Android, there are cases of sessions that differ from that value. Consequently, in some cases, the time difference between each observation is not the same. That results in the need for a superior effort when processing the data. However, that problem was considered to be potentially frequent in real life and should, therefore, also be studied. In any case, it should also be noted that there is already a baseline difference between all the sensors with respect to GPS. This sensor produces data approximately every 10 seconds, which is very different from the frequency of several samples per second present in the remaining used sensors. Even so, in many cases that frequency was greatly extended, even to the point of having sessions without GPS data that would later have to be ignored. Again, that implies a higher research effort.

- Imbalanced data distribution. Given the nature of the actions to be studied, most data belong to the "inactive" activity, as it is much easier to collect samples in that way than in any of the other options. As a general idea, using the main dataset with all sensors and ignoring any session with too much noise or that lasted less than 20 seconds, the data distribution would be as shown in Figure 4.1. Therefore, although imbalance is present and must be taken into account while training the artificial intelligence models, there are plentiful samples in each activity to obtain satisfactory results.

Finally, to prove the potential of the collected datasets, a series of preliminary experiments were carried out on them. For that purpose, Support Vector Machines were used as the machine learning algorithm, as it is a straightforward technique to apply and one of the best-performing in the HAR field. In order to be able to

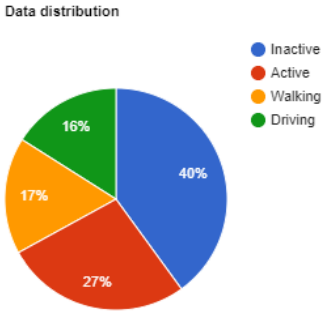


Figure 4.1: Approximate distribution of the usable data collected among the studied activities.

introduce the data into the model, a window size of 20 seconds was established, with overlaps of 19 seconds in each sliding window. In this way, a large number of samples are obtained (around half a million), while maintaining coherence with the long-themed activities to be studied. As for the features to be calculated in each of those windows, simple options such as the mean and standard deviation were chosen, among others. In addition, in order to ensure the existence of GPS data in those windows, it was necessary to replicate them on a second-by-second basis, according to the latest observations found.

All in all, different combinations of hyperparameters were tested until obtaining the results shown in Table 4.1, corresponding to the best one found [Garcia-Gonzalez et al., 2020a]. That combination consisted of the Radial Basis Function (RBF) kernel, specifically when γ was set to 0.01, along with C = 10. There, the average precision obtained with the best combination found is indicated, applying it to each collected dataset, together with its standard deviation. As can be seen, the results are not ideal and clearly indicate that there is significant room for potential improvement. Furthermore, the dataset that does not include the gyroscope seems to yield the best results, which contradicts the positive influence that this sensor appeared to have in other HAR studies. Nonetheless, they prove both the potential of the data and the orientation towards real-life environments, as initially intended.

Acc. + GPS	**Acc. + Magn. + GPS**	**Acc. + Gyro. + Magn. + GPS**
67.53% ±6.33	74.39% ±10.75	69.28% ±15.10

Table 4.1: First mean accuracies achieved for each set of data.

However, for the earlier stated reasons, an in-depth study of machine learning algorithms and their most suitable configurations for the dataset in question was conducted, which is detailed in Section 4.2. Additionally, due to the simple data preparation in this contribution, further investigation was carried out in this regard, exploring new sets of features and window sizes. Moreover, the actual influence of the gyroscope in the experiments was also studied, considering the results presented here.

4.2 Machine learning exploration

This section is heavily based on the contents of the article presented in Appendix A.2.

The preliminary results previously obtained were highly improvable and gave very little information on how to approach the new dataset. Therefore, it was decided to carry out an in-depth exploration of what could be the best machine learning approaches for the new orientation proposed in this Thesis. In order to make the study as detailed as possible, the following objectives were taken into account:

- Using multiple algorithms. In order to carry out a highly detailed study, it was necessary to apply a large number of different algorithms to observe their behaviour with the new dataset. Thus, it was decided to opt for algorithms widely used in HAR such as Support Vector Machines, Decision Trees, Multilayer Perceptron, Naïve Bayes, K-Nearest Neighbours and Random Forest. As a novelty, it was also chosen to include Extreme Gradient Boosting because of its recent great popularity in other fields and its excellent results, despite not being seen so much in HAR.

- Applying different data preparations. Due to the long-themed nature of the activities collected in the new dataset, it is possible that longer time windows may help to classify them better. Therefore, it was also necessary to test the previously arranged algorithms with different window sizes to look for significant differences. Those sizes ranged from 20 to 90 seconds, increasing in increments of 10. In the same vein, it was also decided to try a new set of features, outside of the simple statistics used previously, to see if there was any improvement in that aspect too. Given that, values such as total positive time, number of local minima and total distance travelled, among many others, were calculated. They are summarised in Table 4.2 [Garcia-Gonzalez et al., 2023b].

- Studying the actual influence of the gyroscope on the results. In preliminary experimentation with the new dataset, the best accuracy obtained was for the dataset without the gyroscope. Numerous HAR studies have shown that the gyroscope has a positive influence on the classification of actions. Therefore, given that the prior study did not have many variables to validate that result,

Features			
Primary set	**Proposed additions**		
General	**General**	**Not for GPS**	**Only for GPS**
Mean		Signal magnitude area	
Variance	Energy	Number of zero crosses	
Median absolute deviation	Number of observations	Number of local maxima	Total distance travelled
Maximum	Maximum time gap	Number of local minima	
Minimum	Minimum time gap	Total positive time	
Interquartile range		Total negative time	

Table 4.2: New feature set proposed for the machine learning exploration.

it was decided to repeat all the tests performed on the main dataset in the one without the gyroscope. In this way, it could be confirmed whether or not this sensor was positive for the new orientation proposed in this Thesis.

Anyhow, tree-based algorithms were the best overall performers for all the cases studied. Among them, Random Forest stands out as the one that obtained the best accuracy peaks. Table 4.3 shows the average confusion matrix for the best configuration found for the main dataset. In that scenario, the outcomes correspond to the Random Forest algorithm, employing a window size of 80 seconds (with overlaps of 79 seconds) and the new feature set. Although the classification accuracy improved notably from the preliminary study, some problems exist in correctly differentiating the "active" activity. That is mainly due to the inherently fuzzy nature of that action, where moments of both activity and inactivity are combined. A characteristic example of an action that would be classified as "active" is giving a lecture. In that case, during the lesson explanation, the professor may be walking around the classroom to facilitate student understanding. At the same time, they may be sitting and waiting for the students to perform some related exercise. Therefore, it is reasonable to expect some confusion when classifying that type of activity, especially with the "inactive" and the "walking" activities.

On the other hand, it is also worth noting that the proposed new set of features did not seem to bring any significant improvement compared to the results of the statistically-based one. However, substantial improvements were observed with bigger window sizes. When that value reached around 60 seconds or more, the results were significantly superior to those obtained with smaller window sizes, such as 20 or 30 seconds.

Finally, it could be concluded that the gyroscope did benefit the final results, as demonstrated in other HAR studies.

Nevertheless, given the difficulties encountered in optimally discerning all the activities studied, as well as the inconclusive results observed between the different calculated features, it was decided to look for a distinct approach. To address that, the deep learning algorithms that yielded the most promising results in HAR, namely

	Ground truth				
	Inactive	**Active**	**Walking**	**Driving**	**Precision**
Inactive	19,965	230	261	13	97.54%
Active	888	12,980	1,005	373	85.14%
Walking	24	325	6,043	94	93.17%
Driving	50	44	29	5,157	97.67%
Recall	95.40%	95.59%	82.35%	91.49%	**92.97%**

Table 4.3: Average confusion matrix for the best combination found in the machine learning exploration.

CNN and LSTM, were employed. Consequently, a comprehensive examination of their architecture was conducted to maximise their inherent capabilities. The study involved combining and comparing those algorithms in order to enhance the machine learning models developed in this contribution. Further details regarding that research can be found in Section 4.3.

4.3 Deep learning exploration

This section is heavily based on the contents of the article presented in Appendix A.3.

The application of deep learning in human activity recognition has been steadily increasing in recent years. For that reason, it would not be appropriate to limit exploration to techniques based on pure machine learning. Thus, although the results achieved in the latest experiments were truly satisfactory, deep learning was considered to have the potential to enhance them. The most recent work in this field shows that using models based on Convolutional Neural Networks or Long Short-Term Memory recurrent neural networks generally offer better results than other techniques. Therefore, the main objective of this point was to improve the classification of previously collected data by searching for the best architecture based on deep learning. In addition, during the search, it would be possible to compare those two techniques, both individually and combined, as well as using other variants of the same techniques.

However, the different sampling frequencies present in the dataset (explained in Section 4.1) made the process quite challenging. In previous experiments, by calculating several features on the selected sliding windows, those frequency differences were not as meaningful for classification. Regardless, in the case of deep learning, that feature calculation is done by the model itself, assuming that all observations are equidistant in time. Although applying some rather interpolation

is presumably the most common approach in such problems, it would eventually result in data that would not correspond to reality, invalidating it as a solution. Therefore, given the non-existence of a similar problem in HAR, an experimental solution was chosen. After thorough data analysis, it was found that those frequency changes corresponded in almost every case to two specific values: 20 ms or 200 ms. In other words, the sensors were providing data at intervals of either 20 ms or 200 ms, without taking into account the GPS, which has a wildly different frequency in itself (approximately one value every 10 seconds). Thus, only the closest observations at each 200 ms interval from the session's start were kept in the dataset. Hence, although there is data loss at moments of greater frequency, the actual data corresponding to that time instant are maintained. The number of samples is more than enough to perform a satisfactory classification, so that loss was not considered a big problem.

As for the models finally used, on the CNN side, it was opted to use its separable variant, DS-CNN (Depth-Wise Separable Convolutional Neural Networks), as they are faster than the original ones and produce equivalent results. On the LSTM side, its original form was used, in addition to its bidirectional variant, Bi-LSTM, also widely used in the scientific community, to compare their performances. In this way, five different models were formed. On the one hand, the individual models: DS-CNN, LSTM and Bi-LSTM. On the other hand, their hybrid variants: (DS-CNN)-LSTM and (DS-CNN)-(Bi-LSTM). The latter should yield better results than their original forms, as they can take advantage of the natures of each of the two algorithms and exploit them together. In any case, a comprehensive comparison of the best-performing deep learning techniques in HAR would be achieved, as initially intended. Thus, their viability could also be studied for the new direction proposed in this Thesis.

Eventually, the depicted model architecture shown in Figure 4.2 [Garcia-Gonzalez et al., 2023a] represents the final design. In order to avoid dealing with the problems present in each sensor's nature, it was decided to deal with the data from each of them independently. Then, after applying one of the models indicated in the previous paragraph, the outputs of each branch would be concatenated, resulting in a single final outcome with the data classification. The best configuration found corresponds to the hybrid model of (DS-CNN)-LSTM, with an accuracy of 94.80%, as shown in Table 4.4. These results correspond to the main dataset. Also, it should be noted that the number of samples there is lower compared to the previous work due to the utilisation of a smaller overlap in those experiments. In previous experiments, the overlap was set to the value of the specified window size minus one. However, in order to simplify data management in memory, the current experiments established an overlap equivalent to the window size minus ten. Anyhow, as can be seen, the confusion with the "active" class, present in previous scans, was largely resolved. Therefore, the initial objectives were achieved, improving the classification of the previously collected data.

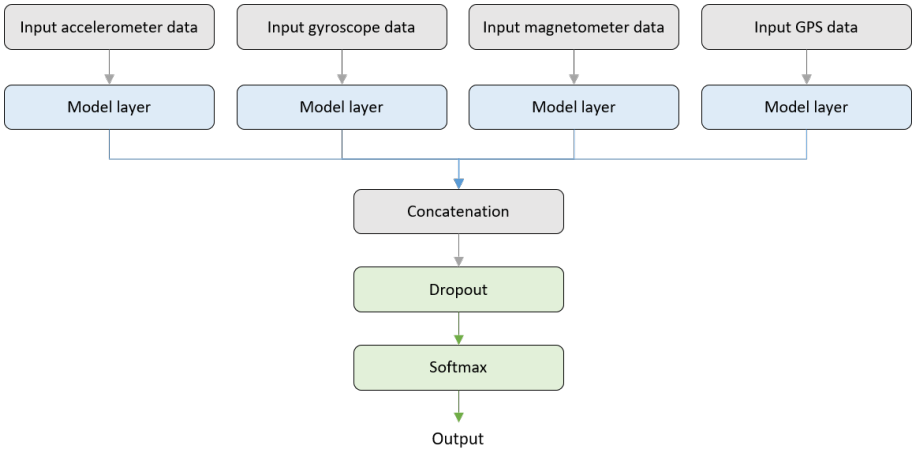


Figure 4.2: General architecture of the whole model used to carry out the deep learning experiments.

	Ground truth				
	Inactive	**Active**	**Walking**	**Driving**	**Precision**
Inactive	1,993.4	42.2	3.8	3.4	97.58%
Active	40.6	1,226.8	51.2	20.3	91.63%
Walking	2.9	45.8	613.7	4.9	91.97%
Driving	11.7	7.5	4.5	515.3	95.60%
Recall	97.31%	92.78%	91.16%	94.74%	**94.80%**

Table 4.4: Average confusion matrix for the best combination found in the deep learning exploration.

Even so, it should be noted that the results presented here correspond to a window size of 90 seconds, the maximum value addressed during the development of this Thesis. Therefore, it is not ruled out that extending the window size may benefit the data classification. However, in doing so, the possibility of classifying actions performed in shorter periods of time is lost, which reduces the variability and the number of samples to be studied. The differences in results between, for example, a 90-second window size and a 60-second window size were statistically similar. Hence, it is perhaps most reasonable to stick with a window size around that one-minute duration.

Still, although the problems in the classification of the "active" activity observed in Section 4.2 were solved to a large extent, there remain certain areas where further

improvements could be executed. Nonetheless, considering the inherently fuzzy nature of that particular activity, it might be appropriate to delve deeper into more precise actions, applying all the knowledge acquired during this Thesis development.

Algorithm	**Window size (s)**	**Accuracy**
SVM	80	86.56% ±11.30%
DT	20	89.99% ±6.13%
MLP	40	86.85% ±6.12%
NB	80	83.27% ±7.78%
KNN	80	89.02% ±8.00%
RF	80	92.97% ±6.23%
XGB	70	92.23% ±7.30%
DS-CNN	90	90.70% ±7.29%
LSTM	90	93.52% ±5.59%
Bi-LSTM	90	93.09% ±5.10%
(DS-CNN)-LSTM	**90**	**94.80% ± 4.09%**
(DS-CNN)-(Bi-LSTM)	90	94.16% ±5.06%

Table 4.5: Comparison of the best results obtained on the main proposed dataset, with the methods used during the Thesis and for the window size that yielded the best performance.

In summary, Table 4.5 provides an overview of the top-performing outcomes achieved by each algorithm implemented on the main dataset proposed in this Thesis, according to their best window size. The deep learning algorithms proved to be the most effective in yielding favorable results, while also acknowledging the noteworthy performance of the tree-based algorithms. Among them, the cases involving LSTM were the most successful, proving to be the most suitable option for the given dataset among all the options studied. Nonetheless, it is noteworthy to observe how the best results, in the machine learning cases, remain within the range of 80 seconds, with a slight decrease in accuracy percentage for window sizes of 90 seconds. Conceivably, if the same specific window size was used in the deep learning experiments, a similar trend could be observed. However, even though the best results were obtained with an 80-second window size for the machine learning cases, the accuracy achieved with the other window sizes does not differ significantly from the best one. To prove that point, Figure 4.3 shows a Tukey test performed on the best case found for RF. Similarly, another Tukey test was conducted for the deep learning cases to observe the same behaviour. In that case, it was performed for all used models, as well as the different window sizes applied (30, 60, and 90 seconds). As seen in Figure 4.4, there are no significant differences between 90 and 60 seconds, but there are differences

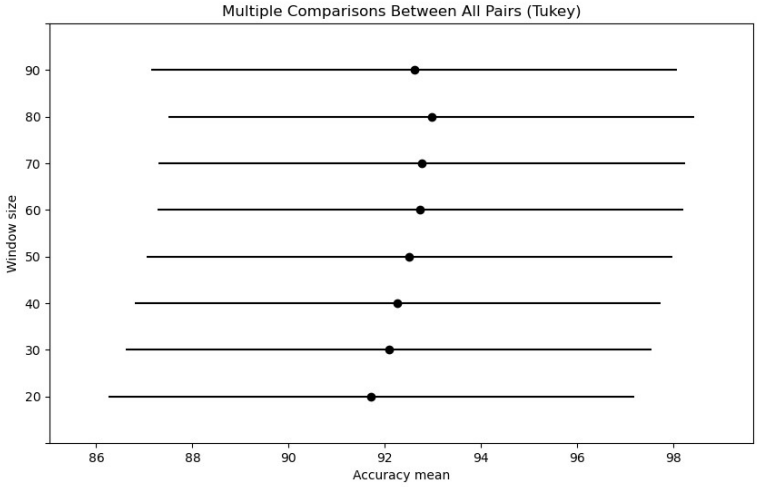


Figure 4.3: Results of the Tukey test performed for all window sizes used with Random Forest, for its best case found.

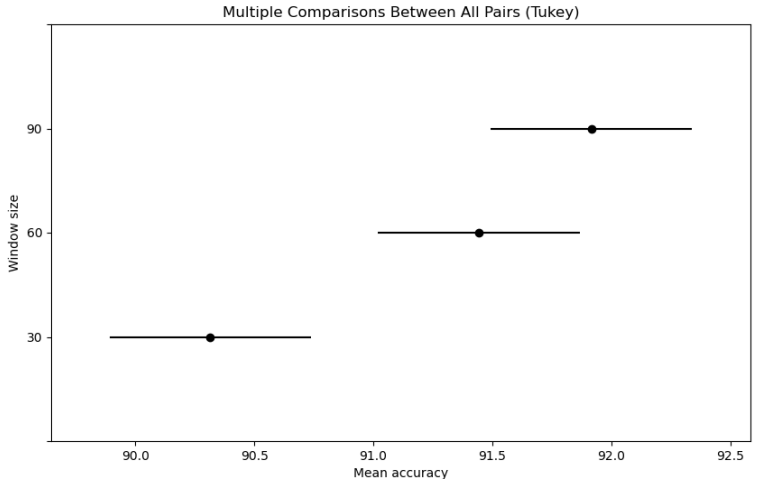


Figure 4.4: Tukey test results for each group of accuracy values referring to each selected window size, for all the deep learning models.

with the 30-second case. That finding, combined with the observations from the machine learning exploration, suggests that using more tightly adjusted window sizes (around one minute) may be sufficient to achieve satisfactory classification results.

Chapter 5

Conclusions and future work

This Thesis has addressed many challenges associated with the quest to orient research in the human activity recognition field towards real-life environments. The conclusions of all the work carried out are summarised below, as well as a number of future lines of research that could be positive in that area.

5.1 Conclusions

The main objective of this Thesis was to orient current research in the human activity recognition field towards real-life environments. Given the lack of data focused in that direction, it was essential to collect a new dataset in order to start that process. In addition, it was also crucial to deal with data from the sensors embedded in today's smartphones, given their prominent use in the current developed world. Therefore, the first contribution of this Thesis focused on that point. The resulting dataset contains information from 19 different individuals, each with distinct physical peculiarities and ways of using their particular smartphone and who performed a range of activities almost freely. In terms of sensors, the accelerometer, gyroscope, magnetometer and GPS were used. In this way, the information collected was diverse enough and realistic to be able to transfer future findings to real-world problems.

However, as it was a very different data collection from any other conducted in this field so far, several unforeseen issues arose. First, not all individuals who participated in the study had all the necessary sensors available on their smartphones. Consequently, in some cases, data from a few individuals are not available. Secondly, the sampling frequency was not always the same for each sensor. Even if the maximum value allowed by Android was set in that respect, there are some cases where the sampling rate changes, which means that more effort has to be put into data processing. Finally, the resulting dataset exhibits a noticeable degree of imbalance towards one of the four activities studied. Although the number of samples

in the rest of the actions is more than enough to carry out a correct classification, it is something that must be considered in the developments to be made on those data.

All in all, the scientific community currently has at its disposal a dataset collected in a real-life environment. Even considering the problems described in the previous paragraph, those challenges might become commonplace in other future datasets that follow a similar orientation. Therefore, even if they require more research effort, they could still result in valuable findings for advancing HAR in that direction. In this way, researchers can draw on such information and make their own developments, optimising them and successfully focusing on more realistic scenarios.

Following that guideline, a comprehensive comparison of multiple artificial intelligence algorithms was carried out, with numerous combinations of hyperparameters and features, as well as different window sizes. In terms of hyperparameters, much information was obtained on which cases favoured classification more, according to the algorithm, although with some arbitrariness depending on the parameter studied. Regarding the features, experimentation was carried out on two sets: the classical ones, which rely on statistics (including mean and standard deviation), and another group that relates more to the distinct aspects of the collected signals. Unfortunately, the results were not entirely conclusive, so it is unclear which would be the most appropriate case for the proposed dataset. Finally, it was found that larger window sizes (around one minute) had a positive influence on the final classification.

Nonetheless, all the experiments carried out on said dataset validated the possibility of accomplishing the orientation proposed in this Thesis. The classification of the activities studied is higher than 90% in most cases, reaching 94.80% in the best case found. In this sense, the best-performing algorithms were those based on deep learning, highlighting the hybrid model resulting from joining Convolutional Neural Networks and recurrent neural networks based on the Long Short-Term Memory technique. Also noteworthy was the performance of tree-based models, especially Random Forest, which obtained results very close to those of the deep learning algorithms. In any case, in virtually all instances, peak accuracy was achieved with the largest window sizes used in this Thesis (between 60 and 90 seconds). With smaller window sizes, the implemented AI systems were not as adept at discerning the distinct features of the long-duration activities studied here. In addition, it is worth highlighting the fact that algorithms based on LSTM yielded the best results. Those networks are renowned for their effective handling of time series data, which, in conjunction with the observed behaviour under different window sizes, demonstrates the significance of appropriately addressing data temporality when classifying such actions. That, in turn, constitutes one of the most common challenges within the HAR field. Still, there is potential for further improvement in the results. After all, the developments carried out in this Thesis, although diverse, are only a part of the large variability that could arise over the years.

Finally, it should also be noted that, during the years of development of this Thesis, more papers have arisen that have made use of the proposed dataset

[Hnoohom et al., 2020, Hu et al., 2023], also with good results. At the same time, other datasets with the same orientation sought here are also starting to emerge [Quan et al., 2022]. Taking into account all the research carried out during this Thesis, together with the last points made here, there is no doubt that the project has been a success. With the possible advances that will occur in the coming years, there is a strong possibility that the accumulated knowledge in HAR could be, eventually, directly applied to real-world scenarios.

5.2 Future work

Even if all the work carried out during the development of this Thesis was finally successful, it is true that some unforeseen problems had to be solved. No matter how efficient and convincing the solution is, there will always be room for improvement. Therefore, the following are some ideas that could be further developed and improved in future lines of research:

- Different approaches for data processing. Given the problem of the inconsistency of the sampling frequency of each sensor in the dataset provided, the solutions in this sense can be very diverse. During the development of this Thesis, an experimental solution was carried out to correct that problem and to be able to continue with the implementation of the models. Although the result is considered satisfactory, it is likely that with different methods the outcomes will be more positive than with the one proposed there. On the other hand, although multiple window sizes were studied during the development of this Thesis, they were still ad hoc to the proposed solutions. For those reasons, a more in-depth exploration of those issues could result in a better performance of the final models that perform the classification of the previously processed data.

- New feature sets. All the features calculated in the work carried out during the development of this Thesis were focused on the time domain, given the problems related to the data discussed above. In the event of a successful resolution to that issue, it is possible that other features from the frequency domain could be effective at improving the final results. In fact, numerous studies in HAR have applied such features in their work, with good results [Seto et al., 2015, Sousa et al., 2017]. Therefore, it is feasible to think that they will work in a similar way with the new orientation proposed in this Thesis.

- Other algorithms and configurations. Nowadays, there are many different artificial intelligence algorithms available. Moreover, the hyperparameter combinations influencing their performance are often quite broad. Also, depending on the final model's architecture and the hybrid models that result

from combining them, the outcomes can be significantly different. Although it is considered that, in this Thesis, a good selection of all those matters has been made, it is still limited, with much room for improvement. It is possible that other configurations could result in a more accurate classification of the data. In fact, the application of models based on the Transformer architecture [Vaswani et al., 2017] could be beneficial, considering their recent remarkable results in numerous areas related to artificial intelligence. Therefore, this point is presented as a further line of future research that could be positive for the project.

- New data. As mentioned above, the activities studied in the collected dataset are considerably generic. Thus, once their potential has been demonstrated and the feasibility of the new orientation proposed here has been seen, it may be time to refine and study new activities. In this way, the idea would be to develop new datasets in which more specific actions are studied. Such activities could be similar to those considered in previous work in HAR, such as raising a hand, standing up or going upstairs. The difference would lie in how the data are collected, which should be as freely and flexibly as possible to bring it as close as attainable to the real world. In such a manner, the final applicability of the systems that could result from such work would be much more practical and direct.

- Testing the developed models on different real-life datasets. Just as previously highlighted, using novel data could yield additional valuable insights. As new datasets collected from real-world scenarios become available, the models developed during the course of this Thesis could be examined in contexts distinct from those studied here. In this way, further progress could be made in this line of inquiry, potentially reaffirming the findings of this research and streamlining the transfer of all the outcomes achieved.

Chapter 6

Research results

All the work done during the Thesis period was validated by publishing different articles in international journals. Specifically, each one links to one of the contributions described in Chapter 4. Those articles are listed in the following, respectively with those contributions:

- Garcia-Gonzalez, D., Rivero, D., Fernandez-Blanco, E., and Luaces, M. R. (2020). **A public domain dataset for real-life human activity recognition using smartphone sensors.** *Sensors*, 20(8):2200. DOI: 10.3390/s20082200.
  IF (JCR): 3.9 (Q2).

- Garcia-Gonzalez, D., Rivero, D., Fernandez-Blanco, E., and Luaces, M. R. (2023). **New machine learning approaches for real-life human activity recognition using smartphone sensor-based data.** *Knowledge-Based Systems*, 262:110260. DOI: 10.1016/j.knosys.2023.110260.
  IF (JCR): 8.8 (Q1).

- Garcia-Gonzalez, D., Rivero, D., Fernandez-Blanco, E., and Luaces, M. R. (2023). **Deep learning models for real-life human activity recognition from smartphone sensor data.** *Internet of Things*, page 100925. DOI: 10.1016/j.iot.2023.100925.
  IF (JCR): 5.9 (Q1).

Among them, it is worth highlighting the first one, where the proposed dataset is published. At the time of writing this Thesis, that work had more than 90 academic citations, proving the interest and its relevance in the HAR scientific community.

Similarly, the dataset gathered for this Thesis was also presented at the XoveTIC 2023 conference. At the time of writing this document, it had not yet been published, so its DOI and other specific information were unknown. In any case, the official document should look similar to the following:

- Garcia-Gonzalez, D., Rivero, D., Fernandez-Blanco, E., and Luaces, M. R. (2023). **Introducing a human activity recognition dataset gathered on real-life conditions.** *Proceedings of VI XoveTIC Conference.*

Apart from that, work has also been carried out which has not yet been published. During the research stay at the Aristotle University of Thessaloniki in Greece, it has been possible to initiate a project related to mobility data from Madrid's Community, in collaboration with Apostolos N. Papadopoulos. The data come mainly from the public transport users of that community. These include urban and interurban buses, suburban trains, metro and tram. From those data, it is possible to develop a system capable of predicting the density of people in a specific area of Madrid. That is particularly attractive for defining new transport lines or introducing reinforcements when certain events occur. Moreover, by employing different clustering techniques, it is possible to delimit the territory into a specific number of areas with which to relate the previously arranged data. Then, based on that information, various artificial intelligence techniques can be applied to obtain the final approximate future value. Currently, it has been possible to develop such a system to make predictions based on previously defined areas with the aforementioned algorithms. Although it is still under development, the results are getting closer and closer to reality and it is hoped to obtain some outstanding merit in the near future.

Furthermore, it is worth noting that this work was initiated as part of the GEMA (GEstión de la MovilidAd) research project, which aimed to address four research challenges: intelligent planning of routes, agendas, and schedules; automatic semantic labelling of trajectories; efficient representation, storage, and exploitation of trajectories; and automated development of Mobile Workforce Management software. Additionally, four Galician SMEs participated in this project, in collaboration with CITIC: Gestora de Subproductos de Galicia S.L., Enxenio S.L., AO Mayores Servicios Sociales S.L., and Taprega Prevención de Riesgos S.L. Therefore, the outcomes of this Thesis may be transferred to those industries.

Bibliography

[Aggarwal and Cai, 1999] Aggarwal, J. K. and Cai, Q. (1999). Human motion analysis: A review. *Computer vision and image understanding*, 73(3):428–440.

[Aggarwal and Xia, 2014] Aggarwal, J. K. and Xia, L. (2014). Human activity recognition from 3d data: A review. *Pattern Recognition Letters*, 48:70–80.

[Anguita et al., 2013a] Anguita, D., Ghio, A., Oneto, L., Parra, X., and Reyes-Ortiz, J. L. (2013a). A public domain dataset for human activity recognition using smartphones. In *Esann*.

[Anguita et al., 2013b] Anguita, D., Ghio, A., Oneto, L., Parra, X., and Reyes-Ortiz, J. L. (2013b). Training computationally efficient smartphone–based human activity recognition models. In *International Conference on Artificial Neural Networks*, pages 426–433. Springer.

[Athey et al., 2019] Athey, S., Tibshirani, J., Wager, S., et al. (2019). Generalized random forests. *Annals of Statistics*, 47(2):1148–1178.

[Attal et al., 2015] Attal, F., Mohammed, S., Dedabrishvili, M., Chamroukhi, F., Oukhellou, L., and Amirat, Y. (2015). Physical human activity recognition using wearable sensors. *Sensors*, 15(12):31314–31338.

[Badshah, 2019] Badshah, M. (2019). Sensor-based human activity recognition using smartphones.

[Banos et al., 2014] Banos, O., Garcia, R., Holgado-Terriza, J. A., Damas, M., Pomares, H., Rojas, I., Saez, A., and Villalonga, C. (2014). mhealthdroid: a novel framework for agile development of mobile health applications. In *Ambient Assisted Living and Daily Activities: 6th International Work-Conference, IWAAL 2014, Belfast, UK, December 2-5, 2014. Proceedings 6*, pages 91–98. Springer.

[Banos et al., 2015] Banos, O., Villalonga, C., Garcia, R., Saez, A., Damas, M., Holgado-Terriza, J. A., Lee, S., Pomares, H., and Rojas, I. (2015). Design, implementation and validation of a novel open framework for agile development of mobile health applications. *Biomedical engineering online*, 14(2):1–20.

[Bao and Intille, 2004] Bao, L. and Intille, S. S. (2004). Activity recognition from user-annotated acceleration data. In *Pervasive Computing: Second International Conference, PERVASIVE 2004, Linz/Vienna, Austria, April 21-23, 2004. Proceedings 2*, pages 1–17. Springer.

[Beddiar et al., 2020] Beddiar, D. R., Nini, B., Sabokrou, M., and Hadid, A. (2020). Vision-based human activity recognition: a survey. *Multimedia Tools and Applications*, 79:30509–30555.

[Bekkar et al., 2013] Bekkar, M., Djemaa, H. K., and Alitouche, T. A. (2013). Evaluation measures for models assessment over imbalanced data sets. *J Inf Eng Appl*, 3(10).

[Bishop et al., 1995] Bishop, C. M. et al. (1995). *Neural networks for pattern recognition*. Oxford university press.

[Breiman, 2001] Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

[Breiman et al., 1984] Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. A. (1984). *Classification and regression trees*. CRC press.

[Chavarriaga et al., 2013] Chavarriaga, R., Sagha, H., Calatroni, A., Digumarti, S. T., Tröster, G., Millán, J. d. R., and Roggen, D. (2013). The opportunity challenge: A benchmark database for on-body sensor-based activity recognition. *Pattern Recognition Letters*, 34(15):2033–2042.

[Chen and Guestrin, 2016] Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794.

[Chen et al., 2017] Chen, Z., Zhu, Q., Soh, Y. C., and Zhang, L. (2017). Robust human activity recognition using smartphone sensors via ct-pca and online svm. *IEEE Transactions on Industrial Informatics*, 13(6):3070–3080.

[Chollet, 2017] Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258.

[Cogswell et al., 2015] Cogswell, M., Ahmed, F., Girshick, R., Zitnick, L., and Batra, D. (2015). Reducing overfitting in deep networks by decorrelating representations. *arXiv preprint arXiv:1511.06068*.

[Cortes and Vapnik, 1995] Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.

[Cunningham and Delany, 2020] Cunningham, P. and Delany, S. J. (2020). k-nearest neighbour classifiers–. *arXiv preprint arXiv:2004.04523*.

[Demrozi et al., 2020] Demrozi, F., Pravadelli, G., Bihorac, A., and Rashidi, P. (2020). Human activity recognition using inertial, physiological and environmental sensors: a comprehensive survey. *IEEE Access*.

[Du et al., 2019] Du, Y., Lim, Y., and Tan, Y. (2019). A novel human activity recognition and prediction in smart home based on interaction. *Sensors*, 19(20):4474.

[Fernandez-Blanco et al., 2020a] Fernandez-Blanco, E., Rivero, D., and Pazos, A. (2020a). Convolutional neural networks for sleep stage scoring on a two-channel eeg signal. *Soft Computing*, 24:4067–4079.

[Fernandez-Blanco et al., 2020b] Fernandez-Blanco, E., Rivero, D., and Pazos, A. (2020b). Eeg signal processing with separable convolutional neural network for automatic scoring of sleeping stage. *Neurocomputing*, 410:220–228.

[Ferrari et al., 2020] Ferrari, A., Micucci, D., Mobilio, M., and Napoletano, P. (2020). On the personalization of classification models for human activity recognition. *IEEE Access*, 8:32066–32079.

[Figueiredo et al., 2019] Figueiredo, J., Gordalina, G., Correia, P., Pires, G., Oliveira, L., Martinho, R., Rijo, R., Assuncao, P., Seco, A., and Fonseca-Pinto, R. (2019). Recognition of human activity based on sparse data collected from smartphone sensors. In *2019 IEEE 6th Portuguese Meeting on Bioengineering (ENBENG)*, pages 1–4. IEEE.

[Fukushima and Miyake, 1982] Fukushima, K. and Miyake, S. (1982). Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition. In *Competition and cooperation in neural nets*, pages 267–285. Springer.

[Garcia-Gonzalez et al., 2020a] Garcia-Gonzalez, D., Rivero, D., Fernandez-Blanco, E., and Luaces, M. R. (2020a). A public domain dataset for real-life human activity recognition using smartphone sensors. *Sensors*, 20(8):2200.

[Garcia-Gonzalez et al., 2023a] Garcia-Gonzalez, D., Rivero, D., Fernandez-Blanco, E., and Luaces, M. R. (2023a). Deep learning models for real-life human activity recognition from smartphone sensor data. *Internet of Things*, page 100925.

[Garcia-Gonzalez et al., 2023b] Garcia-Gonzalez, D., Rivero, D., Fernandez-Blanco, E., and Luaces, M. R. (2023b). New machine learning approaches for real-life human activity recognition using smartphone sensor-based data. *Knowledge-Based Systems*, 262:110260.

[Garcia-Gonzalez et al., 2020b] Garcia-Gonzalez, D., Rivero, D., Fernandez-Blanco, E., and R. Luaces, M. (2020b). A public domain dataset for real-life human

activity recognition using smartphone sensors. Mendeley Data, V2. Available online: https://data.mendeley.com/datasets/3xm88g6m6d/2.

[Gavrila, 1999] Gavrila, D. M. (1999). The visual analysis of human movement: A survey. *Computer vision and image understanding*, 73(1):82–98.

[Grandini et al., 2020] Grandini, M., Bagli, E., and Visani, G. (2020). Metrics for multi-class classification: an overview. *arXiv preprint arXiv:2008.05756*.

[Guan and Plötz, 2017] Guan, Y. and Plötz, T. (2017). Ensembles of deep lstm learners for activity recognition using wearables. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(2):1–28.

[Hassan et al., 2018] Hassan, M. M., Uddin, M. Z., Mohamed, A., and Almogren, A. (2018). A robust human activity recognition system using smartphone sensors and deep learning. *Future Generation Computer Systems*, 81:307–313.

[Hernández et al., 2019] Hernández, F., Suárez, L. F., Villamizar, J., and Altuve, M. (2019). Human activity recognition on smartphones using a bidirectional lstm network. In *2019 XXII Symposium on Image, Signal Processing and Artificial Vision (STSIVA)*, pages 1–5. IEEE.

[Hnoohom et al., 2020] Hnoohom, N., Jitpattanakul, A., and Mekruksavanich, S. (2020). Real-life human activity recognition with tri-axial accelerometer data from smartphone using hybrid long short-term memory networks. In *2020 15th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP)*, pages 1–6. IEEE.

[Hochreiter and Schmidhuber, 1997] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

[Hossin and Sulaiman, 2015] Hossin, M. and Sulaiman, M. (2015). A review on evaluation metrics for data classification evaluations. *International Journal of Data Mining & Knowledge Management Process*, 5(2):1.

[Hu et al., 2023] Hu, L., Zhao, K., Ling, B. W.-K., and Lin, Y. (2023). Activity recognition via correlation coefficients based graph with nodes updated by multi-aggregator approach. *Biomedical Signal Processing and Control*, 79:104255.

[Ignatov, 2018] Ignatov, A. (2018). Real-time human activity recognition from accelerometer data using convolutional neural networks. *Applied Soft Computing*, 62:915–922.

