



Universidad de Valladolid

PROGRAMA DE DOCTORADO EN INGENIERÍA INDUSTRIAL

TESIS DOCTORAL:

**Monitorización y reconocimiento de
actividades de personas mayores en entornos
residenciales**

Presentada por **Raúl Gómez Ramos** para optar al grado
de
Doctor por la Universidad de Valladolid

Dirigida por:
Eduardo Zalama Casanova
Jaime Gómez García-Bermejo
Jaime Duque Domingo

La inteligencia es la capacidad de adaptarse a los cambios.

Stephen Hawking

Agradecimientos

Parece que fue ayer cuando decidí embarcarme en el complicado y apasionante mundo de la investigación, la innovación y el desarrollo. Después de mucho esfuerzo, retos que parecían no tener solución y, pandemia incluida, puedo decir que lo he conseguido. Quiero agradecer la oportunidad que me brindaron Eduardo Zalama, Jaime Gómez y Jaime Duque de formar parte de su grupo de investigación. Es todo un honor compartir trabajos y experiencias con unos auténticos referentes en el ámbito de la robótica y la inteligencia artificial. Tengo que agradecer a mi familia, M^a Carmen, Jesús, Sonia, y a mi pareja, Lorena, todo el apoyo y el cariño que me habéis aportado a lo largo de todos mis estudios. Vuestro sacrificio y esfuerzo me han servido de ejemplo para llegar hasta aquí. A mis amigos y compañeros del Centro Tecnológico CARTIF, gracias por vuestro apoyo y por todo lo que me habéis enseñado.

Resumen

Esta tesis doctoral presenta un sistema de monitorización de las principales actividades de la vida diaria que las personas mayores pueden realizar en sus hogares. Este sistema se ha realizado mediante la aplicación de tecnología no intrusiva que no captura información personal sobre los usuarios. Las técnicas utilizadas para detectar las actividades se pueden clasificar dentro de dos grupos: métodos de aprendizaje supervisado y métodos de aprendizaje no supervisado. La tesis parte del análisis de la información de la base de datos pública Milan mediante la aplicación de métodos de aprendizaje supervisado utilizando modelos de redes neuronales recurrentes y diferentes técnicas de filtrado y mejora de la calidad de los datos, dando como resultado un sistema capaz de detectar en tiempo real las actividades realizadas por un usuario dentro de su vivienda con una alta precisión.

En la tesis se propone la realización de una base de datos propia que almacene la información recogida de una vivienda en la que residen dos usuarios de forma simultánea mediante la aplicación de tres grupos tecnológicos: una red de sensores no intrusivos, un sistema de localización de los usuarios dentro de la vivienda y la información proporcionada por unas pulseras de actividad. Esta base de datos recibe el nombre de SDHAR-HOME y recoge la información de un total de 18 actividades diferentes. Se propone un método de aprendizaje supervisado para analizar los datos de SDHAR-HOME basado en una arquitectura personalizada en la que se aplican tres capas de redes neuronales diferentes: RNN, LSTM y GRU. Se genera un modelo diferente y personalizado para cada usuario, que permite la detección de actividades en tiempo real. Se ha elaborado otro método de detección de ADL's mediante la aplicación de redes neuronales CNN junto con un método de validación CVV-SV, el cual arroja mejores resultados que los métodos previamente propuestos. Finalmen-

te, se ha elaborado un modelo de red neuronal mediante la aplicación de modelos transformer y capas de atención que incrementa la precisión y velocidad del sistema. De forma paralela, la tesis presenta la elaboración de un método de aprendizaje no supervisado, que no necesita una etapa de etiquetado de actividades para entrenar el modelo, lo que genera un sistema replicable en un mayor número de viviendas. Este método está basado en la combinación de dos HMM: uno para filtrar las estancias de los usuarios y otro para proporcionar la actividad.

Para ofrecer servicio y funcionalidad al usuario, se ha desarrollado un ecosistema de inteligencia ambiental que integra un robot social, el cual ofrece diferentes funciones al usuario: ejercicios, juegos, videollamadas o búsqueda y detección en el hogar. El robot se encarga de proporcionar apoyo al usuario en su casa, le ayuda a combatir la soledad y mejora su bienestar y calidad de vida, aportando seguridad en su día a día. El ecosistema se ha probado durante 2 meses de duración en una vivienda real, permitiendo la detección de situaciones peligrosas y elaborando planes personalizados utilizando el robot social. Este sistema establece un mecanismo de intervención temprana, permitiendo de esta manera favorecer y promover la vida independiente de las personas mayores, brindando mayor calidad de vida y seguridad a este grupo en sus hogares.

Abstract

This doctoral thesis presents a monitoring system for the main activities of daily living that older people can perform in their homes. This system has been developed using non-intrusive technology that does not capture personal information about the users. The techniques used to detect activities can be classified into two groups: supervised learning methods and unsupervised learning methods. The thesis begins with the analysis of information from the Milan public database by applying supervised learning methods using recurrent neural network models and different filtering and data quality improvement techniques, resulting in a system capable of real-time detection of activities performed by a user within their home with high precision.

The thesis proposes the creation of a proprietary database, named SDHAR-HOME, which stores information collected from a household where two users residing simultaneously using three technological groups: a network of non-intrusive sensors, a user localization system within the home, and information provided by activity bracelets. This database collects information on a total of 18 different activities. A supervised learning method is proposed to analyse the data from SDHAR-HOME based on a customised architecture applying three layers of different neural networks: RNN, LSTM, and GRU. A different and personalised model is generated for each user, enabling real-time activity detection. Another method for detecting ADLs is developed using CNN neural networks along with a CVV-SV validation method, which provides better results than the previously proposed methods. Finally, a neural network model is developed using transformer models and attention layers to increase the accuracy and speed of the system. In parallel, the thesis presents the development of an unsupervised learning method, which does not require a labeling stage for activities to train the model, thus generating a replicable system in a larger

number of households. This method is based on the combination of two HMMs: one to filter user stays and another to provide activity information.

An ambient intelligence ecosystem has been developed to provide service and functionality to the user. The system integrates a social robot, which offers several functions to the user: exercises, games, video calls, or home search and detection. The robot is responsible for providing support to the user at home, helping to combat loneliness and improve their well-being and quality of life, while also enhancing security in their daily life. The ecosystem has been tested for a duration of 2 months in a real household, allowing the detection of dangerous situations and creating personalised plans using the social robot. This system establishes a mechanism for early intervention, thus promoting and facilitating the independent living of elderly individuals, providing them with a higher quality of life and greater security in their homes.

Índice general

1. Introducción	1
1.1. Motivación	3
1.2. Objetivos	4
1.3. Metodología	5
1.4. Marco de realización	6
1.5. Contribuciones y trabajos publicados	7
1.6. Estructura de la tesis	10
2. Estado de los sistemas de reconocimiento de actividades	13
2.1. Métodos de monitorización de personas mayores	13
2.1.1. Sistemas de reconocimiento por sensores no intrusivos	18
2.1.2. Sistemas de reconocimiento por imágenes	20
2.1.3. Sistemas de reconocimiento por sensores vestibles	23
2.2. Métodos de posicionamiento en interiores	25
2.2.1. Sistemas de posicionamiento por balizas Bluetooth	28
2.2.2. Sistemas de posicionamiento por análisis de señales WiFi	30
2.3. Análisis de patrones de comportamiento y anomalías	30
3. Tecnología propuesta para la monitorización de personas mayores	35
3.1. Diseño del despliegue de monitorización	35
3.2. Sensores no intrusivos	37
3.2.1. Sensores con señales de tipo evento	40
3.2.2. Sensores con señales de tipo ambiental	42
3.3. Balizas Bluetooth	43

3.4.	Pulseras de actividad	45
3.5.	Arquitectura del despliegue	47
4.	Diseño del modelo de reconocimiento de actividades	51
4.1.	Reconocimiento de actividades mediante aprendizaje supervisado . .	51
4.1.1.	Redes neuronales recurrentes (RNN)	54
4.1.2.	Redes neuronales Long Short-Term Memory (LSTM)	55
4.1.3.	Redes neuronales Gated Recurrent Unit (GRU)	59
4.1.4.	Redes neuronales convolucionales (CNN)	61
4.1.5.	Modelos transformadores	63
4.2.	Reconocimiento de actividades mediante aprendizaje no supervisado .	65
4.2.1.	Modelos ocultos de Markov (HMM)	67
4.2.2.	Filtros Bayesianos	69
4.2.3.	Algoritmos de propagación y retro propagación	71
4.2.4.	Algoritmo de Viterbi	74
4.3.	Modelos propuestos para el reconocimiento de actividades	77
4.3.1.	Modelo basado en redes LSTM para viviendas con un usuario	77
4.3.2.	Modelo basado en redes CNN, LSTM y GRU para viviendas con múltiples usuarios	85
4.3.3.	Modelo CVV-SV para viviendas con múltiples usuarios	89
4.3.4.	Modelo transformador para viviendas con múltiples usuarios .	93
4.3.5.	Modelo HMM combinatorio para viviendas con múltiples usua- rios	95
4.3.6.	Modelo HMM jerárquico para viviendas con múltiples usuarios	102
5.	Ecosistema de inteligencia ambiental para el reconocimiento de ac- tividades de personas mayores	105
5.1.	Robótica social aplicada al cuidado de personas mayores	106
5.2.	Arquitectura propuesta para el ecosistema de inteligencia ambiental .	108
5.3.	Soluciones elaboradas para la interacción con el usuario final	112
6.	Experimentación y discusión	117
6.1.	Experimentos de métodos de aprendizaje supervisado	117
6.2.	Experimentos de métodos de aprendizaje no supervisado	136

6.3. Experimentos y resultados del ecosistema de inteligencia ambiental .	142
6.4. Discusión	158
7. Conclusiones y líneas futuras	169
7.1. Conclusiones	169
7.2. Aplicaciones del método propuesto	172
7.3. Trabajos futuros	173
Glosario	175
Bibliografía	179

Índice de figuras

2.1. Tecnologías utilizadas para el reconocimiento de actividades.	18
2.2. Sensores vestibles y su aplicación.	23
2.3. Método de triangulación para obtener la posición de un usuario en función de las distancias Euclídeas a cada baliza BLE.	29
3.1. Red ZigBee: nodos de datos de sensores (3,4 y 5) comunican informa- ción al concentrador (0) usando pasarelas (1 y 2).	39
3.2. Esquema general de la vivienda con la distribución de los sensores no intrusivos.	41
3.3. Ejemplo de sensores con señales de tipo evento.	42
3.4. Ejemplo de sensores con señales de tipo ambiental.	44
3.5. Baliza ESP32: Montaje final y chip.	45
3.6. Diagrama general de los diferentes componentes del despliegue del sistema de monitorización.	47
3.7. Arquitectura del sistema de monitorización y comunicaciones entre los componentes.	48
4.1. Esquema de funcionamiento de un método de aprendizaje supervisado.	52
4.2. Contenido de una red neuronal RNN.	55
4.3. Contenido de una red neuronal LSTM.	56
4.4. Arquitectura de una red neuronal LSTM bidireccional.	59
4.5. Contenido de una red neuronal GRU.	60
4.6. Contenido de una red neuronal CNN.	62
4.7. Esquema de funcionamiento de un modelo transformador.	64

4.8. Esquema de funcionamiento de un método de aprendizaje no supervisado.	66
4.9. Caso de uso de una máquina de 3 estados genérica e implementación del algoritmo de propagación y retro propagación.	72
4.10. Caso de uso de una máquina de 4 estados genérica e implementación del algoritmo de Viterbi.	74
4.11. Etapas de procesamiento y tratamiento del conjunto de datos de entrada para algoritmos de un sólo residente.	81
4.12. Arquitectura de red neuronal basada en capas LSTM bidireccionales para el reconocimiento de ADL's en viviendas con un residente. . . .	82
4.13. Diagrama general del sistema de monitorización de viviendas de un sólo residente: Procesamiento de datos, filtrado y red neuronal.	84
4.14. Proceso de distribución aleatoria de base de datos SDHAR-HOME en conjuntos de datos de entrenamiento, test y validación.	88
4.15. Arquitectura de red neuronal basada en capas de análisis de series temporales para el reconocimiento de ADL's en viviendas con múltiples residentes.	89
4.16. Esquema del modelo de convolución propuesto para realizar HAR utilizando la base de datos SDHAR-HOME.	90
4.17. Esquema del modelo CVV propuesto para clasificar las ADL's del sistema HAR combinado con CNN.	91
4.18. Arquitectura propuesta mediante la aplicación de modelos de redes neuronales transformer basados en capas de múltiples cabezas de atención.	94
4.19. Esquema de la morfología de la vivienda de SDHAR-HOME y las posibles transiciones entre estancias.	98
4.20. Algoritmo basado en HMM para la localización en interiores.	98
4.21. Algoritmo basado en HMM para la detección de ADL's en viviendas con múltiples usuarios.	100
4.22. Arquitectura de modelo de aprendizaje no supervisado basado en HMM's para el reconocimiento de ADL's en viviendas con múltiples residentes.	102

4.23. Modelo de HMMs jerárquico empleado en los despliegues del sistema de reconocimiento de actividades en hogares reales.	103
5.1. Robot social Pepper interactuando con usuarios de la residencia Lacort en Viana de Cega (Valladolid).	107
5.2. Diagrama de la arquitectura propuesta para el ecosistema de inteligencia ambiental para el cuidado de personas mayores.	108
5.3. Componentes del sistema de inteligencia ambiental: módulo HAR, módulo de métricas y anomalías y sistema cognitivo.	113
5.4. Funcionalidades implementadas en el robot social: información, utilidades, ocio y ejercicios.	116
6.1. Evolución de las curvas de aprendizaje del modelo de red neuronal basado en capas LSTM bidireccionales.	119
6.2. Matriz de confusión para el modelo de red neuronal LSTM en viviendas con un usuario.	121
6.3. Evolución de las curvas de aprendizaje del usuario 1 mediante el método de aprendizaje supervisado para viviendas con varios residentes.	124
6.4. Evolución de las curvas de aprendizaje del usuario 2 mediante el método de aprendizaje supervisado para viviendas con varios residentes.	125
6.5. Matriz de confusión para el modelo de red neuronal GRU en viviendas con múltiples usuarios: usuario 1.	128
6.6. Matriz de confusión para el modelo de red neuronal LSTM en viviendas con múltiples usuarios: usuario 2.	129
6.7. Entrenamientos de los dos primeros agrupamientos de k para CVV-SV utilizando <i>early-stopping</i> (Usuario 1).	130
6.8. Matriz de confusión para el modelo CNN con CVV-SV en viviendas con múltiples usuarios: usuario 1.	132
6.9. Matriz de confusión para el modelo CNN con CVV-SV en viviendas con múltiples usuarios: usuario 2.	133
6.10. Matriz de confusión para el modelo transformer en viviendas con múltiples usuarios: usuario 1.	135

6.11. Matriz de confusión para el modelo transformer en viviendas con múltiples usuarios: usuario 2.	136
6.12. Ejemplos de detección de actividad mediante la implementación de métodos de aprendizaje no supervisado basado en las señales de los sensores.	138
6.13. Curvas ROC uno-vs-todos obtenidas de la aplicación de los métodos de aprendizaje no supervisado para la base de datos SDHAR-HOME.	141
6.14. Matriz de confusión para el modelo de aprendizaje no supervisado: usuario 1.	142
6.15. Matriz de confusión para el modelo de aprendizaje no supervisado: usuario 2.	143
6.16. Diagrama de secuencia de actuación del robot social durante el plan de detección de emergencia.	144
6.17. Diagrama de ejecución de planes: despertar e inactividad prolongada.	145
6.18. Robot social Temi V3 asistiendo al usuario en 3 situaciones diferentes: (a) proponiendo ejercicio físico, (b) proponiendo juegos de memoria y (c) asistiendo en la cocina.	148
6.19. Aspecto general de la aplicación web desarrollada para acceder a la información de los usuarios de la vivienda monitorizada.	149
6.20. Diagrama de Gantt resultado del sistema de detección de actividades para una vivienda real.	150
6.21. Diagramas de frecuencias de actividades para las 5 viviendas reales analizadas durante 2 meses.	151
6.22. Gráficas de duración de actividades por día para las 5 viviendas reales analizadas durante 2 meses.	152
6.23. Diagramas de caja y bigotes de duración de actividades por día para las 5 viviendas reales analizadas durante 2 meses.	154
6.24. Resumen de información relativa a una vivienda durante un día completo para realizar el etiquetado manual: gráficas de sensores y salida del modelo.	155
6.25. Matrices de confusión de las actividades detectadas para las 5 viviendas reales.	159