[Janko et al., 2018] Janko, V., Rešçiç, N., Mlakar, M., Drobnič, V., Gams, M., Slapničar, G., Gjoreski, M., Bizjak, J., Marinko, M., and Luštrek, M. (2018). A new frontier for activity recognition: The sussex-huawei locomotion challenge.

In *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*, pages 1511–1520.

[Jeong and Oh, 2021] Jeong, S. and Oh, D. (2021). Development of a hybrid deep-learning model for the human activity recognition based on the wristband accelerometer signals. *Journal of Internet Computing and Services*, 22(3):9–16.

[Ke et al., 2013] Ke, S.-R., Thuc, H. L. U., Lee, Y.-J., Hwang, J.-N., Yoo, J.-H., and Choi, K.-H. (2013). A review on video-based human activity recognition. *Computers*, 2(2):88–131.

[Kohavi et al., 1995] Kohavi, R. et al. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145. Montreal, Canada.

[Krizhevsky et al., 2017] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90.

[Kwapisz et al., 2011] Kwapisz, J. R., Weiss, G. M., and Moore, S. A. (2011). Activity recognition using cell phone accelerometers. *ACM SigKDD Explorations Newsletter*, 12(2):74–82.

[Lago et al., 2019] Lago, P., Takeda, S., Okita, T., and Inoue, S. (2019). Measured: Evaluating sensor-based activity recognition scenarios by simulating accelerometer measures from motion capture. In *Human Activity Sensing*, pages 135–149. Springer.

[Lane et al., 2011] Lane, N., Xu, Y., lu, H., Hu, S., Choudhury, T., Campbell, A., and Zhao, F. (2011). Enabling large-scale human activity inference on smartphones using community similarity networks (csn). pages 355–364.

[Lara and Labrador, 2012] Lara, O. D. and Labrador, M. A. (2012). A survey on human activity recognition using wearable sensors. *IEEE communications surveys & tutorials*, 15(3):1192–1209.

[Lawal and Bano, 2019] Lawal, I. A. and Bano, S. (2019). Deep human activity recognition using wearable sensors. In *Proceedings of the 12th ACM International Conference on PErvasive Technologies Related to Assistive Environments*, pages 45–48.

[LeCun et al., 1999] LeCun, Y., Haffner, P., Bottou, L., and Bengio, Y. (1999). Object recognition with gradient-based learning. In *Shape, contour and grouping in computer vision*, pages 319–345. Springer.

[Li and Wang, 2022] Li, Y. and Wang, L. (2022). Human activity recognition based on residual network and bilstm. *Sensors*, 22(2):635.

[Liashchynskyi and Liashchynskyi, 2019] Liashchynskyi, P. and Liashchynskyi, P. (2019). Grid search, random search, genetic algorithm: A big comparison for nas. *arXiv preprint arXiv:1912.06059*.

[Liu et al., 2021] Liu, R., Ramli, A. A., Zhang, H., Datta, E., Henricson, E., and Liu, X. (2021). An overview of human activity recognition using wearable sensors: Healthcare and artificial intelligence. *arXiv preprint arXiv:2103.15990*.

[Ma, 2021] Ma, F. (2021). Action recognition of dance video learning based on embedded system and computer vision image. *Microprocessors and Microsystems*, 81:103779.

[Mantyjarvi et al., 2001] Mantyjarvi, J., Himberg, J., and Seppanen, T. (2001). Recognizing human motion with multiple acceleration sensors. In *2001 ieee international conference on systems, man and cybernetics. e-systems and e-man for cybernetics in cyberspace (cat. no. 01ch37236)*, volume 2, pages 747–752. IEEE.

[Micucci et al., 2017] Micucci, D., Mobilio, M., and Napoletano, P. (2017). Unimib shar: A dataset for human activity recognition using acceleration data from smartphones. *Applied Sciences*, 7(10):1101.

[Murphy et al., 2006] Murphy, K. P. et al. (2006). Naive bayes classifiers. *University of British Columbia*, 18(60).

[Nielsen, 2016] Nielsen, D. (2016). Tree boosting with xgboost-why does xgboost win" every" machine learning competition? Master's thesis, NTNU.

[Peterson, 2009] Peterson, L. E. (2009). K-nearest neighbor. *Scholarpedia*, 4(2):1883.

[Prechelt, 1998] Prechelt, L. (1998). Early stopping-but when? In *Neural Networks: Tricks of the trade*, pages 55–69. Springer.

[Qi et al., 2019] Qi, W., Su, H., Yang, C., Ferrigno, G., De Momi, E., and Aliverti, A. (2019). A fast and robust deep convolutional neural networks for complex human activity recognition using smartphone. *Sensors*, 19(17):3731.

[Quan et al., 2022] Quan, H., Hu, Y., and Bonarini, A. (2022). Polimi-itw-s: A large-scale dataset for human activity recognition in the wild. *Data in Brief*, 43:108420.

[Quinlan, 1986] Quinlan, J. R. (1986). Induction of decision trees. *Machine learning*, 1(1):81–106.

[Quinlan, 2014] Quinlan, J. R. (2014). *C4. 5: programs for machine learning*. Elsevier.

[Raeiszadeh and Tahayori, 2018] Raeiszadeh, M. and Tahayori, H. (2018). A novel method for detecting and predicting resident's behavior in smart home. In *2018 6th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS)*, pages 71–74. IEEE.

[Ramanujam et al., 2021] Ramanujam, E., Perumal, T., and Padmavathi, S. (2021). Human activity recognition with smartphone and wearable sensors using deep learning techniques: A review. *IEEE Sensors Journal*, 21(12):13029–13040.

[Reiss and Stricker, 2012] Reiss, A. and Stricker, D. (2012). Introducing a new benchmarked dataset for activity monitoring. In *2012 16th international symposium on wearable computers*, pages 108–109. IEEE.

[Reyes-Ortiz et al., 2014] Reyes-Ortiz, J.-L., Oneto, L., Ghio, A., Samá, A., Anguita, D., and Parra, X. (2014). Human activity recognition on smartphones with awareness of basic activities and postural transitions. In *International conference on artificial neural networks*, pages 177–184. Springer.

[Rish et al., 2001] Rish, I. et al. (2001). An empirical study of the naive bayes classifier. In *IJCAI 2001 workshop on empirical methods in artificial intelligence*, volume 3, pages 41–46.

[Ronao and Cho, 2016] Ronao, C. A. and Cho, S.-B. (2016). Human activity recognition with smartphone sensors using deep learning neural networks. *Expert systems with applications*, 59:235–244.

[Sansano et al., 2020] Sansano, E., Montoliu, R., and Belmonte Fernandez, O. (2020). A study of deep neural networks for human activity recognition. *Computational Intelligence*, 36(3):1113–1139.

[Schuster and Paliwal, 1997] Schuster, M. and Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681.

[Seto et al., 2015] Seto, S., Zhang, W., and Zhou, Y. (2015). Multivariate time series classification using dynamic time warping template selection for human activity recognition. In *2015 IEEE Symposium Series on Computational Intelligence*, pages 1399–1406. IEEE.

[Shoaib et al., 2016] Shoaib, M., Bosch, S., Incel, O., Scholten, H., and Havinga, P. (2016). Complex human activity recognition using smartphone and wrist-worn motion sensors. *Sensors*, 16(4):426.

[Sikder et al., 2019] Sikder, N., Chowdhury, M. S., Arif, A. S., and Nahid, A.-A. (2019). Human activity recognition using multichannel convolutional neural network. In *2019 5th Int. Conf. Adv. Electr. Eng.*

[Soleimani and Nazerfard, 2021] Soleimani, E. and Nazerfard, E. (2021). Cross-subject transfer learning in human activity recognition systems using generative adversarial networks. *Neurocomputing*, 426:26–34.

[Solis Castilla et al., 2020] Solis Castilla, R., Akbari, A., Jafari, R., and Mortazavi, B. J. (2020). Using intelligent personal annotations to improve human activity recognition for movements in natural environments. *IEEE Journal of Biomedical and Health Informatics*, pages 1–1.

[Sousa et al., 2017] Sousa, W., Souto, E., Rodrigres, J., Sadarc, P., Jalali, R., and El-Khatib, K. (2017). A comparative analysis of the impact of features on human activity recognition with smartphone sensors. In *Proceedings of the 23rd Brazillian Symposium on Multimedia and the Web*, pages 397–404. ACM.

[Stisen et al., 2015] Stisen, A., Blunck, H., Bhattacharya, S., Prentow, T. S., Kjærgaard, M. B., Dey, A., Sonne, T., and Jensen, M. M. (2015). Smart devices are different: Assessing and mitigatingmobile sensing heterogeneities for activity recognition. In *Proceedings of the 13th ACM conference on embedded networked sensor systems*, pages 127–140.

[Student, 1908] Student (1908). The probable error of a mean. *Biometrika*, pages 1–25.

[Subasi et al., 2018] Subasi, A., Radhwan, M., Kurdi, R., and Khateeb, K. (2018). Iot based mobile healthcare system for human activity recognition. In *2018 15th learning and technology conference (L&T)*, pages 29–34. IEEE.

[Tang et al., 2022] Tang, Y., Zhang, L., Min, F., and He, J. (2022). Multi-scale deep feature learning for human activity recognition using wearable sensors. *IEEE Transactions on Industrial Electronics*.

[Taud and Mas, 2018] Taud, H. and Mas, J. (2018). Multilayer perceptron (mlp). In *Geomatic Approaches for Modeling Land Change Scenarios*, pages 451–455. Springer.

[Teng et al., 2020] Teng, Q., Wang, K., Zhang, L., and He, J. (2020). The layer-wise training convolutional neural networks using local loss for sensor-based human activity recognition. *IEEE Sensors Journal*, 20(13):7265–7274.

[Tukey, 1949] Tukey, J. W. (1949). Comparing individual means in the analysis of variance. *Biometrics*, pages 99–114.

[Ustev et al., 2013] Ustev, Y. E., Durmaz Incel, O., and Ersoy, C. (2013). User, device and orientation independent human activity recognition on mobile phones: Challenges and a proposal. In *Proceedings of the 2013 ACM conference on Pervasive and ubiquitous computing adjunct publication*, pages 1427–1436. ACM.

[Vanneschi et al., 2010] Vanneschi, L., Castelli, M., and Silva, S. (2010). Measuring bloat, overfitting and functional complexity in genetic programming. In *Proceedings of the 12th annual conference on Genetic and evolutionary computation*, pages 877–884.

[Vaswani et al., 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

[Voicu et al., 2019] Voicu, R.-A., Dobre, C., Bajenaru, L., and Ciobanu, R.-I. (2019). Human physical activity recognition using smartphone sensors. *Sensors*, 19(3):458.

[Wan et al., 2020] Wan, S., Qi, L., Xu, X., Tong, C., and Gu, Z. (2020). Deep learning models for real-time human activity recognition with smartphones. *Mobile Networks and Applications*, 25(2):743–755.

[Wang et al., 2019] Wang, Y., Cang, S., and Yu, H. (2019). A survey on wearable sensor modality centred human activity recognition in health care. *Expert Systems with Applications*, 137:167–190.

[Wolpert and Macready, 1997] Wolpert, D. H. and Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE transactions on evolutionary computation*, 1(1):67–82.

[Wu et al., 2015] Wu, Z., Zhang, A., and Zhang, C. (2015). Human activity recognition using wearable devices sensor data.

[Xia et al., 2020] Xia, K., Huang, J., and Wang, H. (2020). Lstm-cnn architecture for human activity recognition. *IEEE Access*, 8:56855–56866.

[Xu et al., 2019] Xu, C., Chai, D., He, J., Zhang, X., and Duan, S. (2019). Innohar: a deep neural network for complex human activity recognition. *Ieee Access*, 7:9893–9902.

[Yang et al., 2015] Yang, J., Nguyen, M. N., San, P. P., Li, X. L., and Krishnaswamy, S. (2015). Deep convolutional neural networks on multichannel time series for human activity recognition. In *Twenty-fourth international joint conference on artificial intelligence*.

[Zainudin et al., 2017] Zainudin, M. S., Sulaiman, M. N., Mustapha, N., and Perumal, T. (2017). Monitoring daily fitness activity using accelerometer sensor fusion. In *2017 IEEE International Symposium on Consumer Electronics (ISCE)*, pages 35–36. IEEE.

Appendix A

Articles

This appendix presents the contents of the articles resulting from all the research carried out, which have been peer-reviewed and published in high-impact journals. Each of them corresponds directly to one of the sections offered in Chapter 4:

- Summarised in Section 4.1 and presented in Appendix A.1:

  Garcia-Gonzalez, D., Rivero, D., Fernandez-Blanco, E., and Luaces, M. R. (2020). **A public domain dataset for real-life human activity recognition using smartphone sensors.** *Sensors*, 20(8):2200. DOI: 10.3390/s20082200.

- Summarised in Section 4.2 and presented in Appendix A.2:

  Garcia-Gonzalez, D., Rivero, D., Fernandez-Blanco, E., and Luaces, M. R. (2023). **New machine learning approaches for real-life human activity recognition using smartphone sensor-based data.** *Knowledge-Based Systems*, 262:110260. DOI: 10.1016/j.knosys.2023.110260.

- Summarised in Section 4.3 and presented in Appendix A.3:

  Garcia-Gonzalez, D., Rivero, D., Fernandez-Blanco, E., and Luaces, M. R. (2023). **Deep learning models for real-life human activity recognition from smartphone sensor data.** *Internet of Things*, page 100925. DOI: 10.1016/j.iot.2023.100925.

sensors

MDPI

Article

A Public Domain Dataset for Real-Life Human Activity Recognition Using Smartphone Sensors

Daniel Garcia-Gonzalez *, Daniel Rivero, Enrique Fernandez-Blanco and Miguel R. Luaces

Department of Computer Science and Information Technologies, University of A Coruna, 15071 A Coruna, Spain; daniel.rivero@udc.es (D.R.); enrique.fernandez@udc.es (E.F.-B.); miguel.luaces@udc.es (M.R.L.)
* Correspondence: d.garcia2@udc.es

Received: 13 March 2020; Accepted: 7 April 2020; Published: 13 April 2020

Abstract: In recent years, human activity recognition has become a hot topic inside the scientific community. The reason to be under the spotlight is its direct application in multiple domains, like healthcare or fitness. Additionally, the current worldwide use of smartphones makes it particularly easy to get this kind of data from people in a non-intrusive and cheaper way, without the need for other wearables. In this paper, we introduce our orientation-independent, placement-independent and subject-independent human activity recognition dataset. The information in this dataset is the measurements from the accelerometer, gyroscope, magnetometer, and GPS of the smartphone. Additionally, each measure is associated with one of the four possible registered activities: inactive, active, walking and driving. This work also proposes asupport vector machine (SVM) model to perform some preliminary experiments on the dataset. Considering that this dataset was taken from smartphones in their actual use, unlike other datasets, the development of a good model on such data is an open problem and a challenge for researchers. By doing so, we would be able to close the gap between the model and a real-life application.

Keywords: HAR; human activity recognition; sensors; smartphones; dataset; SVM

1. Introduction

Giving birth to the knowledge area called human activity recognition (HAR), the accurate identification of different human activities has become a hot research topic. This area tries to identify the action performed by a subject based on the data records from a set of sensors. The recording of these sensors is carried out while the subject performs a series of well-defined movements, such as nodding, raising the hand, walking, running or driving. In this sense, wearable devices, such as activity bracelets or smartphones, have become of great use as sources of this sort of data. This kind of devices, especially the latter ones, provide a broad set of sensors in a convenient size which can be used relatively easy with high-grade performance and accuracy. The researchers use the information about people's behaviors gathered by these sensors to support the demands from domains like healthcare, fitness or home automation [1]. The result from the intersection between the widespread sensing all over the world, due to the smartphones and the models developed from that continuous recording, is a research area that has attracted increasing attention in recent years [2].

The main challenges to be tackled are two: first, managing the vast number of information that the devices can produce, as well as their temporal dependency, and, second, the lack of knowledge about how to relate this data to the defined movements. Some methods have achieved remarkable results in extracting information from these sensors readings [3,4]. However, it is relevant to note that in such studies, the devices have been modified to be carried in a particular way, attached to different body parts, such as waist or wrist. Therefore, the success of those models can be biased using data collected

in such a controlled environment, with specific device orientations and a few activities. Regarding these orientations, this is far from the ideal scenario, as every person may use these devices, especially their smartphones, in many different ways. For the same individual, different clothes may vary the orientation and placement of the device. In the same way, for different individuals, their body shape, as well as their behavior, can make an enormous difference too. In this way, the artificial intelligence (AI) models proposed to date are highly dependent on orientation and placement. For that reason, they cannot be generalized to every kind of user, so there has not been a real transition to real-life, yet. Presently, personalization of AI models in HAR for large numbers of people is still an active research topic [5,6], despite being actively researched for nearly a decade [7,8].

To address the aforementioned issues, this work presents a more realistic dataset which is independent of the device orientation and placement, while it also keeps the independence of the user. Those are the main differences according to data with other works developed so far. Additionally, with the implementation of a simple support vector machine (SVM) model, we present a first model as proof of concept to detect the main activities in the more realistic dataset. In this way, we are laying the foundations for the transition of this type of system into real life.

Therefore, the main contributions of this paper can be summed up as follows:

- Provide and make publicly available a new HAR dataset closer to a realistic scenario (see the files in Supplementary Materials). This dataset is independent of the device orientation and placement, while it is also individual independent.
- The new dataset adds additional signals not very explored until today like the GPS and magnetometer sensor measurements.
- A first reference model is provided for this dataset, after applying a specific sliding window length and overlap.
- A study of the best architecture for longer-themed activities, such as those suggested in our work.

The organization of the rest of the paper is as follows. Section 2 shows some related works on HAR, as well as other datasets used in this field. Section 3 gives a thorough explanation of the dataset arrangement, as well as the data collection process. Section 4 presents and discuss the experimental results obtained on the SVM model we propose, using our custom dataset; while finally, Section 5 contains the conclusions and future work lines.

2. Related Work

Inside HAR knowledge area, other datasets have been previously published. The first one worth to mention, because its widespread use in different works and comparisons, is UCI (University of California, Irvine) HAR dataset. Proposed in [9], the dataset contains data gathered while carrying a waist-mounted smartphone with embedded inertial sensors. The time signals, in this case, were sampled in sliding windows of 2.56 s and 50% overlap between them, as the activities researched are done in short intervals of time: standing, sitting, laying down, walking, walking downstairs and walking upstairs. In this work, they also created an SVM model to be exploited. With a total of 561 features extracted, they got particularly good results, with accuracies, precisions and recalls higher than 90%. However, it is a dataset taken in a laboratory environment, with a particular position and orientation. For that reason, in a realistic environment in which users could use their smartphones in their way, the results obtained would not be trustable.

Apart from the UCI HAR dataset, there is the WISDM (Wireless Sensor Data Mining) one [10], which is also widely used. In this case, the sliding windows chosen were of 10 s, with apparently no overlap applied. They mention that they also worked with 20 s, but the results were much better with the first case. Here, the activities researched were: walking, jogging, ascending stairs, descending stairs, sitting and standing. In their work, they used some WEKA (Waikato Environment for Knowledge Analysis) algorithms like J48 or Logistic Regression to perform some predictions over their data, with quite good outcomes. Nonetheless, it has the same problem as the previous case, so its results could not be taken to a real-life environment either.

Sensors **2020**, *20*, 2200

To highlight these differences, we show in Table 1 a qualitative comparison between these two datasets and the one we propose in this paper.

Table 1. Comparison between datasets: UCI HAR, WISDM and the proposed one.

	UCI HAR	**WISDM**	**Proposed**
Type of actions studied	Short-themed	Short-themed	**Long-themed**
Smartphone orientation and positioning	Fixed	Fixed	**Free**
Different individuals	Yes	Yes	**Yes**
Fixed sensor frequency	Yes	Yes	**No**
Sensors used	Acc. and gyro.	Acc. and gyro.	**Acc., gyro., magn. and GPS**

In the literature, many works tested and validated these datasets. For example, in [11], they made a comparison between Convolutional Neural Networks (CNN), Random Forest, Principal Component Analysis (PCA) and K-Nearest Neighbors (KNN) based algorithms. They concluded that CNN outperforms the rest of the ones they tested, apart from seeing that larger sliding windows did not necessarily improve their behavior. Also, they proposed some CNN architectures, making a comparison between different combinations of hyperparameters and the performance they achieved. Similarly, more recently, [12] also proposed a CNN model to address the HAR problematic, with apparently slightly better results. On the other hand, [13] submitted a combination between feature selection techniques and a deep learning method, concretely a Deep Belief Network (DBN), with some good results, higher than the ones achieved with SVM-based models, which showed to be one of the best algorithms to use in HAR problematics. By contrast, in [14,15] they made comparisons between different feature selections for different widely used machine learning (ML) algorithms in the literature. Results showed that frequency-based features are more feasible, at least for algorithms like SVM or CNN, as they throw the best results.

Furthermore, many other works built their dataset to carry out their research. One of the most interesting ones is [16]. In their work, they propose an online SVM model approach for nine different smartphone orientations. Regarding the data collection, they took it while carrying the mobile in a backpack. On the opposite hand, they also made a comparison between their custom approach and some other generic classifiers, such as KNN, decision trees, and Naive Bayes. These methods, alongside some other techniques like SVM, CNN, Random Forest, and Gradient Boosting, showed to be valid for HAR with a reasonable size of data. In the end, their approach outperformed the rest of the classifiers, but they addressed that the future of HAR would be in deep learning methods, as they seem to get better results in practice. More recent works, like [17,18] show similar results. In these cases, more sensors apart from accelerometer and gyroscope were used, like GPS or magnetometer, showing their potentiality in more long-themed activities like walking or jogging.

Following the same line, other works made their datasets but applying purely Deep Learning methods. In [19], the results show that these methods might be the future for HAR, as their results are very hopeful, at least in the non-stationary activities such as walking or running, as SVM still reigns in short-timed activities such as standing or laying down. More recently, works implementing LSTM (long short-term memory) models are arising. The principal advantage of these implementations is that they take into account past information and, at being a deep learning-based technique, they do not need a prior feature extraction to perform the training. The downside is that they need big datasets to get reliable classification results, as well as more time to be trained and suitable stop criteria to avoid overfitting (and underfitting). For example, in [20,21] we can see this kind of models and with particularly good results. In fact, in [20] they implemented a modification of LSTMs which are called Bi-LSTMs (bidirectional LSTMs). What makes this modification special is that these models can also learn from the future, throwing accuracies of around 95%.

However, as we already addressed in the introduction, all these works depend on a particular device orientation to get these successful results. In [22], the problem of different device orientations,

as well as different smartphone models, was addressed. In this case, they got good results by transforming the phone's coordinate system to the earth coordinate system. Moreover, their results did not show remarkable decreases in accuracy when carrying different smartphone models, but only when the orientation changed. Even so, it does not address the problem that arises when the smartphone is put in different places and not only in the pocket (for example, a bag).

As can be seen, there are problems of lack of realism and applicability in real life of the systems and datasets developed so far in HAR. While the results of many of the models developed in this field are quite promising, their real-life application would probably not be as successful. Therefore, in our work, we are determined to know these problems with the formation of our own more realistic dataset. With a simple SVM model, we could see the performance differences concerning other works and overcome them in future developments, if there are many.

3. Materials and Methods

This part contains a step-by-step description of our work, divided into the following sections. First, Section 3.1 presents the procedure carried out to collect the data. Then, in Section 3.2, we describe how the data was prepared to use once the data collection was over, as well as the features extracted from them. Finally, Section 3.3 offers a summary of the classification algorithm applied.

The dataset and all the resources used in this paper are publicly available (see the files in Supplementary Materials).

3.1. Data Collection

Data collection was made through an Android app developed by the authors that allowed an easy recording, labeling and storage of the data. To do this, we organized an initial data collection that lasted about a month, to see what data we were getting and to be able to do some initial tests on it. Later, we carried out another more intensive collection, over a period of about a week, to alleviate the imbalances and weaknesses found in the previous gathering. Each of the people who took part in the study was asked to set the activity they were going to perform at each moment, through that Android app, before starting the data collection. In this way, once the activity was selected, the gathering of such data was automatically started, until the user indicated the end of such activity. Hence, each stored session corresponds to a specific activity, carried out by a particular individual. Regarding the activities performed, they were four:

- Inactive: not carrying the mobile phone. For example, the device is on the desk while the individual performs another kind of activities.
- Active: carrying the mobile phone, moving, but not going to a particular place. In other words, this means that, for example, making dinner, being in a concert, buying groceries or doing the dishes count as "active" activities.
- Walking: Moving to a specific place. In this case, running or jogging count as a "walking" activity.
- Driving: Moving in a means of transport powered by an engine. This would include cars, buses, motorbikes, trucks and any similar.

The data collected comes from four different sensors: accelerometer, gyroscope, magnetometer and GPS. We selected accelerometer and gyroscope because they are the most used in the literature and the ones that showed the best results. We also added the magnetometer and GPS because we think they could be useful in this problem. In fact, in our case, GPS should be essential to differentiate the activities performed by being able to detect the user's movement speed who carries the smartphone.

We save the data of the accelerometer, the gyroscope and the magnetometer with their tri-axial values. In the case of GPS, we store the device's increments in latitude, longitude and altitude, as well as the bearing, speed and accuracy of the collected measurements. Also, for the accelerometer, we used the gravity sensor, subtracting the last reading of the latter from the observations of the first. In this way, we get clear accelerometer values (linear accelerometer), as they are not affected by the

Sensors **2020**, *20*, 2200

smartphone's orientation. Therefore, we obtain a dataset independent of the place where the individual is, as well as of the device's bearings.

On the other hand, before saving the data locally, a series of filters are applied. In the case of the accelerometer and magnetometer, we use a low-pass filter to avoid too much noise in these sensor's measurements. Concerning the gyroscope, to bypass the well-known gyro drift, a high-pass filter was used instead. Nevertheless, we also had to deal with Android's sensor frequency problem, as we cannot set the same frequency for each one of them. In our case, this is especially problematic, having to join data from very high-frequency sensors such as the accelerometer, with a low-frequency sensor, such as the GPS. From the latter, we obtain new measurements every ten seconds, approximately, compared to the possible ten, or even 50, measurements per second we can get from the accelerometer. Anyhow, given the inability to set a frequency in Android and having to take the values as they are offered by the system itself, there may be gaps in the measurements. These gaps are especially problematic in the case of GPS, where there may be cases where no new measurements were obtained in more than a minute (although perhaps this is mainly due to the difficulty of accessing closed environments). Such gaps also occur in the case of the accelerometer, gyroscope or magnetometer, despite offering about 10, 5 or 8 measurements per second, respectively, in the most stable cases. In these cases, the gaps are between 1 and 5 s, and occur mostly at the start of each data collection session, although much less frequently than with GPS. In this way, in Table 2, we show the average number of recordings per second for each sensor and each activity measured, as well as the resulting average frequency. Below each average value, in a smaller size, we also show the standard deviation for each class. Please note that for moving activities such as "active" or "walking" there is an increase in these measurements, especially with the accelerometer. This is because the smartphone detects these movements and, to get the most information, its frequency is increased automatically to get the maximum number of measurements. However, this increase also occurs during "driving" activity, even more so. Vibrations due to the car use may be the cause of this increase, as they might also be detected by the sensors of the smartphone. Additionally, in "walking" and "active" activities there may be certain inactive intervals (like waiting for a traffic light to go green or just standing doing something, respectively) that lower these averages.

Table 2. Average number of recordings per second for each sensor and each activity measured.

Activity	Accelerometer Hz.	Gyroscope Hz.	Magnetometer Hz.	GPS Hz.
Inactive	11.00 ±16.38	4.66 ±0.74	7.91 ±11.72	0.13 ±0.35
Active	32.55 ±24.80	4.46 ±1.44	9.13 ±13.64	0.06 ±0.23
Walking	31.24 ±27.47	6.24 ±11.86	8.16 ±12.05	0.06 ±0.23
Driving	51.16 ±31.59	4.66 ±2.42	17.00 ±20.01	0.04 ±0.20

In this way, the final distribution of the activities in our dataset is the one shown in Table 3. In this table, we measured the total time recorded, the number of recordings, the number of samples and the percentage of data (this one related to the number of samples), for each of the activities we specified. Here, each recording refers to a whole activity session, since the individuals begin an action until they stop it; while each sample is related to a single sensor measurement. As can be seen, there are less samples on "inactive" activities in proportion to the total time recorded. This is because the frequency of the sensors increases with activities that require more movement, as explained above, so in these cases they remained at a lower value. Therefore, the total percentage of the data may give a wrong view of the total data distribution, once the sliding windows are applied. This is because, by using these windows on which to compute a series of features, the number of samples actually moves into second place, with the total time recorded being the most important value. The more total time recorded,

the more sliding windows computed, and the more patterns for that class. Hence, there would be a much clearer imbalance in the dataset, where "inactive" activity would have three times as many patterns as in the case of "walking". Regarding the number of recordings made, there are far more with the "walking" activity than with the rest. Anyhow, we consider that the dataset remains useful and feasible to implement models that could distinguish these activities. Moreover, the total number of individuals who participated in the study was 19. Therefore, the dataset also contains different kinds of behaviors that end up enriching the possible models developed later.

Table 3. Dataset distribution for each activity measured.

Activity	Time Recorded (s)	Number of Recordings	Number of Samples	Percentage of Data
Inactive	292,213	147	7,064,757	24.25%
Active	178,806	99	8,918,021	30.62%
Walking	98,071	200	4,541,130	15.59%
Driving	112,226	128	8,602,902	29.54%
Overall	681,316	574	29,126,810	100%

On the other hand, there is also another problem in Android, as not all devices contain a gyroscope or a magnetometer to this day. While it is mandatory to have an accelerometer and a GPS, a gyroscope or a magnetometer are not compulsory in older versions of Android. In this way, some of our users took measurements without including these sensors. In Tables 4 and 5, we show the number of samples that do not include a gyroscope or a gyroscope and a magnetometer simultaneously, as the people who did not have a magnetometer did not have a gyroscope either. Something important to highlight in these tables is the difference in the relation between the number of samples and the time recorded compared to the one showed in Table 3. Here, the number of samples is much higher in relation to the time recorded. This may explain the strange data that we pointed out before in Table 2, as the accelerometer may increase more its frequency in general, by becoming the only sensor to detect motion. On another note, the percentages we show in this table are related to the whole amount of data, from Table 3. Fortunately, these percentages are quite low, and the dataset is not as affected by this problem. Anyhow, it will be something to keep in mind when preparing the data to be applied to a future AI model.

Table 4. Dataset distribution for each activity measured without gyroscope.

Activity	Time Recorded (s)	Number of Recordings	Number of Samples	Percentage of Data
Inactive	11,523	8	668,536	2.29%
Active	13,866	7	619,913	2.13%
Walking	4169	15	584,262	2.01%
Driving	25,718	23	3,776,468	12.97%
Overall	55,276	53	5,649,179	19.40%

Table 5. Dataset distribution for each activity measured without gyroscope and magnetometer.

Activity	Time Recorded (s)	Number of Recordings	Number of Samples	Percentage of Data
Inactive	5409	2	269,710	0.93%
Active	10,286	2	90,487	0.31%
Walking	0	0	0	0%
Driving	0	0	0	0%
Overall	25,695	4	360,197	1.24%

3.2. Data Preparation and Feature Extraction

After having collected the data, we proceed to prepare them to be introduced later in the model. To do so, and taking into account the well-known time-series segmentation problem in HAR, we opted

to use sliding windows of 20 s, with an overlap of 19 s (95%). We chose 20 s because it is the most we have seen used in this field. Moreover, we consider that our activities, being long-themed, need a large window size to be correctly detected. We thought even a greater size could be beneficial, but we decided to be conservative and see what happens with a smaller one. As for the overlap, we chose the maximum possible that would allow us to have comfortable handling of the data, as well as a higher number of patterns, with one second between windows. In this way, we get around half a million patterns, on a quite long time window, compared to previous works. Additionally, with this distribution, we hope to get reliable results for the movements we are analyzing, as they are long-themed (inactive, active, walking and driving).

However, to apply these windows, it is first necessary to pre-process the data. The algorithm implemented to do so consists of deleting rows that met one or more of the following properties:

1. GPS increments in latitude, longitude and altitude that are higher than a given threshold, obtained from a prior, and very conservative, data study. We detected that there were occasional "jumps" in our GPS-related values, as some of these observations were outside the expected trajectory. For this reason, we decided to fix a threshold of 0.2 for latitude and longitude increments, and 500 for the altitude ones. In this way, any value that is too far out of line is eliminated, keeping those that are closer to the expected.
2. Timestamps that do not match the structure defined (*yyyy-MM-dd HH:mm:ss.ZZZ*) or that do not correspond to an actual date (year 1970 values, for example).
3. Any misplaced value between timestamp and z-axis magnetometer, which showed to appear in some very few observations at the beginning of the project.

Table 6 shows the mean and standard deviation values of each sensor for each of the activities studied, after the application of this algorithm. To correctly understand the values indicated in this table, it is important to explain what each of these sensors measures. The accelerometer values correspond to the acceleration force applied to the smartphone on the three physical axes (x, y, z), in m/s^2. On the other hand, the gyroscope measures in rad/s the smartphone's rotation speed around each of the three physical axes (x, y, z). Regarding the magnetometer, it measures the environmental geomagnetic field of the three physical axes (x, y, z) of the smartphone, in μT. As for the GPS, its values correspond, on the one hand, to the increments of the values of the geographical coordinates, longitude and latitude, in which the smartphone is located, with respect to the previous measurement. Similarly, the increments in altitude, in meters, were also measured. Then, the values of speed, bearing and accuracy were also taken into account. Speed was measured in m/s and specifies the speed that is taking the smartphone. The bearing measured the horizontal direction of travel of the smartphone, in degrees. Finally, accuracy values refer to the deviation from the actual smartphone location, in meters, where the smaller the value, the better the accuracy of the measurement. Going back to Table 6, in each cell, the values corresponding to the mean are at the top and, at the bottom, in a smaller size, the standard deviation values. Each pair of values corresponds to the set that forms each sensor. In the case of the accelerometer, gyroscope and magnetometer, these refer to the values related to their "X", "Y" and "Z" axes. As for the GPS, this set is formed by the latitude increments (Lat.), the longitude increments (Long.), the altitude increments (Alt.), the speed (Sp.), the bearing (Bear.) and the accuracy (Acc.) of every measurement. Here, it is worth noting some rare data, such as those relating to GPS "inactive" activity, where the values are very high concerning what is expected from such action. In this case, we consider that these values are due to the fact that such activity is carried out in indoor environments, which are not so accessible for GPS. Even so, as can be seen, there are some clear differences between the activities, so the possibilities of identification with future models are more than feasible.

Table 6. Sensor's mean and standard deviation values for each activity measured.

		Activity			
		Inactive	**Active**	**Walking**	**Driving**
Accelerometer	*X*	0.11761 ±0.45934	−0.01338 ±1.30277	0.09425 ±3.33422	−0.04747 ±0.83290
	Y	0.06136 ±0.26764	0.07598 ±1.45440	−0.37604 ±4.35808	−0.12936 ±0.93828
	Z	0.84318 ±2.66926	0.13008 ±1.70294	0.07353 ±4.09859	0.18127 ±1.24042
Gyroscope	*X*	−0.00004 ±0.03828	−0.00001 ±0.36806	0.00760 ±1.31125	0.00080 ±0.19224
	Y	0.00004 ±0.04719	−0.00102 ±0.40959	−0.00020 ±0.89244	0.00277 ±0.19835
	Z	0.00001 ±0.03526	0.00055 ±0.24528	−0.00560 ±0.53685	−0.00243 ±0.16678
Magnetometer	*X*	25.93805 ±56.45617	6.03153 ±30.00980	−0.28182 ±27.03210	−5.96356 ±46.08005
	Y	−19.62683 ±85.70343	−0.02890 ±28.76398	18.73800 ±29.63926	10.73609 ±40.46829
	Z	−56.60425 ±33.19593	9.56310 ±39.76136	0.64541 ±25.55331	−2.93043 ±29.45994
GPS	**Lat.**	0.00075 ±0.00166	0.00112 ±0.00234	0.00047 ±0.00220	0.00175 ±0.00365
	Long.	0.00125 ±0.00285	0.00118 ±0.00314	0.00056 ±0.00300	0.00204 ±0.00420
	Alt.	32.59169 ±53.06269	30.77538 ±48.65634	34.06931 ±42.51933	41.59391 ±54.74934
	Sp.	0.37222 ±0.82495	0.12109 ±0.81007	0.79924 ±0.71835	10.82191 ±11.82733
	Bear.	57.25005 ±105.49576	14.69719 ±56.00693	124.85103 ±119.80663	118.88108 ±118.78510
	Acc.	265.44485 ±494.66499	214.57640 ±429.81169	75.54539 ±259.59907	192.90736 ±508.87285

After applying previous preprocessing, since data collection required the user to tap a button before performing the activity, we eliminated the first five seconds of each activity collection. In the same way, we did so with the final five seconds of each measurement. Hence, we can prevent the future models from ending up learning the movement that precedes the start or the end of the action, such as, for example, putting the smartphone in the pocket or pulling it out. While doing this, we also get rid of those sessions that have quite large gaps between the data (at least five seconds) for any sensor other than the GPS, by considering them as invalid. In this way, in Table 7, the final results after the application of this sliding window and overlap are shown for the samples containing all the sensors. As we already addressed in the previous section, although at the sample level the data may appear lower for activities such as inactive or walking, at the final pattern level the results are much different.

Table 7. Number of patterns for the samples containing all the sensors with a sliding window of 20 s and 19 s overlap.

Activity				
Inactive	**Active**	**Walking**	**Driving**	**Overall**
214,130 (43%)	140,060 (28%)	83,376 (17%)	61,710 (12%)	499,276

Sensors **2020**, *20*, 2200

Later, we had to go through a transformation process to extract the features and apply all the information needed for the classification algorithm. Due to GPS' low frequency, to carry out this feature extraction, it was necessary to previously replicate some of the data stored by this sensor, for each of the windows applied. To do this, if the difference between one observation and the next differed in a longer time than one second, the latter measurement is replicated, with a different timestamp. For this reason, all sessions that do not contain at least one GPS observation are removed from the list of valid ones for this process. We repeat this step until all the windows that may be in the middle are correctly filled. We selected one second as the amount of time to be between each sample, so there is always at least one observation in each of the windows applied, making the final feature extraction match to the data obtained. After that, for each set of measurements, we computed six different types of features, each generating a series of inputs for the AI model. The features used were: mean, variance, median absolute deviation, maximum, minimum and interquartile range, all based in the time domain. All of them were used in previous works like [16], with remarkable results. In this way, we maintain the simplicity of the model, being able to complicate it or change it in future works according to the results we achieve.

3.3. Classification Algorithm

As already indicated in the related work section, there are many kinds of models used in HAR. In our case, we chose to employ an SVM model. Although SVM showed excellent results with rather short-themed activities, we consider it interesting to test it as an initial model in our dataset. It is one of the most used models in HAR, applied in works such as [9,16] and, more recently, in [23], all with outstanding overall performance in this field, as well as being a simple and straightforward AI model.

An SVM is a supervised machine learning model that uses classification algorithms for two-group classification problems. After giving an SVM model tagged training data sets for either category, they can categorize new examples. To do this, the SVM looks for the hyperplane that maximizes the margins between the two classes. In other words, it looks for the hyperplane whose distance from the nearest element in each category is the highest. Hither, non-linearity is achieved through kernel functions, which implicitly map the data to a more dimensional space where this linear approximation is applied. On the other hand, other hyperparameters such as C or gamma also affect the definition of this hyperplane. As for C, it marks the width of the margins of this hyperplane, as well as the number of errors that are accepted. Concerning gamma, it directly affects the curve of the hyperplane, making it softer or more accentuated, depending on the patterns that are introduced into the model.

While SVM is typically used to solve binary classification tasks, it can also be used in multi-class problems. To do this, it is necessary to use a *one-vs-all* or *one-vs-one* strategy. The first case is designed to model each class against all other classes independently. In this way, a classifier is created for each situation. On the other hand, the second case is used to model each pair of classes separately, performing various binary classifications, until a final result is found. In our case, we will be using a one-vs-all approach, as it is the most used one in the literature. For this, we implemented it on Python, using the functions provided with Scikit-learn.

4. Results and Discussion

4.1. Results

To provide reliable results in this dataset to future users, we conducted a series of experiments on it. For this purpose, we applied SVM classifiers, looking for the best kernel between Polynomial, RBF (Radial Basis Function) and Linear SVM. Also, we explored the optimal trade-off parameter C, the bandwidth γ in RBF and Polynomial kernels, as well as the degree in this last one, with the features discussed in the previous section. The reason we selected these kernels was, on the one hand, because the RBF kernel is one of the most used ones in the literature. On the other hand, the linear and

the polynomial ones were also selected to have a basis for comparison. To select the best configuration and architecture of the network, we obeyed the following organization:

1. First, with the whole combination of all sensors, we made a stratified 10-fold with which to have 10 sets with presumably the same number of patterns for each class.
2. Then, we took each of those folds to use them to perform a grid search on their corresponding dataset. To evaluate the resulting predictions, since we use a one-vs-all approach that will have unbalanced data in each sub-classifier, we chose the f1-score metric to minimize this problematic. The f1-score is a measure of the test accuracy, based on the harmonic mean of the precision and the recall metrics. Its formula would be as follows:

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

   With that in mind, it is closely linked to the correct classification of each pattern, not being so influenced by class imbalances. When this happens, accuracy might give an incorrect idea of the model's performance. However, the f1-score will give a slightly smoother value that better represents that model, making it a good option for our grid search. On the other hand, we also set a maximum number of iterations (1000) as a stop criterion, given the high-dimensional data and the scaling problem of SVM. To carry out this process, we selected the following hyperparameters: as kernels, we chose the polynomial, the RBF and the linear ones, because of what we addressed before. As for parameter C, we selected those of the set {1, 10, 100, 1000, 10000}. For the γ parameter, specifically for the RBF and polynomial kernels, we chose those of the set {0.0001, 0.001, 0.01, 0.1, 1}. Concerning the degree parameter for the polynomial kernel, we selected those of the set {1, 2, 3, 4}.
3. Once the grid search is done, we evaluated the results and selected the best combination of hyperparameters for each fold. Then, we tested the best corresponding model.
4. Finally, we studied the impact of the gyroscope and magnetometer, taking advantage of the users that could not include these sensors in their measurements. For this purpose, we prepared three different sets: accelerometer + gyroscope + magnetometer + GPS (all users but the ones missing gyroscope and magnetometer), accelerometer + gyroscope + GPS (all users but the ones missing magnetometer) and accelerometer + GPS (all users).

The first steps of the experiments yielded the results that can be seen in Table 8. In this table, for each cell, we show the average test f1-score obtained (top), as well as its standard deviation (below). As can be seen, the best results correspond, in general, to the RBF kernel, and, more specifically, for cases where g equals 0.1, especially in conjunction with C = 100. With this combination of hyperparameters, we managed to achieve an f1-score of 74.34%.

The average confusion matrix yielded by the third step of the experiments is the one showed in Table 9, along with its particular metrics (recall, precision and accuracy). This result corresponds to an accuracy of 69.28%. As can be seen, the model manages to correctly separate "inactive" events but struggles with the rest, especially with the "active" one. In this case, we think that this is due to the diffusion of this action since it combines both moments of inactivity and movement, in which we may walk from one place to another. On the other hand, we can also see that the activities of "walking" and "driving" are also confused with each other. This was expected considering that most driving took place in an urban environment. In this scenario, there may be traffic jams or moments of less fluidity that may be quite similar, at a sensory level, to the data obtained while performing the "walking" activity, as well as the rest of actions. Anyhow, the GPS is probably very influential in this confusion and it would be interesting to change the related features used to see how they affect the final classification. Maybe greater sliding window sizes or any kind of feature related to the Fourier transform of the signal, to pick up its periodic component, could positively affect the final model.

Sensors **2020**, *20*, 2200

Table 8. Mean f1-scores achieved for each combination of kernel, C, γ and degree hyperparameters in the grid search. The best result found is highlighted in bold.

		C = 1	C = 10	C = 100	C = 1000	C = 10,000
Linear		36.33% ±17.03	38.96% ±12.11	42.58% ±12.54	42.58% ±12.54	42.58% ±12.54
RBF	γ = **0.0001**	5.88% ±4.28	11.08% ±11.08	17.81% ±7.68	39.94% ±18.53	37.38% ±20.70
	γ = **0.001**	15.78% ±14.89	28.26% ±18.39	45.12% ±15.03	41.09% ±9.19	42.75% ±14.17
	γ = **0.01**	59.16% ±7.07	63.21% ±14.51	58.66% ±11.77	65.44% ±16.52	59.24% ±13.19
	γ = **0.1**	68.30% ±10.80	73.33% ±6.62	**74.34%** ±8.26	70.94% ±7.92	69.42% ±9.51
	γ = **1**	63.73% ±10.69	56.88% ±6.21	56.96% ±6.30	56.96% ±6.30	56.96% ±6.30
Poly d=1	γ = **0.0001**	12.89% ±9.60	29.65% ±17.48	37.20% ±18.34	26.37% ±13.73	43.37% ±19.91
	γ = **0.001**	29.39% ±16.57	33.50% ±20.58	34.51% ±15.72	39.10% ±17.36	39.17% ±15.85
	γ = **0.01**	32.08% ±16.92	33.90% ±11.22	40.71% ±19.78	43.18% ±18.73	39.07% ±18.59
	γ = **0.1**	33.21% ±21.96	33.69% ±16.93	40.93% ±15.31	36.65% ±14.93	36.65% ±14.93
	γ = **1**	36.33% ±17.03	38.96% ±12.12	42.58% ±12.54	42.58% ±12.54	42.58% ±12.54
Poly d=2	γ = **0.0001**	7.92% ±4.43	6.85% ±3.62	10.22% ±8.73	5.92% ±5.05	9.69% ±7.02
	γ = **0.001**	10.22% ±8.73	5.92% ±5.05	9.70% ±7.02	12.34% ±6.12	24.01% ±7.49
	γ = **0.01**	9.69% ±7.03	12.27% ±5.78	26.54% ±7.70	22.56% ±5.74	20.64% ±6.93
	γ = **0.1**	23.63% ±7.41	24.40% ±5.83	26.24% ±7.85	26.23% ±7.85	26.23% ±7.85
	γ = **1**	27.35% ±10.83	27.33% ±10.84	27.33% ±10.84	27.33% ±10.84	27.33% ±10.84
Poly d=3	γ = **0.0001**	5.61% ±4.41	6.21% ±4.08	7.45% ±4.55	10.01% ±6.79	10.03% ±6.79
	γ = **0.001**	10.01% ±6.79	10.03% ±6.79	6.60% ±4.46	8.19% ±7.02	20.48% ±12.54
	γ = **0.01**	5.87% ±3.97	19.68% ±13.58	24.29% ±9.31	22.63% ±8.17	16.92% ±7.38
	γ = **0.1**	26.40% ±7.11	17.90% ±8.51	17.60% ±12.79	17.60% ±12.79	17.60% ±12.79
	γ = **1**	17.77% ±8.63	17.77% ±8.63	17.77% ±8.63	17.77% ±8.63	17.77% ±8.63
Poly d=4	γ = **0.0001**	5.87% ±3.31	6.42% ±3.93	9.09% ±3.98	8.91% ±8.84	13.12% ±8.93
	γ = **0.001**	13.12% ±8.93	7.92% ±4.44	6.18% ±3.27	11.03% ±10.01	11.26% ±9.90
	γ = **0.01**	9.16% ±8.76	7.87% ±6.80	6.45% ±3.21	5.52% ±1.81	7.18% ±4.55
	γ = **0.1**	8.71% ±4.79	9.55% ±6.75	9.49% ±6.89	9.49% ±6.89	9.49% ±6.89
	γ = **1**	8.97% ±5.29	8.97% ±5.29	8.97% ±5.29	8.97% ±5.29	8.97% ±5.29

Table 9. Average confusion matrix for the experiments conducted.

	Ground Truth				
	Inactive	**Active**	**Walking**	**Driving**	**Precision**
Inactive	16,787	610	486	90	93.40%
Active	3026	8676	1163	914	62.97%
Walking	1341	3772	5675	1714	45.39%
Driving	259	948	1015	3453	60.58%
Recall	78.40%	61.95%	68.05%	55.96%	69.28%

To a lesser extent, it is also important to note that there are some cases in which some activities are confused as an "inactive" action. This was also relatively expected, as every activity is subject to prolonged stoppages. For example, while acting as "walking" or "driving", traffic lights that force the individual to stop may appear. In these situations, these pauses may be mistaken by the model for cases of pure inactivity. Perhaps the use of other and more specific features could improve the differentiation in all these cases, as well as the use of another type of AI algorithms and bigger sliding window sizes.

Regarding the fourth and last step, we also applied the same algorithm for the rest of the data sets formed, obtaining the results shown in Table 10. Similar to the other tables shown, the average values are on the left side of each cell, while the standard deviations are on the right side, in a smaller size. This comparison is made from the average of the test values yielded by the experiments conducted to each set. As can be seen, the combination of the accelerometer, the magnetometer and the GPS, with the lack of the gyroscope, performs better in comparison with the other two, especially with the case formed only by accelerometer and GPS. However, the expected best result would have been the one that appends the gyroscope too, as in the other works that included it in their studies. Perhaps the fact that we are studying long-themed activities is something in which the gyroscope does not have much of a presence. In addition, the model has more patterns with the winning combination, which could also positively influence the final result.

Table 10. Mean accuracies achieved for each set of data, with the best group result highlighted in bold.

Acc. + GPS.	**Acc. + Magn. + GPS**	**Acc. + Gyro. + Magn.+ GPS**
67.53% ±6.33	**74.39%** ±10.75	69.28% ±15.10

4.2. Discussion

Although the results obtained might not seem as good as those seen so far in the rest of the literature, we consider that they are promising given the problem addressed. The data used are very different from those of the other datasets that currently exist in the field, as well as being much less specific. Therefore, while the results may seem worse, actually they are not comparable. The data collected correspond to different profiles of people, each with their physical peculiarities and ways of using their smartphone. Moreover, the nature of each of the defined activities implies short periods of some of the other actions. For example, within the "active" exercise, there are both moments of inactivity and moments of travel. Within the "walking" activity, there may be stops due to traffic lights or other obstacles encountered along the way. Furthermore, during the action of "driving", it is noteworthy that an urban environment has many peculiarities and stops that can complicate the final classification. Therefore, given these problems and the simplicity of the proposed model, we consider that these results are a relatively good first approximation of what they could be. We believe that perhaps with other types of models also used in this field, such as Random Forest, the results could be improved considerably. Also, through the application of algorithms based on deep learning, such as LSTM, that showed exceptional performance in this domain too. Hence, with this change in the model to be used and the addition of new metrics, we would surely get closer to that real-life environment we are searching.

5. Conclusions and Future Work

In this paper, we presented a dataset for the HAR field. This dataset contains information from 19 different users, each with its own way of using their smartphone, as well as their physical peculiarities. The amount of data is enough to make classifications about them, and the information gathered is realistic enough to be taken to a real-life environment.

Therefore, with the development of this dataset, we hope to alleviate the problems that are seen in other works. While it is true that the final results we got may not be as good as those seen to date, we believe that it will be the beginning of the road to take the models developed for HAR to real life. We also hope that the current confusions of the proposed model, among some of the determined activities, can be overcome in future research. In this way, it would be possible to implement a system capable of correctly detecting a person's movements or activities, regardless of the way they use their smartphone or their physical peculiarities. This could be very interesting for many companies or individuals to be able to monitor or predict the activities performed by a particular individual.

For this reason, we will continue advancing in the same line of work, testing other techniques that also had pretty good results in the field, such as Random Forest, CNN or LSTM. Also, the deletion or the addition of new features, such as those related to the Fourier transform, to search for possible periodic components in the stored signals, could positively affect the final model. In this way, we will be able to compare the results obtained, in search of the best model to solve this problem. In addition, we will also explore the real impact of the sensors used, as well as other possible sliding windows greater sizes and combinations of hyperparameters, in search of improving the best configuration found so far.

Supplementary Materials: The complete dataset, as well as the scripts used on our experiments, are available online at http://lbd.udc.es/research/real-life-HAR-dataset. Similarly, they have also been uploaded to Mendeley Data [24].

Author Contributions: Conceptualization, D.G.-G., D.R. and E.F.-B.; data curation, D.G.-G.; formal analysis, D.G.-G., D.R. and E.F.-B.; funding acquisition, M.R.L.; investigation, D.G.-G.; methodology, D.R., E.F.-B. and M.R.L.; project administration, M.R.L.; resources, M.R.L.; software, D.G.-G.; supervision, D.R., E.F.-B. and M.R.L.; validation, D.G.-G.; visualization, D.G.-G.; writing—original draft preparation, D.G.-G.; writing—review and editing, D.G.-G., D.R. and E.F.-B. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partially funded by Xunta de Galicia/FEDER-UE (ConectaPeme, GEMA: IN852A 2018/14), MINECO-AEI/FEDER-UE (Flatcity: TIN2016-77158-C4-3-R) and Xunta de Galicia/FEDER-UE (AXUDAS PARA A CONSOLIDACION E ESTRUTURACION DE UNIDADES DE INVESTIGACION COMPETITIVAS.GRC: ED431C 2017/58 and ED431C 2018/49).

Acknowledgments: First of all, we want to thank the support from the CESGA to execute the code related to this paper. Also, we would like to thank all the participants who took part in our data collection experiment.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhu, N.; Diethe, T.; Camplani, M.; Tao, L.; Burrows, A.; Twomey, N.; Kaleshi, D.; Mirmehdi, M.; Flach, P.; Craddock, I. Bridging e-health and the internet of things: The sphere project. *IEEE Intell. Syst.* **2015**, *30*, 39–46. [CrossRef]
2. Lara, O.D.; Labrador, M.A. A survey on human activity recognition using wearable sensors. *IEEE Commun. Surv. Tutor.* **2012**, *15*, 1192–1209. [CrossRef]
3. Attal, F.; Mohammed, S.; Dedabrishvili, M.; Chamroukhi, F.; Oukhellou, L.; Amirat, Y. Physical human activity recognition using wearable sensors. *Sensors* **2015**, *15*, 31314–31338. [CrossRef]
4. Shoaib, M.; Bosch, S.; Incel, O.; Scholten, H.; Havinga, P. Complex human activity recognition using smartphone and wrist-worn motion sensors. *Sensors* **2016**, *16*, 426. [CrossRef]
5. Ferrari, A.; Micucci, D.; Mobilio, M.; Napoletano, P. On the Personalization of Classification Models for Human Activity Recognition. *IEEE Access* **2020**, *8*, 32066–32079. [CrossRef]
6. Solis Castilla, R.; Akbari, A.; Jafari, R.; Mortazavi, B.J. Using Intelligent Personal Annotations to Improve Human Activity Recognition for Movements in Natural Environments. *IEEE J. Biomed. Health Inform.* **2020**. [CrossRef] [PubMed]

7. Weiss, G.; Lockhart, J. The Impact of Personalization on Smartphone-Based Activity Recognition. In Proceedings of the AAAI Publications, Workshops at the Twenty-Sixth AAAI Conference on Artificial Intelligence, Toronto, ON, Canada, 22–23 July 2012.
8. Lane, N.; Xu, Y.; lu, H.; Hu, S.; Choudhury, T.; Campbell, A.; Zhao, F. Enabling large-scale human activity inference on smartphones using Community Similarity Networks (CSN). In Proceedings of the 13th International Conference on Ubiquitous Computing, Beijing, China, 17–21 September 2011; pp. 355–364.
9. Anguita, D.; Ghio, A.; Oneto, L.; Parra, X.; Reyes-Ortiz, J.L. A public domain dataset for human activity recognition using smartphones. In Proceedings of the Esann, Bruges, Belgium, 24–26 April 2013.
10. Kwapisz, J.R.; Weiss, G.M.; Moore, S.A. Activity recognition using cell phone accelerometers. *ACM SigKDD Explor. Newsl.* **2011**, *12*, 74–82. [CrossRef]
11. Ignatov, A. Real-time human activity recognition from accelerometer data using Convolutional Neural Networks. *Appl. Soft Comput.* **2018**, *62*, 915–922. [CrossRef]
12. Sikder, N.; Chowdhury, M.S.; Arif, A.S.; Nahid, A.A. Human Activity Recognition Using Multichannel Convolutional Neural Network. In Proceedings of the 5th International Conference on Advances in Electronics Engineering, Dhaka, Bangladesh, 26–28 September 2019.
13. Hassan, M.M.; Uddin, M.Z.; Mohamed, A.; Almogren, A. A robust human activity recognition system using smartphone sensors and deep learning. *Future Gener. Comput. Syst.* **2018**, *81*, 307–313. [CrossRef]
14. Seto, S.; Zhang, W.; Zhou, Y. Multivariate time series classification using dynamic time warping template selection for human activity recognition. In Proceedings of the IEEE Symposium Series on Computational Intelligence, Cape Town, South Africa, 7–10 December 2015; pp. 1399–1406.
15. Sousa, W.; Souto, E.; Rodrigres, J.; Sadarc, P.; Jalali, R.; El-Khatib, K. A comparative analysis of the impact of features on human activity recognition with smartphone sensors. In Proceedings of the 23rd Brazillian Symposium on Multimedia and the Web, Gramado, Brazil, 17–20 October 2017; pp. 397–404.
16. Chen, Z.; Zhu, Q.; Soh, Y.C.; Zhang, L. Robust human activity recognition using smartphone sensors via CT-PCA and online SVM. *IEEE Trans. Ind. Inf.* **2017**, *13*, 3070–3080. [CrossRef]
17. Figueiredo, J.; Gordalina, G.; Correia, P.; Pires, G.; Oliveira, L.; Martinho, R.; Rijo, R.; Assuncao, P.; Seco, A.; Fonseca-Pinto, R. Recognition of human activity based on sparse data collected from smartphone sensors. In Proceedings of the IEEE 6th Portuguese Meeting on Bioengineering (ENBENG, Lisbon, Portugal, 22–23 February 2019; pp. 1–4.
18. Voicu, R.A.; Dobre, C.; Bajenaru, L.; Ciobanu, R.I. Human Physical Activity Recognition Using Smartphone Sensors. *Sensors* **2019**, *19*, 458. [CrossRef] [PubMed]
19. Ronao, C.A.; Cho, S.B. Human activity recognition with smartphone sensors using deep learning neural networks. *Expert Syst. Appl.* **2016**, *59*, 235–244. [CrossRef]
20. Hernández, F.; Suárez, L.F.; Villamizar, J.; Altuve, M. Human Activity Recognition on Smartphones Using a Bidirectional LSTM Network. In Proceedings of the XXII Symposium on Image, Signal Processing and Artificial Vision (STSIVA), Bucaramanga, Colombia, 24–26 April 2019; pp. 1–5.
21. Badshah, M. Sensor-Based Human Activity Recognition Using Smartphones. Master's Thesis, San Jose State University, San Jose, CA, USA, 2019.
22. Ustev, Y.E.; Durmaz Incel, O.; Ersoy, C. User, device and orientation independent human activity recognition on mobile phones: Challenges and a proposal. In Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication, Zurich, Switzerland, 8–12 September 2013; pp. 1427–1436.
23. Ahmed, N.; Rafiq, J.I.; Islam, M.R. Enhanced Human Activity Recognition Based on Smartphone Sensor Data Using Hybrid Feature Selection Model. *Sensors* **2020**, *20*, 317. [CrossRef] [PubMed]
24. Garcia-Gonzalez, D.; Rivero, D.; Fernandez-Blanco, E.; Luaces, M.R. A Public Domain Dataset for Real-Life Human Activity Recognition Using Smartphone Sensors. Available online: https://data.mendeley.com/datasets/3xm88g6m6d/2 (accessed on 18 August 2020).

Knowledge-Based Systems 262 (2023) 110260

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

New machine learning approaches for real-life human activity recognition using smartphone sensor-based data

Daniel Garcia-Gonzalez *, Daniel Rivero, Enrique Fernandez-Blanco, Miguel R. Luaces

Department of Computer Science and Information Technologies, University of A Coruna, CITIC, 15071 A Coruna, Spain

ARTICLE INFO

Article history:
Received 7 April 2021
Received in revised form 4 June 2021
Accepted 2 January 2023
Available online 5 January 2023

Keywords:
HAR
Human activity recognition
Machine learning
Real life
Smartphones
Sensors

ABSTRACT

In recent years, mainly due to the application of smartphones in this area, research in human activity recognition (HAR) has shown a continuous and steady growth. Thanks to its wide range of sensors, its size, its ease of use, its low price and its applicability in many other fields, it is a highly attractive option for researchers. However, the vast majority of studies carried out so far focus on laboratory settings, outside of a real-life environment. In this work, unlike in other papers, progress was sought on the latter point. To do so, a dataset already published for this purpose was used. This dataset was collected using the sensors of the smartphones of different individuals in their daily life, with almost total freedom. To exploit these data, numerous experiments were carried out with various machine learning techniques and each of them with different hyperparameters. These experiments proved that, in this case, tree-based models, such as Random Forest, outperform the rest. The final result shows an enormous improvement in the accuracy of the best model found to date for this purpose, from 74.39% to 92.97%.

1. Introduction

Having become a hot research topic in recent years, human activity recognition (HAR) analyses series of sensor-collected data to identify the actions taken by a person [1–3]. These sensors can be used through wearable devices such as wristbands or, more recently, smartphones. Both cases offer a broad set of sensors that can be used relatively easy, with excellent accuracy and a small size that favours its portability. In addition, this area has many application possibilities in various fields such as health, fitness or even home automation [4–8]. All this, together with the recent application of smartphones in HAR and its global use, make this field a highly attractive option for research [9,10].

There are several research challenges within this field. Firstly, there is the challenge of correctly processing the vast amounts of data that these devices collect, while controlling the temporality of such data. Also, although significant advances have been made [11,12], the relation between these data and most human movements is still not known precisely, making the task even more difficult. Besides, most of the studies carried out to date were done in a laboratory environment, with highly controlled movements and specific placements of the device that collects the data [13,14]. That is interesting in order to see what the approximate relation between the information collected and the action studied is. However, the excellent results seen in these works may not be as good when they are applied outside that highly-controlled environment. That is because, in a daily life environment, people will use and carry the data collection device differently, outside of what was previously examined. In this way, the orientation and placement of the device could vary greatly, even when performing the same action. Also, each person may have many physical peculiarities that could considerably influence the final result as well. In fact, the personalization of AI models in HAR for large numbers of people is something that it is being researched since almost a decade [15–18]. For these reasons, the transfer of this knowledge to real-life remains to be seen.

In this paper, looking to close the gap with the real-life application, a dataset gathered for this purpose was used [19]. For that goal, the dataset was collected using the sensors of several individuals' smartphones, with almost total freedom. In this way, a comparative study between the last results obtained and the current ones is presented. To do so, numerous machine learning algorithms frequently used in HAR were implemented, in search of the best combination between algorithm and hyperparameters. In the same manner, a comparison of the results obtained by using the data taken with all the sensors, as well as with the absence of the gyroscope, is also presented to observe which case behaves better. In this way, we will be able to get even closer to that real-life ideal that is currently being pursued.

* Corresponding author.
E-mail addresses: d.garcia2@udc.es (D. Garcia-Gonzalez), daniel.rivero@udc.es (D. Rivero), enrique.fernandez@udc.es (E. Fernandez-Blanco), miguel.luaces@udc.es (M.R. Luaces).

https://doi.org/10.1016/j.knosys.2023.110260

Thus, the main contributions of this paper are the following:

- A comparison of the main machine learning algorithms applied in the HAR field, using a dataset taken in a real-life environment, unlike in other studies.
- The addition of the Extreme Gradient Boosting (XGB) algorithm to the comparison, not very explored until today in this knowledge area.
- A study of the best model configurations for long-themed activities (such as driving or jogging), based on the indicated dataset, with changes and additions to the last feature set used.
- A review of the gyroscope's real influence to the final results.
- The improvement of the current approaches in this field, oriented towards real life.

The remaining sections of this paper are organized as follows: Section 2 shows some related works on HAR, Section 3 gives a thorough explanation of how the data was prepared, as well as a brief description of every algorithm and metric used, Section 4 presents and discuss the experimental results obtained on the models we propose, and, finally, Section 5 contains the conclusions and future work lines.

2. Related work

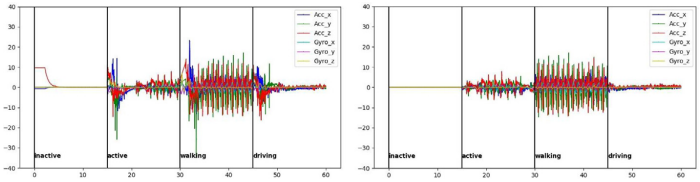
Human activity recognition (HAR) has been studied extensively in recent years and, over the last decade, the continuous flow of works has brought a steady pace of advances. Most of these works were carried out using datasets such as those provided in [20,21], which are two of the most widely used ones in HAR. Both datasets offer a large amount of information about different actions to be exploited, using smartphone sensors such as the accelerometer and the gyroscope. However, for both cases, these data were taken in a laboratory environment. That means that the smartphone was placed in a particular position and the actions performed were highly controlled. An example would be [22], where a comparison is made between different machine learning algorithms, namely, Convolutional Neural Networks (CNN), Random Forest (RF), K-Nearest Neighbours (KNN) and also a feature selection method, Principal Component Analysis (PCA). Among all of them, CNN was the best by far, for which they also contributed different architectures, with several combinations of hyperparameters and the result of each one of them. Besides, they also concluded that with rather large time windows the results did not improve. Likewise, in [23], another CNN model was also proposed for this problem, with slightly better performance. Alternatively, other works such as [12] provided techniques based on deep learning, such as the Deep Belief Network (DBN). Here, after a feature selection process, they also obtain pretty good results, even better than those of the models based in the Support Vector Machine (SVM) algorithm, which proved to be the best to use for the HAR problem. Conversely, research was also done on the selection of features for different machine learning algorithms widely used in HAR. The results of works such as [24,25] showed that the frequency-based parameters are more feasible since they were the ones that showed better results.

However, not all the work relied solely on the accelerometer and gyroscope for its research. Some studies such as [26,27] show high-grade results with the addition of other sensors such as the GPS or the magnetometer. In fact, in these works, they studied long-themed activities such as walking or jogging, which shows the potential of these sensors for this type of actions. On the other hand, in [28], an online SVM model is proposed for nine different smartphone orientations, although all of them are based on leaving the mobile phone in a backpack. They also made a comparison with other methods typically used in HAR, such as KNN, Decision Tree (DT) and Naïve Bayes (NB). All these methods, together with other techniques such as SVM, CNN, Random Forest (RF) and Gradient Boosting (GB), proved to be fully valid in HAR for a reasonable amount of data. At the same time, it also indicates that the application of deep learning techniques could be a very up-and-coming line of research for the HAR field, as some of the best results in practice seem to be obtained with this type of methods.

In the same vein, more recently, research in HAR is focusing more on the application of purely deep learning techniques. One of the first works to apply these techniques was [29], in which a comparison of different architectures for a deep CNN model with other methods widely used in the literature, such as SVM or Multilayer Perceptron (MLP), is presented. Moreover, currently, besides the deep CNN models, much research is being done with models that implement the Long Short-term Memory (LSTM) technique. The main advantage that these implementations have is that they can include information from the past in their training, as well as not needing a previous feature extraction period. However, as a disadvantage, they need a large amount of data to obtain reliable results, as well as requiring an adequate stop criterion to avoid overfitting and underfitting. Some examples of the application of this technique are the works of [30,31], in which excellent results were obtained. Specifically, in [30], a modification of this method was carried out, called Bi-LSTM (bidirectional LSTM), which also manages to learn from the future, throwing accuracies of around 95%. On the other hand, other works have been in charge of comparing in depth the two most used deep learning methods, CNN and LSTM and their variants, in search of the most suitable model for HAR [32,33]. The results show that both techniques have very similar potentials and performances, being probably two of the best options to use for short-themed activities such as sitting or standing. Nonetheless, it seems that some studies suggest that the application of CNNs over LSTMs is favoured, due to its higher speed and its straightforward application [34].

However, despite all the progress mentioned above, all the works share the same problem. That one is no other than the dependency on precise use guidelines for the device to obtain good results. While there are some works such as [35,36] that have addressed this problem, they cannot be considered feasible for real life. In these cases, they obtained good results by transforming the phone's coordinate system to the Earth's coordinate system. In addition, in the case of [35], different models of smartphones were also used, without an apparent drop in the eventual accuracy. In any case, when changing the orientation of the smartphones, the final performance does decrease. Finally, they also do not address the problem of placing the smartphone in different places, like a backpack, and not just in a trouser pocket.

Nonetheless, a very recent work does address this problem rightly [19]. There, a dataset focusing on the application of HAR techniques in real life is presented, which will be used in this paper as well. Also, they did a series of experiments with machine learning techniques, but they are very elementary and could be highly improved. Therefore, this work has as main aim to advance in the resolution of these problems of lack of realism and applicability in real life of all the models developed up to date in HAR. While it is true that the advances made so far are very promising, if those advances were taken into a real-life environment, more than probably they would show a grave detriment on their performance. Hence, with the development of new models, comparing them to each other and building on a much more realistic dataset, it is hoped to surpass the results obtained so far in this transition to a more credible environment.



(a) (b)

Fig. 1. Examples of data taken by the accelerometer and gyroscope of a specific individual's smartphone, during the first 15 seconds of each session, for each of the specified activities, being: (a) Raw data. (b) Data after having been preprocessed.

3. Experimental setup

This section contains a description of all the resources and methods that were used to carry out this work. Firstly, in Section 3.1, the data preprocessing guidelines are presented, as well as the chosen features for the machine learning models. Then, Section 3.2 gives a brief description of each artificial intelligence algorithm used, as well as introducing their most crucial hyperparameters.

3.1. Data preparation and feature extraction

To carry out this project, the dataset published in [19] was used. In that work, the authors gathered information from four different sensors: accelerometer, gyroscope, magnetometer and GPS. Likewise, it also offers datasets in which the gyroscope does not exist, or neither the gyroscope nor the magnetometer exist simultaneously. The last best results came from the case where the gyroscope data were missing. For this reason, in addition to studying all the sensors, it was decided to study this option as well. In this way, a detailed comparison can be made, in search of the most representative set and the real influence of the gyroscope on the final result. Regarding the activities performed, they were four, as said in that work:

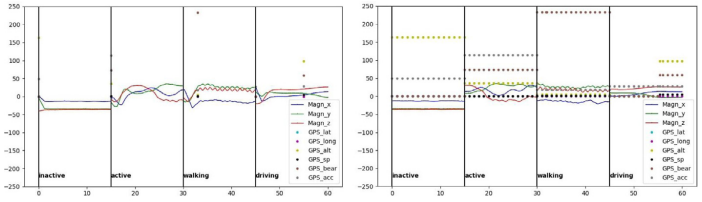
- Inactive: the individual does not have the smartphone on him.
- Active: any action that involves moving, such as cooking or brushing your teeth, but not moving anywhere in particular.
- Walking: any trip made without the use of vehicles, such as walking or running.
- Driving: every kind of journey made utilizing an engine-powered transport, without the need to be the person driving it.

As for the data preparation, it obeyed the same steps as in the original proposal, so the following actions were carried out:

- All outliers in the GPS data that exceeded 0.2 in latitude and longitude increments or 500 in altitude increments were ignored.
- The first and last five seconds of each data collection session were excluded to prevent the model from learning the movements of before and after the activity (putting in or taking the smartphone out of the pocket, for example). Each session corresponds to a whole activity recording, since a person starts an action until they stop it.
- Due to the lack of GPS data in many sessions, they were replicated so that there was always one record per second. For this purpose, if the difference between one observation and another was greater than one second, the last measurement was repeated, with a different timestamp. Similarly, for the same reason, any data session that did not have at least one GPS observation was disregarded.
- Any data session that had long gaps ($>$ 5 seconds) between the accelerometer, gyroscope or magnetometer sensor observations was also ignored.

In order to represent more clearly the arrangement of these data, Figs. 1 and 2 show examples of the data provided by a specific user for each of the indicated activities. In both figures, the image on the left (a) shows the first 15 s of each activity before any preprocessing, while the one on the right (b) shows the result of such preprocessing. The selection of this time interval allows for a straightforward interpretation of the values on a considerable figure size. In order to improve the readability, the figures were divided due to the high variability shown by the average of the values for each sensor. Fig. 1 shows the accelerometer and gyroscope data, while Fig. 2 shows the magnetometer and GPS data. The data for the accelerometer, gyroscope and magnetometer are shown separately for each of their three axes (“Acc”, “Gyro”, and “Magn” on the figures). As for the GPS acronyms, these correspond, on the one hand, to the latitude, longitude and altitude increments between each sample (“GPS_lat”, “GPS_long” and “GPS_alt”, respectively). On the other hand, the speed, bearing and accuracy of each of its measurements correspond to “GPS_sp”, “GPS_bear” and “GPS_acc”, respectively. Note that each GPS measurement is plotted as a single point instead of a continuous signal to highlight its differential behaviour regarding the other sensors. Additionally, each figure contains a series of vertical bars, every 15 s, to delimit the different activities in the same plot. It should be highlighted that these are not continuous signals from one activity to another. Each action is given by different sessions, from the initial moment of data collection until the 15 s have elapsed. As can be seen on the (b) figures, all GPS data is replicated, and the first few seconds of each session are cut out, for the reasons indicated above. Moreover, some clear trends can be observed for each sensor, depending on the activity performed.

Once data were loaded and preprocessed as previously described, the features were extracted. That extraction was based on the application of a sliding window from 20 to 90 s, in increments of 10, with the maximum possible overlap (one second less than each full window size), to have as many samples as possible. Thus, as an example shown in the original work for a window size of 20 s and the dataset containing all the sensors, Table 1 shows the number of available patterns and their distribution by every activity studied. On the other hand, in each of those windows, the features shown in Table 2 were calculated. As can be seen, there is a primary set already proposed in [19] (“Primary set” column) that will be used as a basis for comparison. Similarly, another series of features are also shown (“Proposed additions” column), which were added to the primary set. In this way, it will be possible to examine the differences in performance between the initial set and the one formed by that set plus the proposed collection. Each of the sub-columns that can be observed in this table refers to the sensors to which these features were practised. In the case



(a) (b)

Fig. 2. Examples of data taken by the magnetometer and GPS of a specific individual's smartphone, during the first 15 seconds of each session, for each of the specified activities, being: (a) Raw data. (b) Data after having been preprocessed.

Table 1
Number of patterns available and their distribution by every activity studied, for a window size of 20 seconds and the dataset containing all the sensors.

Activity				
Inactive	**Active**	**Walking**	**Driving**	**Overall**
214,130 (43%)	140,060 (28%)	83,376 (17%)	61,710 (12%)	499,276

Table 2
Feature set used.

Features			
Primary set	Proposed additions		
General	General	Not for GPS	Only for GPS
Mean		Signal magnitude area	
Variance	Energy	Number of zero crosses	
Median absolute deviation	Number of observations	Number of local maxima	Total distance travelled
Maximum	Maximum time gap	Number of local minima	
Minimum	Minimum time gap	Total positive time	
Interquartile range		Total negative time	

of "Not for GPS", they relate to the features that did not make sense to be used for GPS. That is because for the GPS the values are much more separated in time (approximately one value every 10 s) and always remain on the positive side of the signal, as these are absolute increments between observations. Besides, for the same reason, there are not three accurate axes like the X, Y and Z used in the other sensors, so it was also not attainable to use the signal magnitude area feature (SMA). Thus, a specific feature for this sensor was applied, "total distance travelled", which, from the increments of the latitude and longitude values, approximates the distance travelled using a Pythagorean theorem. The resulting value corresponds to the sum of all the hypotenuses, i.e. all the distances calculated in each of the observations.