Índice de tablas

2.1. Sensores, medición, ventajas y desventajas de las tres categorías principales de los métodos de reconocimiento basados en sensores: sensores vestibles (SV), sensores de objetos (SdO) y sensores ambientales (SA).	17
2.2. Principales bases de datos de libre acceso de sistemas de reconocimiento por sensores no intrusivos.	21
2.3. Principales bases de datos de libre acceso de sistemas de reconocimiento por imágenes.	22
2.4. Principales bases de datos de libre acceso de sistemas de reconocimiento por sensores vestibles.	25
2.5. Comparación de estándares de comunicación dentro de los sistemas basados en radiofrecuencia para la localización de usuarios en interiores.	27
2.6. Resumen de principales métodos de detección de anomalías mediante algoritmos ML.	34
3.1. Conjunto de actividades que el sistema es capaz de detectar.	36
3.2. Estudio de los sensores necesarios para detectar las actividades propuestas.	38
3.3. Resumen de los sensores de la instalación: tipos y referencias.	40
4.1. Principales características de la base de datos Milan (CASAS).	78
4.2. Ejemplo de registro de la base de datos Milan durante la realización de una ADL.	79
5.1. Resumen de incidencias, métricas y acciones asociadas del ecosistema de inteligencia ambiental.	115

6.1.	Resultados del modelo supervisado en viviendas con un usuario. . . .	120
6.2.	Resumen del entrenamiento de modelos de aprendizaje supervisado para viviendas con varios usuarios.	126
6.3.	Resultados del modelo supervisado en viviendas con varios usuarios (Usuario 1 - Usuario 2).	127
6.4.	Resultados obtenidos del entrenamiento del modelo basado en redes neuronales CNN con CVV.	131
6.5.	Resultados obtenidos del entrenamiento del modelo basado en redes neuronales transformer con 4 tipos de mapeo: lineal, GRU, LSTM y CNN.	134
6.6.	Duración media de cada una de las ADL's de la base de datos SDHAR-HOME para cada usuario.	139
6.7.	Resultados del modelo no supervisado en viviendas con varios usuarios (Usuario 1 - Usuario 2).	140
6.8.	Resultados del sistema de detección de anomalías y efectividad del sistema cognitivo.	147
6.9.	Media y desviación típica (en horas) de duración de actividades para las 5 viviendas reales analizadas durante 2 meses.	156
6.10.	Resultados de precision, recall y F1-Score de las actividades detectadas para las 5 viviendas reales.	157
6.11.	Resumen de los diferentes métodos de reconocimiento de actividades empleados con la base de datos SDHAR-HOME.	162
6.12.	Comparación de los principales métodos de reconocimiento de actividades mediante técnicas de aprendizaje supervisado.	166
6.13.	Comparación de los principales métodos de reconocimiento de actividades mediante técnicas de aprendizaje no supervisado.	168

Capítulo 1

Introducción

Los sistemas de reconocimiento de actividades (HAR, del inglés *Human Activity Recognition*) son técnicas utilizadas para identificar un conjunto de acciones ejecutadas por uno o varios actores dentro de un entorno [1]. El propósito principal de estos sistemas es reconocer las actividades más comunes que pueden efectuar las personas en escenarios reales [2] con la finalidad de obtener información sobre las diferentes situaciones y circunstancias que pueden darse dentro de una casa. En los últimos años, realizar HAR se ha convertido en un objetivo muy importante en el campo de la computación y en el análisis de señales [3], ya que conduce a una mejora sustancial en diversos campos, como pueden ser la salud, la seguridad o el bienestar. Sin embargo, una de las mayores dificultades que se plantean es conseguir alcanzar este objetivo con la menor carga computacional posible [4], además de conseguir una elevada tasa de acierto en entornos reales [5]. Son muchos los estudios que apuntan que la calidad de vida y la seguridad de las personas mayores o de las personas con cierto nivel de dependencia [6] pueden incrementarse mucho obteniendo sus patrones de comportamiento o si se consigue detectar las diferentes situaciones de peligro que pueden darse dentro del hogar [7].

En la actualidad existen diferentes soluciones tecnológicas para llevar a cabo HAR. El uso de reconocimiento por imágenes proporcionadas por una red de cámaras desplegadas por la casa ofrece buenos resultados [8]. Sin embargo, este tipo de soluciones suelen ser rechazadas por los usuarios finales, ya que compromete en gran medida su privacidad en sus hogares [9]. Del mismo modo ocurre con las soluciones

basadas en el análisis de señales audibles proporcionadas por micrófonos [10]. Otras soluciones proponen realizar HAR analizando las señales proporcionadas por pulseras de actividad o “smartbands” [11]. Sin embargo, las bases de datos construidas con este tipo de señales proporcionan únicamente información sobre actividades físicas, como andar, correr o saltar. Sin embargo, las actividades más interesantes que las personas pueden llevar a cabo en sus casas, y las más interesantes de detectar, son las actividades que se llevan a cabo durante el día a día y que proporcionan información acerca de sus hábitos [12], lo que puede llevar a comprobar si se están siguiendo correctamente las pautas de medicación o si se está comiendo a las horas correctas del día [13].

En la presente tesis se plantea un escenario donde se combinan diferentes tecnologías catalogadas como no intrusivas con el fin de no comprometer la privacidad de los usuarios en sus casas: el análisis de señales proporcionadas por sensores discretos que capturan información del entorno y el análisis de la intensidad Bluetooth generada por el enlace de un sistema de balizas y las pulseras de actividad de los usuarios para conocer su posición dentro de la casa.

Una vez que se cuenta con las señales proporcionadas por la agrupación de las diferentes soluciones tecnológicas, se deben generar modelos de aprendizaje profundo (DL, del inglés *Deep Learning*) con el fin de extraer la actividad que está llevando a cabo cada usuario. Estos modelos se pueden dividir en dos tipos de aprendizaje: supervisado y no supervisado. La tesis lleva a cabo un estudio de los diferentes métodos mediante su aplicación ante datos reales para comprobar su eficiencia, fortalezas y debilidades. Con respecto a los métodos de aprendizaje supervisados, se han realizado diferentes modelos basados en redes neuronales profundas (DNNs, del inglés *Deep Neural Networks*). Para los métodos de aprendizaje no supervisado, se han generado modelos basados en modelos ocultos de Markov (HMM, del inglés *Hidden Markov Models*).

Los resultados obtenidos durante el desarrollo de la tesis muestran una mejora sustancial con respecto a los estudios existentes, además de aportar un mayor número de actividades a detectar y una mejora de la replicabilidad del sistema. Una vez que se cuenta con las actividades y los patrones de comportamiento de los usuarios, el sistema genera una acción en consecuencia. Estas acciones pueden ser recomendaciones

directas al usuario, un efecto sobre la casa o avisos a familiares y/o cuidadores.

1.1. Motivación

En España, el 15.9% de la población vive en zonas rurales cuyos municipios tienen menos de 30 000 habitantes. Concretamente, en la región de Castilla y León, este valor alcanza el 35.9%. De éstos, el 84.2% en Castilla y León y el 59.3% en España reside en municipios de menos de 5000 habitantes, lo que corresponde el 30.23% de la población en Castilla y León y al 9.43% en España.

En la actualidad, el medio rural de España y muchos otros lugares de Europa sufre una serie de problemas estructurales relacionados con el envejecimiento, la despoblación y la falta de servicios públicos, lo que dificulta la provisión de servicios de atención a las personas mayores que viven solas. Esta problemática se agrava en las zonas rurales de Castilla y León debido a su alta tasa de población superior a los 65 años. Además, se espera un intenso crecimiento de este porcentaje. Concretamente, las previsiones indican que en los próximos años, el número de personas mayores de 80 años pase de los 115 000 actuales a 150 000, lo que supone un incremento del 30.4%.

El modelo tradicional de cuidados de las personas mayores se centra en el servicio de ayuda a domicilio, combinado con atención residencial cuando las personas pierden autonomía o requieren apoyos intensos y continuados. Sin embargo, este modelo presenta una serie de debilidades. Los costes de este sistema son muy elevados, sobre todo el porcentaje que necesita atención residencial. No responde al deseo de los usuarios, ya que prefieren vivir en sus casas y su entorno. Además, no ofrecen atención personalizada, lo que genera insatisfacción en los usuarios y/o familiares. La respuesta puede ser lenta y no ejerce función preventiva. Por último, el modelo tradicional de cuidados favorece la dinámica de despoblación del medio rural.

Según la Encuesta Europea de Salud en España del año 2020 [14], el 87% de las personas mayores prefieren vivir en sus casas en lugar de ingresar en centros residenciales. Además, el 88% de las personas mayores en Castilla y León son propietarias de sus viviendas, llegando a alcanzar casi el 100% en el medio rural, lo que provoca un gran sentimiento de afección hacia ellas. Todo esto repercute en la necesidad de

elaborar un modelo de cuidados de personas mayores que favorezca la vida independiente en sus propias viviendas, genere una atención rápida y personalizada y que frene la despoblación de las zonas rurales. Estas necesidades son tratadas mediante la presente tesis.

1.2. Objetivos

El objetivo principal de la presente tesis doctoral es realizar un modelo capaz de reconocer en tiempo real las principales actividades que pueden desempeñar las personas mayores que viven solas en sus casas con el fin de mejorar su seguridad y bienestar y favorecer su independencia. Este objetivo se puede dividir en los siguientes sub-objetivos:

1. Identificación de las principales líneas de investigación actuales sobre inteligencia artificial aplicada al reconocimiento de acciones.
2. Seleccionar una tecnología que proporcione información de valor para poder llevar a cabo el reconocimiento, que no vulnere la privacidad de los usuarios en sus casas y que sea fácil de instalar, y estudiar los parámetros necesarios para conseguir un reconocimiento preciso.
3. Desarrollar un modelo de reconocimiento de actividades capaz de funcionar en tiempo real y que sea totalmente transparente para los usuarios. El modelo no tiene que necesitar la intervención de los usuarios para facilitar la implementación y aceptación del mismo.
4. Conseguir que el sistema sea flexible y adaptable al mayor número de hogares posible. La estructura tiene que ser genérica y modular para que no dependa de las particularidades que pueden darse dentro de los hogares. El sistema tiene que ser capaz de funcionar en hogares donde cohabiten varias personas de forma simultánea y ser susceptible a perturbaciones como visitas de amigos y/o familiares e incluso presencia de mascotas.
5. Elaboración de estrategia de análisis de patrones de comportamiento de los usuarios y ejecución de acciones en consecuencia. El modelo tiene que ser capaz

de detectar e interpretar los patrones de comportamiento de los usuarios, como los horarios a los que se realizan las actividades, para poder extraer métricas y generar actuaciones que mejoren el bienestar del usuario.

6. Experimentación del modelo en hogares reales durante períodos prolongados para poder anticiparse a las diferentes situaciones peligrosas que puedan surgir como consecuencia de caídas o malos hábitos.
7. Integración del modelo de reconocimiento de actividades dentro de un ecosistema de inteligencia ambiental para la atención y el cuidado de personas mayores con la ayuda de robots sociales. Este modelo tiene que ser capaz de facilitar la vida independiente de las personas mayores en sus hogares, con el fin de alargar su permanencia en los mismos. Esto repercute en una mejora de su calidad de vida y su bienestar.

1.3. Metodología

Con el fin de alcanzar los objetivos propuestos, se ha llevado a cabo la siguiente metodología de trabajo:

1. Estudio de la problemática actual sobre la situación de las personas mayores que viven solas en sus casas, a través de distintos proveedores de servicios como puede ser el caso de la Fundación ASPRODES o la Fundación INTRAS. Gracias a este estudio, se han identificado las principales necesidades a cubrir mediante la presente investigación. Con el fin de detectar estas necesidades, se han elaborado una serie de sesiones de cocreación con ambas fundaciones, en la que han participado distintos grupos de usuarios finales aportando ideas y problemáticas detectadas durante su vida diaria.
2. Documentación bibliográfica sobre los principales trabajos de investigación de diferentes autores con objetivos similares a los propuestos en la presente tesis en los últimos años (generalmente desde el año 2019 en adelante). Se ha realizado una recopilación de los principales artículos publicados en revistas de impacto y congresos de alto reconocimiento dentro de la comunidad científica.

3. Estudio sobre la aplicación y funcionamiento del aprendizaje profundo por ordenador. Se ha realizado una amplia documentación e investigación sobre el estado actual de las diferentes técnicas existentes de aprendizaje automático, tanto supervisado como no supervisado, mediante la aplicación de redes neuronales profundas, modelos ocultos de Markov y otras alternativas.
4. Asistencia a eventos orientados al aprendizaje profundo: congresos nacionales e internacionales, cursos online, jornadas de co-creación con los usuarios finales de la solución, realización de publicaciones de los distintos trabajos desarrollados a lo largo de la tesis, para dichos eventos y para revistas de alto valor para la comunidad científica.
5. Desarrollo del contenido de la tesis doctoral. Las soluciones y métodos desarrollados en la presente tesis se han realizado mediante metodología ágil Scrum, definiendo ciclos de trabajo cortos donde poder comprobar los resultados de forma rápida, con pequeños cambios progresivos y permitiendo una detección temprana de ajustes necesarios para dar valor a la solución. El desarrollo del contenido de la tesis puede recogerse en tres etapas: prueba de viabilidad de la idea mediante la aplicación de diferentes técnicas de inteligencia artificial utilizando una base de datos pública, generación de métodos y algoritmos utilizando la instalación de un entorno controlado de un domicilio y, por último, prueba y validación de los métodos usando domicilios reales de personas mayores.

1.4. Marco de realización

La presente tesis doctoral se ha desarrollado de forma conjunta en el Centro Tecnológico CARTIF y en el Departamento de Ingeniería de Sistemas y Automática de la Escuela de Ingenierías Industriales de la Universidad de Valladolid.

El trabajo ha sido financiado gracias a las ayudas para contratos predoctorales para la formación de doctores 2019 PRE2019-088170 vinculada al proyecto de investigación RTI2018-096652-B-I00 del MICINN. Además, esta investigación ha sido parcialmente financiada por el “Programa Retos Investigación del Ministerio de

Ciencia, Innovación y Universidades PID2021-123020OB-I00” y la Gerencia de Servicios Sociales, Consejería de Familia de la Junta de Castilla y León (España) dentro del Plan UE Recuperación, Transformación y Resiliencia de la UE. Este trabajo se ha realizado bajo el proyecto EIAROB: Ecosistema de Inteligencia Ambiental para el apoyo a los cuidados de larga duración en el hogar mediante uso de robots sociales.

1.5. Contribuciones y trabajos publicados

En la presente tesis se propone un modelo para la monitorización de personas mayores en el hogar a partir de sensores ambientales compatibles con los requerimientos deseables de privacidad de la vida cotidiana. La aproximación propuesta supone una mejora respecto a los sistemas tradicionales por cuanto es capaz de detectar patrones de comportamiento y detectar en tiempo real situaciones de riesgo, sin necesidad de un entrenamiento previo supervisado para cada usuario. Además, permite detectar patrones no deseados dentro de la conducta diaria de los usuarios, como por ejemplo alteraciones en las pautas de medicación o falta de alimentación. Por último, puede ser integrado de forma directa en entornos asistidos por robots sociales.

Las principales contribuciones realizadas en la presente tesis se resumen en la siguiente lista:

- Revisión de las tecnologías y metodologías existentes para el reconocimiento de actividades humanas en el hogar, con identificación de sus principales fortalezas y debilidades.
- Instrumentación de una vivienda piloto y elaboración de una base de datos con la información recogida en la misma durante largos periodos de tiempo, para el ensayo y validación de nuestras investigaciones. La base de datos elaborada recibe el nombre de SDHAR-HOME y se corresponde como una de las bases de datos más completas hasta la fecha, orientada a la detección de actividades de personas mayores en sus hogares. Esta base de datos se ha compartido en abierto con la comunidad científica con el fin de que distintos grupos de investigación prueben y mejoren sus algoritmos, permitiendo así enriquecer el valor de la presente tesis.