Regarding the implemented features, it was decided to apply some of those included in the proposed collection, such as "number of zero crosses", "number of local maxima/minima" and "total positive/negative time". This is due to having seen their reliable performance in [37], in which a comparison of different features for HAR was made. The first of those listed refers to the number of times the signal changes from positive to negative or vice versa. All of them were considered attractive given the variability in the activities to be studied. For example, in a case of inactivity, these values should be much lower than those that could be found in a situation where the individual is walking or running. On the other hand, "energy" and "signal magnitude area", are two very common calculations in signals and the HAR field, so it was decided to include them as well. As for "number of observations", "maximum/minimum time gap" and "total distance travelled", these were features that were thought that could be favourable given the peculiarities of this dataset, in order to take advantage of the fact that there are some gaps between the data, mainly in the case of GPS.

3.2. Classification algorithms

Within the scope of HAR, numerous machine learning algorithms can be applied. In this case, it was decided to use the following ones: Support Vector Machine (SVM), Decision Tree (DT), Multilayer Perceptron (MLP), Naïve Bayes (NB), K-Nearest Neighbour (KNN), Random Forest (RF) and Extreme Gradient Boosting (XGB). The selection of these algorithms is due, in most cases, to the fact that they were the most used and with the best results within this field [22,28,29], as seen in the Related Work section. Only the case of XGB would be a novelty, as it has not been seen so much in this area. Anyhow, its addition was considered attractive to the list due to its high popularity in recent years and its outstanding results in many machine learning competitions [38]. Moreover, every algorithm mentioned above was implemented in Python, using the Scikit-learn library [39], as well as the XGBoost one [40] for its own case.

3.2.1. Support Vector Machine

Support Vector Machines (SVM) are machine learning models often used in binary classification problems [41]. This type of models searches for the hyperplane which maximizes the margins between two previously specified and labelled classes. To make this hyperplane non-linear, functions called kernels are used, which are one of the most crucial hyperparameters in SVM. These functions transform non-linear spaces into linear spaces, by changing the dimension in which they are plotted, making

possible the application of this linear approach. Depending on the kernel used (linear, polynomial or radial basis function), the hyperparameters to be applied change. The only fundamental hyperparameter that occurs in any kernel is C, which defines the number of errors that can be accepted by the model, as well as the width of the margins of the resulting hyperplane. In the same way, other fundamental hyperparameters also influence to a great extent the definition of such hyperplane. One of them is gamma (not applicable in linear kernels, among others), which determines the curves that the hyperplane can take, making them more accentuated or softer, depending on the patterns that are introduced into the model. Similarly, for polynomial kernels, the degree of the polynomial broadly affects the curvature that such a hyperplane can take. In fact, for example, if the degree is equal to 1, the result will be equivalent to that of a linear kernel (one straight line).

Usually, SVMs are employed to solve binary classification problems, but they can also be used for multi-class ones. To perform these tasks, it is necessary to choose a *one-vs-one* or *one-vs-all* strategy. In the first case, the classes are modelled in pairs, performing several binary classifications until a final result is obtained. Conversely, in the second case, the models are formed by confronting each class with the rest independently, creating a specific classifier for each situation. In this paper, the one-vs-all approach will be the one implemented, since it is the most used one in the literature [20,42]. In this way, the final result returned will be the average of all the classifiers created in the process.

3.2.2. Decision Tree

Decision Trees (DT) are one of the closest models to human thought, representing knowledge through trees. To do this, they generate a series of rules or questions that they use to predict and classify the data entered. There are numerous tree creation algorithms, including ID3 [43], C4.5 [44] or CART [45]. In this paper, the latter one will be used, as there is a widely accepted version, which is available and on which no modifications have been made for the purpose of comparison. To carry out this creation process, the algorithm follows a series of steps:

1. It starts by looking for the attribute that best defines each of the classes and places it at the top of the tree. This attribute is also known as the root node. To determine the order in which the attributes are evaluated, it uses statistical measures such as information gain. This metric calculates the expected reduction of uncertainty, which is obtained from the division of the dataset into a given attribute.
2. The algorithm then generates a criterion by which it separates the data, depending on the probability distribution of each of the classes in the tree.
3. Finally, it forms branches that split the datasets into subsets known as internal nodes. To evaluate these divisions, the algorithm uses the Gini Index, which provides a score of how good the resulting subsets are. The smaller this value is, the better the division.

Once these steps have been performed, the algorithm repeats the first and second steps until it reaches, in each branch, a leaf node, which is a subset of data that cannot be further divided.

3.2.3. Multilayer Perceptron

The Multilayer Perceptron (MLP) is one of the most widely used neural models nowadays, as well as being one of the first machine learning techniques to appear [46]. Unlike more traditional neuron networks, it can have more than one layer of neurons. For the simplest case, it would consist of three different layers, where the first one would be the input layer, followed by the hidden layer and ending with the output layer. The data are entered by the input layer, taking the predictions in the output layer. The hidden layers can be multiple, depending on how complex the model needs to be for the specified problem. Each layer is represented as follows:

$$y = f(W \times x + b) \tag{1}$$

The letter "f" would be the activation function, which is responsible for describing the input–output relations in a non-linear way. In this way, the model has more power to be more flexible in the description of arbitrary associations. On the other hand, "W" refers to the layer weights, which change as errors are found, by adding the learning rate, which can be constant or dynamic. Similarly, "x" would correspond to the input data vector of the previous network and "b" would be the bias vector, which is an additional set of weights with which to allow the layer to produce a series of output data. In order to carry out the training of the network, it is necessary to define a loss function. This loss will be high if the class predictions do not correspond to the ground truth, and will be low if they do. In this way, the layer weight values (W) would be added to this loss. The idea is that during the training of the model this loss value will be low. For this purpose, functions called optimizers are used, which look for the appropriate values of the weights with which to lower this value. In this paper, the Adaptive Moment Estimation (Adam) function will be used [47], as it is the more recommended one for large datasets. Moreover, to avoid overfitting, these algorithms use an alpha parameter that penalizes weights with large magnitudes.

3.2.4. Naïve Bayes

The Naïve Bayes (NB) classifiers are a collection of classification algorithms based on Bayes' Theorem [48]. This theorem expresses the conditional probability of an event A given B, from the conditional probability of B given A and the marginal probability of A. This definition is represented in the Bayes' Rule:

$$\Pr(A|B) = \frac{\Pr(B|A)\Pr(A)}{\Pr(B|A)\Pr(A) + \Pr(B|\neg A)\Pr(\neg A)} \tag{2}$$

Thus, it is not a single algorithm, but a family of algorithms that share a common principle, which is that in each pair of classified features, each one is independent of the other. The main differences between each of the algorithms of this family are based on the assumption they make regarding the distribution of $\Pr(B|A)$. The continuous values associated with each feature are assumed to follow a specific distribution, such as the Gaussian one, a given multinomial distribution or Bernoulli's multivariate event model, where the features introduced are independent booleans (binary variables) [49]. In this paper, this last assumption will be used, since the rest do not apply to our problem, or offered preliminary results far below what it is considered randomness (50% success rate). On the other hand, although the assumptions made by this kind of methods may seem very simple, the truth is that this kind of algorithms have worked well in many tasks. Moreover, it is an extremely fast classifier compared to other types of more sophisticated machine learning algorithms, so it is considered that it is worth trying.

3.2.5. K-Nearest Neighbour

The K-Nearest Neighbour (KNN) algorithm is supervised and instance-based, so it needs the data entered to be pre-labelled, as well as not being able to create a model explicitly [50,51]. Instead, it memorizes the training instances that are used as a basis for the prediction phase. The most crucial point in this algorithm is the selection of the "K" number, which represents the number of neighbours that are taken into account in the neighbourhood to

classify the previously specified groups. In this way, the algorithm follows a series of steps defined for each of the observations in the data:

1. The distances between the selected observation and all other observations in the dataset are calculated. This distance can be understood as a similarity measure between these elements. It is calculated using a predefined function, such as the Euclidean or the Manhattan distance.
2. Then, the closest K-elements are selected and a majority vote is taken among them. The dominant class will be the one deciding the final classification, depending also on the weights given to each of these classes.

One of the most prominent problems in KNN is the immense amount of memory and time required as the selected dataset grows. That is because they need to evaluate every observation in the data, so if the number of features and data is very high, the computational resources required for their training can be quite significant. Nevertheless, it is considered as an algorithm that can produce great results, being also easy to understand and to implement.

3.2.6. Random Forest

Models based on Random Forest's (RF) algorithm are among the most popular nowadays [52,53]. Through the creation of multiple decision trees from previously tagged data, they can produce very robust models. That is because, by having to create different trees, they can select the best possible solution in a much more general and flexible way, as well as also reducing overfitting by not having a single decision tree. Thus, the main work of the algorithm is divided into the following steps:

1. First, you start by selecting various subsets randomly over the given dataset.
2. Then, the algorithm will build decision trees for each of these examples, following the steps described in 3.2.2. The number of decision trees constructed will be given from the number of estimators hyperparameter, which is specified previously.
3. Once the trees have been created, the resulting prediction is obtained from each of them. At this point, a vote is also taken on each of these resulting values, where the dominant class will decide the final result.
4. Finally, the most voted class is selected as the final result of the prediction.

When making predictions with the model already created, this algorithm is usually much slower than the rest. That is because of having to average the outcomes of each of the trees that make up the final model. Even so, it is widely used today because it is capable of creating very robust models with a high-grade performance, being also faster to train than many of the other artificial intelligence algorithms used nowadays.

3.2.7. Extreme Gradient Boosting

Although Extreme Gradient Boosting (XGB) is not an algorithm by itself but a refined implementation of the Gradient Boosting algorithm [40], it is worth to be considered. The main reason is that this approach has won several competitions and has recurrently offered very competitive results in the related literature [38]. The implementation provides a more efficient and flexible method by parallelizing the tree boosting process. Concerning the Gradient Boosting Machine (GBM), it is an algorithm that seeks the production of a model through the formation of numerous "weak" prediction models, usually decision trees. For this purpose, decision trees are created in a stage-wise fashion, sequentially, following the same lines listed in 3.2.2. As in Random Forest, a number of estimators hyperparameter is used to determine the number of trees to be created. The idea is to seek the progressive improvement of the final model. To do this, a loss function is defined that evaluates the performance of the last tree created, and which, presumably, will progressively decrease as all the observations in the trees built are better classified. That results in a final model that is much more robust and easy to tune, as well as offering excellent results. However, it can be quite sensitive to overfitting and noise, so it is important to be careful when training it.

Table 3
Binary confusion matrix example.

		Model output	
		False	**True**
Ground truth	**False**	TN	FP
	True	FP	TP

3.3. Evaluation metrics

One of the most elementary and easily interpreted metrics is the confusion matrix. A confusion matrix is a table that facilitates the visualization of the performance of a classification model from a set of test data. A simple example would be the one showed in Table 3. From this, we can draw many widely used terms to evaluate these confusion matrices, such as: precision, recall, accuracy and F_1-score [54].

Precision and recall are the metrics used to measure the quality and quantity of the classifications made, respectively. Precision measures the number of true positives, divided by the total number of positive results returned. Concerning recall, it measures the number of true positives, divided by the number of correct results that should have been returned. Their formulas would be as follows:

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

Any other way, accuracy and F_1-score metrics are used to know the performance of a model in test. The first consists of the measurement of all correctly identified cases, while the second is based on the harmonic mean of the recall and precision metrics. Their formulas would be as follows:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{5}$$

$$F_1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{6}$$

However, in a multi-class problem such as the one in this paper, the way these metrics are computed changes. As a model example, it can be seen how these values would be calculated in Table 4, compared to the binary case of the previous example. Thus, to calculate the overall precision and recall of the whole model, the final value is obtained from various types of averaging, among which the following stand out: *micro* and *macro* [55]. The first one considers the total of TP, FN and FP to calculate the metric, so it is suitable for problems with mutually exclusive classes. As for macro, it returns the average of calculating the metric for each label, regardless of the proportion of each of them in the dataset. Concerning accuracy, it is usually calculated in the same way as in the latter case.

On the other hand, F_1-score has more ways of weighting its result to evaluate multi-class classification problems. In addition

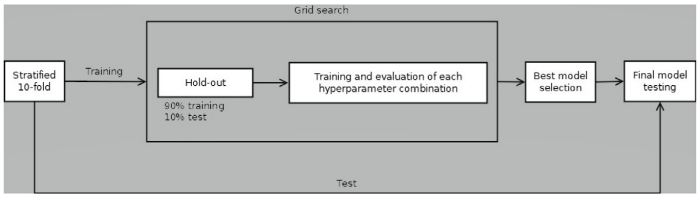


Fig. 3. Methodology followed for each algorithm and window size specified.

Table 4
TP, TN, FP and FN calculations for the "Class 1" class of a multi-class confusion matrix example.

		Model output		
		Class 1	Class 2	Class 3
Ground truth	**Class 1**	TP		FP
	Class 2 Class 3	FN		TN

to the two most commonly used ones discussed above, there is a variant of macro, *macro-weighted* (provided by the Scikit-learn library), which does take into account the data proportions by averaging the precisions and recalls of each of the classes involved.

Although accuracy is the most widely used measure globally, F_1-score is also closely linked to the correct classification of groups, but it is not as influenced by possible imbalances between classes in the datasets [56]. In fact, when this occurs, the accuracy could give an incorrect impression of the final results. In this paper, the F_1-score will be used as the initial assessment metric, specifically with the macro-weighted case of averaging, as the proportions of the data are quite inclined towards one of the classes involved. Anyhow, the final results will be shown mainly with the accuracy metric, since it is the most common one for comparison with the rest of the works.

3.4. Validation and optimization techniques

One of the most widely used model validation techniques in the field of machine learning is *cross-validation* [57]. Before entering the data in the model to be trained, the data is divided into training and test. This elementary division of data is also called *hold-out*. In this way, there is a set of data destined to train the model and another subset that serves to test its performance, with data a priori unknown to it. One of the most common ways of making this division is employing the *k-fold cross-validation* technique. Here, the aim is to partition the original data set into "k" equal sized subsamples. Then, one of these subsamples is selected as the validation set to test the model, being the rest of subsamples used to train it. This process is repeated "k" times until all subsets are selected as a test once. Finally, the results obtained by the model are averaged and the relevant performance evaluation metrics are calculated. In this paper, a variant of this technique will be used, called *stratified k-folding*. This alternative seeks that the proportion of each class in each of the subsets created is practically the same. In this way, the existing imbalance on the dataset and its possible influences on the model performance are avoided.

As for the optimization of the models to be used, there is a technique widely employed in the field called *grid search* [58]. A grid search is a process that consists of an exhaustive search for the best combination of hyperparameters introduced to a model using a particular algorithm. Here, each of the possibilities of the set of hyperparameters previously indicated is tested. It is a long and expensive process, but with which it is possible to know the best possible performance of the model to be used. In this paper, the F_1-score commented in 3.3 will be used as the evaluation metric for these combinations. Also, some of these combinations will be trained not once, but 50 times. This is because non-deterministic algorithms such as Multilayer Perceptron (MLP), Random Forest (RF) and Extreme Gradient Boosting (XGB) will be used. The nature of these types of algorithms means that there is always a small random component that affects their final result. Thus, by averaging the values obtained in each of these repetitions, it is possible to get a reliable outcome.

In the present paper, these techniques will be used together to find the most robust and optimal model for these problems, as shown in Fig. 3. For this purpose, an initial 10-fold will be applied to the dataset. Each of the training parts of each fold will be introduced later in the grid search. To validate the performance of each of these models, the data of the corresponding fold will be divided applying a hold-out. Hence, 90% of the data will be for training and the resulting 10% for testing. It was decided to apply this technique instead of the traditional k-fold cross-validation due to the high number of experiments to be performed in the grid search. In this way, we obtain the final results in a reasonable time, as we do not have to evaluate each of the folds. Besides, the high number of patterns available (around 450,000 for each fold) could lead to greater redundancy in the data if we applied another 10-fold. Finally, the best model selected by the grid search will be tested again, this time with the corresponding part for testing of the initial 10-fold. Thus, the models are tested with truly unseen data during their training, making the final result more realistic concerning their data generalization capabilities.

4. Results and discussion

Within this section, all the results of the experiments carried out will be displayed. First, in Section 4.1, all the data obtained will be represented, with their corresponding graphs and evaluation metrics. Then, in Section 4.2, the main observations and considerations of the results obtained will be discussed.

4.1. Results

After having prepared the data, applying the selected window sizes with the previously discussed features (Table 2), a series of experiments were conducted on them. As described before, the sliding window sizes ranged from 20 to 90 s, in increments of 10, with the maximum possible overlap (one second less than each full window size). To do this, the most applied machine learning algorithms in HAR were used, which are: Support Vector Machine (SVM), Decision Tree (DT), Multilayer Perceptron (MLP), Naïve Bayes (NB), K-Nearest Neighbour (KNN) and Random Forest (RF), with the addition of Extreme Gradient Boosting (XGB). In order to get the best possible results for each of them, it was decided to explore the best architecture for each of them beforehand. For

Table 5
Chosen hyperparameters to perform further grid search for each machine learning algorithm.

Algorithms	Hyperparameters
SVM	Kernel = {linear, RBF (Radial Basis Function), polynomial} C = {1, 10, 100, 1000, 10000} Gamma (not applicable for linear kernels) = {0.0001, 0.001, 0.01, 0.1, 1} Degree (only applicable for polynomial kernels) = {1, 2, 3, 4} Maximum iterations = 1000
DT	Maximum depth = {5, 8, 15, 30, None} Leaf minimum size = {1, 2, 4, 8, 16, 32, 64} Minimum size for node division = {2, 5, 10}
MLP	Hidden layers and units size = {(5,), (10,), (20,), (30,), (50,), (70,), (100,) (5, 5), (10, 10), (20, 20), (30, 30), (50, 50), (70, 70), (100, 100)} Activation function = {tanh (Hyperbolic Tangent), ReLU (Rectified Linear Unit)} Alpha = {0.0001, 0.05, 0.1}
NB	It has no specific hyperparameters. In our case, we used the model based on Bernoulli, since the rest, after some preliminary tests, were not applicable to this problem.
KNN	Number of neighbours = {3, 5, 7, 11, 15, 19, 24, 29, 34} Weights = {uniform, distance} Metrics = {Euclidean, Manhattan} Leafs size = {30, 50, 100}
RF	Number of estimators = {100, 250, 500, 1000} Maximum depth = {5, 12, 25, 50} Leaf minimum size = {1, 2, 4} Minimum size for node division = {2, 5, 10}
XGB	Number of estimators = {100, 300, 500, 800, 1200} Maximum depth = {5, 8, 15, 30, None} Minimum size for node division = {1, 3, 5}

Table 6
Accuracy results comparison between algorithms and sliding window sizes for the initial feature set and the complete dataset.

	Window size							
	20	30	40	50	60	70	80	90
SVM	69.28% ±15.10%	78.07% ±10.37%	81.30% ±8.79%	79.52% ±10.57%	78.84% ±8.54%	79.45% ±9.77%	81.23% ±8.53%	80.90% ±9.02%
DT	88.17% ±12.47%	85.79% ±16.40%	88.12% ±8.83%	86.71% ±13.47%	87.82% ±13.00%	86.17% ±14.37%	87.57% ±12.81%	89.91% ±7.15%
MLP	86.46% ±6.30%	86.80% ±6.02%	86.85% ±6.12%	86.59% ±6.61%	86.65% ±7.11%	86.39% ±7.69%	86.57% ±7.90%	85.47% ±8.65%
NB	78.11% ±6.93%	78.68% ±6.61%	79.09% ±6.73%	79.48% ±6.92%	79.66% ±6.95%	80.01% ±7.08%	80.09% ±7.08%	79.62% ±6.81%
KNN	85.68% ±7.20%	86.30% ±6.20%	86.83% ±6.34%	86.32% ±6.48%	86.56% ±6.50%	86.84% ±6.77%	86.99% ±6.57%	87.09% ±6.76%
RF	91.78% ±5.20%	92.27% ±5.58%	92.36% ±5.74%	**92.56%** **±5.92%**	92.55% ±5.99%	92.29% ±5.86%	92.37% ±5.80%	92.28% ±6.50%
XGB	90.58% ±7.57%	90.47% ±9.09%	91.42% ±7.62%	91.36% ±7.76%	91.80% ±8.06%	92.23% ±7.30%	91.21% ±6.98%	91.30% ±7.09%

this reason, different values for the most crucial hyperparameters of each of them were selected, in search of the best possible combination between them. To choose the best final result, the next steps were followed, for each of the previously specified datasets:

1. The first step was to carry out a stratified 10-fold to have ten different datasets, with approximately the same pattern distribution for each class.
2. Then, with each of the previous folds, a grid search was carried out to find the best combination of hyperparameters of each algorithm. Because the dataset used has some unbalance towards the class of "inactive", the resulting predictions were evaluated using the F_1-score metric, offering a value that should better represent the performance of the model. On the other hand, concerning the hyperparameters used for each of the selected machine learning algorithms, they were all arranged in Table 5. The intention was to have a wide range of hyperparameters, to know the best possible option. Therefore, attempts were made to increase the number of possibilities in those that were more influential in the final result. However, the overall complexity of each training increases with higher amounts of hyperparameters, so it was necessary to be conservative in general. Any other hyperparameters not listed there were chosen and set to the default value, after some preliminary experiments and confirmation of good performance. In this way, the number of experiments is adequate to find the optimal model in a reasonable time.
3. For the MLP and RF algorithms, given their non-deterministic nature, the previous step was repeated 50 times, for each of the ten initial folds. In this way, we avoid that their random behaviour affects the final result, averaging all the obtained ones. On the other hand, although the XGB algorithm also should have this non-deterministic nature, for the Python package used in this work the results are always the same, so it was not necessary to perform these repetitions.
4. Once the grid search was completed, the results were evaluated and the best combination of hyperparameters for each of the folds was selected. Then, each of the best models selected was tested with truly unseen data from the initial 10-fold.

Table 7
Accuracy results comparison between algorithms and sliding window sizes for the proposed feature set and the complete dataset.

	Window size							
	20	30	40	50	60	70	80	90
SVM	80.77% ±12.64%	82.00% ±13.45%	82.15% ±14.12%	83.38% ±10.85%	83.59% ±11.84%	85.12% ±11.55%	86.56% ±11.30%	85.98% ±11.18%
DT	89.99% ±6.13%	89.92% ±6.62%	87.95% ±10.18%	88.27% ±11.01%	87.68% ±12.17%	86.94% ±14.31%	89.63% ±8.38%	88.26% ±10.26%
MLP	83.76% ±10.37%	84.00% ±10.49%	84.24% ±10.38%	84.60% ±10.22%	84.73% ±10.32%	84.32% ±10.79%	84.96% ±10.37%	84.43% ±10.09%
NB	81.69% ±7.20%	82.21% ±7.16%	82.49% ±7.16%	82.77% ±7.32%	82.86% ±7.60%	83.06% ±7.67%	83.27% ±7.78%	82.72% ±7.68%
KNN	87.62% ±7.37%	88.27% ±7.02%	88.72% ±6.99%	88.99% ±6.96%	88.76% ±7.94%	88.80% ±8.01%	89.02% ±8.00%	88.03% ±7.94%
RF	91.71% ±5.47%	92.08% ±5.49%	92.26% ±5.68%	92.51% ±5.86%	92.73% ±5.98%	92.77% ±6.16%	**92.97%** ±**6.23%**	92.61% ±6.60%
XGB	88.32% ±12.29%	88.99% ±11.66%	88.87% ±12.61%	88.78% ±14.26%	89.72% ±11.54%	89.55% ±12.54%	90.38% ±9.64%	91.15% ±7.01%

Once the steps indicated in the previous paragraph have been carried out, for each of the algorithms and sliding window sizes used, the results shown in the Tables 6 and 7 were obtained. The value below each accuracy result refers to its standard deviation. These values correspond to the initial and proposed sets of features, respectively. As can be seen, the tree-based models, DT, RF and XGB, work considerably better than the rest, since they are the only ones capable of approaching and even surpassing the 90% of success. The cases of RF and XGB stand out even more because they are less variable than DT. However, as it is logical, the computational complexity of these algorithms is much higher than in the case of DT. On the other hand, the size of the windows is not as influential as it was thought at first, except for SVM, where the fluctuations are quite broad, especially in the step of 20 to 30 s. For that reason, although the best value is obtained with a 80-second window, the model might perform similarly with a window such as the 20-second one, since the difference in accuracy is pretty low in most cases. In this way, they would be easier to apply to a real-life environment, as they can be used more finely with the activities that are being carried out at each moment. As for the differences between each of the sets of features, the truth is that these were much smaller than expected. In fact, after performing a T-test between each pair of values of both groups, significant differences were only found in one case. This only case corresponds to the SVM algorithm, specifically for the 30-second window size. For the rest, the p-values were all above 0.1, which means that the results are statistically similar. However, some clear improvements are visible for the proposed set, as in the cases of SVM, NB and KNN. On the contrary, for MLP and XGB, it seems that the results are slightly worse. The rest of the algorithms remain with very similar values to each other, especially in the case of RF. Additionally, the test results between each pair of results in Tables 6 and 7 seem to indicate that the new features do not add information that would significantly increase accuracy, not making it possible to reach a reliable conclusion. Moreover, in the same way, in Fig. 4 the F_1-scores resulting from each of the algorithms and window sizes used are shown, for each of the sets of features applied. Both metrics are displayed to, on the one hand, show the numerical accuracy results in tables as a comparison with other works. On the other, it is also possible to show the differences between each case in a much more visual way through the F_1-score graphs.

On the other hand, the results obtained for the dataset that did not include the gyro, both with the old and the new features, are also shown in Tables 8 and 9, respectively. As in the previous case, a T-test was also carried out to check the differences between the sets of features introduced in the models. Once again, the only case where significant differences can be seen is with SVM and a window size of 30 s. Therefore, the results are, in general, statistically similar, so we cannot claim that they are different. In any case, the observations from before are repeated, with improvements in SVM, NB and KNN, as well as a worsening in MLP and XGB. However, in this case, DT worsens slightly. Anyhow, as can be seen, the results are, in general, somewhat worse than those obtained with all the sensors. Although there is indeed some case with some improvement, like the one found in [19], with SVM and 20 s of a window, it is considered that the models work better with the complete case. In this way, the gyroscope does provide a slight improvement in the models, as shown in previous works that included it in their tests. Similarly, the F_1-scores for this case are also shown in Fig. 5.

Regarding the winning hyperparameter combinations of each algorithm, for each window size and dataset, the best values of F_1-score obtained for each of those combinations are shown in Fig. 6. Here, the "Feature dataset" field refers to the combination of the set of features and the dataset used in each case, as shown in Table 10. Thus, the numbers 1 and 2 would correspond to the dataset containing all sensors, while 3 and 4 to the dataset not including gyroscope data. At the same time, the numbers 1 and 3 would also correspond to the initial set of features, while 2 and 4 to that proposed in this paper. As can be seen, no single winning hyperparameter combination is obtained for each algorithm, but different sets depending on the case study. Therefore, the data displayed there for each of the applied algorithms are detailed below:

- SVM. The RBF kernel was by far the most selected, with only a few cases of grade 3 polynomial kernels in the larger window sizes when using the proposed feature set. The linear kernel was always lower in overall performance. As for C, the values fluctuated a lot, but seemed to settle more for the intermediate values of 10, 100 and 1000, leaving quite apart from the value 1. Finally, with gamma something similar happened, although the extreme values of 1 and 0.0001 were practically never selected, being clearer the dominance of the rest of the values, more or less in equal parts.
- DT. The maximum depths in DT always remained at the low values of 5 and 8, except for some isolated cases for the complete dataset with the primary feature set (case 1). In the latter case, the chosen value was 15. As for the minimums per leaf and to divide the node, their values were very arbitrary and all were selected more or less equally, so it is not possible to draw a reliable conclusion.
- MLP. The case of (100,) was by far the most selected for the size of the hidden layers. However, there were also

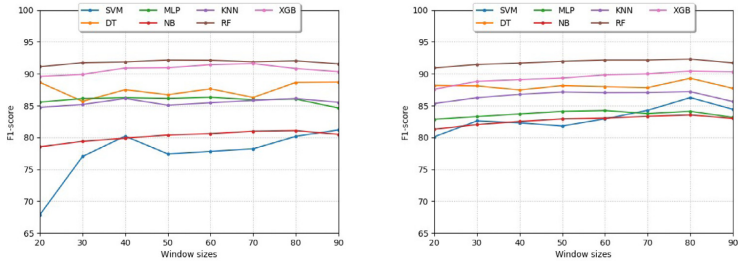


Fig. 4. F_1-scores for: (a) The initial feature set and the complete dataset. (b) The proposed feature set and the complete dataset.

Table 8
Accuracy results comparison between algorithms and sliding window sizes for the initial feature set and the dataset missing the gyroscope's measurements.

	Window size							
	20	30	40	50	60	70	80	90
SVM	74.39% ±10.75%	73.47% ±9.66%	76.77% ±10.98%	81.00% ±9.41%	80.66% ±12.96%	82.56% ±7.68%	80.70% ±9.22%	81.77% ±9.28%
DT	82.74% ±13.36%	83.67% ±13.72%	87.07% ±8.23%	87.06% ±9.19%	87.88% ±8.75%	88.82% ±6.29%	88.60% ±7.44%	87.28% ±8.93%
MLP	86.35% ±4.95%	86.70% ±4.97%	86.46% ±5.22%	86.63% ±6.06%	86.88% ±6.56%	87.16% ±6.85%	87.00% ±7.10%	87.01% ±7.42%
NB	80.23% ±7.30%	80.39% ±7.27%	80.62% ±7.50%	81.24% ±7.32%	80.94% ±7.35%	80.97% ±7.63%	81.32% ±7.63%	81.42% ±7.13%
KNN	84.61% ±6.15%	86.10% ±5.13%	86.28% ±5.57%	86.81% ±5.38%	86.84% ±5.52%	86.68% ±5.85%	86.92% ±5.88%	86.18% ±9.49%
RF	89.34% ±6.67%	89.75% ±7.27%	90.03% ±7.64%	90.45% ±7.44%	90.62% ±7.52%	90.63% ±7.92%	**90.76%** ±**8.01%**	90.36% ±8.06%
XGB	87.40% ±10.57%	89.21% ±7.34%	89.75% ±7.22%	90.35% ±6.80%	90.27% ±7.41%	89.53% ±8.61%	89.75% ±8.06%	90.50% ±6.62%

Table 9
Accuracy results comparison between algorithms and sliding window sizes for the proposed feature set and the dataset missing the gyroscope's measurements.

	Window size							
	20	30	40	50	60	70	80	90
SVM	80.65% ±11.65%	81.34% ±9.45%	80.66% ±11.48%	82.57% ±9.56%	82.77% ±8.71%	83.71% ±7.96%	84.64% ±7.86%	84.88% ±7.75%
DT	82.15% ±12.88%	81.97% ±12.50%	84.77% ±10.49%	84.71% ±12.07%	84.04% ±12.62%	84.64% ±12.34%	85.55% ±11.90%	86.66% ±9.72%
MLP	84.07% ±7.82%	84.57% ±8.02%	85.03% ±7.98%	85.45% ±7.68%	85.69% ±7.54%	85.51% ±7.82%	85.94% ±7.55%	86.17% ±7.42%
NB	81.49% ±6.44%	82.18% ±6.70%	82.53% ±6.93%	82.60% ±7.23%	82.72% ±7.34%	82.92% ±7.66%	83.04% ±8.04%	82.75% ±7.72%
KNN	85.43% ±6.49%	86.25% ±6.50%	86.96% ±6.34%	87.16% ±6.17%	87.47% ±6.05%	87.45% ±6.04%	87.60% ±5.64%	86.73% ±6.05%
RF	88.94% ±6.22%	89.55% ±6.24%	90.17% ±6.29%	90.19% ±7.15%	90.41% ±7.51%	90.50% ±7.85%	**90.62%** ±**7.97%**	90.22% ±7.90%
XGB	85.57% ±11.17%	84.96% ±13.02%	87.33% ±11.12%	86.15% ±12.98%	86.57% ±12.75%	87.44% ±11.37%	88.33% ±10.61%	87.32% ±10.79%

some cases of (70,), for the 90-second windows, and (5,) for the 20-second ones (for the primary feature set). As for the activation functions, the tanh (Hyperbolic Tangent) case dominated for the primary feature set (cases 1 and 3), while ReLU (Rectified Linear Unit) was always the chosen function in the proposed one (cases 2 and 4). Besides, the ReLU was also always chosen for the 20 and 30-second windows of the cases 1 and 3. Regarding the alpha values, generally, those of 0.1 were chosen much more, with some appearances of 0.05 and only one choice of 0.0001.

- NB. No hyperparameters to choose.
- KNN. Here, the number of neighbours was always 34 for the smaller window sizes (until 40 or 50 s). For the following ones, the number was regularly diluted to the lowest values in the list with the size of 90 s, except for the complete dataset with the primary feature set (case 1), where these high values were relatively constant. As for the weight, it was pretty arbitrary in all cases, so it is not possible to draw a reliable conclusion. Concerning the chosen metric, it was always Manhattan, in absolutely all measurements. Finally,

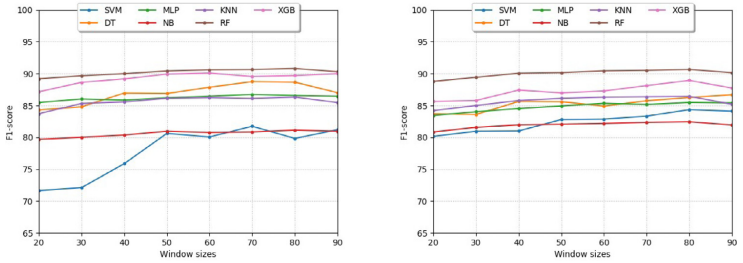


Fig. 5. F_1-scores for: (a) The initial feature set and the dataset missing the gyroscope's measurements. (b) The proposed feature set and the dataset missing the gyroscope's measurements.

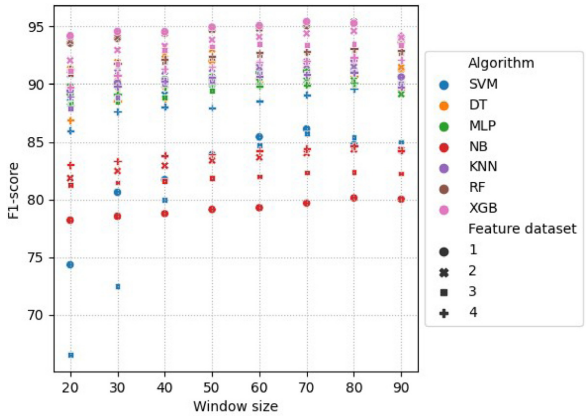


Fig. 6. F_1-scores for each winning hyperparameter combination of each algorithm used.

the size of the leaves did not seem to have any influence whatsoever on the results, as the accuracy was always the same for any of the three values studied.

- RF. In the vast majority of cases, the number of estimators remained at the value of 1000 and, to a lesser extent, 500. The value of 1000 dominated mostly in cases where the proposed feature set was applied (cases 2 and 4). On the other hand, the maximum depth was more inclined towards the mean values of 12 and 25, with some cases of 50. The dominance of these mean values was clearest in cases 2 and 4, as with the previous parameter. As for the minimum per leaf, the most selected value was 1, although in case 1, 4 was the most dominant by far. Finally, the minimum to split the node does not seem to be entirely conclusive, as all cases were selected more or less equally, with some tendency towards the lower values of 2 and 5. The value of 10 was only ever selected for cases 3 and 4 (non-gyroscope ones).
- XGB. In this case, the values are pretty arbitrary. However, there does seem to be a trend towards the number of 1200 estimators compared to the rest, especially with the non-gyroscope dataset. As far as the maximum depth is concerned, 5 and 8 were the most widely used values. Finally, the minimum to divide the node was centred much more on the lower numbers of 1 and 3.

To select the best-resulting model, a Critical Difference diagram was carried out, as shown in Fig. 7. This diagram was constructed from all the datasets, features and window sizes used, with the best values obtained for each case (the ones showed in Tables 6, Table 7, Tables 8 and 9). As it is shown in the aforementioned figure, RF, XGB, DT and MLP models appear to be statistically equivalent. From these four, given the results, it was decided to select RF, because it has the highest accuracy peaks and is less computationally complex than XGB. Additionally, although it requires more time and computational resources than MLP and DT, its performance is considerably better, as well as being a more advanced version of the latter one.

As can be seen in all the tables and figures shown, the best model obtained is the one thrown by the Random Forest algorithm, for 80-second time windows and for the proposed set of features. This case yields the average confusion matrix shown in Table 11, along with its particular metrics (recall, precision and accuracy). The model manages to correctly classify all activities, although some problems with the "active" action are visible. That is because this activity is very diffuse, and can include both moments of activity and inactivity while the individual remains "active". Thus, some confusion can be expected from the classifier with the rest of the activities concerning this one. Although there is still room for improvement, it is considered that the classifier

Table 10
Combinations of dataset and feature set used.

	Dataset		Features	
	Complete	Missing gyroscope	Primary set	Proposed additions
Case 1	X		X	
Case 2	X			X
Case 3		X	X	
Case 4		X		X

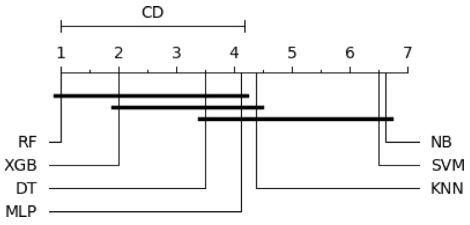


Fig. 7. Critical Difference diagram made from all the results obtained with all the algorithms used.