- Investigación en reconocimiento de actividades mediante aprendizaje supervisado y propuesta de un modelo funcional conexionista orientado al reconocimiento de actividades en tiempo real, por medio de técnicas de aprendizaje profundo. El método propuesto está basado en modelos de redes neuronales recurrentes (RNN), convolucionales (CNN), long short-term memory (LSTM), gated recurrent unit (GRU) y modelos transformer. A su vez, se han implementado técnicas de aprendizaje profundo que, hasta la fecha, no se han aplicado nunca en el reconocimiento de actividades de personas mayores en sus hogares, como el método CVV-SV.
- Investigación en reconocimiento de actividades mediante aprendizaje no supervisado y propuesta de un método basado en el modelado estadístico de secuencias por medio de modelos ocultos de Markov (HMM). El modelo propuesto no necesita una etapa de entrenamiento previa al funcionamiento real, por lo que mejora su alcance y facilita su despliegue en los hogares.
- Adopción de un enfoque general adaptado tanto al caso de un único residente, el más abordado en las investigaciones precedentes, como al de múltiples residentes que conviven simultáneamente en una misma vivienda. Los sistemas de localización en interiores proporcionan información adicional de relevancia, especialmente en este segundo caso.
- Creación de un conjunto de técnicas de equilibrado y filtrado de datos encaminadas a mejorar la precisión de los modelos planteados. Estas técnicas se encargan de equilibrar los modelos desarrollados en la presente tesis para que sean capaces de detectar todas las actividades que las personas mayores pueden realizar en sus viviendas.
- Desarrollo de algoritmos que se encargan de compartir, entre los usuarios de una misma vivienda, los patrones de realización de las actividades durante la etapa de entrenamiento. Estos algoritmos incrementan la precisión de los modelos de reconocimiento de actividades.
- Diseño de una arquitectura de ecosistema ambiental inteligente en la que se combinan sensores, actuadores, robots sociales y los modelos de reconociemien-

to de actividades propuestos, para el cuidado y la atención de las personas mayores.

- Validación de las investigaciones contra bases de datos públicas existentes, habiéndose obtenido tasas de acierto superiores a las reportadas en investigaciones precedentes.
- Validación del modelo propuesto tanto en la vivienda piloto desarrollada en la presente investigación, como en 5 viviendas de usuarios reales, en el marco de una investigación de largo alcance encaminada a una mejora progresiva del bienestar de las personas mayores en nuestra sociedad.

Durante la realización de la tesis, se han publicado un total de 3 artículos en revistas de alto impacto de cuartiles Q1-Q2 del JCR, donde se han detallado los resultados obtenidos en la presente investigación. Las publicaciones son las siguientes:

- “Daily Human Activity Recognition Using Non-Intrusive Sensors” [15] en publicación “Sensors”, MDPI, Agosto de 2021 (Factor de Impacto: 3.900 en SCI-JCR (Q1) en 2022), DOI: 10.3390/s21165270
- “SDHAR-HOME: A Sensor Dataset for Human Activity Recognition at Home” [16] en publicación “Sensors”, MDPI, Octubre de 2022 (Factor de Impacto: 3.900 en SCI-JCR (Q1) en 2022), DOI: 10.3390/s22218109. Este artículo pertenece al Special Issue: “Deep Learning Methods for Human Activity Recognition and Emotion Detection”.
- “An Unsupervised Method to Recognise Human Activity at Home Using Non-Intrusive Sensors” [17] en publicación “Electronics”, MDPI, Noviembre de 2023 (Factor de Impacto: 2.900 en SCI-JCR (Q2) en 2022), DOI: 10.3390/electronics12234772. Este artículo pertenece al Special Issue: “Ubiquitous Sensor Networks II”.

A su vez, se ha presentado la siguiente publicación en congreso internacional:

- “Ambient Intelligence Ecosystem for Elderly Pattern Detection and Care Using Social Robots” en el congreso “The 17th IEEE International Conference on Internet of Things (iThings2024)”, Copenhague, Agosto de 2024.

También se ha presentado en los siguientes congresos nacionales:

- “Sistema de monitorización no intrusiva para el reconocimiento de actividades de la vida diaria en entornos multiusuario” en las “Jornadas Nacionales de Robótica, Educación y Bioingeniería”, Málaga, Mayo de 2022.
- “Mejora en la clasificación de actividades mediante redes de convolución y CVV-SV” en las “Jornadas Nacionales de Robótica y Bioingeniería”, Madrid, Junio de 2023.
- “Sistema de monitorización no intrusiva para vivienda de personas mayores” en las “Jornadas Nacionales de Robótica y Bioingeniería”, Madrid, Junio de 2023.
- “Comportamiento de un modelo recurrente-transformador para la detección de actividades humanas mediante sensores desplegados en una vivienda” en el congreso “Simposio de Robótica, Bioingeniería y Visión por Computador 2024”, Badajoz, Mayo de 2024.
- “Detección de actividades mediante modelos ocultos de Markov jerárquicos” en las “Jornadas de Automática 2024”, Málaga, Septiembre de 2024.
- “Segmentación semántica bajo paradigma one-shot learning utilizando SAM y CP-CVV” en las “Jornadas de Automática 2024”, Málaga, Septiembre de 2024.

Por último, esta tesis ha sido finalista del concurso 3MT (Three Minute Thesis) 2022, organizado por la Escuela de Doctorado de la Universidad de Valladolid, con la intervención titulada “Como en casa en ningún sitio”. Otro mérito significativo es haber recibido el “Premio Iberdrola al Mejor Proyecto de Investigación Universitaria 2022” expedido por el Suplemento Innovadores El Mundo en Junio de 2022.

1.6. Estructura de la tesis

La tesis se ha estructurado en un total de seis capítulos. El primero de ellos se corresponde con la presente introducción.

En el Capítulo 2 se recopilan un conjunto de trabajos que comparten diversas similitudes con los objetivos buscados en la presente tesis y que han servido para construir la base de esta investigación. Se exponen trabajos relacionados con sistemas de monitorización de personas mayores existentes, comprendiendo soluciones basadas en reconocimiento por imágenes, señales proporcionadas por pulseras de actividad o sensores no intrusivos. A su vez, se han recogido trabajos cuyo objetivo es posicionar a los usuarios dentro de los hogares mediante diferentes tecnologías como balizas Bluetooth o WiFi. Por último se han recogido trabajos que buscan detectar patrones de comportamiento en usuarios y anomalías dentro de estos patrones.

El Capítulo 3 muestra el diseño del despliegue tecnológico que se ha elaborado para realizar la monitorización de las personas mayores y el reconocimiento de sus actividades diarias.

En el Capítulo 4 se detallan los diferentes métodos de aprendizaje profundo que se han utilizado para elaborar el sistema de reconocimiento de actividades. En este capítulo se realiza una síntesis de los principales métodos de aprendizaje supervisado utilizados para realizar HAR basados en redes neuronales, aportando métodos desarrollados en la presente tesis para abordar el reconocimiento tanto en viviendas con un usuario como en viviendas con múltiples usuarios simultáneos. Este capítulo presenta los diferentes métodos de aprendizaje no supervisado utilizados para realizar HAR en viviendas con múltiples usuarios, aportando sus ventajas e inconvenientes para cada caso.

En el Capítulo 5 se muestra el ecosistema de inteligencia ambiental elaborado como resultado de la aplicación del sistema de reconocimiento de actividades junto con otros módulos funcionales, como puede ser la robótica social. En este Capítulo se puede observar la arquitectura elaborada para proporcionar apoyo a los usuarios en sus viviendas, así como las acciones que toma el sistema para mejorar su calidad de vida y su bienestar en sus hogares.

El Capítulo 6 expone el conjunto de experimentos de los diferentes algoritmos que se han desarrollado durante la elaboración del sistema propuesto, realizando una discusión objetiva de los resultados que se obtienen de estos experimentos.

El Capítulo 7 realiza una discusión de las deducciones y aportaciones obtenidas a lo largo del desarrollo de la tesis, así como de las posibles futuras investigaciones

que se pueden llevar a cabo.

Por último, se incluye un apartado donde se muestran los trabajos referenciados a lo largo de la tesis.

Capítulo 2

Estado de los sistemas de reconocimiento de actividades

En el presente capítulo se aborda el estado del arte de los diferentes métodos de monitorización analizados y utilizados en esta tesis. Se presentan: los principales métodos de monitorización de personas mayores (Sección 2.1), los principales métodos de posicionamiento en interiores (Sección 2.2) y, por último, los sistemas de análisis de patrones de comportamiento de usuarios y sus anomalías (Sección 2.3).

2.1. Métodos de monitorización de personas mayores

La necesidad de implementar sistemas de monitorización de las principales actividades diarias (ADL, del inglés *Activity Daily Living*) que pueden desempeñar las personas mayores en sus casas se ha convertido en un campo de investigación de especial interés. En los últimos años, en muchos países de la Unión Europea, el número de personas mayores que residen solas en sus casas ha aumentado de forma exponencial. Concretamente en España, según un estudio del año 2020, se registraron un total de 4 849 900 personas que viven solas. De este número, un total de 2 131 400 tenían 65 años o más [18]. La tendencia es creciente, puesto que supone un aumento del 6.1 % con respecto a los datos del año anterior. Según Flores y Villegas en [19], vivir solo puede producir sensación de soledad y puede suponer un problema para

las personas mayores debido a la degradación de funciones cognitivas. Acordarse de tareas sencillas, como tomarse las medicinas a la hora correspondiente o tener unas pautas correctas de alimentación puede suponer un problema debido al envejecimiento. Este envejecimiento puede ocasionar cambios en la salud mental de las personas mayores, pudiendo producir pérdidas de memoria y de atención, lo que repercute negativamente en su sensación de bienestar [20].

Debido a esta problemática, surge la necesidad de elaborar un sistema de monitorización de ADL's que se encargue de extraer información sobre el día a día de las personas mayores. Obtener pautas de comportamiento anómalas y situaciones peligrosas en tiempo puede permitir la ejecución de una intervención temprana [21].

Según Chin y Bee en [22], el reconocimiento de actividades se puede clasificar en función de la tecnología utilizada para obtener información sobre el entorno. Siguiendo ese planteamiento, los autores proponen tres grupos tecnológicos:

1. Cámaras RGB: Estos sistemas aportan mucha información sobre el entorno, ya que pueden analizar la cercanía de los usuarios con objetos relacionados con la actividad que están realizando [23]. Este tipo de tecnología es generalmente rechazado por los usuarios debido a la sensación de ausencia de privacidad que otorga su uso.
2. Sensores de profundidad: Este tipo de sensores también son conocidos como RGBD [24]. Esta tecnología proporciona la distancia a la que se encuentra el usuario del dispositivo o de cualquier otro objeto mediante el análisis de la profundidad de los puntos de la imagen utilizando técnicas de luz estructurada o calculando tiempos de vuelo [25]. Esta información es útil a la hora de realizar la monitorización, ya que se pueden obtener posturas y movimientos. Estos sensores normalmente van acompañados de la captación de imágenes RGB. Un ejemplo de dispositivo puede ser las cámaras Kinect de Microsoft [26].
3. Sensores vestibles. Para realizar la monitorización con este tipo de sensor, normalmente es necesario adherir uno o varios de estos dispositivos al cuerpo de los usuarios. Estos dispositivos suelen ir acompañados de los siguientes grupos de sensores: acelerómetros 3D-axiales, giroscopios o etiquetas RFID [27]. Dentro de este grupo de pueden incluir los teléfonos inteligentes o *smartphones*.

Otros autores prefieren generalizar y distinguir unicamente dos grupos de sistemas de monitorización de actividad, tal y como comentan L. Ming Dang et al. en [28]. Los autores dividen los sistemas en: métodos de reconocimiento basados en sensores y métodos de reconocimiento basados en visión. Dentro de la primera categoría, los autores hacen referencia a los sensores vestibles [29], la capacidad de sensorizar objetos cuya intervención se encuentre ligada a una actividad particular [30] [31] y a los sensores ambientales, con el fin de obtener información del entorno donde se está realizando la actividad [32]. En la Tabla 2.1 se puede observar un resumen de los principales componentes de los sistemas de reconocimiento de actividad basados en sensores, indicando sus principales ventajas y desventajas. A su vez, dentro de la categoría de los métodos de reconocimiento basados en visión, los autores realizan la siguiente división: datos obtenidos mediante cámaras RGB y RGB con sensores de profundidad (RGBD) [33] [34]. Estos métodos son fiables y proporcionan resultados generalmente buenos, pero tienen el inconveniente de su dificultad a la hora de desplegar la solución y su baja tasa de aceptación por parte de los usuarios, ya que comprometen en gran medida su privacidad dentro de su propio hogar. A pesar de todo esto, existen numerosas bases de datos públicas que contienen imágenes tomadas de usuarios desarrollando distintas actividades, lo que supone un beneficio a la hora de desarrollar soluciones basadas en DL.

Categoría	Sensor	Medición	Ventajas y Desventajas	
			Ventajas	Desventajas
SV	Acelerómetro	Aceleración (fuerza y gravedad)	Económico, duradero, alta sensibilidad, alta impedancia y alta respuesta en frecuencia.	Error de histéresis, sensible a la temperatura y eficiencia disminuida con el tiempo.

Categoría	Sensor	Medición	Ventajas y Desventajas	
			Ventajas	Desventajas
	Giroscopio	Velocidad angular	Rápido, mide el movimiento rotacional y con mayor resolución en comparación con otros sensores de fuerza o inclinación.	Costoso, dependencia de la rotación de la tierra y no mide el movimiento lineal en ninguna dirección.
	GPS	Información de geo-localización, velocidad y tiempo	Gratuito y mide directamente la posición global tridimensional.	No funciona en interiores.
SdO	RFID	Frecuencia de radio	Fácil de instalar, seguro y puede almacenar hasta 2 KB de datos.	Costoso, sensible a interferencias electromagnéticas externas y de alcance limitado.
	WiFi	Señal inalámbrica	Fácil de instalar, seguro y el rango de señal inalámbrica es mejor que el RFID.	Requiere más recursos.

Categoría	Sensor	Medición	Ventajas y Desventajas	
			Ventajas	Desventajas
SA	Presión	Presión	Alta relación señal-ruido, interfaz en tiempo real y menos intervención del usuario.	Las medidas varían mucho en función de la localización donde esté ubicado el sensor.
	Temperatura	Temperatura	Contacto directo, económico y respuesta rápida.	Mucha inercia en las medidas.
	Sonido	Presión del aire	Asequible y respuesta rápida.	Requiere más memoria, tiene un alcance limitado, rechazo de los usuarios (muy intrusivo).
	Radar	Detección de cualquier objeto dentro de los conos de detección del radar	Proporciona información precisa de distancia y velocidad.	Costoso, baja resolución espacial y alcance limitado.

Tabla 2.1: Sensores, medición, ventajas y desventajas de las tres categorías principales de los métodos de reconocimiento basados en sensores: sensores vestibles (SV), sensores de objetos (SdO) y sensores ambientales (SA).

Según lo visto en trabajos similares al propuesto en la presente tesis, se propone dividir los sistemas de reconocimiento de actividad para personas mayores en función de los siguientes grupos tecnológicos (ver Figura 2.1): sensores no intrusivos, imágenes y sensores vestibles.