Table 11
Average confusion matrix for the best combination found.

	Ground truth				Precision
	Inactive	Active	Walking	Driving	
Inactive	19,965	230	261	13	97.54%
Active	888	12,980	1,005	373	85.14%
Walking	24	325	6,043	94	93.17%
Driving	50	44	29	5,157	97.67%
Recall	95.40%	95.59%	82.35%	91.49%	92.97%

Table 12
Average confusion matrix for the 20-second window size option of the best case found.

	Ground truth				Precision
	Inactive	Active	Walking	Driving	
Inactive	20,451	328	190	22	97.43%
Active	852	13,089	1,359	493	82.88%
Walking	51	474	6,700	106	91.39%
Driving	58	115	90	5,550	95.48%
Recall	95.51%	93.45%	80.35%	89.94%	91.71%

achieved far exceeds what was expected, with a resulting accuracy of 92.97%, a much higher value than that of 74.39% achieved in other works [19].

That was the highest result among all the combinations of feature set, window size and dataset. However, after performing a Tukey test, it was clear that there were no significant differences among the different window sizes, as it is evident if we take a look at Fig. 8. Thus, any window size would be feasible to be selected as the best, depending on the problem in which it would be used. In this case, it is considered that 20-second windows would be more than enough to obtain good results, since it would allow a more suitable classification of the activities to be studied by being able to separate them into 20-second intervals. The average confusion matrix for this case would be the one shown in Table 12. As can be seen, the most fundamental differences lie in the "active" activity, as noted above. Even so, the results are statistically similar to those of the best case found with window sizes of 80 s, so it is considered feasible to select this one option preferably.

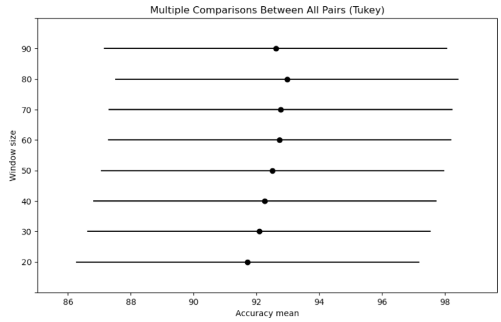


Fig. 8. Results of the Tukey test performed for all window sizes used with Random Forest, for the complete dataset and proposed feature set (case 2).

4.2. Discussion

The results obtained in this paper manage to advance to a great extent towards that real-life environment that is so much sought after. The resulting accuracy of the best model found is exceedingly superior to the best obtained so far, from 74.39% to 92.97%. Besides, several findings have been made about the given dataset, as it appears that the size of the sliding window is not as crucial as first thought. Consequently, the results are quite similar between all the window sizes used in the vast majority of cases.

On the other hand, it has also been possible to reinforce the demonstration of the advantages of using the gyroscope as opposed to not using it in HAR, resulting in a better performance than when it is not used. However, in an effort to further improve the final results, it has not been possible to enhance the primary set of features used in [19]. The additions proposed in this paper have not been entirely conclusive, since the comparison between using each set has quite similar performances to each other, with improvements and worsenings depending on the algorithm and window size used. While there are clear cases of an upgrade such as SVM and NB, the same cannot be said of MLP, XGB and even DT. Perhaps the most robust examples in this sense were KNN and RF, with pretty slight variations in both cases. Probably the proposed features are adding noise to some algorithms and hence are not entirely favourable.

In any case, given the results, it can be concluded that the best algorithms to use in this matter are the tree-based ones (DT, RF and XGB), since they have been the ones that have given the best outcomes with a considerable difference from the rest. Even so, MLP and KNN deserve special mention, since its results have been kept only a little below the latter.

Also, on the other hand, it should be noted that there are still problems in optimally discerning the activity of "active", unlike the rest, where the percentage of success is very high. Although this activity is very diffuse and can be easily confused with cases of "inactive" or "walking" (especially with this one),

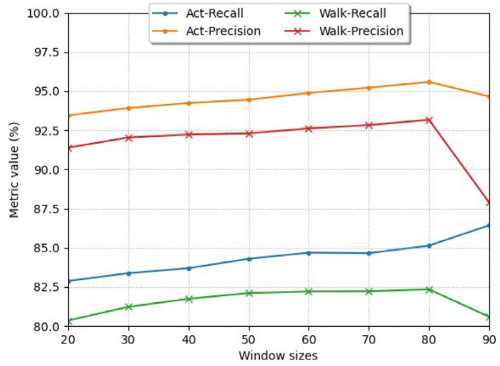


Fig. 9. Recall and precision values for the "active" and "walking" activities and every sliding window size used.

it is possible that with a more in-depth study for this case the final solution will be found. In fact, looking at the confusion matrices for each of the window sizes of the case indicated in Table 11, a relevant trend is indeed observed. With smaller window sizes, this confusion is more pronounced, with more samples misclassified among these activities. Similarly, with larger sizes, this confusion is milder. This is shown in Fig. 9. However, as previously discussed, window size did not end up being a crucial parameter for the overall performance of the models studied, although it seems to be a trend to have the highest accuracy peaks on the 80-second windows. Additionally, as can be seen in that figure, there is a clear downgrade when the window size reaches the value of 90. Perhaps using other more influential features for this activity could make it possible to obtain the optimal model for this case. Also, the application of other types of algorithms, such as LSTM and CNN, characteristic of the deep learning aspect, could improve the performance, as they are giving outstanding results in HAR in recent years.

5. Conclusions and future work

In this work, a series of experiments were carried out in search of the improvement of the last model developed in HAR, for a dataset oriented to the introduction of this problem in a real-life environment. By using various machine learning algorithms, different features and much larger sliding windows, the results were severely improved. In addition, it has been observed that the preliminary results on the dataset in which gyroscope's measurements were missing were inconclusive. This sensor finally proved to improve the final performance in a general way in all the algorithms used, as in the rest of the works that included it in their experiments.

Unfortunately, the proposed set of features has not obtained such good results. The performance of the algorithms with this set and the primary one is quite similar in most cases, with ups and downs depending on the algorithm to be used, so its impact is not entirely conclusive. Perhaps it is necessary to think of other types of features that could improve the classification of the "active" activity, which, although the current models differentiate it relatively well, is quite improvable. Also, it could be interesting to make some kind of evaluation and selection of features, applying, for example, a Principal Component Analysis (PCA), to see which ones are the most suitable.

Another point worth to be mentioned is the fact that the size of the windows does not have as much influence on the final results as was first thought. For that reason, a window of 20 s could be perfectly chosen (at least in the best case found, for the Random Forest algorithm), as it would be easier to apply in a real-life environment. In this way, it could be possible to detect and classify each activity more finely, as well as making it possible to get these results in a shorter time since the start of the action.

For the aforementioned reasons, it is considered that the results can still be upgraded. Perhaps also with the application of algorithms purely focused on deep learning, such as CNN or LSTM, which are the two most widely used algorithms in recent years in HAR and which, apparently, are offering the best results nowadays. For this purpose, the exploration will continue in order to improve these results in the future, probably with the algorithms mentioned above, in search of the optimal model.

CRediT authorship contribution statement

Daniel Garcia-Gonzalez: Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Daniel Rivero:** Conceptualization, Formal analysis, Methodology, Supervision, Writing – review & editing. **Enrique Fernandez-Blanco:** Conceptualization, Formal analysis, Methodology, Supervision, Writing – review & editing. **Miguel R. Luaces:** Funding acquisition, Project administration, Resources, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We would like to thank the support given by CESGA and CITIC to execute the code related to this paper.

Funding

This research was partially funded by MCIN/AEI/10.13039/501100011033, NextGenerationEU/PRTR, FLATCITY-POC, Spain [grant number PDC2021-121239-C31]; MCIN/AEI/10.13039/501100011033 MAGIST, Spain [grant number PID2019-105221RB-C41]; Xunta de Galicia/FEDER-UE, Spain [grant numbers ED431G 2019/01, ED481A 2020/003, ED431C 2022/46, ED431C 2018/49 and ED431C 2021/53]. Funding for open access charge: Universidade da Coruña/CISUG.

Appendix. Supplementary materials

The complete dataset, as well as the scripts used to preprocess its data, are available online at http://lbd.udc.es/research/real-life-HAR-dataset. Similarly, these have also been uploaded to Mendeley Data [59]. Additionally, the code used to carry out the experiments in this work is available online at http://gitlab.lbd.org.es/dgarcia/new-machine-learning-har.

References

[1] E. Kim, S. Helal, D. Cook, Human activity recognition and pattern discovery, IEEE Pervasive Comput. 9 (1) (2009) 48–53.
[2] J.K. Aggarwal, L. Xia, Human activity recognition from 3d data: A review, Pattern Recognit. Lett. 48 (2014) 70–80.
[3] E. Soleimani, E. Nazerfard, Cross-subject transfer learning in human activity recognition systems using generative adversarial networks, Neurocomputing 426 (2021) 26–34.

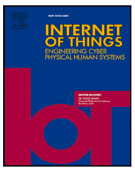
[4] C. Torres-Huitzil, A. Alvarez-Landero, Accelerometer-based human activity recognition in smartphones for healthcare services, in: Mobile Health, Springer, 2015, pp. 147–169.
[5] A. Zahin, R.Q. Hu, et al., Sensor-based human activity recognition for smart healthcare: A semi-supervised machine learning, in: International Conference on Artificial Intelligence for Communications and Networks, Springer, 2019, pp. 450–472.
[6] J. Manjarres, P. Narvaez, K. Gasser, W. Percybrooks, M. Pardo, Physical workload tracking using human activity recognition with wearable devices, Sensors 20 (1) (2020) 39.
[7] N. Zhu, T. Diethe, M. Camplani, L. Tao, A. Burrows, N. Twomey, D. Kaleshi, M. Mirmehdi, P. Flach, I. Craddock, Bridging e-health and the internet of things: The sphere project, IEEE Intell. Syst. 30 (4) (2015) 39–46.
[8] Y. Du, Y. Lim, Y. Tan, A novel human activity recognition and prediction in smart home based on interaction, Sensors 19 (20) (2019) 4474.
[9] O.D. Lara, M.A. Labrador, A survey on human activity recognition using wearable sensors, IEEE Commun. Surv. Tutor. 15 (3) (2012) 1192–1209.
[10] F. Demrozi, G. Pravadelli, A. Bihorac, P. Rashidi, Human activity recognition using inertial, physiological and environmental sensors: a comprehensive survey, 2020, arXiv preprint arXiv:2004.08821.
[11] M. Shoaib, S. Bosch, O. Incel, H. Scholten, P. Havinga, Complex human activity recognition using smartphone and wrist-worn motion sensors, Sensors 16 (4) (2016) 426.
[12] M.M. Hassan, M.Z. Uddin, A. Mohamed, A. Almogren, A robust human activity recognition system using smartphone sensors and deep learning, Future Gener. Comput. Syst. 81 (2018) 307–313.
[13] F. Attal, S. Mohammed, M. Dedabrishvili, F. Chamroukhi, L. Oukhellou, Y. Amirat, Physical human activity recognition using wearable sensors, Sensors 15 (12) (2015) 31314–31338.
[14] C. Xu, D. Chai, J. He, X. Zhang, S. Duan, Innohar: a deep neural network for complex human activity recognition, Ieee Access 7 (2019) 9893–9902.
[15] N. Lane, Y. Xu, H. lu, S. Hu, T. Choudhury, A. Campbell, F. Zhao, Enabling large-scale human activity inference on smartphones using community similarity networks (CSN), in: UbiComp'11 - Proceedings of the 2011 ACM Conference on Ubiquitous Computing, 2011, pp. 355–364.
[16] G. Weiss, J. Lockhart, The impact of personalization on smartphone-based activity recognition, in: AAAI Publications, Workshops At the Twenty-Sixth AAAI Conference on Artificial Intelligence, 2012.
[17] A. Ferrari, D. Micucci, M. Mobilio, P. Napoletano, On the personalization of classification models for human activity recognition, IEEE Access PP (2020) 1.
[18] R. Solis Castilla, A. Akbari, R. Jafari, B.J. Mortazavi, Using intelligent personal annotations to improve human activity recognition for movements in natural environments, IEEE J. Biomed. Health Inf. (2020) 1.
[19] D. Garcia-Gonzalez, D. Rivero, E. Fernandez-Blanco, M.R. Luaces, A public domain dataset for real-life human activity recognition using smartphone sensors, Sensors 20 (8) (2020) 2200.
[20] D. Anguita, A. Ghio, L. Oneto, X. Parra, J.L. Reyes-Ortiz, A public domain dataset for human activity recognition using smartphones, in: Esann, 2013.
[21] J.R. Kwapisz, G.M. Weiss, S.A. Moore, Activity recognition using cell phone accelerometers, ACM SigKDD Explor. Newsl. 12 (2) (2011) 74–82.
[22] A. Ignatov, Real-time human activity recognition from accelerometer data using convolutional neural networks, Appl. Soft Comput. 62 (2018) 915–922.
[23] N. Sikder, M.S. Chowdhury, A.S. Arif, A.-A. Nahid, Human activity recognition using multichannel convolutional neural network, in: 2019 5th Int. Conf. Adv. Electr. Eng, 2019.
[24] S. Seto, W. Zhang, Y. Zhou, Multivariate time series classification using dynamic time warping template selection for human activity recognition, in: 2015 IEEE Symposium Series on Computational Intelligence, IEEE, 2015, pp. 1399–1406.
[25] W. Sousa, E. Souto, J. Rodrigres, P. Sadarc, R. Jalali, K. El-Khatib, A comparative analysis of the impact of features on human activity recognition with smartphone sensors, in: Proceedings of the 23rd Brazillian Symposium on Multimedia and the Web, ACM, 2017, pp. 397–404.
[26] J. Figueiredo, G. Gordalina, P. Correia, G. Pires, L. Oliveira, R. Martinho, R. Rijo, P. Assuncao, A. Seco, R. Fonseca-Pinto, Recognition of human activity based on sparse data collected from smartphone sensors, in: 2019 IEEE 6th Portuguese Meeting on Bioengineering, ENBENG, IEEE, 2019, pp. 1–4.
[27] R.-A. Voicu, C. Dobre, L. Bajenaru, R.-I. Ciobanu, Human physical activity recognition using smartphone sensors, Sensors 19 (3) (2019) 458.
[28] Z. Chen, Q. Zhu, Y.C. Soh, L. Zhang, Robust human activity recognition using smartphone sensors via CT-PCA and online SVM, IEEE Trans. Ind. Inform. 13 (6) (2017) 3070–3080.
[29] C.A. Ronao, S.-B. Cho, Human activity recognition with smartphone sensors using deep learning neural networks, Expert Syst. Appl. 59 (2016) 235–244.
[30] F. Hernández, L.F. Suárez, J. Villamizar, M. Altuve, Human activity recognition on smartphones using a bidirectional LSTM network, in: 2019 XXII Symposium on Image, Signal Processing and Artificial Vision, STSIVA, IEEE, 2019, pp. 1–5.
[31] M. Badshah, Sensor-based human activity recognition using smartphones, 2019.
[32] S. Wan, L. Qi, X. Xu, C. Tong, Z. Gu, Deep learning models for real-time human activity recognition with smartphones, Mob. Netw. Appl. (2019) 1–13.
[33] W. Qi, H. Su, C. Yang, G. Ferrigno, E. De Momi, A. Aliverti, A fast and robust deep convolutional neural networks for complex human activity recognition using smartphone, Sensors 19 (17) (2019) 3731.
[34] Q. Teng, K. Wang, L. Zhang, J. He, The layer-wise training convolutional neural networks using local loss for sensor-based human activity recognition, IEEE Sens. J. 20 (13) (2020) 7265–7274.
[35] Y.E. Ustev, O. Durmaz Incel, C. Ersoy, User, device and orientation independent human activity recognition on mobile phones: Challenges and a proposal, in: Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication, ACM, 2013, pp. 1427–1436.
[36] V. Janko, N. Rešçiç, M. Mlakar, V. Drobnič, M. Gams, G. Slapničar, M. Gjoreski, J. Bizjak, M. Marinko, M. Luštrek, A new frontier for activity recognition: The sussex-huawei locomotion challenge, in: Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, 2018, pp. 1511–1520.
[37] S. Rosati, G. Balestra, M. Knaflitz, Comparison of different sets of features for human activity recognition by wearable sensors, Sensors 18 (12) (2018) 4189.
[38] D. Nielsen, Tree boosting with xgboost-why does xgboost win" every" machine learning competition?, (Master's thesis), NTNU, 2016.
[39] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al., Scikit-learn: Machine learning in python, J. Mach. Learn. Res. 12 (2011) 2825–2830.
[40] T. Chen, C. Guestrin, Xgboost: A scalable tree boosting system, in: Proceedings of the 22nd Acm Sigkdd International Conference on Knowledge Discovery and Data Mining, 2016, pp. 785–794.
[41] C. Cortes, V. Vapnik, Support-vector networks, Mach. Learn. 20 (3) (1995) 273–297.
[42] R. Rifkin, A. Klautau, In defense of one-vs-all classification, J. Mach. Learn. Res. 5 (2004) 101–141.
[43] J.R. Quinlan, Induction of decision trees, Mach. Learn. 1 (1) (1986) 81–106.
[44] J.R. Quinlan, C4. 5: Programs for Machine Learning, Elsevier, 2014.
[45] L. Breiman, J. Friedman, C.J. Stone, R.A. Olshen, Classification and Regression Trees, CRC Press, 1984.
[46] H. Taud, J. Mas, Multilayer perceptron (MLP), in: Geomatic Approaches for Modeling Land Change Scenarios, Springer, 2018, pp. 451–455.
[47] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014, arXiv preprint arXiv:1412.6980.
[48] I. Rish, et al., An empirical study of the naive Bayes classifier, in: IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence, Vol. 3, (22) 2001, pp. 41–46.
[49] K.P. Murphy, et al., Naive Bayes Classifiers, 18, (60) University of British Columbia, 2006.
[50] L.E. Peterson, K-nearest neighbor, Scholarpedia 4 (2) (2009) 1883.
[51] P. Cunningham, S.J. Delany, K-nearest neighbour classifiers–, 2020, arXiv preprint arXiv:2004.04523.
[52] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.
[53] S. Athey, J. Tibshirani, S. Wager, et al., Generalized random forests, Ann. Statist. 47 (2) (2019) 1148–1178.
[54] M. Hossin, M. Sulaiman, A review on evaluation metrics for data classification evaluations, Int. J. Data Min. Knowl. Manag. Process 5 (2) (2015) 1.
[55] M. Grandini, E. Bagli, G. Visani, Metrics for multi-class classification: an overview, 2020, arXiv preprint arXiv:2008.05756.
[56] M. Bekkar, H.K. Djemaa, T.A. Alitouche, Evaluation measures for models assessment over imbalanced data sets, J. Inf. Eng. Appl. 3 (10) (2013).
[57] R. Kohavi, et al., A study of cross-validation and bootstrap for accuracy estimation and model selection, in: Ijcai, Vol. 14, (2) Montreal, Canada, 1995, pp. 1137–1145.
[58] P. Liashchynskyi, P. Liashchynskyi, Grid search, random search, genetic algorithm: A big comparison for nas, 2019, arXiv preprint arXiv:1912.06059.
[59] D. Garcia-Gonzalez, D. Rivero, E. Fernandez-Blanco, M. R. Luaces, A public domain dataset for real-life human activity recognition using smartphone sensors, 2020, Mendeley Data, V2, Available online: https://data.mendeley.com/datasets/3xm88g6m6d/2.

Internet of Things 24 (2023) 100925

Contents lists available at ScienceDirect

Internet of Things

journal homepage: www.elsevier.com/locate/iot



Research article

Deep learning models for real-life human activity recognition from smartphone sensor data

Daniel Garcia-Gonzalez *, Daniel Rivero, Enrique Fernandez-Blanco, Miguel R. Luaces

Department of Computer Science and Information Technologies, University of A Coruna, CITIC, 15071 A Coruna, Spain

ARTICLE INFO

Dataset link: http://lbd.udc.es/research/real-life-HAR-dataset, https://data.mendeley.com/datasets/3xm88g6m6d/2, http://gitlab.lbd.org.es/dgarcia/deep-learning-models-har

Keywords:
HAR
CNN
LSTM
Real life
Smartphones
Sensors

ABSTRACT

Nowadays, the field of human activity recognition (HAR) is a remarkably hot topic within the scientific community. Given the low cost, ease of use and high accuracy of the sensors from different wearable devices and smartphones, more and more researchers are opting to do their bit in this area. However, until very recently, all the work carried out in this field was done in laboratory conditions, with very few similarities with our daily lives. This paper will focus on this new trend of integrating all the knowledge acquired so far into a real-life environment. Thus, a dataset already published following this philosophy was used. In this way, this work aims to be able to identify the different actions studied there. In order to perform this classification, this paper explores new designs and architectures for models inspired by the ones which have yielded the best results in the literature. More specifically, different configurations of Convolutional Neural Networks (CNN) and Long-Short Term Memory (LSTM) have been tested, but on real-life conditions instead of laboratory ones. It is worth mentioning that the hybrid models formed from these techniques yielded the best results, with a peak accuracy of 94.80% on the dataset used.

1. Introduction

Research in the field of human activity recognition (HAR) has shown stable progress in recent times. With the rise of wearable devices (mainly bracelets) and, above all, smartphones, it is feasible to think about the possibility of transferring the work carried out in this area to a large part of the world's population. To this end, sensor data from these devices are analysed in search of the classification of actions performed by a particular individual [1–3]. In this way, the applications of the work carried out in this field are multiple, from healthcare [4–6] to fitness [7,8], as well as more specific cases such as home automation [9]. For all these reasons, and thanks to the high portability and accuracy of the sensors of these devices, researchers find in HAR an incredibly tempting research opportunity [10–12].

However, there are some problems that need to be tackled. Firstly, there is the need to handle the temporality of the data, which is especially difficult when dealing with the large amount of information these devices produce. While it is true that previous works have made significant advances [13–15], there are still many activities whose relationship with prior data is still unclear. In addition, most of those works are carried out in a laboratory environment, with a series of pretty specific conditions that are not entirely feasible to transfer to real life. Although these works are helpful to get an approximate idea of the information collected and the actions performed, their outstanding results for the cases studied are very relative. One of the main issues is that the orientation and positioning of the device during the experimentation time can notably affect the final result [16]. Most researchers work with

* Corresponding author.
E-mail addresses: d.garcia2@udc.es (D. Garcia-Gonzalez), daniel.rivero@udc.es (D. Rivero), enrique.fernandez@udc.es (E. Fernandez-Blanco), miguel.luaces@udc.es (M.R. Luaces).

https://doi.org/10.1016/j.iot.2023.100925
Received 20 June 2023; Received in revised form 28 July 2023; Accepted 30 August 2023
Available online 9 September 2023

a smartphone around the waist [17] or using a wristband or bracelet [18]. Those developments could lead to remarkably reduced performances if they are applied to other datasets, especially real-life ones. In fact, specifically for the case of smartphones in everyday life, each person carries and uses them in a different way. That would highly affect the data provided by their sensors, as the previously mentioned orientation and positioning would vary considerably. Even when using different models of smartphones, differences in final measurements may occur [19]. Moreover, in addition to the latter, each individual has a series of physical peculiarities that could also influence the final result, even when using the device in the same way and performing the same action [20]. In fact, this problem has been studied for years, in order to personalise artificial intelligence models for large numbers of people [21,22].

For all those reasons, this work sought to help close that gap between all the acquired knowledge in HAR and its application in real life. To this end, a dataset already formed for this purpose was used [23]. The data was taken from the personal smartphone sensors of 19 individuals, ensuring that the actions performed were as similar as possible to those in their daily lives. To date, some work has been done using such a dataset, but using techniques related to traditional machine learning [24,25]. In this way, a study was carried out on the most suitable configurations for the deep learning algorithms that are yielding the best results in HAR: Convolutional Neural Networks (CNN) and Long-Short Term Memory models (LSTM) [26,27]. To this end, based on these techniques, new architectures have been designed to exploit this real-life data. In such a manner, it is hoped to obtain results that, if not ideal, are close to the optimum that is being sought.

Therefore, the key findings of this paper can be condensed into the following statements:

- An in-depth exploration of the most appropriate configurations for CNN and LSTM networks with real-world data in the HAR domain, using smartphone sensors.
- A new architecture to exploit HAR data taken from different smartphone sensors in a daily life environment, applying deep learning algorithms.
- The use of much more straightforward models than those used in previous real-life HAR domain work, without the need to manually compute its features.
- The improvement of the current results and approaches applying deep learning to a real-life HAR dataset.

The rest of the paper is organised as follows: Section 2 focuses on the evolution of HAR and the most relevant and recent work in this area, Section 3 describes the deep learning algorithms selected to perform all the related experiments, Section 4 depicts the preparation of the data and highlights the evaluation and validation techniques used, together with the proposed architectures and models, Section 5 discusses the main results of the work, and finally, Section 6 contains a series of conclusions and possible lines of future work.

2. Related work

This section is divided into two distinct parts. Firstly, in Section 2.1, a detailed comparison is made between the main datasets used by the scientific community and the one used in this paper. Then, in Section 2.2, a series of notable recent works that made use of those datasets are presented.

2.1. Smartphone datasets

Over the past decade, there have been numerous contributions to human activity recognition (HAR), leading to continuous advancements in the field. These developments have been supported by various datasets used as benchmarks to validate experiments and expand knowledge in the domain [28]. The data within these datasets originate from distinct wearable devices, such as activity wristbands, heart rate monitors, and more recently, smartphones. Among the latter, the UCI HAR dataset was the most widely used one by the scientific community [29]. It focused on activities like walking, sitting, and going upstairs, using data from the accelerometer and gyroscope of a specific smartphone. In addition, 30 participants were involved in the study, placing the smartphone on the left side of their waist. Each activity was performed for a few seconds to collect relevant features. Finally, the output data were sampled at a frequency of 50 Hz, and all the data collection took place in a laboratory setting.

The WISDM dataset [30] is another widely used dataset for human activity recognition, alongside the UCI HAR dataset. The activities included in this dataset are highly similar to those found in the UCI HAR one. Additionally, both datasets involve studying activities performed for several seconds. However, the main difference lies in the placement of the smartphone. In the case of the WISDM dataset, the smartphone was positioned in one of the front trouser pockets of each of the 29 participants who took part in the study. Unlike the UCI HAR dataset, the WISDM one only uses accelerometer data, sampling them at a fixed frequency of 20 Hz. As with the previous dataset, the data collection process for the WISDM dataset was also carried out under controlled laboratory conditions.

Similarly, the HHAR dataset [19] gathered data from eight smartphones and four smartwatches. The smartphones included four different models, while the smartwatches consisted of two distinct types. To collect the data, each participant had the smartphones securely placed in a pouch attached to their waist, and two smartwatches were worn on each wrist. The study involved only nine individuals as participants. As for the activities performed in this dataset, these were basic examples like walking, cycling, or running, but they were recorded over a more extended period of five minutes. Unlike the previous datasets mentioned, the data collection for HHAR did not take place in a laboratory setting. Instead, participants were instructed to follow specific routes within designated

Table 1
Comparison of the main HAR datasets based on smartphone sensor data, along with the real-life one used in this paper.

Dataset	Sensor(s) used	Activities recording time	Number of subjects	Sampling frequency	Device(s) used	Device placement	Environment
UCI HAR	Acc. and gyro.	Few seconds	30	50 Hz	1 smartphone	Left belt	Controlled
WISDM	Acc.	Few seconds	29	20 Hz	1 smartphone	Front pants leg pocket	Controlled
HHAR	Acc. and gyro.	5 min	9	Variable	8 smartphones and 4 smartwatches	Waist and wrist	Semi-controlled
UniMiB SHAR	Acc.	Fixed flow duration	30	50 Hz	1 smartphone	Trouser front pockets	Controlled
Real-life dataset	Acc., gyro., magn. and GPS	Free	19	Variable	19 personal smartphones	Free	Free

timeframes. Regarding the sampling rate, efforts were made to use the maximum value supported by Android. However, there was finally some variability in the sampling rates recorded during the study.

Another noteworthy dataset is the UniMiB SHAR one [31]. For data collection, a specific smartphone was positioned in the front trouser pocket of each of the 30 participants. Unlike some previous datasets, only accelerometer data were used, sampled at a fixed frequency of 50 Hz. Regarding the activities studied, these encompassed walking, standing up, running, jumping, and various others. The entire data collection process was conducted under controlled laboratory conditions, with researchers guiding the participants through specific activities.

As can be seen, the mentioned datasets exhibit significant differences among them. However, they all share a limitation in their data-gathering conditions. Specifically, the measuring devices were fixed to specific body parts, and the activities were performed in predetermined ways for set durations. To address this limitation, the current paper employed a real-life dataset in which the participants carried out the specified activities in a more natural and unrestricted way. In addition, in this dataset, data were collected from participants' personal smartphones, allowing them to carry out and measure the actions as they do regularly, with the smartphone positioned in their preferred habitual manner. In this way, Table 1 presents a summary of key information from each discussed dataset, comparing them to the one used in this work. Note that the abbreviations used in the table correspond to accelerometer (acc.), gyroscope (gyro.), and magnetometer (magn.). There, several distinctions are evident with the real-life dataset. Firstly, including the GPS sensor in data collection is a significant difference. This sensor's ability to detect speed and orientation could be beneficial for classifying the activities under study. Moreover, another noteworthy contrast is the variability in the sensors' sampling frequencies, which deviates from most datasets found in the literature. Unlike other measurement devices that can be set to a specific frequency value, smartphones lack this consistency throughout the data collection process, even if the highest value supported by the smartphone's operating system is set. This variability may not pose a problem for short and controlled data collection, as corrupted data will be minimal. However, for longer durations, such as in the chosen dataset, that needs to be considered, and appropriate data processing will be required. Furthermore, as for the number of participants and the use of different device models, higher variability would be preferred to ensure a more reliable representation of real-world contexts. In this way, the main difference lies in how the data were collected in a free environment with no specific conditions, making the proposals using other datasets less applicable to real-life scenarios.

2.2. Latest approaches

The introduction of wearable devices and widespread smartphone usage has significantly boosted the development of HAR. Since then, there has been a continuous rise in the diversity, improvement, and optimisation of artificial intelligence models that use this type of data. Following a chronological order, in the first years of the last decade, many works focused mainly on the exploitation of Support Vector Machines (SVM), as they seemed to be the models that yielded the best results for this subject [32,33]. Later, other possibilities began to be explored, as in the case of [34]. There, a comparison was carried out between other machine learning algorithms that also get good results in more fields, such as K-Nearest Neighbours (KNN), Multi-layer Perceptron (MLP) or the ones based on Bayes' Theorem. However, SVM still presented the best results to that date. In fact, later, another paper was also published in which an analysis of the principal machine learning algorithms used globally was also carried out, together with SVM [35]. Once again, SVM proved to be the most suitable for HAR. However, they also carried out a study on the influence of smartphone orientations on the returned data. The results showed that variations in this respect could significantly affect the final results. In the same way, work was also carried out on selecting the most convenient features to train these models, such as [36,37]. The results shown by these works proved that frequency-based parameters seemed to be the most suitable for HAR, having the highest percentage of correctness in the trained classifiers.

More recently, other works have been carried out in which deep learning approaches have been applied. Some of the most relevant ones are those of [38,39], for proving the high-grade results obtained by applying deep learning techniques, specifically Convolutional Neural Networks (CNN), on data from the HAR domain. In this way, a detailed comparison between different machine learning algorithms, combined with some custom features, is subsequently presented in [14]. The algorithms used were: CNN, Random Forest (RF) and KNN. Out of all the methods tested, CNN yielded significantly superior results. As a result, they also conducted an extensive analysis to determine the optimal architectures and configurations for this particular case. Since then,

although more traditional algorithms have continued to be used, deep learning became the most chosen option by researchers to solve this problem. In addition to the excellent results, the fact that no manual feature selection is required for some deep learning algorithms, like CNN, makes them even more attractive. Along the same lines, in addition to CNNs, models based on the Long Short-Term Memory (LSTM) technique began to be used to a large extent. That is due to the usual treatment of the temporality of the data in HAR. LSTMs are known for being models capable of including information from the past in their training, which is very positive for optimally classifying the data. Nonetheless, a drawback of these methods is their requirement for a substantial amount of data and time to attain appropriate training, which makes them quite different from CNNs. A few instances that demonstrate the application of this technique are represented in [26,40–42], where excellent results were obtained. In fact, in [41,42], a variant of this technique is presented, capable of achieving even better results than in its original form. This variant is called Bidirectional LSTM (Bi-LSTM) and can store data both from the past and the future (assuming that LSTMs usually store it unidirectionally from the past), adding it to the learning during the training of the models that implement it. Compared to its original form, the disadvantage is its higher complexity, as it has to study two time directions instead of one, leading to even longer training times. Given that, work in HAR is currently focused on the use of CNN and LSTM and their variants, in search of the most suitable model, as both techniques yield outstanding results in this field [15,43–45]. In any case, while both techniques are suitable for brief activities such as sitting or hand raising, it seems that, in general, there is a slight bias towards CNN over LSTM, given its speed and ease of implementation [46].

On the other hand, not all the research carried out was based primarily on the accelerometer and gyroscope sensors. Examples such as [47,48] prove the potential of other sensors, such as the magnetometer or GPS, with excellent results when added to their studies. More specifically, these sensors seem to work well with diverse types of long-themed activities, such as walking or running, as shown in these studies.

However, although all those works served to expand the knowledge in HAR, their advances would not be sufficient to be transferred to an application in everyday life. They have obtained their data in very controlled environments, with pretty specific instructions, so it is not feasible to expect the same good results if we transfer the proposed models to real life. While there are some works such as [49,50] that have tackled this problem, there is still a long way to go. In these cases, they achieved good results by transforming the smartphone's coordinate system into the Earth's one. Anyhow, their performance drops when changing the device's orientation device. In addition, they neither take into account the possible different placements of the smartphone, resulting in the same problems as in the other works.

Fortunately, very recent works have been published that seek to fill that gap between the laboratory models and their real-life applications [23,25]. Specifically, in [23], a dataset was published expressly focused on solving this problem, which will be used in this paper. In [25], the same dataset was also used, with an in-depth study of different machine learning algorithms and configurations, with a particularly significant improvement in the initial results. Other recent work using the same dataset, such as [24], is also worth mentioning, with a graph-theory approach based on Random Forest. However, those works still fall in manual feature engineering, oppositely to deep learning which pursues the automation of this part. Given the latest trends and advances made in HAR, algorithms such as CNN or LSTM should result in an optimisation of the final performance. In fact, there is already a paper using such algorithms on that dataset [51], but they do not mention how the data were preprocessed to feed the proposed models. Likewise, they also do not match the percentage of data per class they present in comparison with the original dataset. The class that should have the highest number of samples is presented as one of the classes with the fewest. Finally, they only use the accelerometer from the four original sensors. For all these reasons, it is impossible to reproduce their experiments as the specific conditions under which they were carried out are unknown. Hence, it is not possible to compare it with the present paper. Anyhow, it is considered that their approaches can be vastly improved, following a much more suitable methodology for such a dataset. Therefore, this paper aims to get the best model for the given dataset, in a quest to move towards that highly pursued real-life ideal.

3. Deep learning

Within the field of human activity recognition, the artificial intelligence algorithms used are very diverse. However, inside the deep learning area, two models stand out above the rest: Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM). Therefore, these two were used to compound the proposed models of this work. All their implementation was carried out entirely in Python, using the Tensorflow and Keras libraries [52,53]. Additionally, for the cases that implement LSTM, the cuDNN library [54] was used to take advantage of the speed of the GPUs available to carry out the experiments of this work.

3.1. Convolutional Neural Network

Convolutional Neural Networks (CNN) [55,56] are one of the most widely used models nowadays. Since the gradient modification carried out in [57], they have become a state-of-the-art model to extract information in almost any area of knowledge. These networks consist of a series of layers formed by a set of neurons or filters that receive different pieces of information as input. In this way, each filter is fed with different data from a sliding window or kernel over the initial signal or image. Unlike traditional neural networks, the weights of each of these filters are the same [58]. Therefore, the output ($X^{(l)}$) is the convolution of the input features ($X^{(l-1)}$) with a set of learnable filters ($W^{(l)}$), to which biases ($b^{(l)}$) are added. Finally, an activation function ($g^{(l)}$) is applied. The most commonly used one in HAR research (and the one selected for this paper) is the Rectified Linear Unit (ReLU), which returns

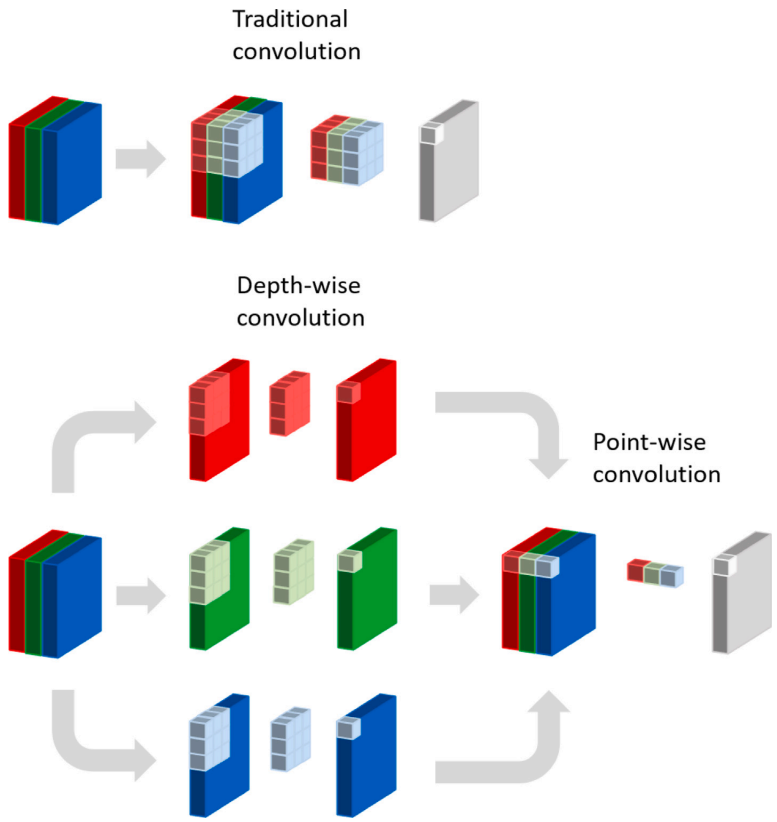


Fig. 1. Comparison of a traditional convolution and its equivalent Depth-wise Separable convolution.

0 when it receives a negative value or the value itself if it is positive. In such a way, the whole process results in the equation below (note that here the symbol "×" reflects a convolution):

$$X^{(l)} = g^{l}(X^{(l-1)} \times W^{(l)} + b^{(l)}) \tag{1}$$

That scheme can be repeated several times, where each layer will extract more features from the information already acquired in previous layers.

Moreover, for this paper, a MaxPooling layer was added after each convolutional layer. Such layers are used to down-sample the spatial dimensions of the input data, retaining the most relevant features. To do that, they divide the input data into a set of non-overlapping rectangular regions, outputting the maximum value of each one. In this case, these regions were implemented with a size of 2, as seen in many other works in HAR [59,60]. That makes the resulting models more robust to possible changes or distortions in the data and reduces the computational time required to train them [61].

Once the features have been extracted from the input matrix and transferred through each layer, they are fed into a fully connected perceptron (Dense layer). As for the final prediction and the probability vector $p_t = [p_{t_1}, p_{t_2}, \ldots, p_{t_k}] \in \mathbb{R}^k$, the softmax function was used, which converts the input values into a probability distribution, with values between 0 and 1. These input values would be the output values of the previously mentioned perceptron (z), giving rise to the following operation:

$$p_{t_i} = \frac{e^{z_i}}{\sum_{j=1}^{k} e^{z_j}} \tag{2}$$

Then, the results obtained would be returned directly, selecting the label with the highest probability after the softmax.

However, for this paper, it was opted to use the Depth-wise Separable Convolutional Neural Networks (DS-CNN) variant [62]. The choice of this variant is mainly due to its higher speed and efficiency compared to its original form. That is particularly appealing considering the large number of patterns to be used in this work. Moreover, it is starting to be applied in the most recent HAR studies,

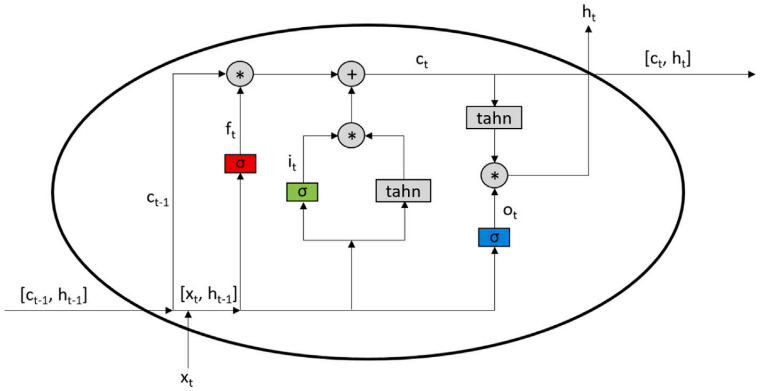


Fig. 2. Example of a LSTM unit, as shown in [40] (weight matrices and bias not displayed).

with excellent results [63]. Nevertheless, this modification is known for drastically reducing the requirements by significantly cutting down the number of necessary parameters [64]. To do that, the kernel is applied separately on each of the available channels of the input signal, rather than on all of them at once. This convolution would work the same way as the traditional one but using fewer features in each case. Then, the information obtained for each channel is combined through another convolution, projecting the resulting data onto a new feature map. The difference here is that the latter is carried out as a point-wise convolution (i.e. 1×1 convolution). As shown in Fig. 1, this ensues in fewer operations by integrating the data from the different channels. In this way, the computations are done with much less data and an equivalent outcome to traditional CNNs.

3.2. Long Short-Term Memory

Unlike their precursor, the Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM) networks [65] are a type of system capable of selectively remembering or forgetting data. To this end, they perform a series of slight modifications to the data they use, based on so-called cell states. For ease of understanding, an example of an LSTM unit is shown in Fig. 2. As can be seen, the typical LSTM network consists of a series of memory blocks called cells, between which two different states are transferred: the cell state (c) and the hidden state (h). In order for these blocks to be able to remember data, they implement a structure consisting of three different gates, as detailed below:

1. Forget Gate (the red one in Fig. 2). It removes all information that is no longer relevant for learning. To do that, the input data of the current time (x_t) and the hidden state of the previous cell (h_{t-1}) are multiplied by their correspondent weight matrix (W). Also, a bias (b) is added to the operation to get a better fit of the data. That constructs a regulatory filter, which is represented by the resulting sigmoidal function σ that follows:

$$f_t = \sigma\left(W_{xf} \times x_t + W_{hf} \times h_{t-1} + b_f\right) \tag{3}$$

   That would result in a value between 0 and 1. When multiplied by the cell state, it decides whether that information should be continued or not.
2. Input Gate (the green one in Fig. 2). It is responsible for adding relevant information to the model and filtering out any that may be redundant. To this end, another sigmoidal function is constructed, multiplied by a hyperbolic tangent one ($tanh$) that outputs the data between -1 and 1. In this way, the tanh function decides which data can be added later to the model, using a sum operation with the information of the forget gate. These functions are represented as follows:

$$i_t = \sigma\left(W_{xi} \times x_t + W_{hi} \times h_{t-1} + b_i\right) \tag{4}$$

$$c_t' = tanh\left(W_{hc} \times h_{t-1} + W_{xc} \times x_t + b_c\right) \tag{5}$$

3. Output Gate (the blue one in Fig. 2). This gate decides which outcome to keep, regarding that not all information flowing through the cell state may be adequate. In much the same way as before, sigmoidal and hyperbolic tangent functions are multiplied to filter these data. These functions are shown below:

$$o_t = \sigma\left(W_{xo} \times x_t + W_{ho} \times h_{t-1} + b_o\right) \tag{6}$$

$$c_t'' = tanh(c_t) \tag{7}$$

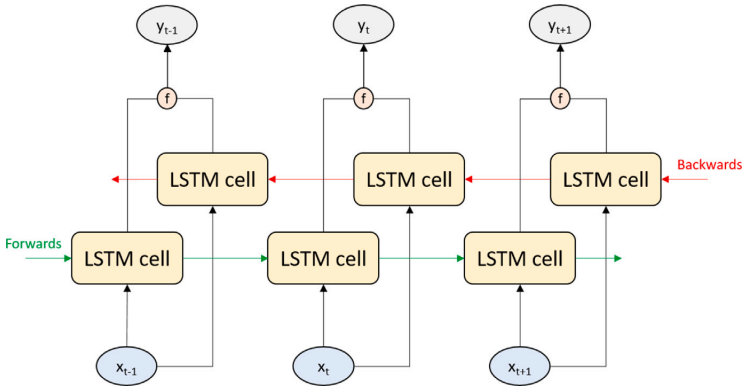


Fig. 3. Example of a Bi-LSTM network.

In this way, new cell and hidden states are obtained. Then, they are transferred to the next unit, repeating the process discussed above. These states are calculated as follows:

$$c_t = f_t \times c_{t-1} + i_t \times c'_t \tag{8}$$

$$h_t = o_t \times c''_t \tag{9}$$

As for the prediction and the probability vector $p_t = [p_{t_1}, p_{t_2}, \ldots, p_{t_k}] \in \mathbb{R}^k$, these are calculated from the resulting hidden state (h_t). This forms a softmax function (s), already commented in 3.1, which results in the following equation:

$$p_t = s(W_{hk} \times h_t + b_k) \tag{10}$$

Finally, the class label k_t is assigned to the one with the highest value in the vector of probabilities.

In the present work, in addition to traditional LSTMs, their bidirectional variant (Bi-LSTMs) was also used. This modification was formerly presented for the predecessor RNNs [66], but it can be used in the same way in a multitude of networks. The difference that characterises this variant is that it makes networks capable of storing data in both directions, usually by adding the future case (assuming that LSTMs usually store data unidirectionally from the past). This peculiarity, coupled with the fact that they are recently being used in the field with high-quality results, makes them a pretty attractive option for this work. In order to carry out this modification, two different LSTM models are trained, one that explores the input data (x) backwards and one that does the same but forwards, as shown in the example Bi-LSTM network in Fig. 3. During each model training, in each time step, a merging stage (f) is performed to mix the outputs obtained. That step can be carried out in different ways, but the most common and the one that was implemented in this work will be that of concatenation. In such a way, the output (y) of the first model is concatenated with the second model's. That ensures the latter can allow for both signal directions in the following time steps.

3.2.1. Hybrid models

A hybrid model refers to a model that combines different types of machine learning or deep learning algorithms. One of the most prominent examples in the HAR field is the combination of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) models. That is due to the peculiarities of each of them. On the one hand, CNNs try to reflect the spatial features of the data introduced into them. On the other hand, LSTM models look for these elements in the temporal section of the data that feeds them. Therefore, if the aim is to classify data with different signal distributions and time intervals, these algorithms combined could significantly improve performance.

Thus, in this work, those algorithms were combined using the variants discussed in the previous sections: Depth-wise Separable Convolutional Neural Networks (DS-CNN) and Bidirectional Long Short-Term Memory (Bi-LSTM) models. That led to the following hybrid models: (DS-CNN)-LSTM and (DS-CNN)-(Bi-LSTM). Hence, the spatial features extracted by the CNNs can be further exploited by the LSTMs, merging them with the temporal characteristics that can be derived by the latter. That should result in a substantial improvement in the final performance, although the corresponding execution times also increase with higher model complexity.

The way these models are assembled differs slightly from their individual cases. For this paper, the DS-CNN layers were always applied first, before the LSTM-based ones, to properly exploit the features as discussed in the previous paragraph. Thus, in hybrid models with more than one layer, a (DS-CNN)-(DS-CNN)-LSTM-LSTM style structure will be followed, without interleaving independent models. With this architecture, the outputs of the last MaxPool performed in the DS-CNNs will be the inputs of the LSTM-based models. Similarly, the outputs of the uttermost LSTM will be the outputs of the full hybrid model.

4. Methodology

This section explains in detail all the techniques and resources employed in this work. Firstly, Section 4.1 discusses how the data was processed and prepared for input into the subsequent artificial intelligence models. Then, Section 4.2 presents the different evaluation metrics used in this work. After that, Section 4.3 outlines various techniques to validate and improve the generalisation of the resulting models. Finally, Section 4.4 introduces the proposed models' architecture and configurations.

4.1. Data preparation

As previously mentioned, the dataset presented in [23] was used to carry out this work. Here, it is also worth noting that the data collection aimed to include individuals with diverse characteristics, encompassing physical diversity, smartphone usage patterns, and device models. Consequently, the study involved 19 participants, aged approximately 25 to 50 years, to ensure a wide range of behavioural patterns contributing to the development of future models. However, gender diversity is limited, with only two women among the participants. Nevertheless, participants' physical characteristics, habits, and preferences regarding smartphone use and positioning display considerable variation. Hence, while there is potential for improvement in variability, significant diversity remains present. As for the sensors used, there were four: accelerometer, gyroscope, magnetometer and GPS. Nonetheless, what makes the dataset most remarkable is that the individuals who took part in the data gathering were given almost total freedom, only having to use a custom Android app to start or stop the concerned activity. These activities were the following, as discussed in that work:

- Inactive: not carrying the smartphone on you at all.
- Active: any activity with movement, but without moving to a specific point in time. That would include activities such as: giving a lecture, cleaning the house or being at a concert.
- Walking: any trip made on foot, whether it is a regular walk or a jog.
- Driving: all journeys made via motorised transportation, without requiring the traveller to be the driver.

Concerning data preprocessing, almost the same dynamics as in the original work were followed, carrying out the following operations:

- Every outlier found in the GPS data was removed. That is the measurements that surpassed 0.2 decimal degrees on latitude and longitude increments between observations or 500 m in the case of altitude. Given its sampling rate, these measures seem unreal to accomplish for any living being.
- The first and last five seconds of each session were eliminated to avoid confusion during the training of the deep learning models. These time intervals correspond to the stages in which individuals picked up or put away the smartphone at the start or end of the action. Therefore, they were not relevant to the activity in question. Note that each session corresponds to an independent data gathering, from the moment when an individual begins an action until they finish it.
- The GPS data are largely sparse in each session, mainly because of the long waiting time between observations (>10 s). In the original paper, if there is more than one second between samples, the first one is replicated second by second, with a different timestamp, until this time difference does not prevail, in both directions. However, in the present work, since the sliding windows will move 10 s at a time, such replication was done every ten seconds instead of only one. Anyhow, every session without any GPS observations was discarded.
- Any session with substantial time gaps without observations (>5 s) was considered corrupt and, therefore, was ignored. Note that this does not include GPS data.

To prove the importance of this preprocessing, an example of how the different sensors behave in each of the specified activities can be seen in Fig. 4. Those examples correspond to the first 15 s of different sessions taken by one specific individual in the study. The selection of this time interval is due to the fact that it allows each activity and sensor behaviour to be illustrated easily on a single figure. Note that each subfigure displayed there corresponds to data taken while performing one of the four studied activities: inactive (*a*), active (*b*), walking (*c*) and driving (*d*). Also, to represent all the values on the same scale, the values corresponding to the GPS were divided by 10. Similarly, the values for the magnetometer and accelerometer were also divided by a value of 5 and 2, respectively. In this way, it is possible to easily observe the changes occurring in each sensor, for each specified action. For the accelerometer, gyroscope and magnetometer, those data are displayed for each of its three axes: *Acc_x*, *Acc_y* and *Acc_z*, *Gyro_x*, *Gyro_y* and *Gyro_z* and *Magn_x*, *Magn_y* and *Magn_z*, respectively. As can be seen in each subfigure, there are evident irregularities for the first seconds of each session, which only add noise to their interpretation. Concerning GPS, the increments in latitude, longitude and altitude from the last observation are displayed (*GPS_lat*, *GPS_long* and *GPS_alt*), together with speed, bearing and accuracy of the current measurement (*GPS_sp*, *GPS_bear* and *GPS_acc*). For this particular case, it is possible to see the small number of observations recorded, compared to the other sensors ones represented, highlighting the need to replicate them. Anyhow, distinct patterns can be discerned for each sensor, contingent on the executed activity. However, it is worth noting that, although considering those evident differences, this may not be the case with other types of actions. After all, although four activities are being studied, all of them encompass a variety of diverse actions, except for the "inactive" activity. For instance, washing dishes or teaching a class could exhibit significantly different signals despite both falling under the "active" activity category. Similarly, the same could apply to "walking" and "driving" activities. In the former case, differences might arise within the same activity if the session involves going

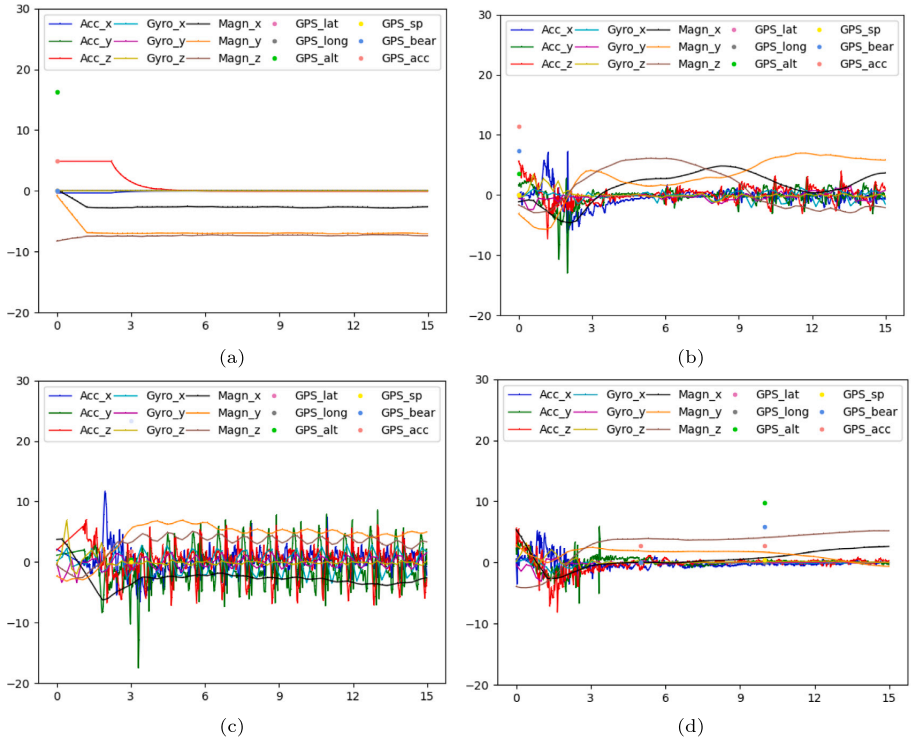


Fig. 4. Raw (scaled) data captured by a single individual's smartphone sensors, within the first 15 *s* of various example sessions, for each designated activity, being: (a) Inactive activity. (b) Active activity. (c) Walking activity. (d) Driving activity.

Table 2
Sensor's average sampling frequency and their respective standard deviation values for each activity measured (in Hz), after data preprocessing.

	Activity			
	Inactive	Active	Walking	Driving
Accelerometer	9.51	32.30	28.29	37.04
	±13.92	±23.49	±24.29	±20.94
Gyroscope	4.67	4.45	6.34	4.70
	±0.72	±1.45	±12.13	±2.62
Magnetometer	7.66	8.23	6.41	7.00
	±11.28	±12.38	±8.66	±9.94
GPS	0.01	0.03	0.07	0.13
	±0.09	±0.16	±0.26	±0.34

for a walk or jogging. As for the latter, substantial differences could appear depending on whether the individual is driving their own car or taking public transportation. In this way, although it is believed that these trends could also be present in other data sessions, they should be approached with a measure of caution.

Following prior data preprocessing, it was decided to apply 30, 60 and 90-s sliding windows, with an overlap of 20, 50 and 80 s, respectively (moving the window 10 s at a time). The selection of these time intervals and no others is due to the performances observed in other works using the same dataset, such as [25]. It is considered that, with this selection, it is possible to see the general behaviour of the proposed models to see if there is any trend in the results towards larger or smaller window sizes. Furthermore, it can be considered a reasonable amount of time given the long-themed nature of the activities included in this dataset, without being too broad or limited. In fact, if it were, it would not be possible to separate and identify the actions correctly, and there could be periods of inactivity in an activity that is supposed to be entirely associated with "walking", for example, by going for a random walk and stopping at a traffic light.

Table 3
Number of available patterns and their distribution among the above-mentioned activities, for an overlap of 10 s less than each full window size.

Window size	Activity				
	Inactive	Active	Walking	Driving	Overall
30	21,152	13,778	7823	6109	48,862
	43%	28%	16%	13%	
60	20,836	13,487	7204	5716	47,243
	44%	29%	15%	12%	
90	20,486	13,223	6732	5439	45,880
	44%	29%	15%	12%	

However, in order to feed the prepared data into the deep learning models, it was necessary to perform another series of operations. With the considerable differences observed in the frequency of each sensor for each activity studied, it is unattainable to transfer the data directly to the model. Table 2 shows the values of this frequency for each sensor and activity performed after preparing the data as detailed above. As can be seen, there are very abrupt cases, especially with the accelerometer, since it drastically changes its sampling frequency when any movement or vibration is detected. For that reason, for example, the frequency is much higher for the case of "walking" compared to "inactive". Likewise, there is a substantial change in the gyroscope's frequency for the "walking" activity, compared to the rest. Anyhow, the differences are slighter than with the accelerometer, as it focuses on changes in the smartphone's orientation and not on any movement or vibration that may exist. As for the magnetometer and GPS, their variations are more arbitrary, although there is a tendency to get more GPS measurements as the travelling speed increases. In addition, each smartphone may have slight differences for the same observation [19], which may also affect this sampling rate. In fact, some of these differences were thought to be mainly due to the behaviour of some sensors in moments of high or low movement [23].

Nonetheless, upon further exploration, those changes seem to be also somewhat arbitrary. Indeed, a peculiar behaviour was observed for the accelerometer, gyroscope and magnetometer. The data provided by these sensors are generally given either every 20 ms or every 200 ms. In the few cases in which this is not the case, it is by a slight difference, with a frequency closer to 10 ms, or approximately 180 ms, depending on the case. Moreover, these differences do not seem to correspond to any specific individual or activity, as they may be noted even during the same data collection session from a specific one. Therefore, the hypothesis is, apart from the movements and vibrations commented on before, that there could be some settings on the individuals' smartphones affecting these sampling rates. For example, the trigger of automatic battery saving when reaching a certain threshold, even if all permissions were activated for the data collection Android app used for such work.

For all those reasons, it was necessary to transform the data so that each sensor had the same sampling rate across all associated observations. For that purpose, one possibility could be to apply linear interpolations, as they were the most commonly used operation in the field when the context required it [15,67,68]. Only in [19] was some exploration with other more complex interpolations like the quadratic or cubic ones, but without an in-depth investigation. Regarding the values of this sampling rate, no clear consensus has been found in the scientific community for smartphone sensors. Some researchers say that around 2–3 Hz is the most appropriate [69]. Others prefer to set it between 0–15 Hz [11], or even up to 50 Hz in some situations [70]. Anyhow, they all were studies carried out in controlled laboratory environments and without handling sensors with frequencies as different as GPS's. Therefore, given the distinct and scarce approaches in the HAR literature, an experimental solution was chosen. The present work deals with a singular case in which sampling rates stabilise around a value every 20 ms or 200 ms (50 Hz and 5 Hz, respectively). Given that, it was considered that the most appropriate approach would be to fix this frequency at 5 Hz, always selecting the closest real value to each time instant, every 200 ms. In this way, the observations to be introduced later in the proposed models would be completely real, without the modification they could suffer when going through a traditional interpolation. In addition, the temporal error that could be accumulated for cases that do not strictly follow these dynamics would be small, given the little and unusual changes that occur at these frequencies. Thus, when data are given every 20 ms, only those observations that correspond to the time instants that occur every 200 ms would be selected, ignoring the rest. When the data are given every 200 ms, it would be only necessary to pick those observations that correspond to each 200 ms advance in time. Although it is true that with this approach a considerable amount of existing patterns from the original dataset are eliminated, it is considered the most appropriate choice for the problem to be solved, given the circumstances. Concerning GPS, although it is not particularly affected by that problem, it does have a very high and irregular frequency, compared to the rest of the sensors used. Hence, following the same idea as before, it was decided to set the frequency at 0.1 Hz (one value every 10 s). That is possible thanks to the replication carried out before, detailed at the beginning of this section.

After carrying out all the steps discussed above, the total number of patterns is shown in Table 3 for each proposed window size. As can be seen, there is a clear imbalance towards the "inactive" activity, probably due to the ease of collecting this type of data compared to the rest. Even so, it is considered that the overall number of patterns in each class is sufficient to perform a satisfactory classification, as seen already in other works using the same dataset [23,25].

4.2. Evaluation metrics

In order to easily view and evaluate each model classification, the most widely used option by the scientific community is the confusion matrix. From this matrix, many metrics can be extracted. Since this paper deals with a multi-class case, a *one-versus-all*

strategy was followed to reduce them to a binary type. Thus, each class is analysed separately comparing it with the rest together. From there, some of the most elementary metrics that can be extracted are the number of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN), for a given class. In addition, from these values, other representative metrics can also be calculated, such as precision, recall, accuracy, and F_1-score [71]. Among those metrics, the most commonly used one to measure the performance of a test model is accuracy. For its calculus, the percentage of correctly identified cases out of the total is measured using the following formula:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{11}$$

However, there are cases where the latter metric can lead to some bias, especially on an imbalanced dataset. For this very reason, when there is a considerable imbalance in the data, the F_1-score [72] metric is usually also shown. To measure it, precision and recall are combined in a harmonic mean, with precision being the ratio of the TP to all cases labelled as positive by the model (TP + FP), while recall refers to the division of the same TP by the total number of positives in the ground truth (TP + FN). Given that, its formula would be as follows:

$$F_1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{12}$$

For this paper, the results will always be evaluated based on the accuracy metric to be able to compare it properly with the rest of the works carried out on the same dataset. Anyhow, given the clear imbalance present in the data, the F_1-score will also be shown for the most representative cases to get a closer look at the actual performance of the models. Since this paper presents a multi-class problem, an averaging process is necessary to get the overall value of this metric. Given that, the *macro* strategy, which returns the mean value obtained by computing the metric for each label individually, was followed.

4.3. Validation techniques

Cross-validation [73] is regarded as one of the most reliable methods for validation. Before data is fed directly into the model, it undergoes a process of division into training and testing. In such a way, the model will be able to use one subset of the data only for training and another for testing, the latter a priori unknown to the model. The most prevalent way to carry out this division is through *k-fold cross-validation*. This technique consists of partitioning the original dataset into a number k of subsets of equal size. One of these partitions will form the test set of the model, while the rest will be used for training. Then, the procedure will be reiterated k times, ensuring that each subset has been designated as a test set once. Finally, after feeding the models with each partition, the outcomes are averaged, and the pertinent metrics are computed. In such a manner, the random component of splitting the original set in training and testing only once, which could lead to unreliable results, is largely avoided. For this paper, a modified version of this approach, known as *stratified k-fold cross-validation*, was employed. This alternative aims to ensure the same percentages of class representativeness in all the partitions carried out. Hence, it can mitigate the influences of the present imbalance in the initial dataset. All things considered, in this work, a stratified 10-fold was applied, splitting the data into training and test, with a distribution of 90%, and 10% for each subset formed, respectively.

Nonetheless, with that approach, one of the most common issues in any work related to machine learning may arise. That is the overfitting problem [74,75]. A model is said to be overfitting a dataset when instead of extracting information from patterns, it mainly memorises them. This problem is even bigger in deep learning because of the increase in the number of weight drives, which considerably expands the memorising capacity of the network. To alleviate this issue as much as possible, for each training set, 11.11% of the data included therein has been assigned to a validation set. Thus, the general distribution for each fold would be 80% training, 10% validation and 10% test. In this way, during training phase, the model's performance is tested against the validation set. That yields a loss value that evaluates the classification at that point, based on the sum of the errors obtained for each sample. The lower the value, the better the classification, a priori. However, this value may reach a point where the improvements are almost imperceptible, leading the model to a clear case of overfitting. To avoid that, an early-stopping function was applied to each model [76]. This kind of function seeks to interrupt training when that point is reached, returning the best weights obtained by the model so far. For this work, training will be interrupted when the model has not improved the last best loss value for 20 iterations (out of a fixed total of 100 iterations). In this way, although it is impossible to guarantee that training does not stop at a local minimum, the generalisation capacity of the model may be improved.

In addition to the above, a Dropout layer was also set up for each proposed model. This layer dumps part of the outputs, forcing the model to rely on other connections. In this way, the generalisability of the model increases considerably. This layer was applied for 50% of the input units and placed just before the final output. Both the quantity and the placement selected for these layers are those commonly used in the literature [77].

4.4. Proposed approach

As discussed before, the algorithms selected for this work were Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) models, given their outstanding results in the field. Specifically, for the CNN case, Depth-wise Separable Convolutional Neural Networks (DS-CNN) were used, to speed up the experiments without affecting performance. On the other hand, in the case of LSTM, its bidirectional variant (Bi-LSTM) was also used, as it was considered that it could provide good results for the problem to be solved, given its recent applications.

Table 4
Training hyperparameters.

Hyperparameter	Value
Batch size	32
Layers	[1, 2]
Neurons (or CNN filters)	[16, 32, 64]
Kernel (CNN only)	[3, 5, 7]
Padding (CNN only)	Same
Activation	ReLU
Optimiser	ADAM
Loss	Cross-entropy
Iterations	≤100
Early stopping	20

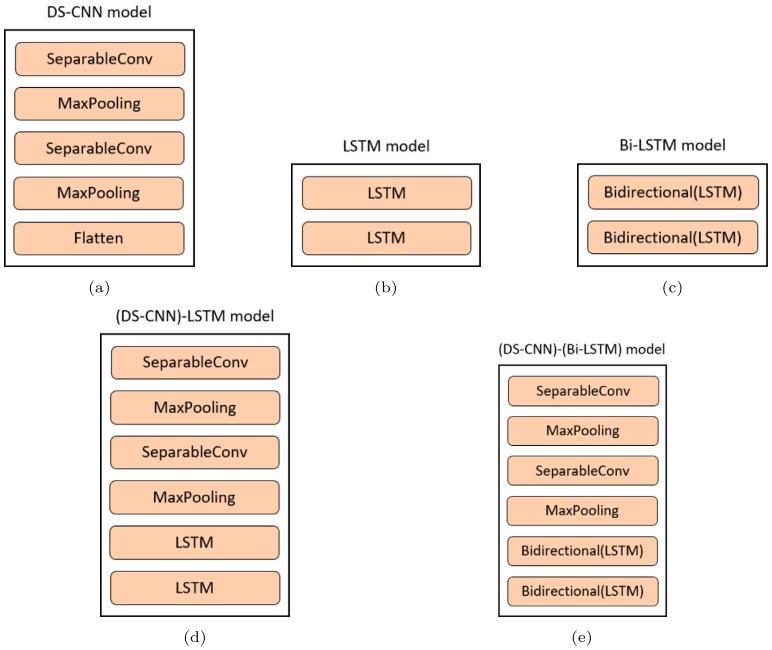


Fig. 5. Implementation of each individual algorithm used, for the case of having a number of layers equal to two, being: (a) DS-CNN model. (b) LSTM model. (c) Bi-LSTM model. (d) (DS-CNN)-LSTM model. (e) (DS-CNN)-(Bi-LSTM) model.

Nonetheless, given the prominent use of those algorithms simultaneously in the literature, it was also decided to do the same for this paper. In addition to using those algorithms individually, they were also combined, resulting in (DS-CNN)-LSTM and (DS-CNN)-(Bi-LSTM) hybrid models, as discussed in Section 3.2.1. In this way, it is possible to make a comparison between all proposed models, observing in detail the advantages and disadvantages of each one.

Concerning the hyperparameters used for each of those models, they are shown in Table 4. The batch size was set to 32. After a few preliminary explorations with higher values (64, 128, 256, 512 and even 1024), this was the best trade-off between efficiency and accuracy. Consequently, considering that the changes in classification accuracy were negligible between 32 and the rest, it was decided to discard them. Regarding the rest of the hyperparameters, note that in the hybrid models, a layer number of one would correspond to a total of two layers (one per individual network). Likewise, two layers would result in a total of four layers. For example, when we join a DS-CNN with an LSTM with a layer number of 2, it would look like this: (DS-CNN)-(DS-CNN)-LSTM-LSTM. That is two layers for DS-CNN and two for LSTM, with the DS-CNN layers always going before the LSTM ones, as said in Section 3.2.1. Therefore, it was not considered to explore with more layers, as this would remarkably increase the complexity of the models. As for the neurons, a similar combination as in [59] was used, but without going that further. Concerning kernel size, once again, the variety used in [59], with fewer options, was the one applied. In addition, the padding of the DS-CNNs was set to "same" to be able to perform convolutions of the desired size. This parameter allows the algorithm to fill with zeros evenly around the signal,

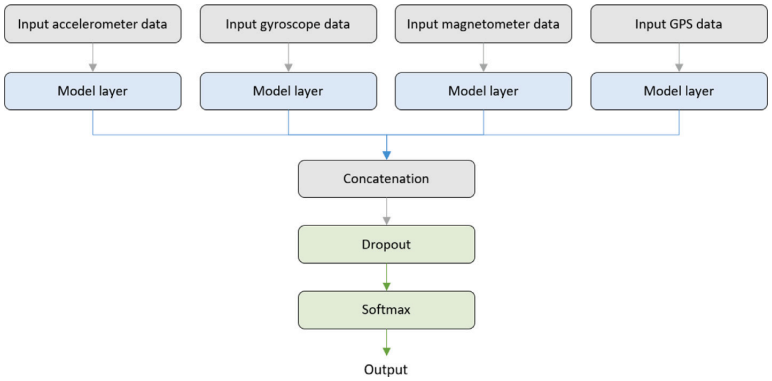


Fig. 6. General architecture of the whole model used to carry out the experiments.

allowing the input dimensions to match the output dimensions. As for the activation, optimiser and loss functions, the most widely used in the literature were the ones applied: Rectified Linear Unit (ReLU), Adaptive Moment Estimation (ADAM) and cross-entropy, respectively. Finally, a number of 100 iterations was set, with an early stop of 20 if the validation loss did not improve. Thus, the model gets enough time to recognise the patterns while avoiding overfitting. Any other parameters that might be present were kept by default.

With respect to the architecture followed to implement those models, you may see the implementation followed for each algorithm individually in Fig. 5. Note that, in the case represented there, all models are shown with two layers to visualise the most complex versions of each kind. There, each subfigure shows a different algorithm, from top (input layer) to bottom (output layer). In the case of DS-CNN (*a*), each model is formed by a convolutional layer (SeparableConv), followed by a MaxPooling layer, as seen in the rest of CNN works in HAR. Also, in the end, regardless of the previous number of layers, a Flatten layer is added. That is to format the resulting feature map to allow for being consumed by subsequent layers. To do that, the input data is converted into a one-dimensional array. Concerning LSTMs (*b*), they are formed only by the layer that implements this algorithm, in this case following cuDNN's implementation to accelerate its training time. For its bidirectional variant (*c*), this layer is enclosed by the wrapper that adds this functionality (Bidirectional()). Finally, for the hybrid models, (DS-CNN)-LSTM (*d*) and (DS-CNN)-(Bi-LSTM) (*e*), the previous cases are combined. In this way, the layers corresponding to DS-CNN are added first, which will subsequently feed those based on LSTM. As the resulting DS-CNN feature maps can already be consumed directly by LSTM, it is not necessary to add a Flatten layer for these hybrid cases.

In such wise, the entire final model consists of four different inputs, one for each sensor used in the dataset, as shown in Fig. 6. In this manner, it is possible to use each sensor's data, while avoiding dealing with the peculiarities of each one by focusing on one particular sensor at a time. Thus, the data measured by each sensor is transferred through the CNN and LSTM networks, as appropriate. As for the model layers represented there, the same one is always applied for each of the four branches. For example, if a DS-CNN model is used for the accelerometer data, the same is implemented for the rest of the sensors. Note that each of those networks would correspond to the models shown in Fig. 5. That results in different outputs for each sensor, depending on the particularities encountered in each case. Then, in order to be able to combine everything in the same model, the outputs of each of these branches are concatenated in a single layer. After that, to avoid overfitting and increase the generalisation capacity of the implemented models, a Dropout layer was added, affecting 50% of the input units, as commented in Section 4.3. Finally, a Softmax layer was set to obtain the desired output with the four activities to be studied.

5. Results and discussion

This section shows all the results obtained from the proposed experiments. On the one hand, Section 5.1 indicates the performance of each experiment, with its corresponding outcomes. Then, Section 5.2 introduces a series of comments and observations on the obtained results.

5.1. Results

With the models discussed in Section 4.4, the results shown in Tables 5, 6 and 7 were obtained. There, the values corresponding to the average accuracy of every possible combination of hyperparameters are represented, with their standard deviation below. Those hyperparameter combinations correspond to the number of layers (L), the number of neurons in each layer (N) and the kernel size (K). Moreover, each table corresponds to a specific sliding window size: 30, 60 and 90 s, respectively, as pointed out previously.

Table 5
Accuracy results obtained detailed for a window size of 30 s.

		N = 16		N = 32		N = 64	
		L = 1	L = 2	L = 1	L = 2	L = 1	L = 2
DS-CNN	K = 3	89.21% ±8.17%	88.02% ±10.37%	87.95% ±9.26%	88.30% ±9.66%	89.10% ±7.06%	89.15% ±8.95%
	K = 5	88.64% ±7.80%	88.39% ±9.22%	87.59% ±8.66%	88.68% ±8.95%	88.73% ±7.85%	88.08% ±9.54%
	K = 7	88.33% ±8.01%	87.95% ±8.70%	89.23% ±7.36%	88.57% ±9.52%	88.00% ±8.68%	87.50% ±11.53%
LSTM		90.99% ±6.99%	91.07% ±5.88%	**93.15% ± 4.52%**	91.49% ±5.58%	91.91% ±6.63%	90.06% ±7.36%
Bi-LSTM		91.91% ±4.76%	90.33% ±8.21%	89.46% ±8.72%	90.99% ±6.73%	91.22% ±5.86%	90.09% ±8.85%
DS-CNN-LSTM	K = 3	92.15% ±7.09%	90.47% ±7.71%	91.15% ±7.54%	90.38% ±8.91%	91.85% ±6.34%	91.20% ±7.88%
	K = 5	91.58% ±7.22%	90.04% ±8.2%	91.07% ±7.79%	88.96% ±10.05%	91.64% ±6.73%	92.46% ±5.61%
	K = 7	91.75% ±5.62%	90.84% ±7.04%	91.76% ±6.21%	91.23% ±6.89%	90.36% ±8.39%	89.05% ±9.43%
DS-CNN-Bi-LSTM	K = 3	91.88% ±7.24%	90.40% ±7.29%	91.64% ±7.10%	90.47% ±8.26%	91.49% ±7.26%	90.64% ±8.70%
	K = 5	92.56% ±5.84%	90.31% ±9.21%	90.92% ±7.20%	89.87% ±8.20%	91.77% ±6.94%	91.24% ±8.48%
	K = 7	90.63% ±8.16%	91.63% ±6.48%	90.99% ±7.47%	90.45% ±7.35%	91.32% ±7.42%	90.51% ±8.02%

Table 6
Accuracy results obtained detailed for a window size of 60 s.