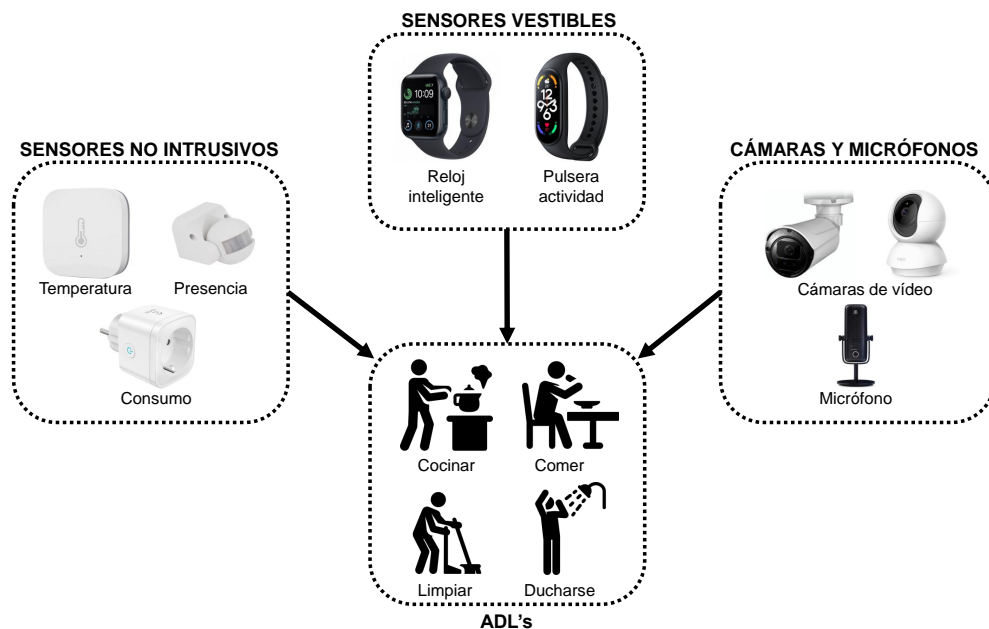


Figura 2.1: Tecnologías utilizadas para el reconocimiento de actividades.

2.1.1. Sistemas de reconocimiento por sensores no intrusivos

Existen numerosos estudios de sistemas de reconocimiento de actividades mediante la implementación de sensores no intrusivos que han repercutido en la generación de bases de datos públicas donde poder consultar la información recogida. Un ejemplo es el trabajo realizado por Tapia et al. en [35], en el cual han desarrollado un Living Lab en Cambridge en el que se cuenta con las siguientes salas: sala de estar, cocina y dormitorio. En este Living Lab han desarrollado diversos estudios (basados en sensores, entrevistas u observaciones directas) con el fin de obtener las ADL's de los diferentes usuarios que colaboran en el estudio. Los autores han utilizado sensores no intrusivos como sensores de temperatura y humedad en las diferentes estancias o sensores de contacto en el refrigerador.

Un estudio que aborda el reconocimiento de actividades en el hogar es el trabajo desarrollado por Alemdar et al. en [36]. Los autores elaboraron una base de datos pública conocida como ARAS (Activity Recognition with Ambient Sensing, del inglés *Activity Recognition with Ambient Sensing*), en la que se centran en recopilar información de dos viviendas reales en las que conviven varios residentes durante

un total de dos meses. Durante este período de tiempo, los autores son capaces de etiquetar 27 actividades diferentes en total utilizando la información recopilada por 20 sensores no intrusivos. Para realizar el reconocimiento de actividades, los autores utilizan varios métodos basados en técnicas DL: modelos basados en HMM y validación cruzada para proporcionar una tasa de reconocimiento media del 61.5% para la vivienda A y un 76.2% para la vivienda B.

Otro estudio similar a ARAS es el trabajo realizado por Cook et al. en [37] (CASAS). Los autores también proponen una base de datos pública en la que recogen información de múltiples usuarios y despliegan una red de sensores no intrusivos. Dentro de este estudio, se han sensorizado un total de 7 viviendas con un número de sensores que abarca desde los 20 sensores no intrusivos hasta los 86. La duración de sus estudios también es variable, ya que proponen una recogida de datos desde los 2 meses hasta un total de 8 meses. Para todas sus instalaciones, los autores realizan un etiquetado de 11 ADL's. Para realizar el reconocimiento de actividades, los autores utilizan algoritmos de SVM (Máquinas de Vectores de Soporte, del inglés *Support Vector Machines*) [38] y son capaces de ofrecer una tasa de reconocimiento del 84.14%.

ARAS y CASAS son los trabajos más similares al propuesto en la presente tesis debido a su requerimiento con respecto a número de usuarios, duración del estudio, número de actividades y número de sensores. La base de datos ARAS no cuenta con un sistema de triangulación de la posición de los usuarios, y tampoco captura señales fisiológicas. Con respecto a la base de datos CASAS, el sistema propuesto por Cook et al. ofrece bajas tasas de reconocimiento para ciertas ADL's como comer o relajarse (inferiores al 50%) mediante el uso de algoritmos SVM. Sin embargo, existen otros trabajos con requerimientos menores. El estudio realizado por Alshammari et al. en [39] revela la creación de la base de datos SIMADL en la que realizan un estudio de una instalación de sensores no intrusivos en los que residen 7 usuarios de forma no simultánea. Para ello, los autores utilizan un total de 29 sensores no intrusivos en un total de 63 días registrando un total de 5 ADL's y anomalías, como dejarse la televisión encendida o el frigorífico abierto. Las actividades registradas durante este período fueron: comer, dormir, trabajar, ocio o aseo. Otro trabajo interesante puede ser el desarrollado en la Universidad de Amsterdam (UvA) en el año 2008 por

Van Kasteren et al. en [40]. En este trabajo, los autores realizaron tres despliegues de 14 sensores (sensores de contacto en puertas o el frigorífico) en los que se recogen un total de 2120 eventos durante 28 días de forma ininterrumpida con el fin de obtener un total de 10 ADL's. Los autores obtienen una tasa de reconocimiento del 79.4% mediante la aplicación de HMM y un 70.8% utilizando métodos de campo aleatorio condicional (CRF, del inglés *Conditional Random Field*). Lago et al. en [41] han desarrollado en su trabajo una base de datos conocida como ContextAct@A4H, compuesta por información de 219 sensores y actuadores durante 1 mes de monitorización de una persona de 28 años. Los autores obtienen una tasa de acierto en el reconocimiento que va del 14% para actividades como lavar los platos, hasta el 98% para actividades como aseo personal, utilizando modelos de lógica temporal basados en reglas programadas. En la Tabla 2.2 se resumen las principales bases de datos de libre acceso con información de sensores no intrusivos junto a sus principales características. Posteriormente, en la Sección 6.1 se va realizar una comparación de los resultados obtenidos en la presente tesis con respecto a otros trabajos existentes que utilicen las mismas bases de datos.

Los sistemas de reconocimiento mediante sensores no intrusivos presentan como ventaja la facilidad en la recogida de información de la vivienda. Normalmente, estos sensores suelen ser fáciles de instalar ya que esta tecnología suele ser inalámbrica. Otra ventaja es que no necesitan intervención por parte de los usuarios, lo que facilita su uso de la vida diaria. Sin embargo, la desventaja de este tipo de tecnología es que no es capaz de discriminar la posición exacta de los usuarios dentro de la vivienda si residen múltiples usuarios de forma simultánea. Del mismo modo, no se puede conocer con exactitud el usuario que está realizando una determinada acción en la vivienda.

2.1.2. Sistemas de reconocimiento por imágenes

Los métodos de reconocimiento de actividades mediante el análisis por imágenes son métodos ampliamente utilizados en la actualidad, aunque son sistemas poco aceptados por parte de los usuarios finales. El principal motivo por el cual los usuarios rechazan este tipo de sistemas se debe a la tecnología empleada para realizar la captura de datos. Estos sistemas se caracterizan por utilizar cámaras (RGB y

Base de datos	Casas	Sensores	Actividades	Duración
ARAS [36]	2	20	27	2 meses
CASAS [37]	7	20-86	11	2-8 meses
SIMADL [39]	1	29	5	63 días
UvA [40]	3	14	10	28 días
ContextAct@A4H [41]	1	219	7	1 mes

Tabla 2.2: Principales bases de datos de libre acceso de sistemas de reconocimiento por sensores no intrusivos.

de profundidad) para captar información sobre la actividad [42]. Generalmente los parámetros más importantes que se recogen de las imágenes son los siguientes [43]: postura de la persona realizando la actividad mediante la recogida de la posición y orientación de su cuerpo [44], objetos que intervienen durante la actividad [45] e intervención de otras personas [46]. Aunque su uso para viviendas particulares sea poco aceptado, existen numerosos estudios que siguen generando bases de datos públicas en la actualidad para generar un amplio volumen de datos para la comunidad científica con el fin de mejorar e implementar los algoritmos DL actuales [47].

Una de las bases de datos con mayor volumen de información y, a su vez, una de las más utilizadas dentro de la comunidad científica es NTU RGB+D [48]. Esta base de datos cuenta con un total de 60 actividades diferentes, en las que 40 clases se corresponden con ADL's. En cambio, 9 clases se corresponden con actividades relacionadas con la salud (como puede ser caídas o estornudos) y otras 11 clases son interacciones entre usuarios (como puede ser besos o abrazos). Los autores han compuesto esta base de datos mediante la captura de imágenes y videos utilizando una cámara Kinect v2. Otra base de datos muy utilizada es UCF 50 dataset [49]. Esta base de datos cuenta con un total de 50 categorías recogidas en vídeos web en malas condiciones, con poca iluminación y con movimientos aleatorios de cámara, lo que supone un desafío mayor a la hora de realizar el reconocimiento de actividad. En la Tabla 2.3 se puede observar un resumen general donde aparecen las principales bases de datos públicas de imágenes de actividades junto con sus principales características.

Existen otras numerosas aplicaciones donde se utilizan los sistemas de reconocimiento por imágenes fuera de su uso en viviendas particulares [55]. Por ejemplo, en el mundo del deporte, los sistemas de reconocimiento por imágenes son muy efectivos.

Base de datos	Nº Actividades	Movimiento de cámara	Fondo
UCF11 [50]	11	Dinámico	Si
UCF50 [49]	50	Dinámico	Si
NTU RGB+D [48]	60	Estático	Si
KTH [51]	6	Estático	Si
Weizmann [52]	10	Estático	No
IXMAS [53]	14	Estático	No
UCF sports [54]	9	Dinámico	Si

Tabla 2.3: Principales bases de datos de libre acceso de sistemas de reconocimiento por imágenes.

Estas técnicas se utilizan para analizar un amplio número de largos partidos para aprender a tomar decisiones [56]. Otro caso es el trabajo realizado por Ullah et al. en [57]. Los autores utilizaron modelos DL basados en redes neuronales convolucionales para extraer información sobre las acciones de los jugadores e implementaron mecanismos para obtener su correlación temporal. Los sistemas de reconocimiento por imágenes también se emplean en el campo de la educación. Lim et al. en [58] fueron capaces de analizar el comportamiento de los estudiantes en las aulas mediante el tratamiento de las imágenes. Si los estudiantes estaban de pie acercándose a los pupitres, el sistema reconoce que el estudiante ha entrado en la clase. Por el contrario si se levanta del pupitre para salir por la puerta, el sistema reconoce que abandona la clase. Por último, es importante resaltar los avances en el campo del entretenimiento. Wang et al. en [59] elaboraron un modelo basados en un clasificador 3D mediante redes neuronales convolucionales para clasificar las acciones realizadas por los actores durante una serie de películas cinematográficas.

Este tipo de tecnología presenta como principal ventaja el gran volumen de información que los dispositivos son capaces de capturar. Analizando la información recogida por una cámara, es posible conocer el patrón de movimientos del usuario, objetos que intervienen en las acciones o participación de varios usuarios. Sin embargo, la principal desventaja de esta tecnología es su nivel de aceptación por parte de los usuarios finales, ya que compromete en gran medida su privacidad dentro de la vivienda.

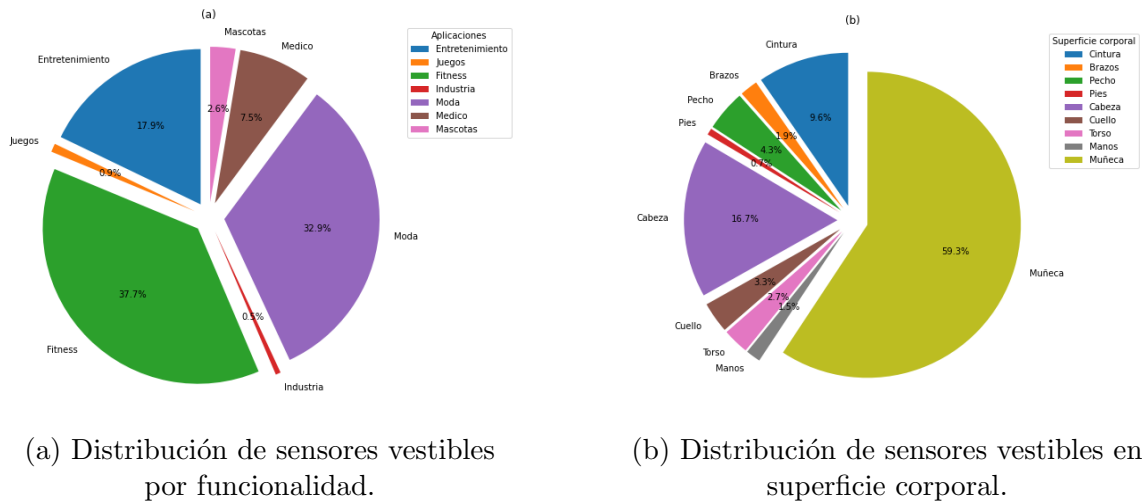


Figura 2.2: Sensores vestibles y su aplicación.

2.1.3. Sistemas de reconocimiento por sensores vestibles

Los sensores vestibles (o dispositivos wearables en inglés) permiten numerosas aplicaciones debido a su extenso uso y la capacidad tecnológica que presentan. Por ejemplo, esta tecnología es ampliamente utilizada en la monitorización del estado de salud, reconocimiento de actividad física o interfaz humano-máquina [60]. Todas estas aplicaciones son posibles debido al elevado número de sensores de baja energía que componen este tipo de dispositivos. En la Figura 2.2 aparece un resumen del reparto del uso de los sensores vestibles en función de su aplicación (Figura 2.2a) y el reparto del uso de los sensores vestibles en función de su distribución en el cuerpo (Figura 2.2b) en función de un estudio realizado por la “Wearable Technology Database” [61] en el año 2022. De estos gráficos se puede deducir que la aplicación más frecuente de este tipo de dispositivos se destina al ejercicio físico, mientras que la parte del cuerpo más utilizada para portar estos dispositivos son las muñecas.

Los dispositivos vestibles están compuestos por sensores internos que proporcionan información relevante a la acción que se esté realizando en ese momento. Estos dispositivos están compuestos principalmente por acelerómetros y giroscopios que proporcionan las características de posición, orientación y velocidad de la parte del cuerpo a la que acompañan [62]. Mediante el análisis temporal de esta información es posible extraer la actividad que está realizando la persona utilizando técnicas de

DL [63]. Sin embargo, estos sistemas de reconocimiento generalmente sirven para obtener actividades de carácter físico. Es decir, estos sistemas reconocen si la persona se encuentra sentada, tumbada, andando, corriendo o subiendo escaleras. Este tipo de actividad es adecuado para obtener parámetros y hábitos físicos. Sin embargo, esta información no suele ser útil para identificar ADL's.