		N = 16		N = 32		N = 64	
		L = 1	L = 2	L = 1	L = 2	L = 1	L = 2
DS-CNN	K = 3	89.64% ±7.67%	89.38% ±9.00%	88.35% ±8.74%	89.86% ±9.51%	89.13% ±8.00%	88.74% ±9.39%
	K = 5	89.18% ±7.99%	89.62% ±8.72%	89.38% ±8.00%	88.94% ±9.57%	88.31% ±8.88%	89.98% ±8.74%
	K = 7	89.64% ±8.25%	89.10% ±10.22%	88.31% ±8.77%	89.80% ±8.56%	88.29% ±8.38%	89.38% ±9.08%
LSTM		91.14% ±8.36%	92.89% ±5.07%	92.15% ±7.26%	93.11% ±6.50%	92.39% ±7.21%	92.99% ±5.04%
Bi-LSTM		91.70% ±8.02%	92.02% ±7.22%	92.75% ±6.58%	92.17% ±7.07%	92.61% ±6.04%	91.89% ±7.30%
DS-CNN-LSTM	K = 3	92.64% ±6.42%	92.71% ±6.87%	93.14% ±6.71%	92.83% ±6.04%	93.09% ±6.14%	92.89% ±5.97%
	K = 5	93.04% ±5.65%	91.55% ±8.37%	93.01% ±7.30%	91.57% ±9.27%	92.55% ±6.47%	91.72% ±7.66%
	K = 7	**93.49% ± 5.34%**	92.09% ±7.71%	93.24% ±6.39%	92.33% ±7.67%	92.42% ±9.11%	92.54% ±7.08%
DS-CNN-Bi-LSTM	K = 3	92.63% ±7.25%	91.68% ±7.50%	92.81% ±6.95%	91.66% ±8.16%	91.64% ±9.05%	90.80% ±8.98%
	K = 5	92.37% ±7.17%	92.32% ±7.08%	92.69% ±5.72%	91.91% ±6.57%	92.38% ±7.67%	91.37% ±8.09%
	K = 7	92.93% ±7.11%	92.48% ±6.57%	92.69% ±7.59%	90.95% ±8.73%	91.89% ±7.18%	90.39% ±8.99%

As can be seen, the accuracies obtained, in general, are higher than those obtained in other works on the same dataset [23,25]. Given the results, the use of deep learning algorithms can be considered one of the best options to exploit such data, especially those based on CNN and LSTM, as proved in recent works in the literature. As for the performance of the models depending on the particular algorithm selected and a specific set of hyperparameters, there do not seem to be very noticeable differences. Their results are objectively constant for each algorithm in the three tables. However, some contrasts are worth noting. Firstly, there are some differences if we look at the values obtained by each algorithm independently. In general, the best results are obtained by the hybrid algorithms mentioned above, closely followed by those based on LSTM, but worsening slightly when only DS-CNN is used. Considering these results, for the dataset used, it appears that the LSTM-based algorithms perform better than the CNN-based algorithms. That seems to indicate that the time component of the signals is more important than the features themselves. To validate these differences, a Tukey test was performed between each group of results, for each algorithm and window size indicated. The

Table 7
Accuracy results obtained detailed for a window size of 90 s.

		N = 16		N = 32		N = 64	
		L = 1	L = 2	L = 1	L = 2	L = 1	L = 2
DS-CNN	K = 3	90.27% ±7.74%	90.32% ±8.54%	89.73% ±7.99%	90.31% ±8.48%	89.76% ±8.09%	89.33% ±9.88%
	K = 5	89.54% ±7.99%	89.51% ±10.5%	89.65% ±8.54%	89.27% ±9.47%	89.28% ±10.09%	89.78% ±9.56%
	K = 7	90.70% ±7.29%	89.25% ±9.40%	88.43% ±10.38%	89.50% ±9.38%	89.89% ±8.17%	89.17% ±8.86%
LSTM		91.88% ±7.14%	91.92% ±6.81%	91.69% ±8.82%	93.52% ±5.59%	92.15% ±7.07%	91.28% ±8.55%
Bi-LSTM		93.09% ±5.10%	92.35% ±6.62%	91.60% ±8.25%	92.12% ±8.23%	91.82% ±8.07%	92.47% ±8.67%
DS-CNN-LSTM	K = 3	93.20% ±5.31%	92.65% ±8.44%	93.60% ±5.96%	92.68% ±7.84%	93.34% ±5.67%	92.77% ±8.30%
	K = 5	93.57% ±5.14%	92.86% ±8.10%	94.78% ±4.64%	92.62% ±9.16%	92.93% ±6.91%	91.48% ±9.28%
	K = 7	93.62% ±7.10%	91.56% ±10.29%	94.19% ±5.72%	92.72% ±7.80%	**94.80%** **±4.09%**	90.88% ±9.79%
DS-CNN- Bi-LSTM	K = 3	93.68% ±6.63%	93.37% ±7.39%	92.72% ±7.36%	91.26% ±8.81%	93.98% ±5.31%	90.73% ±9.33%
	K = 5	92.93% ±7.07%	93.10% ±6.59%	92.05% ±8.90%	93.32% ±7.50%	92.80% ±7.55%	93.04% ±6.71%
	K = 7	92.80% ±8.09%	94.16% ±5.06%	93.51% ±6.28%	93.37% ±7.68%	93.26% ±6.67%	92.58% ±7.99%

Table 8
Overall accuracy results obtained for each window size.

	Window size		
	30	60	90
DS-CNN	88.41% ±8.93%	89.17% ±8.79%	89.65% ±8.97%
LSTM	91.44% ±6.31%	92.44% ±6.72%	92.07% ±7.44%
Bi-LSTM	90.67% ±7.39%	92.19% ±7.07%	92.24% ±7.61%
(DS-CNN)-LSTM	91.00% ±7.63%	92.60% ±7.12%	**93.01%** **± 7.49%**
(DS-CNN)-(Bi-LSTM)	91.04% ±7.66%	91.98% ±7.66%	92.93% ±7.40%

outcomes of these tests are shown in Fig. 7. Note that the widths of the confidence intervals are plotted at 95%, calculated from Tukey's Q value, by default. As previously mentioned, every table showed significant differences in the performance of the DS-CNN algorithm and any of the other four models. However, between the hybrid models and those based solely on LSTM, there appears to be statistical equivalence. Similarly, for the groups of results concerning each individual hyperparameter, after applying another Tukey test, no statistical differences were observed between them.

Moreover, it is also possible to observe how those values change notably depending on the selected window size. Table 8 shows the mean values of the accuracy obtained for each selected algorithm and window size, in a general way, showing in small, below each value, its standard deviation. Likewise, Table 9 shows the mean F_1-score values. As can be seen, these values are higher when the window size is larger (60 and 90 s) compared to those corresponding to a window size of 30 s. In the same way as before, another Tukey test was performed for the selected window sizes (30 and 90, 30 and 60 and 60 and 90), with their corresponding detailed performances from the tables above. The results of this test can be seen in Fig. 8. Only in the case of 60 and 90 s was the p-value greater than 0.1, so no statistically significant difference was found between both sets. However, for the 30-s case, there is a statistical difference with either of the other two values. That reaffirms the hypothesis already shown in previous works such as [25], where larger window sizes obtained superior results. In fact, if we go at the very nature of the activities studied in the dataset used, they have a long-themed character, so it is logical to think that longer time intervals positively affect the classification of their corresponding data.

All things considered, a peak performance of 94.80% accuracy and 94.27% F_1-score is achieved, corresponding to the (DS-CNN)-(LSTM) model, with a window size of 90 s, a single layer, 64 neurons and a kernel size of 7. The average confusion matrix corresponding to this case can be seen in Table 10, along with its particular metrics (recall, precision and accuracy). As can be seen, the model is able to classify any of the four activities with great accuracy, although there are slight problems with the correct identification of the "active" class. This activity is quite fuzzy, as it can accommodate actions where there may be periods of

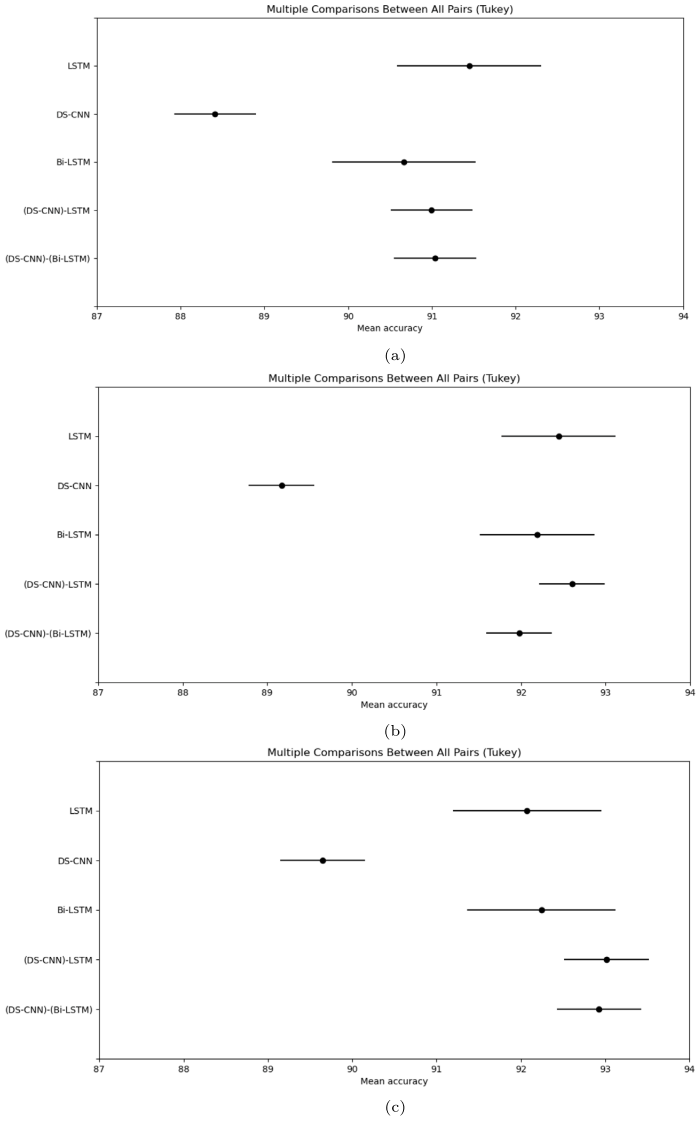


Fig. 7. Tukey test results for each group of accuracy values referring to each implemented algorithm, for each selected window size: (a) 30 s. (b) 60 s. (c) 90 s.

inactivity or where the individual is walking, which could lead to misclassification into these classes. An example of an action that could fall into this class and could be easily confused would be giving a lecture. This action alternates between times when the person may be walking (moving around the classroom) or sitting at the computer. In the first case, that walking moment could result in classifying samples of "active" as "walking". In the second case, sitting without moving at all is very similar to not having a cell phone on you, which could lead to misclassifying this action as "inactive". Even with the "driving" activity there could be confusion, since sitting in the car waiting at a red light, without moving, could also be difficult to classify correctly, even taking into account the vibrations of motor vehicles. Therefore, it is considered that, despite the discrepancies observed, the model is capable

Table 9
Overall F_1-score results obtained for each window size.

	Window size		
	30	60	90
DS-CNN	88.05% ±7.68%	88.25% ±8.95%	88.84% ±8.93%
LSTM	91.17% ±5.12%	92.52% ±5.70%	92.38% ±5.83%
Bi-LSTM	90.61% ±5.78%	92.28% ±5.77%	92.21% ±6.67%
(DS-CNN)-LSTM	90.60% ±6.64%	92.43% ±6.03%	**92.95% ± 6.22%**
(DS-CNN)-(Bi-LSTM)	90.67% ±6.71%	91.82% ±6.92%	92.87% ±6.25%

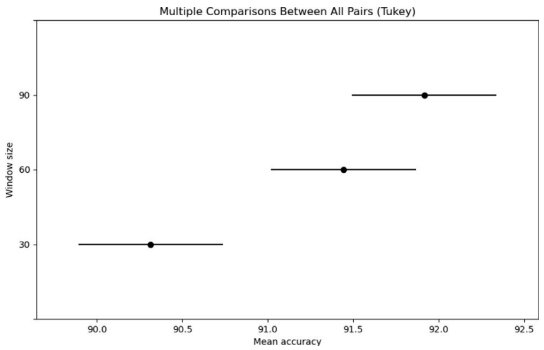


Fig. 8. Tukey test results for each group of accuracy values referring to each selected window size.

Table 10
Average confusion matrix for the best combination found.

	Ground truth				Precision
	Inactive	Active	Walking	Driving	
Inactive	1993.4	42.2	3.8	3.4	97.58%
Active	40.6	1226.8	51.2	20.3	91.63%
Walking	2.9	45.8	613.7	4.9	91.97%
Driving	11.7	7.5	4.5	515.3	95.60%
Recall	97.31%	92.78%	91.16%	94.74%	**94.80%**

of classifying the data exceptionally well, improving the results obtained with the most traditional machine learning techniques, going from 92.97% accuracy to 94.80%.

In any case, given the statistical equivalences observed previously, any model, except the DS-CNN, with a window size of 60 or 90 s, could be chosen as the preferred solution to the required classification. Thus, if the least complex option were sought, among all the statistically equivalent ones, an LSTM model with a single layer and 16 neurons could be sufficient for the problem to be solved. Likewise, a window size of 60 s could be chosen, since it would enable a more fitting classification of the activities under examination by permitting their segregation into 60-s intervals. Table 11 depicts the average confusion matrix for this particular option. As can be seen, the classification is similar to that of the best case obtained, but the confusion with the "active" class is more accentuated. In light of that, it is possible to select this choice as preferred.

Furthermore, in addition to all the experiments conducted with the proposed approach, it was decided to perform an ablation study. In this case, this investigation involved isolating each of the initial branches of the general model. Thus, the outputs of each one go directly to the subsequent layers of Dropout and Softmax, bypassing the concatenation layer. The aim is to observe the approximate influence of each sensor on the final classification based on their individual results. In this way, experimentation was done only with the best-case scenario found previously, corresponding to the confusion matrix in Table 10. As a result, more specific results are obtained while avoiding overloading the paper with extensive tables. With this configuration, the results shown in Table 12 were obtained. As can be seen, the accelerometer and gyroscope were by far the most accurate sensors. It is worth noting

Table 11
Average confusion matrix for the least complex case and statistically equivalent to the best one found.

	Ground truth				Precision
	Inactive	Active	Walking	Driving	
Inactive	1927.6	41.6	63.7	2.2	94.72%
Active	139.6	1231.6	38.2	26.6	85.77%
Walking	10.3	55.3	612.3	8.7	89.18%
Driving	6.1	20.2	6.2	534.1	94.26%
Recall	92.51%	91.32%	84.99%	93.44%	**91.14%**

Table 12
Results from the ablation study, for each individual sensor and with the best configuration found in prior experiments.

	Accuracy	F_1-score
Accelerometer	91.70% $\pm$6.77	90.55% $\pm$6.34
Gyroscope	90.55% $\pm$6.93	87.56% $\pm$9.42
Magnetometer	80.69% $\pm$14.30	80.69% $\pm$14.30
GPS	81.96% $\pm$14.09	83.70% $\pm$11.77

that even on their own, they can surpass accuracies of 90%. However, both the magnetometer and GPS yielded considerably lower results. Nevertheless, they manage to secure 80% accuracy. This is quite acceptable considering the nature of these sensors, which, while beneficial for HAR, do not adapt as well to this field as the accelerometer or gyroscope. All in all, although the individual results exhibit notable disparities, it is crucial to acknowledge that the measurements from each sensor could hold considerable value contingent upon the specific context. Significantly, some sensors might prove more suitable than others, depending on the movement type to be analysed. As a result, given the inherent variability present in the used dataset, the combination of all the sensors yields the best outcomes achieved to date for this particular dataset.

5.2. Discussion

The outcomes of this paper proved that deep learning algorithms are one of the best options in HAR, even in real-life environments such as the one discussed here. The accuracy obtained with the best combination of hyperparameters improves on that obtained with the most traditional machine learning algorithms, from 92.97% to 94.80%. Table 13 shows the comparison of the best results obtained with the methods used in the present paper, with respect to those of other papers that also used the same dataset. As can be observed, the resulting hybrid model of combining DS-CNN and LSTM yielded the most exceptional outcomes, using a window size of 90 s. The superiority of the proposed method over previous approaches may be attributed to several factors. Firstly, it could be due to the choice of feature set. In earlier machine learning endeavours, this process was manually conducted, and the selection of features might not have been the most suitable for the problem at hand. In contrast, the deep learning algorithms presented here automatically perform feature selection, which could ultimately lead to improved results. In addition, combining the chosen sensors and enabling them to analyse data individually, before concatenating their evaluations, proved advantageous for this dataset. It should be noted that, in previous works, this evaluation was performed jointly, assessing all sensors simultaneously. Finally, the intrinsic nature of LSTM, capable of retaining information from the past, also appears highly suitable for HAR, given the obtained results. In this way, considering the window size as well, it could be concluded that this combination of peculiarities presents, to date, the optimal model for the used dataset.

In the previous section, it was possible to observe how the models based solely on DS-CNN had a lower performance than the rest of the models used. Nonetheless, it should be noted that the running times of this case are much lower than those of the other algorithms implemented. To highlight that, Fig. 9 shows a comparison of the average execution time (in seconds) of each algorithm over all the experiments carried out. All these operations were performed on NVIDIA A100 40 GB GPUs. Hence, although the results are objectively worse, it could be a good option when the available time is much more limited. However, it is curious to observe how these times are longer for the individual cases based on LSTM, compared to the hybrid models that have higher complexity. Probably, the feature extraction carried out by DS-CNNs, in addition to improving the final classification in these models, may also be helping to reach a convergence point more quickly.

As already observed in other works on the same dataset like [25], there seems to be a certain tendency to improve classification with larger window sizes. That is confirmed by the results obtained in this paper, where window sizes of 60 and 90 s performed objectively better than the 30-s case. However, there did not appear to be any fundamental difference in the other hyperparameters (number of layers, number of neurons and kernel size). Therefore, as discussed in the previous section, the less complex 60-s case could be used preferably instead of the best combination one.

Moreover, it is meaningful to highlight the findings from the ablation study. As expected, the accelerometer and gyroscope outperformed the magnetometer and GPS. Nevertheless, it is significant to see these results validated in a real-life dataset, as this

Table 13
Comparison of the best results obtained with the methods used in the present paper, with respect to those of other papers that worked with the same dataset.

Work	Algorithm	Window size (s)	Eval. method	Accuracy
[23]	Support Vector Machine	20	10-fold	69.28% ±15.10%
[25]	Support Vector Machine	80	10-fold	86.56% ±11.30%
	Decision Tree	20	10-fold	89.99% ±6.13%
	Multilayer Perceptron	40	10-fold	86.85% ±6.12%
	Naïve Bayes	80	10-fold	83.27% ±7.78%
	K-Nearest Neighbour	80	10-fold	89.02% ±8.00%
	Random Forest	80	10-fold	92.97% ±6.23%
	Extreme Gradient Boosting	70	10-fold	92.23% ±7.30%
This work	DS-CNN	90	10-fold	90.70% ±7.29%
	LSTM	90	10-fold	93.52% ±5.59%
	Bi-LSTM	90	10-fold	93.09% ±5.10%
	(DS-CNN)-LSTM	**90**	**10-fold**	**94.80% ± 4.09%**
	(DS-CNN)-(Bi-LSTM)	90	10-fold	94.16% ±5.06%

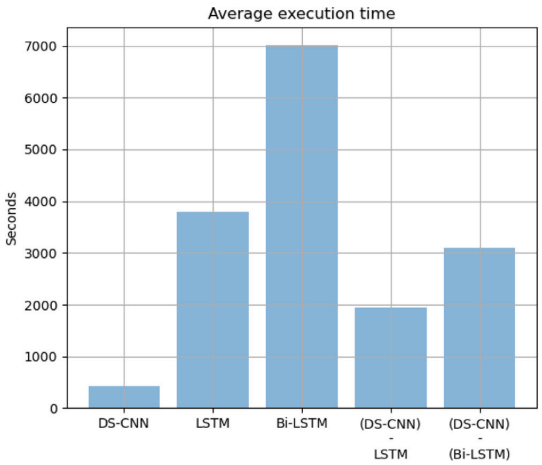


Fig. 9. Average execution time (in seconds) required to complete each of the experiments carried out by each of the implemented models.

confirmation had not been previously conducted. Additionally, the experiments with the comprehensive model demonstrated how combining these sensors, each with its unique characteristics and measurements, positively influenced the overall results.

Furthermore, it is also worth noting that there is still confusion with the "active" class. As previously mentioned, this class encompasses a multitude of actions that could be pretty fuzzy for the classification carried out by the model. Within an "active" session, there may be periods of inactivity or when the individual is walking, which may be detected by the model and marked as an incorrect activity. In any case, these confusions are minor, and they may be simply limitations in the dataset itself. Anyhow, it is feasible to think that these results could be improved, perhaps with other ways of preprocessing the data or with algorithms that may arise in the following years.

6. Conclusions and future work

This paper presents a brand-new set of experiments in human activity recognition (HAR) from smartphone sensor data from activities performed in a real-life environment. To carry them out, the deep learning algorithms that are yielding the best results in this area were applied: Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) models. By comparing their variants, combining them and making an exhaustive study of the different hyperparameters to be used, it was possible to improve the results previously achieved with the same dataset and more traditional machine learning techniques.

The results show that the most suitable models to exploit such data are those based on LSTMs, especially in conjunction with CNNs. However, at the hyperparameter level, no notable differences were observed concerning performance with different numbers of layers, neurons or kernel size. Nonetheless, improvements were detected when the window sizes were wider. When these window sizes were presented in time intervals of 60 or 90 s, the results improved substantially, compared to those obtained with window

sizes of 30 s. The activities studied in the used dataset have a long-themed nature, so it is plausible to think that longer time intervals may ease the classification of the samples fed to the implemented models.

Furthermore, it is also worth acknowledging the results of the ablation study. The accelerometer and gyroscope have indeed shown more robust performance in HAR, yielding high-grade results. That might lead to anticipate that their accuracies would surpass those of the magnetometer and GPS. However, this had not been confirmed until now on a real-life dataset. Additionally, the combination of all four sensors, each with its own distinctive characteristics that could influence the outcomes more or less positively depending on the study context, resulted in the best performance achieved to date for this dataset.

Moreover, it is also worth noting that the "active" class remains the most difficult to classify. Anyhow, in this case, the confusions are significantly lower than in other works. Given the fuzzy nature with which this activity was defined, it is possible that the results cannot be improved much further and that this is a restriction of the dataset used. Perhaps it is time to sharpen the focus and tackle much more specific activities, allowing the transfer of the acquired knowledge to everyday environments with better precision.

In any case, different data treatments could lead to better results. After all, in order to balance the sampling rates of the sensors used to collect the data, an experimental solution had to be implemented, discussed in detail in Section 4.1. A much more thorough exploration of how to address this issue, perhaps with the application of different types of interpolations or specific treatments for each kind of signal, could further refine the proposed models for the dataset used.

On the contrary, while following the same approach, the stratified 10-fold cross-validation applied to the data could have been conducted differently. As a result, data from the same individual could potentially appear in both the training and test sets using this methodology. However, it is worth noting that the data exhibit considerable variability in the different actions to be performed, as mentioned in Section 4.1. As a consequence, the impact of this separation may not be substantial, and the results could be reasonably consistent with those achieved in this paper. Nevertheless, the outcome of implementing a system that guarantees such differentiation remains uncertain. Therefore, even though the performance of the proposed models is considered outstanding, further studies could be conducted to explore and identify the best model and treatment for real-life datasets.

CRediT authorship contribution statement

Daniel Garcia-Gonzalez: Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Daniel Rivero:** Conceptualization, Formal analysis, Methodology, Supervision, Writing – review & editing. **Enrique Fernandez-Blanco:** Conceptualization, Formal analysis, Methodology, Supervision, Writing – review & editing. **Miguel R. Luaces:** Funding acquisition, Project administration, Resources, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The original complete dataset, as well as some of the scripts used to preprocess its data, can be found online at http://lbd.udc.es/research/real-life-HAR-dataset. Likewise, these have also been uploaded to Mendeley Data (https://data.mendeley.com/datasets/3xm88g6m6d/2). In addition, the code used for the entire data preparation and experimentation in this work is available online at http://gitlab.lbd.org.es/dgarcia/deep-learning-models-har.

Acknowledgments

We express our gratitude to CITIC and CESGA for their assistance in executing the code associated with this paper.

Funding

This research was partially funded by MCIN/AEI/10.13039/501100011033, NextGenerationEU/PRTR, FLATCITY-POC, Spain [grant number PDC2021-121239-C31]; MCIN/AEI/10.13039/501100011033 MAGIST, Spain [grant number PID2019-105221RB-C41]; Xunta de Galicia/FEDER-UE, Spain [grant numbers ED431G 2019/01, ED481A 2020/003, ED431C 2022/46, ED431C 2018/49 and ED431C 2021/53]. Funding for open access charge: Universidade da Coruña/CISUG.

References

[1] J.K. Aggarwal, L. Xia, Human activity recognition from 3d data: A review, Pattern Recognit. Lett. 48 (2014) 70–80.
[2] Y. Wang, S. Cang, H. Yu, A survey on wearable sensor modality centred human activity recognition in health care, Expert Syst. Appl. 137 (2019) 167–190.
[3] E. Soleimani, E. Nazerfard, Cross-subject transfer learning in human activity recognition systems using generative adversarial networks, Neurocomputing 426 (2021) 26–34.
[4] A. Subasi, M. Radhwan, R. Kurdi, K. Khateeb, IoT based mobile healthcare system for human activity recognition, in: 2018 15th Learning and Technology Conference (L&T), IEEE, 2018, pp. 29–34.
[5] F. Demrozi, G. Pravadelli, A. Bihorac, P. Rashidi, Human activity recognition using inertial, physiological and environmental sensors: a comprehensive survey, IEEE Access (2020).
[6] R. Liu, A.A. Ramli, H. Zhang, E. Datta, E. Henricson, X. Liu, An overview of human activity recognition using wearable sensors: Healthcare and artificial intelligence, 2021, arXiv preprint arXiv:2103.15990.
[7] F. Attal, S. Mohammed, M. Dedabrishvili, F. Chamroukhi, L. Oukhellou, Y. Amirat, Physical human activity recognition using wearable sensors, Sensors 15 (12) (2015) 31314–31338.
[8] M.S. Zainudin, M.N. Sulaiman, N. Mustapha, T. Perumal, Monitoring daily fitness activity using accelerometer sensor fusion, in: 2017 IEEE International Symposium on Consumer Electronics (ISCE), IEEE, 2017, pp. 35–36.
[9] M. Raeiszadeh, H. Tahayori, A novel method for detecting and predicting resident's behavior in smart home, in: 2018 6th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS), IEEE, 2018, pp. 71–74.
[10] O.D. Lara, M.A. Labrador, A survey on human activity recognition using wearable sensors, IEEE Commun. Surv. Tutor. 15 (3) (2012) 1192–1209.
[11] M.M. Hassan, M.Z. Uddin, A. Mohamed, A. Almogren, A robust human activity recognition system using smartphone sensors and deep learning, Future Gener. Comput. Syst. 81 (2018) 307–313.
[12] Y. Tang, L. Zhang, F. Min, J. He, Multi-scale deep feature learning for human activity recognition using wearable sensors, IEEE Trans. Ind. Electron. (2022).
[13] M. Shoaib, S. Bosch, O. Incel, H. Scholten, P. Havinga, Complex human activity recognition using smartphone and wrist-worn motion sensors, Sensors 16 (4) (2016) 426.
[14] A. Ignatov, Real-time human activity recognition from accelerometer data using convolutional neural networks, Appl. Soft Comput. 62 (2018) 915–922.
[15] K. Xia, J. Huang, H. Wang, LSTM-CNN architecture for human activity recognition, IEEE Access 8 (2020) 56855–56866.
[16] P. Lago, S. Takeda, T. Okita, S. Inoue, Measured: Evaluating sensor-based activity recognition scenarios by simulating accelerometer measures from motion capture, in: Human Activity Sensing, Springer, 2019, pp. 135–149.
[17] I.A. Lawal, S. Bano, Deep human activity recognition using wearable sensors, in: Proceedings of the 12th ACM International Conference on PErvasive Technologies Related to Assistive Environments, 2019, pp. 45–48.
[18] S. Jeong, D. Oh, Development of a hybrid deep-learning model for the human activity recognition based on the wristband accelerometer signals, J. Internet Comput. Serv. 22 (3) (2021) 9–16.
[19] A. Stisen, H. Blunck, S. Bhattacharya, T.S. Prentow, M.B. Kjærgaard, A. Dey, T. Sonne, M.M. Jensen, Smart devices are different: Assessing and mitigatingmobile sensing heterogeneities for activity recognition, in: Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems, 2015, pp. 127–140.
[20] E. Sansano, R. Montoliu, O. Belmonte Fernandez, A study of deep neural networks for human activity recognition, Comput. Intell. 36 (3) (2020) 1113–1139.
[21] N. Lane, Y. Xu, H. lu, S. Hu, T. Choudhury, A. Campbell, F. Zhao, Enabling large-scale human activity inference on smartphones using community similarity networks (CSN), in: UbiComp'11 - Proceedings of the 2011 ACM Conference on Ubiquitous Computing, 2011, pp. 355–364.
[22] A. Ferrari, D. Micucci, M. Mobilio, P. Napoletano, On the personalization of classification models for human activity recognition, IEEE Access 8 (2020) 32066–32079.
[23] D. Garcia-Gonzalez, D. Rivero, E. Fernandez-Blanco, M.R. Luaces, A public domain dataset for real-life human activity recognition using smartphone sensors, Sensors 20 (8) (2020) 2200.
[24] L. Hu, K. Zhao, B.W.-K. Ling, Y. Lin, Activity recognition via correlation coefficients based graph with nodes updated by multi-aggregator approach, Biomed. Signal Process. Control 79 (2023) 104255.
[25] D. Garcia-Gonzalez, D. Rivero, E. Fernandez-Blanco, M.R. Luaces, New machine learning approaches for real-life human activity recognition using smartphone sensor-based data, Knowl.-Based Syst. (2023) 110260.
[26] S. Mekruksavanich, A. Jitpattanakul, Biometric user identification based on human activity recognition using wearable sensors: An experiment using deep learning models, Electronics 10 (3) (2021) 308.
[27] I.U. Khan, S. Afzal, J.W. Lee, Human activity recognition via hybrid deep learning based model, Sensors 22 (1) (2022) 323.
[28] E. Ramanujam, T. Perumal, S. Padmavathi, Human activity recognition with smartphone and wearable sensors using deep learning techniques: A review, IEEE Sens. J. 21 (12) (2021) 13029–13040.
[29] D. Anguita, A. Ghio, L. Oneto, X. Parra, J.L. Reyes-Ortiz, A public domain dataset for human activity recognition using smartphones, in: Esann, 2013.
[30] J.R. Kwapisz, G.M. Weiss, S.A. Moore, Activity recognition using cell phone accelerometers, ACM SigKDD Explor. Newsl. 12 (2) (2011) 74–82.
[31] D. Micucci, M. Mobilio, P. Napoletano, Unimib shar: A dataset for human activity recognition using acceleration data from smartphones, Appl. Sci. 7 (10) (2017) 1101.
[32] D. Anguita, A. Ghio, L. Oneto, X. Parra, J.L. Reyes-Ortiz, Training computationally efficient smartphone–based human activity recognition models, in: International Conference on Artificial Neural Networks, Springer, 2013, pp. 426–433.
[33] J.-L. Reyes-Ortiz, L. Oneto, A. Ghio, A. Samá, D. Anguita, X. Parra, Human activity recognition on smartphones with awareness of basic activities and postural transitions, in: International Conference on Artificial Neural Networks, Springer, 2014, pp. 177–184.
[34] Z. Wu, A. Zhang, C. Zhang, Human activity recognition using wearable devices sensor data, 2015.
[35] Z. Chen, Q. Zhu, Y.C. Soh, L. Zhang, Robust human activity recognition using smartphone sensors via CT-PCA and online SVM, IEEE Trans. Ind. Inform. 13 (6) (2017) 3070–3080.
[36] S. Seto, W. Zhang, Y. Zhou, Multivariate time series classification using dynamic time warping template selection for human activity recognition, in: 2015 IEEE Symposium Series on Computational Intelligence, IEEE, 2015, pp. 1399–1406.
[37] W. Sousa, E. Souto, J. Rodrigres, P. Sadarc, R. Jalali, K. El-Khatib, A comparative analysis of the impact of features on human activity recognition with smartphone sensors, in: Proceedings of the 23rd Brazillian Symposium on Multimedia and the Web, ACM, 2017, pp. 397–404.
[38] J. Yang, M.N. Nguyen, P.P. San, X.L. Li, S. Krishnaswamy, Deep convolutional neural networks on multichannel time series for human activity recognition, in: Twenty-Fourth International Joint Conference on Artificial Intelligence, 2015.
[39] C.A. Ronao, S.-B. Cho, Human activity recognition with smartphone sensors using deep learning neural networks, Expert Syst. Appl. 59 (2016) 235–244.
[40] Y. Guan, T. Plötz, Ensembles of deep lstm learners for activity recognition using wearables, Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 1 (2) (2017) 1–28.
[41] F. Hernández, L.F. Suárez, J. Villamizar, M. Altuve, Human activity recognition on smartphones using a bidirectional LSTM network, in: 2019 XXII Symposium on Image, Signal Processing and Artificial Vision (STSIVA), IEEE, 2019, pp. 1–5.

[42] Y. Li, L. Wang, Human activity recognition based on residual network and BiLSTM, Sensors 22 (2) (2022) 635.
[43] W. Qi, H. Su, C. Yang, G. Ferrigno, E. De Momi, A. Aliverti, A fast and robust deep convolutional neural networks for complex human activity recognition using smartphone, Sensors 19 (17) (2019) 3731.
[44] S. Wan, L. Qi, X. Xu, C. Tong, Z. Gu, Deep learning models for real-time human activity recognition with smartphones, Mob. Netw. Appl. 25 (2) (2020) 743–755.
[45] E. Shalaby, N. ElShennawy, A. Sarhan, Utilizing deep learning models in CSI-based human activity recognition, Neural Comput. Appl. 34 (8) (2022) 5993–6010.
[46] Q. Teng, K. Wang, L. Zhang, J. He, The layer-wise training convolutional neural networks using local loss for sensor-based human activity recognition, IEEE Sens. J. 20 (13) (2020) 7265–7274.
[47] J. Figueiredo, G. Gordalina, P. Correia, G. Pires, L. Oliveira, R. Martinho, R. Rijo, P. Assuncao, A. Seco, R. Fonseca-Pinto, Recognition of human activity based on sparse data collected from smartphone sensors, in: 2019 IEEE 6th Portuguese Meeting on Bioengineering (ENBENG), IEEE, 2019, pp. 1–4.
[48] R.-A. Voicu, C. Dobre, L. Bajenaru, R.-I. Ciobanu, Human physical activity recognition using smartphone sensors, Sensors 19 (3) (2019) 458.
[49] Y.E. Ustev, O. Durmaz Incel, C. Ersoy, User, device and orientation independent human activity recognition on mobile phones: Challenges and a proposal, in: Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication, ACM, 2013, pp. 1427–1436.
[50] V. Janko, N. Rešçiç, M. Mlakar, V. Drobnič, M. Gams, G. Slapničar, M. Gjoreski, J. Bizjak, M. Marinko, M. Luštrek, A new frontier for activity recognition: The sussex-huawei locomotion challenge, in: Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, 2018, pp. 1511–1520.
[51] N. Hnoohom, A. Jitpattanakul, S. Mekruksavanich, Real-life human activity recognition with tri-axial accelerometer data from smartphone using hybrid long short-term memory networks, in: 2020 15th International Joint Symposium on Artificial Intelligence and Natural Language Processing (ISAI-NLP), IEEE, 2020, pp. 1–6.
[52] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al., Tensorflow: A system for large-scale machine learning, in: 12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16), 2016, pp. 265–283.
[53] F. Chollet, et al., Keras, 2015, https://keras.io.
[54] S. Chetlur, C. Woolley, P. Vandermersch, J. Cohen, J. Tran, B. Catanzaro, E. Shelhamer, Cudnn: Efficient primitives for deep learning, 2014, arXiv preprint arXiv:1410.0759.
[55] K. Fukushima, S. Miyake, Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition, in: Competition and Cooperation in Neural Nets, Springer, 1982, pp. 267–285.
[56] Y. LeCun, P. Haffner, L. Bottou, Y. Bengio, Object recognition with gradient-based learning, in: Shape, Contour and Grouping in Computer Vision, Springer, 1999, pp. 319–345.
[57] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Commun. ACM 60 (6) (2017) 84–90.
[58] E. Fernandez-Blanco, D. Rivero, A. Pazos, Convolutional neural networks for sleep stage scoring on a two-channel EEG signal, Soft Comput. 24 (2020) 4067–4079.
[59] J. Zhu, H. Chen, W. Ye, A hybrid CNN–LSTM network for the classification of human activities based on micro-Doppler radar, IEEE Access 8 (2020) 24713–24720.
[60] S.K. Challa, A. Kumar, V.B. Semwal, A multibranch CNN-BiLSTM model for human activity recognition using wearable sensor data, Vis. Comput. (2021) 1–15.
[61] J. Nagi, F. Ducatelle, G.A. Di Caro, D. Cireşan, U. Meier, A. Giusti, F. Nagi, J. Schmidhuber, L.M. Gambardella, Max-pooling convolutional neural networks for vision-based hand gesture recognition, in: 2011 IEEE International Conference on Signal and Image Processing Applications (ICSIPA), IEEE, 2011, pp. 342–347.
[62] F. Chollet, Xception: Deep learning with depthwise separable convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1251–1258.
[63] D. Alghazzawi, O. Rabie, O. Bamasaq, A. Albeshri, M.Z. Asghar, Sensor-based human activity recognition in smart homes using depthwise separable convolutions, Hum. Cent. Comput. Inf. Sci. 12 (2022) 50.
[64] E. Fernandez-Blanco, D. Rivero, A. Pazos, EEG signal processing with separable convolutional neural network for automatic scoring of sleeping stage, Neurocomputing 410 (2020) 220–228.
[65] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780.
[66] M. Schuster, K.K. Paliwal, Bidirectional recurrent neural networks, IEEE Trans. Signal Process. 45 (11) (1997) 2673–2681.
[67] Y. Zhao, R. Yang, G. Chevalier, X. Xu, Z. Zhang, Deep residual bidir-LSTM for human activity recognition using wearable sensors, Math. Probl. Eng. 2018 (2018).
[68] J. Huang, S. Lin, N. Wang, G. Dai, Y. Xie, J. Zhou, TSE-cnn: A two-stage end-to-end CNN for human activity recognition, IEEE J. Biomed. Health Inform. 24 (1) (2019) 292–299.
[69] Y. Chen, Y. Xue, A deep learning approach to human activity recognition based on single accelerometer, in: 2015 IEEE International Conference on Systems, Man, and Cybernetics, IEEE, 2015, pp. 1488–1492.
[70] J.-L. Reyes-Ortiz, L. Oneto, A. Sama, X. Parra, D. Anguita, Transition-aware human activity recognition using smartphones, Neurocomputing 171 (2016) 754–767.
[71] M. Hossin, M. Sulaiman, A review on evaluation metrics for data classification evaluations, Int. J. Data Min. Knowl. Manag. Process 5 (2) (2015) 1.
[72] M. Bekkar, H.K. Djemaa, T.A. Alitouche, Evaluation measures for models assessment over imbalanced data sets, J. Inf. Eng. Appl. 3 (10) (2013).
[73] R. Kohavi, et al., A study of cross-validation and bootstrap for accuracy estimation and model selection, in: Ijcai, Vol. 14, Montreal, Canada, 1995, pp. 1137–1145.
[74] L. Vanneschi, M. Castelli, S. Silva, Measuring bloat, overfitting and functional complexity in genetic programming, in: Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation, 2010, pp. 877–884.
[75] M. Cogswell, F. Ahmed, R. Girshick, L. Zitnick, D. Batra, Reducing overfitting in deep networks by decorrelating representations, 2015, arXiv preprint arXiv:1511.06068.
[76] L. Prechelt, Early stopping-but when? in: Neural Networks: Tricks of the Trade, Springer, 1998, pp. 55–69.
[77] G.E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, R.R. Salakhutdinov, Improving neural networks by preventing co-adaptation of feature detectors, 2012, arXiv preprint arXiv:1207.0580.