En la actualidad, existen multitud de bases de datos de libre acceso con información relevante a sensores vestibles. Por ejemplo, Zhang et al. en [64] desarrollaron en el año 2012 una base de datos conocida como USC-HAD basada en un total de 5 sensores vestibles. Esta base de datos contiene 12 actividades etiquetadas. Sin embargo, estas actividades son de tipo físico, como sentarse en una silla, andar o correr, y la duración de esta base de datos es muy reducida ya que contiene únicamente un total de 6 horas de registros. Otra base de datos interesante es la desarrollada por Tapia et al. en [65] cuyo nombre es MIT PlaceLab Dataset. Esta base de datos combina dos tipos de tecnologías: sensores vestibles y sensores no intrusivos (sensores de iluminación y proximidad). Este estudio se realizó en el año 2006 y se llevó a cabo en dos casas reales. Esto supone una recogida de datos en condiciones no ideales, es decir, fuera de un laboratorio controlado. A diferencia de USC-HAD, esta base de datos sí que contempla ADL's, como puede ser hacer la cama o limpiar la cocina, aunque faltan aún bastantes actividades importantes del día a día. La base de datos OPPORTUNITY [66] es una de las más utilizadas dentro de la comunidad científica. Está compuesta por un total de 72 sensores en los que se combinan sensores vestibles y sensores no intrusivos en 15 instalaciones diferentes. En este estudio participaron un total de 12 usuarios diferentes que no vivían de forma simultánea. Los estudios tienen una duración de 25 horas, lo que supone una desventaja a la hora de buscar patrones de comportamiento. Otra base de datos muy similar y reciente es MARBLE [67] cuyo estudio se realizó el año 2021. Utiliza los mismos tipos de sensores que OPPORTUNITY aunque su duración es menor (únicamente 16 horas de toma de datos). Otra base de datos cuya duración es muy reducida es Cogent-House [68] ya que su duración se limita a los 23 minutos. La base de datos ADL [69] está compuesta únicamente por las medidas de 6 giroscopios que portan 10 voluntarios durante 2 horas, con el fin de desarrollar 9 categorías diferentes de ADL's. Por último, es necesario mencionar la base de datos CSL-SHARE [70] debido a su extendido

Base de datos	Casas	Sensores	Actividades	Duración
USC-HAD [64]	1	5	12	6 horas
MIT PlaceLab [65]	2	77-84	10	2-8 meses
OPPORTUNITY [66]	15	72	18	25 horas
MARBLE [67]	1	8	13	16 horas
Cogent-House [68]	1	12	11	23 minutos
ADL [69]	1	6	9	2 horas
CSL-SHARE [70]	1	10	22	2 horas

Tabla 2.4: Principales bases de datos de libre acceso de sistemas de reconocimiento por sensores vestibles.

uso. Se formó en el año 2021 y rápidamente empezó a utilizarse debido a su amplio número de actividades etiquetadas (un total de 22 categorías diferentes). Esta base de datos está compuesta por un total de 10 sensores vestibles durante una duración de 2 horas. En la Tabla 2.4 aparece un resumen de las principales bases de datos públicas compuestas por sensores vestibles y sus características más relevantes.

Esta tecnología proporciona mucha información respecto a la actividad física de los usuarios y su estado de salud. Del mismo modo, proporciona información sobre la posición y orientación de la parte del cuerpo en la que se implemente el dispositivo. Sin embargo, estos dispositivos no aportan información sobre la realización de ADL's por sí solos, ya que son independientes del resto de información de la vivienda.

2.2. Métodos de posicionamiento en interiores

Durante la implementación de los sistemas de monitorización de actividades en viviendas reales, es necesario tener en cuenta que se pueden encontrar dos escenarios: viviendas con un único residente y viviendas donde residen varios usuarios de forma simultánea. En el primer caso la posición de la persona se puede determinar mediante el análisis de las señales proporcionadas por los sensores no intrusivos, concretamente mediante los sensores de movimiento [71]. El segundo caso es más complejo de tratar, ya que es necesario contar con información relevante de la posición de cada usuario para inferir la actividad que está realizando en el momento de la detección [72]. De esta problemática surge la necesidad de contar con la posición en tiempo real de cada usuario dentro de la vivienda.

Existen diferentes tecnologías capaces de abordar la localización en interiores. Normalmente, se considera que un sistema de localización en interiores para el cuidado de personas mayores es efectivo cuando proporciona una precisión entre 0.5 m y 1 m. Los métodos actuales más fiables para alcanzar esta precisión son los sistemas basados en radiofrecuencia [73]. Este método de comunicación se caracteriza por ser inalámbrico, cuya frecuencia de ondas electromagnéticas oscilan entre los 3 kHz y los 300 GHz [74]. El valor de esta frecuencia afecta a la capacidad de cobertura y penetración ante obstáculos. Una de las principales ventajas de este tipo de comunicación es su amplia cobertura inalámbrica, lo que supone una fortaleza a la hora de realizar los despliegues en viviendas cuyo tamaño puede ser muy variable. A continuación se van a detallar las principales tecnologías y protocolos para llevar a cabo la localización de los usuarios utilizando sistemas basados en radiofrecuencia [75]:

1. Bluetooth de baja energía: BLE, del inglés *Bluetooth Low Energy*. Se considera a este protocolo como uno de los estándares de conectividad de menor consumo de energía en la actualidad. La frecuencia de las ondas electromagnéticas de este protocolo ronda los 2.4 GHz y es capaz de transmitir información con una velocidad de 24 Mbps con un alcance de entre 70 y 100 metros. Una de las ventajas que proporciona este tipo de protocolo es la posibilidad de alimentación mediante baterías, lo que facilita el despliegue de la tecnología en las viviendas [76].
2. Red de área local inalámbrica: WLAN, del inglés *Wireless Local Area Network*. Protocolo de acuerdo al estándar IEEE 802.11 que, de forma inicial, funciona con una frecuencia de onda electromagnética de 2.4 GHz. Sin embargo, en la actualidad, este protocolo también es capaz de trabajar con una frecuencia de 5 GHz para aumentar su velocidad de transferencia, reducir el ruido y minimizar las interferencias [77]. Este protocolo es flexible, cubre un amplio rango de distancia y los nodos que componen la red pueden comunicarse sin ningún tipo de restricción.
3. Zigbee: Este protocolo está basado en el estándar IEEE 802.15.4. Su frecuencia depende de la región en la que se encuentre desplegado. Por ejemplo, en Europa cuenta con una frecuencia de 868 Hz, mientras que en Estados Unidos o

Estándar	Alcance (m)	Precisión	Velocidad	Frecuencia
BLE	70-100	Baja	1-3 Mbps	2.4 GHz
WLAN	50-100	Moderada	1-1.3 Gbps	2.4 GHz / 5 GHz
Zigbee	10-100	Moderada	20-250 Kbps	868 Hz
UWB	0.5-10	Alta	Hasta 6.8 Gbps	2.5-10.6 GHz

Tabla 2.5: Comparación de estándares de comunicación dentro de los sistemas basados en radiofrecuencia para la localización de usuarios en interiores.

Australia funciona a 915 MHz. En otras regiones puede alcanzar una frecuencia de 2.4 GHz. Este protocolo, al igual que el BLE anteriormente mencionado, se considera como un protocolo de bajo consumo de energía, lo que supone una ventaja a la hora de realizar sus despliegue por la posibilidad que ofrece de utilizar baterías. Sin embargo, este protocolo no es fiable para distancias amplias [78].

4. Tecnología de banda ultraancha: UWB, del inglés *Ultra-wideband*. Se trata de una tecnología emergente de bajo consumo con una elevada velocidad (6.8 Gbps). Su frecuencia de onda electromagnética es mayor de 2.5 GHz, alcanzando en algunas ocasiones los 10.6 GHz. Ofrece una precisión muy elevada, que puede alcanzar los 20 cm de error [79]. Sin embargo, al tratarse de una tecnología tan novedosa en la actualidad, existen pocos dispositivos en el mercado que sean capaces de admitir este protocolo, lo que dificulta su implementación. Además, el costo del despliegue de este tipo de soluciones suele ser más elevado.

En la Tabla 2.5 se resumen las principales características de los estándares de comunicación para los sistemas basados en radiofrecuencia con el fin de realizar el posicionamiento de usuarios en interiores. Según las prestaciones y los trabajos existentes sobre localización en interiores, las dos tecnologías más utilizadas en la actualidad son el BLE y el WLAN [80].

En la actualidad, los métodos de localización mediante dispositivos BLE son los más extendidos debido a la facilidad a la hora de encontrar tecnología que cuente con este tipo de conectividad aunque cuente con la tasa de precisión más baja. En el mercado existen multitud de alternativas para poder registrar las posiciones de los usuarios dentro del hogar. Por ejemplo, el uso de sensores vestibles como pulseras de

actividad o dispositivos smartwatch está muy extendido dentro de la población. Otros dispositivos de tipo medallón también cuentan con este tipo de conectividad, aunque resulten menos cómodos de portar dentro del domicilio. Las soluciones de tipo WLAN se están extendiendo gradualmente debido a que proporciona mejor precisión que las soluciones BLE. Sin embargo, presentan la desventaja de que son muy susceptibles al ruido generado por otros dispositivos WLAN. Por ejemplo, en edificios con muchas viviendas y multitud de dispositivos WLAN pueden producir inestabilidades en el sistema de localización en interiores. En cuanto a las soluciones Zigbee, proporcionan buena precisión pero su uso en cuanto a sensores portables no está extendido en el mercado. Por último, la tecnología UWB se está extendiendo cada vez más en el mercado. Actualmente este tipo de tecnología mantiene unos precios elevados para su despliegue en un gran número de viviendas, aunque proporciona la tasa de precisión más alta en comparación al resto de tecnologías.

2.2.1. Sistemas de posicionamiento por balizas Bluetooth

El estándar BLE fue introducido por Apple en el año 2013 y se convirtió en una de las principales tecnologías a la hora de realizar localización de personas en interiores [81]. El trabajo realizado por Bahl et al. en [82] propone una solución basada en el cálculo de la distancia euclídea para obtener la distancia observada entre distintas intensidades de señal a un punto concreto de una estancia. En la Figura 2.3 se puede observar un ejemplo de la obtención de la posición del usuario. La localización de la persona se corresponde con el punto del mapa cuyas coordenadas están calculadas sobre cada una de las distancias euclídeas d_i a cada baliza BLE.

Otros autores, como Ma et al. en [83] propusieron mejorar este método de localización basado en el cálculo de las distancias euclídeas mediante la implementación del algoritmo K-Nearest Neighbor realizando una ponderación de la función Gaussiana y provocó una mejora de los resultados globales y una mejora en la estabilidad del posicionamiento. Otros autores optan por utilizar algoritmos de DL para resolver el problema del posicionamiento en interiores [84]. Por ejemplo, Chai et al. en [85] proponen un método probabilístico basado en histogramas que permite estimar la localización utilizando mapas medidos previamente.

Este tipo de tecnología se ha utilizado en numerosos campos de aplicación. Por

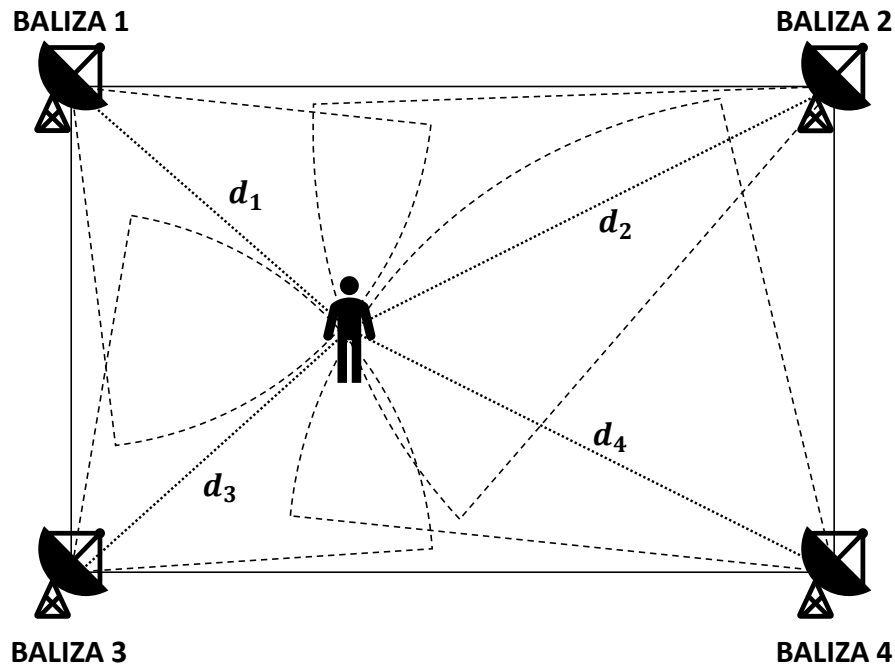


Figura 2.3: Método de triangulación para obtener la posición de un usuario en función de las distancias Euclídeas a cada baliza BLE.

ejemplo, Spachos et al. en [86] crearon un sistema de localización mediante balizas BLE para identificar las posiciones de los visitantes que entraban en un museo inteligente. Montoliu et al. en [87] proponen un sistema de posicionamiento para localizar las posiciones de personas mayores en escenarios reales utilizando una red de balizas y relojes inteligentes, objetivo que se asemeja al de la presente tesis.

La principal ventaja del uso de esta tecnología es su fácil accesibilidad, ya que numerosos dispositivos comerciales incorporan la tecnología BLE, como relojes y pulseras de actividad. Además, para los usuarios resulta más cómodo llevar una pulsera o reloj que otro tipo de dispositivos, como botones de emergencia o medallones. Sin embargo, el principal inconveniente de este tipo de tecnología es su precisión, ya que para obtener una tasa de acierto elevada, se necesita sobredimensionar la instalación para localizar el dispositivo.

2.2.2. Sistemas de posicionamiento por análisis de señales WiFi

La tendencia de utilizar sistemas de posicionamiento de usuarios en interiores mediante el análisis de señales WiFi se ha incrementado en los últimos años debido a su extensión en todo tipo de lugares. Por ejemplo, es una tecnología extendida en hogares, centros comerciales, oficinas, hospitales y hoteles [88]. Belmonte et al. en [89] y [90] proponen un sistema basado en una instalación de emisores y receptores WiFi desplegados en 5 hogares diferentes, con el que obtienen unos resultados que varían desde el 71.07% al 92% de precisión en el reconocimiento. Una de las desventajas de estos sistemas es la pérdida de precisión por utilizar las señales sin aplicar ningún tratamiento previo. Por esta razón, Poulou et al. en [91] proponen un modelo híbrido de DL en el que utiliza mapas de calor de la intensidad de las señales WiFi que proporcionan los módulos emisores, para entrenar algoritmos basados en redes neuronales convolucionales. Otro trabajo que utiliza modelos DL mediante la aplicación de redes neuronales convolucionales es el propuesto por Ibrahim et al. en [92], el cual proporciona una precisión estable del 91.42% en el reconocimiento. El trabajo realizado por Kim et al. en [93] usa también DL pero, a diferencia de los trabajos anteriormente mencionados, además de proporcionar la posición del usuario, es capaz de distinguir la planta en la que se encuentra ya que el estudio se realiza en un edificio de múltiples alturas.

Los sistemas de posicionamiento mediante análisis de señales WiFi son más precisos que los sistemas por BLE, ya que son capaces de proporcionar con mayor exactitud la sala en la que se encuentra el usuario. Sin embargo, este tipo de tecnología es más difícil de desplegar, y el coste de su despliegue es más elevado.

2.3. Análisis de patrones de comportamiento y anomalías

De forma paralela al reconocimiento de ADL's de las personas mayores, es importante tener en cuenta aquellos patrones de comportamiento que se alejan de la secuencia normal de actividades que realiza esa persona en un día normal. Esta al-

teración del patrón de comportamiento *estándar* puede estar relacionado de forma directa con un problema de salud o un indicador de deterioro cognitivo [94]. Por ejemplo, las personas mayores que sufren demencia suelen presentar patrones de comportamiento irregulares y con poca estabilidad en el tiempo [95]. Paudel et al. en [96] llegan a la conclusión de que las anomalías se pueden dividir en los siguientes 3 subgrupos:

1. Actividad o actividades inusualmente cortas o largas. Un ejemplo puede ser un incremento del tiempo a la hora de ducharse o una disminución de las horas de sueño normales para ese usuario.
2. Comportamiento con desviación de la duración típica. Si una persona está acostumbrada a despertarse a una hora determinada y lleva varios días con un desfase de varias horas, puede suponer que padece algún problema de salud.
3. Alteración de su secuencia de actividades. Si la persona está acostumbrada a ducharse antes de ir a dormir y ese hábito cambia, puede suponer un indicador de que algo está ocurriendo.

En cambio, Novák et al. en [97] clasifican las anomalías en función de cómo utilizan el tiempo para cada actividad.

1. Actividades realizadas en horarios poco comunes. Por ejemplo, si la persona decide ver la televisión en lugar de dormir durante la noche.
2. Actividades demasiado largas. Por ejemplo, el usuario tarda demasiado en hacer la comida, lo que puede ser un indicador de que se ha caído o ha sufrido un accidente.
3. Actividades demasiado cortas. Por ejemplo, si la persona duerme demasiado poco o si se levanta antes de cumplir las horas recomendadas de sueño.

Es necesario tener en cuenta que la duración de las actividades puede sufrir variaciones debido a otro tipo de causas, como deterioro físico o cognitivo. También es posible que, puntualmente, la duración de una actividad puede verse afectada debido a la complejidad de la tarea. Por ejemplo, para la actividad de cocinar, la

duración puede alargarse si el usuario elige preparar una comida que necesita mucha elaboración.

La mayor parte de los estudios que se centran en los análisis de patrones de comportamiento y sus anomalías toman como hipótesis principal el hecho de que las personas mayores realizan sus ADL's siguiendo unos patrones estables. Según esta hipótesis, se busca detectar las anomalías modelando en primer lugar los patrones de comportamiento y, después, se detectan las desviaciones con respecto a estos patrones. Existen numerosos métodos que se utilizan para modelar los patrones de comportamiento utilizando algoritmos de aprendizaje automático (ML, del inglés *Machine Learning*). El grupo basado en métodos generativos es uno de los más utilizados. Un clasificador generativo aprende el modelo para generar los datos estimando las distribuciones de este modelo [98]. A este grupo pertenecen los métodos HMM [99] o los CRF [100]. Estos métodos son buenos para bases de datos con poca información o que presenten un volumen de datos de entrenamiento insuficiente. Se caracterizan por tener en cuenta el contexto temporal de los datos y por seguir una progresión secuencial. Los métodos probabilísticos también se utilizan para detectar patrones de comportamiento, pero presentan un mal funcionamiento si el volumen de datos es pequeño. Dentro de este grupo se encuentran las distribuciones acumulativas [101] o los métodos Bayesianos [102]. En cambio, son métodos muy fiables ante actividades de larga duración. Los métodos discriminativos tienen como objetivo buscar los límites entre las clases que analiza el modelo, no se centran tanto en detectar como se generan las ADL's. A este conjunto de métodos pertenecen los algoritmos SVM [103] [104]. Su principal ventaja es que son métodos que no necesitan un amplio volumen de datos de entrenamiento. Sin embargo, para conjunto de datos grandes, ofrece malos resultados ya que proporciona una alta tasa de falsos positivos. Los métodos de *clustering* también se utilizan mucho, ya que se corresponden con un método de aprendizaje no supervisado por lo que no necesitan tener etiquetados los valores de salida. Los algoritmos de *Random Forest* pertenecen a este grupo de métodos [105]. La ventaja que presentan estos algoritmos a la hora de realizar un despliegue en una vivienda real es que no dependen de la ubicación donde se encuentren instalados los sensores o su tipo. Sin embargo, el número de clusters no es fácil de determinar y son modelos sensibles a la escala de los datos de entrada. Los

métodos basados en grafos permiten analizar y explotar la relación entre sensores. Sirven como entradas a algoritmos SVM, HMM o CRF [106]. Se utilizan mucho para extraer información de los sensores de presencia. Sin embargo, estos métodos no tienen en cuenta la inferencia temporal de los datos. Por último, los métodos basados en reglas son sistemas que tienen en cuenta la información del experto. Son métodos fáciles de tratar pero son muy susceptibles a señales ruidosas [107]. En la Tabla 2.6 se puede encontrar un resumen de todos los métodos analizados con sus principales ventajas, desventajas, algoritmos y trabajos relacionados.

Por otra parte, con el fin de detectar las anomalías y los cambios de comportamiento de los usuarios, existen trabajos que han desarrollado algoritmos capaces de detectar alteraciones específicas. Por ejemplo, Saives et al. en [109] utilizan la base de datos pública Domus [110] para evaluar la evolución de usuarios con demencia mediante el análisis de la frecuencia de las actividades y su repetitibilidad. Novak et al. en [97] utilizan la base de datos pública MavHome, desarrollada por Cook et al. en [111] en el año 2003 para detectar anomalías temporales con el fin de aplicarlo al cuidado de personas mayores que viven solas. Esta aplicación es la misma que plantea Elbayoudi et al. en [112] utilizando su propia base de datos. Para ello, los autores se centran en el análisis de las localizaciones donde se realizan las ADL's. Por último, es importante resaltar el trabajo desarrollado por Ordoñez et al. en [102] en el año 2015, ya que utilizan su propia base de datos para detectar los cambios en la duración total de cada actividad para usuarios que padecen Parkinson [113].

Método	Algoritmos	Trabajos relacionados	Ventajas	Inconvenientes
Generativos	HMM, CRF	[99], [100]	Bueno para bases de datos pequeñas y datos de entrenamiento insuficientes.	Métodos poco flexibles, susceptibles a ruido.
Probabilísticos	Métodos Bayesianos, distribuciones acumulativas	[101], [102]	Buen rendimiento ante actividades largas.	No es adecuado para conjuntos de datos pequeños.
Discriminativos	SVM	[103], [104]	Necesitan un volumen reducido de datos para funcionar.	Cantidad de falsos positivos alta ante grandes volúmenes de datos.
Clustering	Random Forest	[105], [108]	Independencia del tipo, posición y relación entre sensores.	Modelos sensibles a la escala de datos de entrada.
Basados en grafos	Entradas a HMM, SVM o CRF.	[106]	Muy utilizado para utilizar sensores de presencia.	No tienen en cuenta inferencia temporal de datos.
Basados en reglas	Reglas de experto	[107]	Modelos sencillos y fáciles de tratar.	Muy susceptibles a señales ruidosas, eficiencia pobre.

Tabla 2.6: Resumen de principales métodos de detección de anomalías mediante algoritmos ML.

Capítulo 3

Tecnología propuesta para la monitorización de personas mayores

En este capítulo se abordan las tecnologías propuestas para realizar el reconocimiento de actividades de las personas mayores. Se presenta: una descripción general sobre las actividades que se pretenden detectar y el despliegue realizado (Sección 3.1), la red de sensores no intrusivos que captan la información del entorno (Sección 3.2), el conjunto de balizas para obtener la posición de los usuarios (Sección 3.3), las pulseras inteligentes que proporcionan información sobre la actividad de los usuarios (Sección 3.4) y, por último, un análisis sobre la arquitectura del sistema de monitorización (Sección 3.5).

3.1. Diseño del despliegue de monitorización

Con el fin de probar la validez del sistema y corroborar la robustez de la idea planteada en la presente tesis, se ha realizado un despliegue de la tecnología abordada en este capítulo en un entorno controlado. De esta forma, se puede depurar su funcionamiento de cara al despliegue en los hogares de los usuarios finales.

Las condiciones de toma de datos se han llevado a cabo en un hogar donde residen al mismo tiempo dos personas adultas junto con su mascota. Además, otro aspecto

ACTIVIDADES		
Actividad en el baño	Tareas del hogar	Cocinar
Lavar los platos	Vestirse	Comer
Lavar la ropa	Hacer comida simple	Salir de casa
Cuidados de mascota	Leer	Relajarse
Ducharse	Dormir	Tomar medicinas
Ver la televisión	Trabajar	Otros

Tabla 3.1: Conjunto de actividades que el sistema es capaz de detectar.

importante es el hecho de que reciben visitas frecuentes, lo que puede suponer ruido para el sistema de detección de actividad. Para construir una base de datos sólida con la que probar los algoritmos de detección, el conjunto de muestras se ha tomado durante un periodo de 2 meses sin interrupción. Durante este periodo, es necesario considerar que el entorno puede sufrir cambios, como el movimiento de objetos en su interior, del mismo modo que la red puede sufrir alteraciones (caídas, interrupciones, ...). Todos estos eventos y alteraciones no suponen un efecto negativo, sino todo lo contrario, ya que agregan valor a la base de datos reproduciendo de forma realista las condiciones que pueden suceder en la vida real en los hogares de los usuarios finales. El hecho de recibir visitas, que los usuarios tengan una mascota o que en diferentes momentos coincidan varios usuarios de forma simultánea son situaciones que se pueden dar en cualquier hogar. Todas estas alteraciones posibles suponen un ruido que el sistema de detección tiene que ser capaz de filtrar.

En la Tabla 3.1 se puede observar un resumen de las principales actividades que pueden desempeñar las personas mayores que viven solas en sus casas. Estas actividades se han deducido a partir del estudio de la bibliografía vigente (e.j. [114]) y a partir de la colaboración con las entidades Fundación INTRAS y ASPRODES mencionadas en la introducción. De la tabla se puede deducir que cada actividad es independiente, y cada usuario sólo puede estar desempeñando una única actividad en cada momento. Además hay que tener en cuenta que ambos usuarios pueden estar desempeñando la misma actividad al mismo tiempo, incluso en estancias diferentes. La actividad “Cocinar” se corresponde con el intervalo de tiempo que uno de los usuarios está elaborando una comida elaborada, en la que es necesario que esté usando la vitrocerámica, lo que normalmente produce alteraciones ambientales en

un área cercana (como puede ser el incremento de temperatura o humedad en la cocina). Al mismo tiempo, la actividad “Hacer comida simple” se corresponde con la elaboración de una comida sencilla y que lleve poco tiempo y esfuerzo, como puede ser hacer un sandwich o preparar una ensalada. Por último, la actividad “Otros” se corresponde con el intervalo de tiempo en el que el usuario no está realizando ninguna de las otras actividades contempladas. Por ejemplo, puede estar deambulando por la casa, hablando por teléfono, etc.

Durante los 2 meses de la toma de datos, ambos usuarios han etiquetado las actividades que han ido desempeñando. Para ello, se han utilizado un conjunto de tags de comunicación de campo cercano (NFC, del inglés *Near Field Communication*) [115]. Los usuarios, pasando su teléfono móvil por encima de los tags NFC, consiguen registrar la hora y la actividad que están realizando, quedando todos los registros localizados en la base de datos. Además, diariamente, uno de los usuarios se encarga de revisar las actividades etiquetadas con el fin de realizar las correspondientes correcciones en caso de ser necesario.

Se ha realizado un estudio preliminar para saber qué sensores son necesarios para detectar las actividades propuestas en la Tabla 3.1. En la Tabla 3.2 aparecen las distintas estancias que componen la vivienda, así como la distribución de actividades más comunes en esas estancias. Además, en la tabla se puede visualizar una idea general de los sensores mínimos necesarios para poder detectar cada una de las actividades y el lugar donde tiene que ir colocado cada sensor.

3.2. Sensores no intrusivos

Para obtener información sobre el entorno en el que residen los usuarios y capturar los diferentes eventos que suceden dentro del hogar, se ha escogido una tecnología catalogada como no intrusiva para no causar rechazo por parte de los usuarios, ya que no capturan información íntimamente ligada a las personas. Esta tecnología no intrusiva está compuesta por sensores que pueden clasificarse en dos subgrupos:

1. Sensores con señales de tipo evento: Este conjunto de sensores se encarga de capturar los diferentes eventos instantáneos que tienen lugar dentro de la casa. Por ejemplo: apertura de puertas, vibraciones en sillas o presencia en una

ESTANCIA	ACTIVIDAD	SENSORES	LOCALIZACIÓN
Dormitorio	Dormir	Presencia	Pared
		Vibración	Cama
		Iluminación	Pared
	Vestirse	Presencia	Pared
		Contacto	Armario
		Vibración	Armario
	Leer	Presencia	Pared
		Iluminación	Pared
Baño	Tomar medicinas	Presencia	Pared
		Contacto	Cajón
	Actividad en el baño	Presencia	Pared
		Iluminación	Pared
	Ducharse	Presencia	Pared
		Iluminación	Pared
Temp.+Hum.	Pared		
Cocina	Cocinar	Presencia	Pared
		Temp.+Hum.	Pared
		Contacto	Despensas
	Lavar los platos	Presencia	Pared
		Contacto	Lavavajillas
	Hacer comida simple	Presencia	Pared
		Contacto	Despensas
	Comer	Presencia	Pared
		Iluminación	Pared
		Vibración	Silla
	Cuidados de mascota	Presencia	Pared
		Vibración	Cuenco
Lavar la ropa	Presencia	Pared	
	Consumo	Lavadora	
Estudio	Trabajar	Presencia	Pared
		Vibración	Silla
Salón	Ver la televisión	Presencia	Pared
		Consumo	Televisión
		Vibración	Sofá
	Relajarse	Presencia	Pared
Vibración		Sofá	
Entrada	Salir de casa	Presencia	Pared
		Contacto	Puerta
	Tareas del hogar	Presencia	Pared
		Contacto	Armario

Tabla 3.2: Estudio de los sensores necesarios para detectar las actividades propuestas.

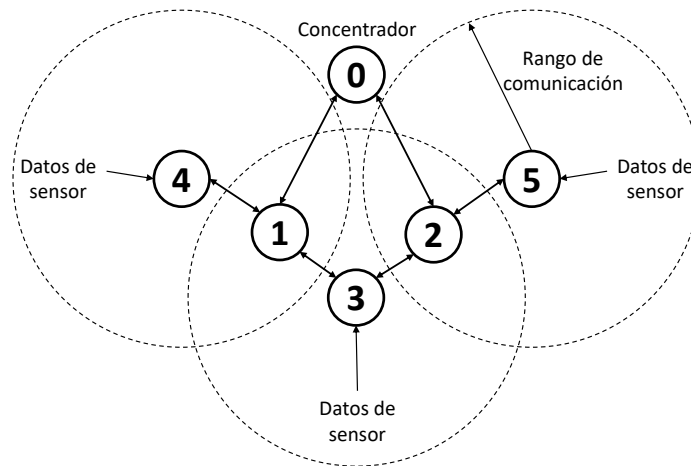


Figura 3.1: Red ZigBee: nodos de datos de sensores (3,4 y 5) comunican información al concentrador (0) usando pasarelas (1 y 2).

determinada estancia. Para que se dispare uno de los eventos anteriormente mencionados, cualquiera de los usuarios tiene que estar realizando una acción. Hay que tener en cuenta que estos eventos también pueden ser disparados por mascotas o visitantes.

2. Sensores con señales de tipo ambiental: Este conjunto de sensores se encarga de registrar diferentes condiciones de la vivienda que pueden estar asociadas o no a la acción de un usuario. Por ejemplo: consumo de un electrodoméstico, variación de temperatura o humedad de una determinada habitación. Estos sensores son sensibles a variaciones estacionales ya que, por ejemplo, el rango de temperatura es diferente en verano y en invierno.

Los sensores se han escogido de tal forma que no causen rechazo ante los usuarios finales. Son sensores discretos, no necesitan cableado para evitar obra civil y son fácilmente instalables.

Se ha escogido utilizar la tecnología ZigBee para la comunicación de los sensores, ya que se trata de una comunicación de bajo coste energético, escalable y muy extendida en la actualidad [116]. El protocolo ZigBee opera bajo el estándar de comunicación inalámbrica IEEE 802.15.4 [117].

En la Figura 3.1 se puede observar un esquema del funcionamiento del protocolo

TIPO DE SENSOR	REFERENCIA	NÚMERO
Aqara Sensor de movimiento (M)	RTCGQ11LM	8
Aqara Sensor de puertas y ventanas (C)	MCCGQ11LM	8
Aqara Sensor de temperatura y humedad (TH)	WSDCGQ11LM	2
Aqara Sensor de vibración (V)	DJT11LM	11
Xiaomi Mi ZigBee Enchufe inteligente (P)	ZNCZ04LM	2
Xiaomi MiJia Sensor de intensidad lumínica (L)	GZCGQ01LM	2
		TOTAL = 33

Tabla 3.3: Resumen de los sensores de la instalación: tipos y referencias.

ZigBee para 3 nodos emisores de datos. En este caso, los nodos 3, 4 y 5 se corresponden con los sensores que envían información del entorno. Sin embargo, estos nodos no son capaces de enviar la información al concentrador central (nodo 0) ya que se encuentra fuera de su rango de comunicación. Por lo tanto, utilizan dos nodos (1 y 2) como pasarelas, ya que estos si que se encuentran dentro del rango de comunicación con el concentrador. Esto supone una ventaja para su despliegue en viviendas, ya que cada casa tiene su topología y sus dimensiones [118].