Appendix B

Resumen extendido en castellano

Este apéndice resume los contenidos de esta Tesis en castellano. En primer lugar, la Sección B.1 explica el contexto del campo del reconocimiento de actividades humanas y las motivaciones para realizar este trabajo. Seguidamente, la Sección B.2 enumera los principales objetivos del proyecto. Tras ello, la Sección B.3 sintetiza los principales logros y aportes de la Tesis de manera concisa. Finalmente, la Sección B.4 describe una serie de conclusiones del trabajo, seguido de una lista de posibles líneas de trabajo futuro en la Sección B.5.

B.1 Motivación

La capacidad de identificar de manera fiable y automática los movimientos realizados por un ser humano es un desafío que ha sido objeto de gran cantidad de investigación en las últimas décadas. El principal interés radica en las múltiples aplicaciones que podrían tener los sistemas que detecten dichas acciones. Por ejemplo, dentro del mundo de la salud [Subasi et al., 2018, Demrozi et al., 2020, Liu et al., 2021] y el fitness [Attal et al., 2015, Zainudin et al., 2017], sería posible conocer los movimientos que realiza un individuo de cara a poder realizar un diagnóstico más adecuado. Además, también sería posible llevar un tratamiento con un control más exhaustivo y cómodo para ambas partes. Por otro lado, también pueden aplicarse los avances en este campo directamente a la domótica [Raeiszadeh and Tahayori, 2018, Du et al., 2019] o al ocio [Ma, 2021], ya que sería posible automatizar y desencadenar acciones basadas en los movimientos realizados por el individuo. Para detectar estas acciones, se pueden utilizar tanto cámaras de vídeo [Ke et al., 2013, Beddiar et al., 2020] como sensores de movimiento que puedan llevar encima los individuos [Aggarwal and Xia, 2014, Wang et al., 2019,

Soleimani and Nazerfard, 2021]. En cuanto a estos últimos, los más comunes son el acelerómetro y el giroscopio. El primero de ellos se utiliza para detectar vibraciones o pequeños movimientos en el individuo. En cuanto al segundo, su cometido es el de medir las diferentes oscilaciones o giros que se puedan producir. Antiguamente, estos sensores eran mucho más caros y menos accesibles. No obstante, desde la aparición de los dispositivos wearables y, sobre todo, los smartphones, el campo del reconocimiento de actividades humanas (HAR) ha experimentado un acelerón importante. Esto se debe principalmente a que dichos dispositivos ya llevan implantados de base ese tipo de sensores. Además, hace más de una década, estos dispositivos se han vuelto enormemente populares en el mundo desarrollado, hasta el punto de que muchas personas llevan una pulsera de actividad en la muñeca y, sobre todo, un smartphone en el bolsillo, que utilizan a diario. Este hecho hace que los costes de investigar en este campo sean mucho menores, con un acceso mucho más sencillo a sensores de alta precisión. Así, los investigadores encuentran en HAR una opción muy atractiva en la que poner su granito de arena [Lara and Labrador, 2012, Hassan et al., 2018, Tang et al., 2022].

Sin embargo, hay una serie de problemáticas que hay que tener en cuenta. En primer lugar, es necesario llevar un control exhaustivo de la temporalidad de los datos, lo cual es especialmente complicado al trabajar con grandes cantidades de información como las que producen los sensores comentados anteriormente. Si bien ya se han hecho grandes avances en este respecto [Shoaib et al., 2016, Qi et al., 2019, Xia et al., 2020], todavía existen actividades en las que la relación entre la acción y sus datos todavía no ha sido del todo resuelta. Esto se debe principalmente a las condiciones en las que dichos estudios fueron elaborados. En general, en estos trabajos, el individuo que realiza la acción lo hace de una forma muy controlada, con unas indicaciones muy concretas de cómo llevarla a cabo [Xu et al., 2019]. Además, el dispositivo de medición utilizado se coloca en un lugar específico y de una forma determinada, como en la muñeca [Lawal and Bano, 2019] o la cintura [Jeong and Oh, 2021]. Si bien es cierto que en estos casos se ha conseguido resolver con gran precisión la gran mayoría de las acciones estudiadas, dichos resultados no serían del todo fiables si se aplicaran en un entorno de la vida real. En la vida diaria, esas condiciones tan específicas no se dan habitualmente, por lo que el conocimiento adquirido no podría ser extrapolado directamente a un entorno más cotidiano y realista. Por ejemplo, una misma acción no tendría por qué arrojar los mismos datos en diferentes individuos [Lago et al., 2019]. En el caso de los smartphones, su uso y su manera de transportarlos difieren para cada persona. Dichas variaciones afectarían notablemente a las mediciones registradas por los sensores del dispositivo. Incluso diferentes modelos de smartphone podrían arrojar datos ligeramente diferentes [Stisen et al., 2015]. Este punto no sería tan crítico utilizando pulseras de actividad, ya que irían siempre en la muñeca. Sin embargo, su uso es mucho menor que en el caso del smartphone. A día de hoy, la gran mayoría de las personas disponen de un smartphone que transportan a todas partes y el utilizar cualquier otro tipo

de dispositivo wearable, como las pulseras, es más una elección personal. Por otro lado, la realización de la acción no tiene por qué ser exactamente igual en todo el mundo. Aunque teóricamente debería serlo, puede haber pequeñas variaciones que podrían llevar a confusión a la hora de realizar la clasificación. Por ejemplo, al caminar, no todo el mundo se inclina de la misma manera o desplaza las piernas del mismo modo. De hecho, algunas de estas diferencias podrían venir también dadas por la propia diversidad física de cada individuo, incluso utilizando el dispositivo de la misma manera [Sansano et al., 2020]. Sin ir más lejos y utilizando el mismo ejemplo anterior, la longitud o el ancho de la pierna de cada persona al caminar podría dar lugar a variaciones en las mediciones si se llevara el dispositivo, por ejemplo, en el bolsillo del pantalón. Es más, esa misma colocación podría dar lugar a resultados diferentes si estamos hablando de, por ejemplo, unas mallas ajustadas en comparación con unos pantalones cargo holgados. De hecho, esta problemática de la personificación de los modelos de clasificación para grandes cantidades de personas es algo que también se lleva estudiando bastante en los últimos años [Lane et al., 2011, Solis Castilla et al., 2020, Ferrari et al., 2020].

Por todas estas razones, la motivación de esta Tesis radica en la dificultad actual de aplicar, fuera de un entorno de laboratorio, todos los avances conseguidos hasta ahora en este campo. Antes de iniciar esta Tesis, no existía ningún conjunto de datos realista. Por lo tanto, es necesario publicar nuevos conjuntos de datos, con información proveniente de acciones más realistas y flexibles. En consecuencia, se podría iniciar un estudio exhaustivo sobre cómo abordarlos correctamente. Para ello, sería esencial analizar la idoneidad de todas las técnicas de inteligencia artificial empleadas anteriormente en HAR. El objetivo sería identificar las opciones más apropiadas y modificarlas en consecuencia para adaptarlas al contexto específico. En este sentido, vale la pena señalar que obtener un algoritmo óptimo para todas las situaciones no es factible. Al centrarse en dominios concretos, como la nueva orientación propuesta, se vuelve más fácil mejorar los resultados para ese caso particular. Sin embargo, dicha mejora puede implicar un rendimiento inferior en otros escenarios, tal y como demuestran los teoremas No Free Lunch (NFL) [Wolpert and Macready, 1997]. Además, sería necesario tener en cuenta que esos datos podrían presentar características únicas que aún no se han observado en anteriores desarrollos de HAR. Así, también sería crucial investigar la metodología más adecuada para procesar esos datos, estudiando la mejor manera de prepararlos para alimentar los modelos pertinentes. Como resultado, todo el progreso alcanzado en dichos conjuntos de datos podría aplicarse directamente a entornos de la vida real, según las actividades estudiadas en ellos.

B.2 Objetivos

Teniendo en cuenta todo lo comentado en las secciones anteriores, el objetivo principal de esta Tesis es impulsar la orientación de la investigación global en el campo del

reconocimiento de las actividades humanas hacia entornos de la vida real. Para ello, es esencial poder contar con nuevos datos provenientes de entornos más realistas que puedan ser explotados por toda la comunidad científica. Todo el conocimiento adquirido hasta ahora en el campo podría aplicarse a ese nuevo enfoque, adaptándolo en consecuencia. De este modo, el primer objetivo de esta Tesis consiste en realizar una revisión bibliográfica en profundidad que abarque todo el ámbito HAR. El propósito es identificar los aspectos cruciales que deben tenerse en cuenta para llevar a cabo avances relevantes en este campo. Teniendo esto en cuenta, el siguiente paso consistirá en elaborar un conjunto de datos nuevo, en el que los individuos que aporten sus mediciones puedan hacerlo de una manera mucho más libre, según las peculiaridades de cada uno. Después, a partir de dicho conjunto de datos, será necesario buscar la mejor forma de abordarlo, desde técnicas tradicionales de machine learning hasta las arquitecturas más recientes basadas en deep learning. Con eso en mente, se deducen una serie de desafíos de investigación que constituirán el núcleo de esta Tesis y que se resumen en los siguientes cuatro puntos:

- **Revisión bibliográfica exhaustiva de todo el ámbito HAR.** Para llevar a cabo desarrollos relevantes en dicho ámbito, es esencial conocer de primera mano todo lo que haya sido realizado previamente por la comunidad científica. Si bien es cierto que dichos trabajos fueron realizados en condiciones diferentes a las perseguidas en esta Tesis, sus hallazgos podrían aportar gran valor igualmente. Al fin y al cabo, es necesario conocer las maneras más adecuadas de procesar datos procedentes de sensores de smartphone, así como las problemáticas más comunes a tener en cuenta y cómo resolverlas. Del mismo modo, es crucial mantenerse al día tanto de las tendencias actuales como de la evolución seguida en el ámbito, de cara a no caer en los mismos errores y poder detectar nuevas oportunidades de investigación.

- **Establecimiento de las pautas necesarias para la elaboración de un conjunto de datos más realista.** Tal y como se comentó en la Sección B.1, la orientación actual de todos los estudios en HAR imposibilita su aplicación directa sobre entornos de la vida real en general. Con el fin de intentar iniciar la reorientación de la investigación hacia esa problemática, se necesita realizar una recolección de datos que pueda ser explotada por toda la comunidad científica. Para ello, se utilizarán los smartphones personales de diferentes individuos, de manera que cada uno de ellos pueda utilizarlo tal y como lo hace de forma habitual. En cuanto a las actividades a estudiar, el objetivo es fijar un grupo con suficiente diversidad entre ellas pero sin ser tan específicas como las de los estudios que se están llevando a cabo actualmente. Así, se puede establecer un punto de partida con el que estudiar el potencial de esta nueva orientación y que después pueda enfocarse en acciones más concretas, según corresponda. De este modo, se conseguiría un conjunto de datos más realista que los elaborados hasta ahora, con mayor libertad y variabilidad en los datos estudiados.

- **Estudio de la aplicabilidad de las técnicas de machine learning y deep learning más utilizadas en HAR para entornos de la vida real.** Una vez tomados los datos, es necesario estudiar la evolución de las técnicas de machine learning y deep learning aplicadas a una temática HAR. A partir de ese estudio, se seleccionarán una serie de técnicas que, según los datos obtenidos en el punto anterior, tuvieran el mejor potencial de obtener buenos resultados. Al mismo tiempo, se buscará la comparación entre ellos, con diferentes configuraciones de hiperparámetros y características, con el fin de obtener la mayor cantidad de información posible. Además, la investigación también buscará explorar la aplicación de técnicas alternativas que se utilizan con menos frecuencia en HAR pero que tienen el potencial de contribuir de manera positiva a la nueva dirección que se persigue en esta Tesis.
- **Búsqueda de las aproximaciones más adecuadas para abordar los retos derivados de la explotación de datos HAR procedentes de la vida cotidiana.** Recolectar datos de manera libre y flexible difiere notablemente de cómo se haría a partir de condiciones de laboratorio. Los imprevistos que podrían surgir al construir un conjunto de datos que siga las pautas propuestas en esta Tesis podrían ser abundantes. Además, debido a la ausencia de conjuntos de datos basados en HAR que sigan dicha orientación dentro de la comunidad científica, será esencial investigar y resolver cualquier problema que pueda surgir durante el proceso de recolección de datos. Esto requiere la identificación y resolución de esos desafíos en tiempo real, pudiendo ser completamente diferentes a cualquiera visto anteriormente. Para lograrlo, se llevará a cabo un estudio exhaustivo para determinar los métodos óptimos para procesar el conjunto de datos propuesto, así como explorar diversas configuraciones y arquitecturas para los modelos de inteligencia artificial seleccionados.

B.3 Contribuciones

A partir de los objetivos comentados en la Sección B.2, ha sido posible avanzar en la investigación en el campo del reconocimiento de actividades humanas, orientando todos los hallazgos hacia un entorno de la vida real, tal y como se destacó dicha necesidad anteriormente. Por lo tanto, las contribuciones se centraron en crear un conjunto de datos más realista y buscar los mejores modelos para clasificar dichos datos, desde los más tradicionales hasta los más actuales, siguiendo los objetivos establecidos al comienzo de la Tesis.

B.3.1 Recolección de datos de la vida real

Sin una recolección de datos que siguiese el enfoque propuesto en esta Tesis era imposible seguir los objetivos propuestos inicialmente. Este motivo, sumado además

a la inexistencia de conjuntos de datos basados en HAR con dicha orientación en la comunidad científica, propició la creación de uno nuevo que pudiese ser explotado posteriormente por todos los investigadores del campo. De forma resumida, a continuación se presentan las características más destacadas del conjunto de datos resultante:

- Smartphones como dispositivo de recolección. Los smartphones actuales presentan sensores de alta precisión muy accesibles para cualquier investigador del ámbito. En este caso, se utilizaron el acelerómetro, el giroscopio, el magnetómetro y el GPS incluidos en los smartphones personales de cada individuo participante en la recogida de datos. Además, debido a su uso global en comparación con cualquier otro tipo de dispositivo wearable, los posibles hallazgos futuros podrían tener un alcance mucho más amplio.

- Diferentes perfiles de participantes. El conjunto de datos resultante proviene de 19 personas diferentes, cada uno con diferentes peculiaridades, desde la diversidad física de cada uno hasta el uso y modelo de los smartphones personales de cada uno. Esto aumenta la variabilidad en los datos, resultando en más casos de estudio.

- Actividades más genéricas. A diferencia de otros conjuntos de datos basados en HAR, las actividades aquí estudiadas suelen realizarse durante períodos más largos de tiempo. Las cuatro que fueron analizadas fueron las siguientes:

  - Inactivo: toda acción en la que no se lleve el smartphone encima.
  - Activo: cualquier movimiento sin ir a un lugar específico. Ejemplos: cepillarse los dientes, bailar en un concierto o jugar a algún videojuego.
  - Andando: desplazamientos sin vehículo motorizado. Por ejemplo, correr se clasificaría como "andando".
  - Conduciendo: viajes en un vehículo motorizado sin necesidad de ser el conductor. Por ejemplo, viajar en autobús se clasificaría como "conduciendo".

  Esto brinda la oportunidad de observar, de manera preliminar, si la nueva orientación propuesta en esta Tesis es factible en este campo. Luego, según los avances realizados y el contexto definitivo, se podrían estudiar nuevas actividades más específicas basadas en el conocimiento previo.

- Mayor libertad y realismo al realizar las acciones. A los participantes del estudio solamente se les pidió que realizaran las acciones especificadas, tal y como las harían de forma habitual, pero iniciando y finalizando la acción de forma específica a partir de su smartphone personal. Así, se implementó una aplicación sencilla para Android desde la cual los diferentes usuarios podían iniciar y finalizar sus sesiones de recolección, así como enviar toda

esta información a un servidor dedicado de recopilación de datos. Cada sesión consistió en realizar una acción específica durante la duración de la misma, desde el momento en que se inició la acción hasta que terminó, ambos presionando el botón correspondiente.

Desafortunadamente, el conjunto de datos resultante presenta algunas peculiaridades que dificultan los desarrollos que se podrían realizar a partir del mismo. Todas estas problemáticas surgieron como imprevistos a la hora de realizar la recogida de datos. Esto se debe a la novedad de este tipo de recolección, ya que fue llevada a cabo de forma muy diferente a cómo sería a partir de condiciones de laboratorio. A continuación, se comentan brevemente cada una de ellas:

- Falta de sensores. No todos los smatphones que participaron en el conjunto de datos tenían los mismos sensores disponibles. De las 19 personas que participaron en la recopilación de datos, cinco personas tuvieron ese problema, con la ausencia de al menos uno de los sensores utilizados.

- Diferencias en la frecuencia de muestreo. Aún procurando establecer la frecuencia de cada sensor al máximo valor permitido por Android, hay intervalos que difieren de ese valor. Esto significa que, en algunos casos, la diferencia de tiempo entre cada observación no es la misma.

- Desbalanceo en los datos. Dada la naturaleza de las acciones a estudiar, un porcentaje considerable de los datos pertenece a la actividad de "inactivo", debido a que es mucho más fácil recopilar muestras de esa manera que en cualquiera de las otras opciones. Aún así, esa tendencia no es demasiado acentuada y hay patrones suficientes en cada actividad para obtener resultados satisfactorios.

Con todo, se consideró que estas problemáticas podrían ser frecuentes en la vida real y, por tanto, ser algo habitual en este tipo de recogida de datos. Si bien suponen un esfuerzo mayor en el procesamiento de los datos, son casos de estudio que conviene analizar. De este modo, el conjunto de datos propuesto presenta varios retos de investigación en los que profundizar en futuros desarrollos.

B.3.2 Exploración de los datos

Una vez recogidos los datos, se comenzó un estudio exhaustivo para conocer los modelos de inteligencia artificial más prometedores para clasificar dicha información. Durante el mismo, también se investigaron las aproximaciones más adecuadas para preparar y procesar los datos propuestos. En primer lugar, este estudio se enfocó en el machine learning tradicional, para observar las peculiaridades de los modelos que más se han estado utilizando a lo largo de toda la investigación en HAR. De cara a hacer el estudio lo más detallado posible, se tuvieron en cuenta los objetivos que se

enumeran a continuación. En cada uno de ellos, también se indican brevemente los hallazgos más destacados:

- Utilización de múltiples algoritmos. Para poder realizar un estudio muy detallado, fue necesario aplicar una buena cantidad de algoritmos diferentes para observar sus comportamientos con el nuevo conjunto de datos. De este modo, se decidió optar por algoritmos muy utilizados en HAR como Support Vector Machines (SVM), Decision Trees (DT), Multilayer Perceptron (MLP), Naïve Bayes (NB), K-Nearest Neighbours (KNN) y Random Forest (RF). Como novedad, se optó también por incluir el Extreme Gradient Boosting (XGB) por su reciente gran popularidad en otros campos y sus excelentes resultados, a pesar de no verse tanto en HAR.

  Al final, los algoritmos basados en árboles fueron los que obtuvieron mejores resultados en todos los casos estudiados. Entre ellos, Random Forest destaca como el que obtuvo los mejores picos de precisión, con un 92.97% de acierto en su mejor caso.

- Aplicación de diferentes preparaciones de los datos. Debido a la naturaleza de las actividades recolectadas en el nuevo conjunto de datos, es posible que ventanas de tiempo más amplias ayuden a clasificarlas de mejor manera. Por ello, fue necesario también probar los algoritmos dispuestos previamente con diferentes tamaños de ventana para ver si se encontraban diferencias significativas. Siguiendo la misma línea, también se decidió realizar una comparación entre un conjunto de características principalmente estadístico, con otro que incluyese peculiaridades más concretas de cada señal. Así, se calcularon valores como el tiempo total positivo, número de mínimos locales, distancia total viajada, entre muchos otros.

  Con todo, cabe señalar que no se encontraron diferencias significativas entre los conjuntos de características indicados. Sin embargo, sí se observaron mejoras sustanciales con mayores tamaños de ventana. Cuando el valor alcanzaba alrededor de 60 segundos o más, los resultados fueron significativamente superiores a los obtenidos con tamaños de ventana más pequeños, de 20 o 30 segundos.

- Estudio de la influencia real del giroscopio en los resultados. En numerosos estudios de HAR se demostró que este sensor influía positivamente en la clasificación de las acciones. Para confirmar que dicho sensor también presentaba el mismo comportamiento en la nueva orientación propuesta en esta Tesis, se optó por repetir todas las pruebas realizadas sobre el conjunto de datos, excluyendo las mediciones del giroscopio.

  Finalmente, se pudo concluir que el giroscopio sí influía positivamente en los resultados finales, tal y como se demostró en otros estudios de HAR.

Por otro lado, dado que la aplicación de deep learning en HAR no ha dejado de aumentar en los últimos años, no sería adecuado limitar la exploración a técnicas basadas en el machine learning más tradicional. De este modo, aunque los resultados conseguidos en los últimos experimentos fueron muy satisfactorios, se consideró que podrían ser mejorados mediante la aplicación de deep learning. Las aportaciones más determinantes de esta segunda parte del estudio se presentan en los siguientes puntos:

- Implementación de una solución experimental para el tratamiento de los datos. Las diferentes frecuencias de muestreo presentes en el conjunto de datos propuesto complicaron esta exploración. Esto se debe a que los modelos utilizados en este caso calculan las características por ellas mismas, asumiendo que los patrones son equidistantes a nivel temporal. Por ello, fue necesario buscar la manera de establecer dicha equidistancia. De esta manera, se observó que los sensores proporcionaban, generalmente, datos en intervalos de 20 ms o 200 ms. Esto es, sin tener en cuenta el GPS, que tiene una frecuencia muy diferente al resto (aproximadamente un valor cada 10 segundos). De este modo, sólo se conservaron en el conjunto de datos las observaciones más cercanas a cada intervalo de 200 ms desde el inicio de la actividad. Así, aunque hay pérdida de datos en momentos de mayor frecuencia, se mantiene la información real correspondiente a ese instante de tiempo, sin rellenar huecos de forma "artificial". Además, el número de muestras es más que suficiente para realizar una clasificación satisfactoria, por lo que dicha pérdida no se consideró un gran problema.

- Comparación exhaustiva de los mejores algoritmos de deep learning en el HAR actual. La investigación más reciente en HAR demuestra que el uso de modelos basados en redes de neuronas convolucionales (CNN) o redes de neuronas recurrentes basadas en la técnica de Long Short-Term Memory (LSTM) generalmente ofrecen mejores resultados que otros métodos. Por ello, se seleccionaron ambos algoritmos para este estudio. Más concretamente, se optó por la utilización de la variante separable de CNN, DS-CNN (Depth-Wise Separable Convolutional Neural Networks), ya que son más rápidas y producen resultados equivalentes a los modelos originales. En cuanto a LSTM, se utilizó su forma original y su variante bidireccional, Bi-LSTM, ampliamente utilizada en la comunidad científica, para comparar su rendimiento. De esta manera, se formaron cinco modelos diferentes (tres individuales y dos híbridos, respectivamente): DS-CNN, LSTM, Bi-LSTM, (DS-CNN)-LSTM y (DS-CNN)-(Bi-LSTM). Entre ellos, las variantes híbridas fueron las que arrojaron los mejores resultados, con un pico de precisión de 94.80% en el mejor caso encontrado.

- Presentación de una nueva arquitectura para explotar los datos propuestos. Para evitar lidiar con los problemas presentes en la naturaleza de cada sensor,

se decidió tratar los datos de cada uno de ellos de forma independiente. De este modo, se formó una rama independiente para cada uno de los conjuntos de datos correspondientes a cada sensor, utilizando uno de los modelos comentados en el párrafo anterior en cada una de dichas ramas. Así, se entrenan cuatro modelos de forma simultánea, cada uno con los datos correspondientes a uno de los sensores utilizados, pero empleando la misma configuración técnica. Después, las salidas de cada una de estas ramas se concatenan, resultando en un único valor final con la clasificación de los datos.

En resumen, la Tabla B.1 proporciona una visión general de los mejores resultados logrados por cada algoritmo implementado para el conjunto de datos propuesto en esta Tesis, según su mejor tamaño de ventana. Los algoritmos de deep learning demostraron ser los más efectivos para obtener resultados favorables, destacando al mismo tiempo el notable desempeño de los algoritmos más tradicionales basados en árboles. Entre ellos, los casos que involucran a LSTM fueron los más exitosos, demostrando ser la opción más adecuada para el conjunto de datos dado, entre todas las opciones estudiadas.

Algorithm	**Window size (s)**	**Accuracy**
SVM	80	86.56% ±11.30%
DT	20	89.99% ±6.13%
MLP	40	86.85% ±6.12%
NB	80	83.27% ±7.78%
KNN	80	89.02% ±8.00%
RF	80	92.97% ±6.23%
XGB	70	92.23% ±7.30%
DS-CNN	90	90.70% ±7.29%
LSTM	90	93.52% ±5.59%
Bi-LSTM	90	93.09% ±5.10%
(DS-CNN)-LSTM	**90**	**94.80% ± 4.09%**
(DS-CNN)-(Bi-LSTM)	90	94.16% ±5.06%

Table B.1: Comparación de los mejores resultados obtenidos en el conjunto de datos propuesto, con los métodos utilizados durante la Tesis y para el tamaño de ventana que obtuvo el mejor rendimiento.

B.4 Conclusiones

El objetivo principal de esta Tesis era el de orientar la investigación actual en el campo del reconocimiento de actividades humanas hacia entornos de la vida real.

Dada la inexistencia de datos enfocados en este sentido, era fundamental recolectar un nuevo conjunto de datos con el que poder iniciar dicho proceso. Además, también era crucial tratar con datos provenientes de los sensores incluidos en los smartphones de ahora, dado su popular uso en el mundo desarrollado actual. Por ello, la primera contribución de esta Tesis se centró en este punto. El conjunto de datos resultante contiene información de 19 individuos diferentes, cada uno con diferentes peculiaridades físicas y formas de utilizar su smartphone personal y que realizaron una serie de actividades con casi total libertad. En cuanto a los sensores, se utilizaron el acelerómetro, el giroscopio, el magnetómetro y el GPS. De este modo, se consiguió que la información recolectada tuviese suficiente variabilidad y fuera lo bastante realista para poder transferir los futuros hallazgos hacia problemáticas del mundo real.

Sin embargo, al tratarse de una recolección de datos muy diferente a todas las realizadas hasta el momento en este ámbito, aparecieron una serie de imprevistos. Primero, no todos los individuos que participaron en el estudio disponían de todos los sensores necesarios en su smartphone. Esto hace que en algunos casos no se pueda disponer de los datos provenientes de algunos pocos individuos. En segundo lugar, la frecuencia de muestreo no es siempre la misma para cada sensor. Aún habiendo fijado el máximo valor permitido por Android en este aspecto, existen algunos casos en los que dicha frecuencia cambia, lo que obliga a destinar un esfuerzo mayor al procesamiento de los datos. Por último lugar, el conjunto de datos resultante presenta cierto desbalanceo hacia una de las cuatro actividades estudiadas. Aunque la cantidad de patrones en el resto de acciones es más que suficiente para llevar a cabo una correcta clasificación de los datos, es algo que hay que considerar en los desarrollos que se hagan sobre dichos datos.

Con todo, la comunidad científica dispone actualmente de un conjunto de datos formado en un entorno de la vida real. Aún teniendo en cuenta las problemáticas descritas en el anterior párrafo, éstas podrían ser algo habitual en el resto de conjuntos de datos futuros que podrían surgir siguiendo una orientación similar. Por lo tanto, aunque requieran un esfuerzo de investigación mayor, podrían resultar en hallazgos valiosos para el desarrollo de HAR en esa dirección. De este modo, los investigadores pueden aprovechar dicha información y realizar sus propios desarrollos, optimizándolos y pudiéndolos enfocar con éxito en escenarios más realistas.

Siguiendo esta pauta, se llevó a cabo una comparación exhaustiva de múltiples algoritmos de inteligencia artificial, con numerosas combinaciones de hiperparámetros y características, así como diferentes tamaños de ventana. En cuanto a los hiperparámetros, se consiguió mucha información sobre qué casos favorecían más la clasificación según el algoritmo, aunque con cierta arbitrariedad según el parámetro estudiado. Con respecto a las características, se realizaron experimentos con dos conjuntos: las clásicas, basadas en estadísticas (incluyendo la media y la desviación típica), y otro grupo que se relaciona más con los aspectos distintivos de las señales recopiladas. Desafortunadamente, los resultados no fueron del todo concluyentes,

por lo que no queda claro cuál sería el caso más adecuado para este conjunto de datos. Por último, se pudo afirmar que los tamaños de ventana más grandes (alrededor de un minuto) influían positivamente en la clasificación final.

En cualquier caso, todos los experimentos realizados sobre dicho conjunto de datos validaron la posibilidad de llevar a cabo la orientación propuesta en esta Tesis. La clasificación de las actividades estudiadas es superior al 90% en la mayoría de los casos, llegando hasta un 94.80% en el mejor caso encontrado. En este sentido, los algoritmos que mejor funcionaron fueron los basados en deep learning, con el modelo híbrido resultante de unir redes de neuronas convolucionales y redes de neuronas recurrentes basadas en la técnica de Long Short-Term Memory. También cabe destacar el rendimiento de los modelos basados en árboles, especialmente Random Forest, que obtuvieron resultados muy cercanos a los arrojados por los algoritmos de deep learning. De todas formas, en prácticamente todos los casos, la máxima precisión se alcanzó con los tamaños de ventana más grandes utilizados en esta Tesis (entre 60 y 90 segundos). Con tamaños de ventana más pequeños, los sistemas de inteligencia artificial implementados no fueron tan hábiles a la hora de discernir las características distintivas de las actividades de larga duración aquí estudiadas. Además, cabe destacar el hecho de que los algoritmos basados en LSTM hayan arrojado los mejores resultados. Estas redes son famosas por su eficacia en el manejo de series temporales de datos, lo que, junto con el comportamiento observado con diferentes tamaños de ventana, demuestra la importancia de abordar adecuadamente la temporalidad de los datos a la hora de clasificar este tipo de acciones. Esto, a su vez, constituye uno de los retos más comunes en HAR. Aún así, todavía existe potencial para mejorar los resultados. Al fin y al cabo, los desarrollos llevados a cabo en esta Tesis, aunque diversos, conforman solamente una parte de la gran variabilidad que podría surgir a lo largo de los años.

Finalmente, también cabe destacar que, durante los años de desarrollo de esta Tesis, han aparecido más trabajos que han hecho uso de los datos aquí publicados [Hnoohom et al., 2020, Hu et al., 2023], también con buenos resultados. Al mismo tiempo, también están empezando a surgir otros conjuntos de datos con la misma orientación buscada aquí [Quan et al., 2022]. Teniendo en cuenta toda la investigación realizada durante esta Tesis, sumado a los últimos puntos aquí dispuestos, no cabe duda de que el proyecto ha sido un éxito. Con los posibles avances que vayan surgiendo en los próximos años, es muy posible que se pueda transferir directamente, por fin, todo el conocimiento adquirido en HAR en los últimos años hacia el mundo real.

B.5 Trabajo futuro

Aunque todo el trabajo realizado durante el desarrollo de esta Tesis fuese finalmente exitoso, es cierto que fueron apareciendo algunos imprevistos que hubo que ir resolviendo. No importa cuán eficiente y satisfactoria sea la solución aportada, que

siempre habrá margen de mejora. Por ello, a continuación se enumeran algunas ideas que podrían ser profundizadas y mejoradas en líneas de investigación futuras:

- Diferentes maneras de procesar los datos. Dada la problemática de la inconsistencia de la frecuencia de muestreo de cada sensor en el conjunto de datos aportado, las soluciones en este sentido pueden ser muy diversas. Durante el desarrollo de esta Tesis, se llevó a cabo una solución experimental con la que corregir dicho problema y poder continuar con la implementación de los modelos. Aunque el resultado se considera satisfactorio, es probable que con diferentes aproximaciones el resultado sea más positivo que el propuesto aquí. Por otro lado, aunque durante el desarrollo de la Tesis se estudiaron múltiples tamaños de ventana, no dejaron de ser *ad hoc* a dichas soluciones propuestas. Por estos motivos, una exploración más en profundidad en estos temas podría resultar en un mejor rendimiento de los modelos finales que realicen la clasificación de los datos procesados previamente.

- Nuevos conjuntos de características. Todas las características calculadas en los trabajos realizados durante el desarrollo de esta Tesis se centraron en el dominio del tiempo, dadas las problemáticas de los datos comentadas anteriormente. En caso de resolver dichos problemas con éxito, es posible que otras características más centradas en el dominio de la frecuencia pudiesen ser positivas para mejorar los resultados. De hecho, numerosos estudios en HAR aplicaron dichas características en sus trabajos, con buen desenlace [Seto et al., 2015, Sousa et al., 2017]. Por lo tanto, es factible pensar que funcionarán de manera similar con la orientación propuesta en esta Tesis.

- Otros algoritmos y configuraciones. A día de hoy, los algoritmos de inteligencia artificial disponibles son muy numerosos y diferentes. Además, la combinatoria de todos los hiperparámetros que influyen en su rendimiento suele ser muy amplia. Asimismo, dependiendo de la arquitectura del modelo final y de los modelos híbridos que resultan de combinarlos entre sí, los resultados pueden ser muy diferentes. Aunque en esta Tesis se considera que se ha hecho una selección acertada de todas estas cuestiones, no deja de ser algo limitado con mucho margen de mejora. Es posible que otro tipo de configuraciones resulten en una clasificación más precisa de los datos. De hecho, la aplicación de modelos basados en la arquitectura Transformer [Vaswani et al., 2017] podría ser beneficiosa, considerando sus recientes excelentes resultados en numerosas áreas relacionadas con la inteligencia artificial.

- Nuevos datos. Como ya se comentó anteriormente, las actividades estudiadas en el conjunto de datos recolectado son bastante genéricas. Por ello, una vez demostrado su potencial y vista la viabilidad de la nueva orientación aquí propuesta, quizá sea momento de hilar más fino y estudiar nuevas actividades.

De esta manera, la idea consistiría en elaborar nuevos conjuntos de datos en los que se estudien actividades más concretas. Dichas acciones podrían ser similares a las estudiadas en otros trabajos anteriores en HAR, como levantar la mano, ponerse de pie o subir escaleras. La diferencia radicaría en la manera de recolectar los datos, que debería ser con la mayor libertad y flexibilidad posible para acercarla tanto como sea factible al mundo real. De este modo, la aplicabilidad final de los sistemas que podrían resultar de dichos trabajos sería mucho más práctica y directa.

- Probar los modelos desarrollados en distintos conjuntos de datos de la vida real. Siguiendo la línea del punto anterior, el uso de nuevos datos podría aportar información adicional valiosa. A medida que se disponga de nuevos conjuntos de datos recogidos en escenarios del mundo real, los modelos desarrollados a lo largo de esta Tesis podrían ser analizados en contextos distintos a los estudiados aquí. De este modo, se podría seguir avanzando en esta línea de investigación, reafirmando potencialmente las conclusiones de este trabajo y agilizando la transferencia de todos los resultados obtenidos.