En la Tabla 3.3 aparece un resumen del número total de sensores que se han empleado en la instalación y su referencia.

En la Figura 3.2 se puede observar un esquema general de la vivienda junto con la distribución de la red de sensores.

3.2.1. Sensores con señales de tipo evento

Los sensores con señales de tipo evento son los encargados de detectar interacciones de los usuarios con el entorno. Dentro de este grupo se encuentran los siguientes sensores: sensores de presencia, sensores de contacto y sensores de vibración. A continuación, se van a detallar cada uno de los sensores (ver Figura 3.3):

1. Sensor de presencia: Este tipo de sensor se utiliza para detectar la presencia de los usuarios dentro de las diferentes estancias de la vivienda. Aunque no son capaces de distinguir el usuario que se está moviendo por la habitación, son fiables a la hora de detectar cualquier movimiento. Se han escogido los sensores de presencia pasivos infrarrojos (PIR, del inglés *Passive Infrared*),

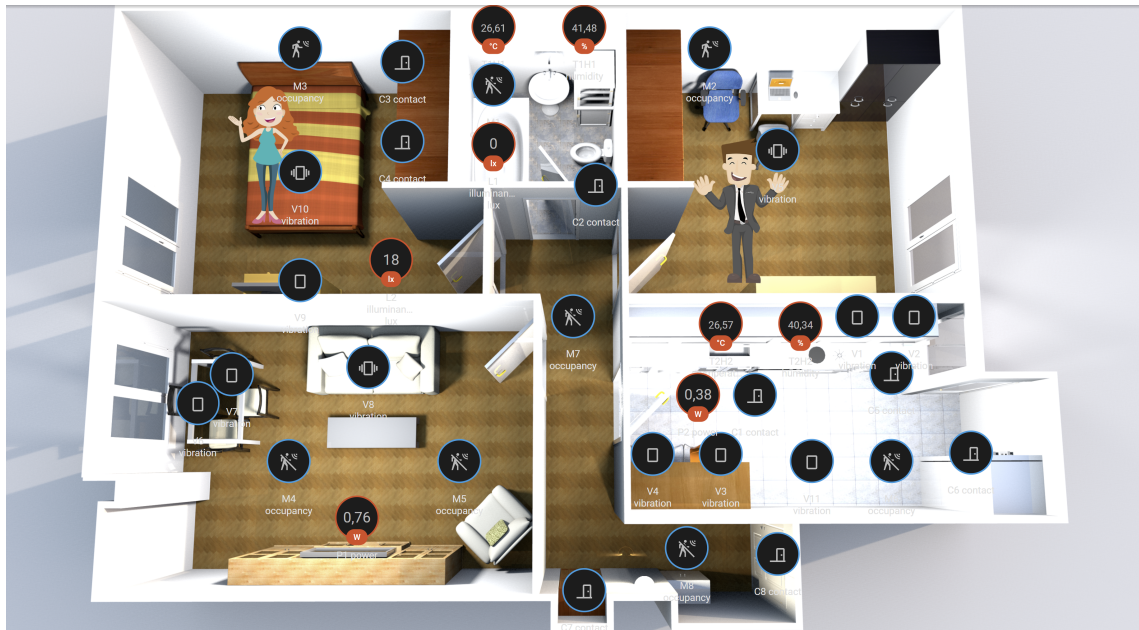


Figura 3.2: Esquema general de la vivienda con la distribución de los sensores no intrusivos.

que se encargan de recoger la variación de radiaciones infrarrojas de su rango de acción [119]. En el despliegue se han utilizado un total de 8 sensores de este tipo, uno para cada estancia. Para el salón y el pasillo se han utilizado 2 para cada uno, con el fin de dividir las estancias en dos zonas. Estos sensores normalmente se colocan en las paredes, a ser posible en zonas altas apuntando hacia el suelo (ver Figura 3.3a).

2. Sensor de contacto: Este tipo de sensor se utiliza para detectar apertura de puertas, ventanas o cajones dentro de la vivienda. Por ejemplo, se puede utilizar para detectar la interacción con distintos electrodomésticos, como el frigorífico o microondas. A su vez, también se puede utilizar para detectar la apertura de la puerta de la calle o el cajón de las medicinas. En el despliegue se han utilizado un total de 8 sensores de este tipo. El sensor viene dividido en dos partes, la parte mas ancha se corresponde con el receptor de señal y la interfaz de comunicación, mientras que la parte más pequeña se corresponde con el imán que dispara el evento y que, normalmente, se coloca en la parte móvil (ver Figura 3.3b).



Figura 3.3: Ejemplo de sensores con señales de tipo evento.

3. Sensor de vibración: Este tipo de sensor se utiliza para detectar interacciones con diferentes muebles de la vivienda. Debido a su alta sensibilidad, es capaz de detectar pequeñas vibraciones al interactuar, por ejemplo, con la cama o con una silla. También son capaces de detectar aperturas de armarios o cajones. Debido a su pequeño tamaño, es muy fácil de instalar y esconder (ver Figura 3.3c).

3.2.2. Sensores con señales de tipo ambiental

Los sensores con señales de tipo ambiental son los encargados de detectar los cambios y alteraciones que sufre el entorno de la vivienda. Dentro de este grupo se encuentran los siguientes sensores: sensores de temperatura y humedad, sensores de consumo eléctrico y sensores de intensidad lumínica. A continuación, se van a detallar cada uno de los sensores (ver Figura 3.4):

1. Sensor de temperatura y humedad: Este sensor se encarga de registrar las variaciones de temperatura y humedad de la estancia en la que está colocado. Normalmente estos sensores se colocan en la cocina y en el cuarto de baño con el fin de reconocer si los usuarios están cocinando o duchándose, ya que en ambas actividades se produce un incremento rápido de la humedad y la temperatura del ambiente. En el despliegue se han utilizado un total de 2 sensores de este tipo. Se colocan cerca de la vitrocerámica en la cocina y cerca de la ducha en el cuarto de baño (ver Figura 3.4a).

2. Sensor de consumo eléctrico: Este sensor se encarga de registrar el consumo eléctrico del electrodoméstico con el que esté asociado. Existen actividades que están ligadas directamente con el uso de un determinado electrodoméstico, por eso es importante detectar las variaciones de consumo para obtener encendidos u apagados. En el despliegue se han utilizado un total de 2 sensores de este tipo: uno colocado en el enchufe de la televisión para conocer con detalle el uso de este dispositivo y otro en el enchufe de la lavadora para detectar la actividad de lavar la ropa. Este tipo de sensor es muy fácil de instalar, ya que sólo hay que introducirlo como intermediario entre el enchufe de la pared y el electrodoméstico seleccionado. Además, este sensor incluye protección frente a sobrecorriente y sobretensiones, lo que incrementa el nivel de seguridad del uso del electrodoméstico (ver Figura 3.4b).
3. Sensor de intensidad lumínica: Este tipo de sensor se encarga de detectar el nivel de luz de una determinada estancia. En el despliegue se han utilizado un total de 2 sensores de este tipo: uno en el dormitorio y otro en el cuarto de baño. Las señales de estos sensores sirven para detectar si se está realizando alguna actividad en esta estancia. Los sensores de presencia son capaces de detectar si hay un usuario moviéndose en una habitación, por ejemplo, en el dormitorio. De esta forma, combinando la señal de un sensor de presencia con la información de un sensor de intensidad lumínica, se puede saber por ejemplo si el usuario está durmiendo (movimiento en el dormitorio pero luz apagada). Son sensores de pequeño tamaño que pueden situarse en cualquier superficie expuesta a la luz (ver Figura 3.4c).

3.3. Balizas Bluetooth

El inicio o el final de una actividad se puede detectar mediante el análisis de las señales proporcionadas por los sensores no intrusivos. Sin embargo, en viviendas donde residan varios usuarios, no se puede deducir la persona que está realizando la actividad. Por esta razón, el despliegue cuenta con una red de balizas con dos funcionalidades básicas: clasificar la actividad e identificar la posición de cada usuario

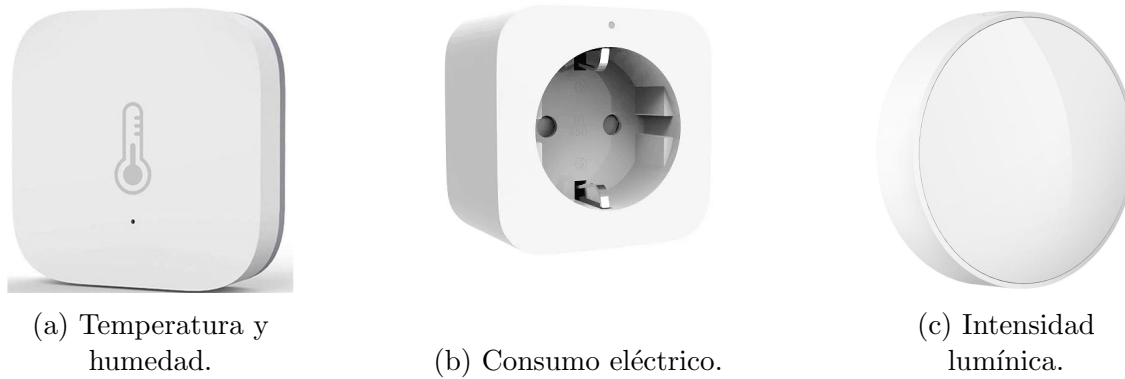


Figura 3.4: Ejemplo de sensores con señales de tipo ambiental.

dentro de la casa. Para poder triangular la posición de los usuarios, se ha colocado una baliza en cada una de las estancias de la casa [80]. Las balizas se encargan de medir la intensidad de la señal Bluetooth que emite cualquier dispositivo que cuente con este protocolo. De esta manera, cada usuario lleva consigo una pulsera de actividad o reloj inteligente que emite mensajes por Bluetooth para que las balizas puedan localizar al usuario. Se ha escogido utilizar pulseras inteligentes para asegurar que los usuarios lleven el dispositivo el mayor tiempo posible (excepto cuando sea necesario cargar su batería), ya que existen otras soluciones que también cuentan con el protocolo Bluetooth (etiquetas, teléfonos móviles y similares) [120]. Por lo tanto, se ha escogido utilizar los dispositivos Xiaomi Mi Smart Band 4 ya que son dispositivos económicos y ampliamente extendidos en el mercado [121].

Las balizas Bluetooth se han desarrollado sobre un chip ESP32 ya que se trata de un microcontrolador de bajo precio y ampliamente utilizado para desarrollos IoT [122]. Una de las ventajas que presentan los microcontroladores ESP32 frente a otras soluciones similares en el mercado es que incorporan los 3 tipos de conectividad más empleados en IoT actualmente [123]: WiFi, Bluetooth y BLE [124]. El software que se ha utilizado en el ESP32 es un desarrollo del grupo ESPresense [125]. El software cuenta con un filtro de Kalman [126] para reducir el ruido que puede existir entre las medidas de las diferentes balizas, ya que existe un margen de error al tratar de medir la posición de un objeto que se encuentra en movimiento. Cada una de las balizas miden la intensidad de la señal Bluetooth que proporcionan las pulseras de

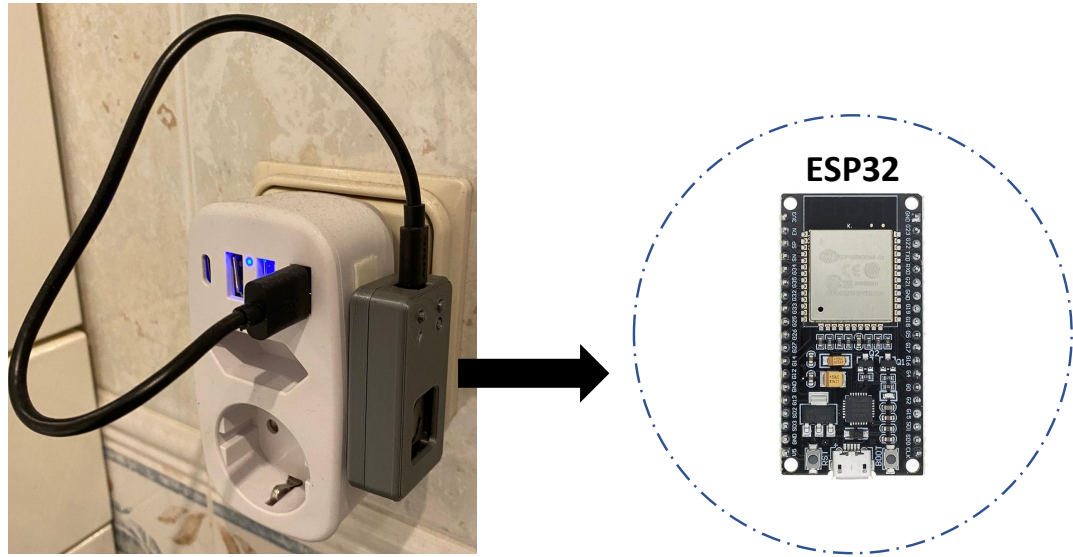


Figura 3.5: Baliza ESP32: Montaje final y chip.

actividad y envían la información mediante mensajes MQTT (del inglés *Message Queuing Telemetry Transport*) [127] al concentrador principal. En la Figura 3.5 se puede observar un esquema donde aparece una imagen del montaje de la baliza final junto con un detalle donde aparece el chip ESP32.

3.4. Pulseras de actividad

Las pulseras de actividad escogidas para el despliegue, además de ser de utilidad para obtener la ubicación de los usuarios tal y como se ha mencionado en la Sección 3.3, también proporcionan información muy valiosa acerca de los parámetros de actividad física de los usuarios. Las pulseras seleccionadas proporcionan información sobre los acelerómetros y giroscopios en los 3 ejes del sistema cartesiano, lo que repercute que en todo momento se sabe la orientación y velocidad de la mano. Además, cuenta con sensores fotoeléctricos que informan sobre el ritmo cardíaco del portador [128]. A continuación se exponen todos los parámetros que proporcionan estas pulseras:

1. Ritmo cardíaco: Las pulseras de actividad cuentan con sensores fotoeléctricos en la parte inferior en los que, mediante la diferencia entre una pequeña señal de luz emitida hacia la muñeca y la recibida por los sensores, se puede estimar la frecuencia cardíaca. Esto se debe a la capacidad de absorción de luz que tiene la sangre, a mayor frecuencia cardíaca, mayor es el bombeo de sangre y se absorbe mayor cantidad de luz.
2. Datos de acelerómetros y giroscopios: Por defecto, las pulseras de actividad no publican estos datos, ya que significa un mayor consumo de batería debido al flujo de información. A pesar de esto, se ha modificado el protocolo de emisión de información para poder compartir estos datos ya que la orientación y velocidad de las manos puede aportar mucha información para determinar la actividad que está realizando el usuario [129].
3. Calorías y grasa quemada: Estos parámetros se obtienen mediante el análisis de los acelerómetros y giroscopios previamente mencionados. Esta información es útil para conocer los hábitos de la persona y obtener el grado de sedentarismo.
4. Estado de la batería: La pulsera de actividad publica el porcentaje de batería que le queda, así como si está cargando o necesita cargarse.
5. Pasos y metros recorridos: Estos parámetros se obtienen de una forma similar a los de calorías y grasa quemada. Se obtiene mediante el análisis de las señales de los acelerómetros y giroscopios. Esta información es útil para conocer el nivel de actividad física de la persona.

El funcionamiento normal de las pulseras de actividad comerciales se basa en la conexión pulsera-teléfono móvil mediante Bluetooth. Una vez que el teléfono móvil tiene la información, se la comunica a la nube del fabricante (en este caso Xiaomi). Por esta razón, se ha incorporado un dispositivo intermediario que actúa con el mismo rol que el teléfono móvil para poder recoger los datos en bruto para, posteriormente, almacenarlo en la base de datos privada del despliegue.

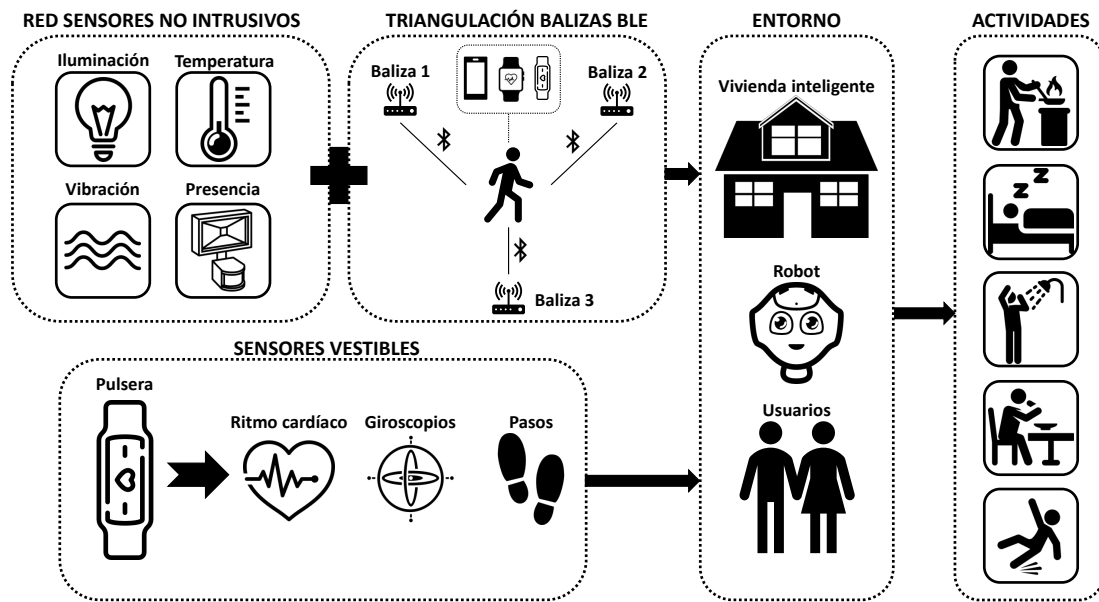


Figura 3.6: Diagrama general de los diferentes componentes del despliegue del sistema de monitorización.

3.5. Arquitectura del despliegue

En la Figura 3.6 se puede observar un esquema general de todos los componentes que conforman el despliegue del sistema de monitorización mencionados en los anteriores apartados: la red de sensores no intrusivos, la triangulación de los usuarios mediante balizas Bluetooth y la información recogida por las pulseras de actividad.

Todo el sistema de monitorización se engloba dentro de la arquitectura que se puede comprobar en la Figura 3.7.

Todo el sistema de control se encuentra centralizado en un modulo concentrador de señales. Para ello, se ha escogido una NVIDIA Jetson Nano debido a su alto rendimiento y su elevada capacidad de análisis de datos [130]. Además, este equipo cuenta con una unidad de procesamiento gráfico (GPU, del inglés *Graphics Processing Unit*) propia, siendo esto un beneficio a la hora de procesar los datos del sistema y realizar las predicciones de las actividades en tiempo real [131]. Para controlar el sistema, se ha instalado el sistema operativo Home Assistant [132], ya que proporciona las siguientes funcionalidades: enlazar sensores con el controlador, proporcionar inter-

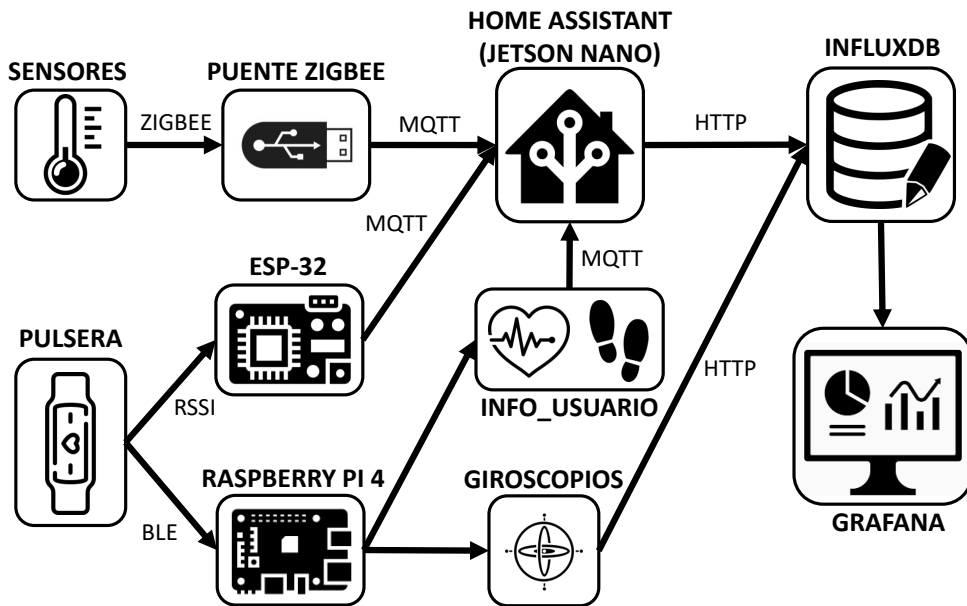


Figura 3.7: Arquitectura del sistema de monitorización y comunicaciones entre los componentes.

faz visual para verificar el correcto funcionamiento del sistema de monitorización y exportación de información a base de datos. El sistema operativo Home Assistant recibe la información del sistema de monitorización mediante el protocolo MQTT.

La red de sensores no intrusivos se comunican mediante el protocolo de comunicación ZigBee. Para poder recoger estas señales mediante el Home Assistant, se ha implementado un hardware intermediario (ConBee II) que transforma las señales ZigBee en mensajes MQTT. Para el sistema de triangulación por balizas Bluetooth, los dispositivos ESP32 pueden emitir directamente el mensaje MQTT a la Jetson Nano. Sin embargo, para obtener la información de las pulseras de actividad, es necesario poder recoger las señales Bluetooth que emiten estos dispositivos. Como la Jetson Nano no proporciona interfaz Bluetooth, se ha incorporado al sistema un concentrador adicional que se encargue únicamente de recoger la información de las pulseras de actividad. Para ello, se ha escogido utilizar una Raspberry Pi 4 dado que sí soporta este protocolo de comunicación y proporciona buen rendimiento ante este flujo de información [133]. La información recogida por las pulseras se envía de dos formas:

1. Ritmo cardíaco, calorías, grasa quemada, pasos, metros recorridos y estado de batería: Esta información se envía mediante mensajes MQTT en tiempo real al Home Assistant, permitiendo de esta manera su visualización en tiempo real en el interfaz gráfico.
2. Datos de giroscopios y acelerómetros: El flujo de información generado es demasiado elevado como para mandarlo en tiempo real al Home Assistant. Esto podría colapsar el sistema de comunicación y algunos datos podrían perderse. Por esta razón, una vez al día, todos estos datos se envían directamente a la base de datos (normalmente por la noche, ya que es el intervalo de tiempo donde el sistema de monitorización recoge menos datos). Este envío se realiza mediante el protocolo de transferencia de hipertexto (HTTP, del inglés *Hypertext Transfer Protocol*).

Toda la información del sistema de monitorización se almacena en una base de datos InfluxDB localizada en un servidor externo a la vivienda. Se ha escogido utilizar una base de datos InfluxDB debido a las siguientes ventajas [134]: se trata de una base de datos de código abierto que está optimizada para procesar series de datos temporales, ofrece integración con la herramienta de visualización Grafana, proporciona soporte ante un elevado número de lenguajes de programación, es fácilmente instalable y existe una extensa documentación sobre su uso y mantenimiento [135]. La herramienta Grafana se ha utilizado para visualizar los datos históricos recogidos en la base de datos. Se trata de una herramienta de código abierto en la que los usuarios pueden crear paneles de visualización personalizados de manera flexible [136].

Capítulo 4

Diseño del modelo de reconocimiento de actividades

En el presente capítulo se abordan los diferentes métodos de inteligencia artificial y DL empleados durante la investigación y ejecución de la tesis. En primer lugar (Sección 4.1), se presentan los principales métodos de aprendizaje supervisado que se han empleado en los modelos HAR propuestos en la presente tesis. A continuación, en la Sección 4.2 se exponen los métodos de aprendizaje no supervisado utilizados para elaborar más modelos HAR. Por último, en la Sección 4.3 se detallan los modelos elaborados para realizar el reconocimiento de actividades, mediante técnicas de aprendizaje supervisado y no supervisado. Todos estos modelos propuestos son válidos para alcanzar los objetivos planteados en la presente tesis doctoral, y la elección del modelo a utilizar dependerá del usuario a analizar, tal y como se discute en la Sección 6.4.

4.1. Reconocimiento de actividades mediante aprendizaje supervisado

Los métodos de aprendizaje supervisado son modelos que necesitan conocer la salida que tienen que generar para un conjunto de entradas durante la etapa de entrenamiento [137]. Este requisito es necesario cuando se utilizan DNNs, como las redes neuronales convolucionales (CNN) [138] [139] o redes neuronales recurrentes

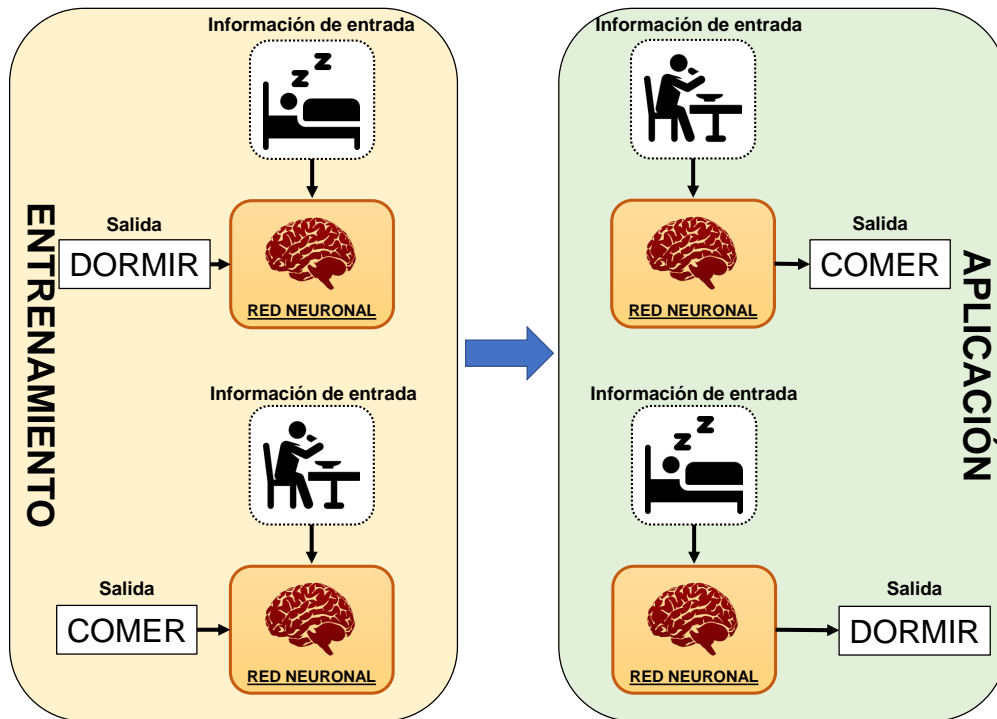


Figura 4.1: Esquema de funcionamiento de un método de aprendizaje supervisado.

(RNN) [140]. Para realizar HAR, el modelo necesita conocer las actividades que se están desarrollando para un conjunto de entradas específico, etiquetado que tiene que hacer manualmente el usuario o un experto colaborador. Esto puede resultar costoso y tedioso, ya que normalmente requiere recopilar datos durante un largo periodo de tiempo para lograr un aprendizaje satisfactorio (semanas o incluso meses). Así pues, aunque este método de aprendizaje es posible en teoría y proporciona buenos resultados, puede resultar difícil de aplicar en la práctica. Además, si el sistema se traslada a otra casa, con otras personas y otros sensores, suele ser necesario un proceso de reentrenamiento. La situación se complica aún más en el caso de que haya más de un usuario viviendo en la misma casa, ya que los sensores no son capaces de recoger información inherente a un usuario concreto.

En la Figura 4.1 se expone un ejemplo sencillo del funcionamiento interno de un método de aprendizaje supervisado. El modelo serviría en este caso para diferenciar dos clases de ADL's: comer y dormir. En la imagen se pueden diferenciar claramente dos etapas:

1. Etapa de entrenamiento: En esta etapa el modelo únicamente se encarga de aprender las características más importantes de los datos de entrada para la clase que se le está especificando. Por lo tanto, en esta etapa, el modelo no genera ningún tipo de salida.
2. Etapa de aplicación: Una vez que se ha realizado la etapa de aprendizaje, el modelo sólo necesita los datos de entrada para generar salidas. Si la etapa de entrenamiento ha contado con el suficiente volumen de datos de calidad, la eficacia en la aplicación será alta.

Es importante recalcar que el modelo necesita un amplio volumen de datos, siempre y cuando estos datos sean de calidad. Por ejemplo, si el modelo solamente se entrena con la misma ADL, durante la etapa de aplicación no será capaz de distinguir ADL's que nunca ha visto. Del mismo modo, si solamente se ha entrenado con actividades realizadas con el mismo patrón, el modelo funcionará mal si intenta distinguir la misma actividad realizada de forma diferente. Por todo esto, es muy importante realizar un reparto correcto de los datos entre las dos etapas para contar con el suficiente volumen de variantes.

En la presente tesis se han estudiado diferentes métodos de aprendizaje supervisado para su posterior aplicación con datos reales. Los métodos de aprendizaje supervisado analizados son los siguientes:

1. Redes neuronales recurrentes: RNN, del inglés *Recurrent Neural Network*. Los modelos basados en RNNs son variantes de los modelos de redes neuronales convencionales, que tratan con series temporales y almacenan cierto conocimiento de eventos pasados.
2. Redes neuronales LSTM: redes con memoria a largo o a corto plazo, del inglés *Long Short-Term Memory*. Los modelos de redes neuronales basados en capas LSTM son un tipo de RNN con una serie de modificaciones con la finalidad de evitar el fenómeno de desvanecimiento de gradiente que sufren los métodos basados en RNNs [141].
3. Redes neuronales GRU: redes de unidades recurrentes con compuertas, del inglés *Gated Recurrent Unit*. Los modelos basados en redes neuronales GRU

son una variante de las redes RNN convencionales, bastante similares a las redes LSTM en cuanto a que resuelven el problema del desvanecimiento de gradiente, con la particularidad de que son menos pesadas computacionalmente que las redes LSTM.

4. Redes neuronales convolucionales: CNN, del inglés *Convolutional Neural Network*. Los modelos basados en redes neuronales CNN cuentan con diferentes capas de filtros de convolución, que se encargan de extraer las principales características de los datos de entrada y, a medida que se implementan capas, la dimensión del vector de salida se va reduciendo [142].
5. Redes neuronales transformadoras: redes neuronales transformadoras (transformers) comúnmente utilizadas para el análisis de secuencias basadas en modelos de atención. Este tipo de modelos buscan relaciones entre los datos de entrada y sus salida correspondiente, y no necesariamente tienen que ser secuenciales como sucede en los modelos recurrentes. Las “cabezas” de atención que componen el modelo buscan correlaciones entre los datos y, cuantas más cabezas se utilicen, mayor es la capacidad de extracción de relaciones. [143]

A continuación, en los siguientes apartados, se va a detallar el funcionamiento interno y el fundamento matemático de cada uno de los métodos mencionados.

4.1.1. Redes neuronales recurrentes (RNN)

Las redes neuronales RNN son capaces de generar una secuencia de salidas en función de la información temporal extraída de un conjunto de entradas con dependencias temporales. A su vez, son capaces de aplicar múltiples operaciones básicas a cada una de las entradas de forma simultánea. La memoria se actualiza paso a paso entre los distintos estados ocultos de la red, lo que permite retener información pasada para aprender estructuras temporales y dependencias entre los datos a lo largo del tiempo [144]. Este tipo de redes se utiliza ampliamente en tareas de procesamiento del lenguaje natural y reconocimiento del habla.

La Figura 4.2 muestra el funcionamiento interno de una red RNN. Para un instante de tiempo t , la red RNN toma la secuencia de entrada x_t y el correspondiente

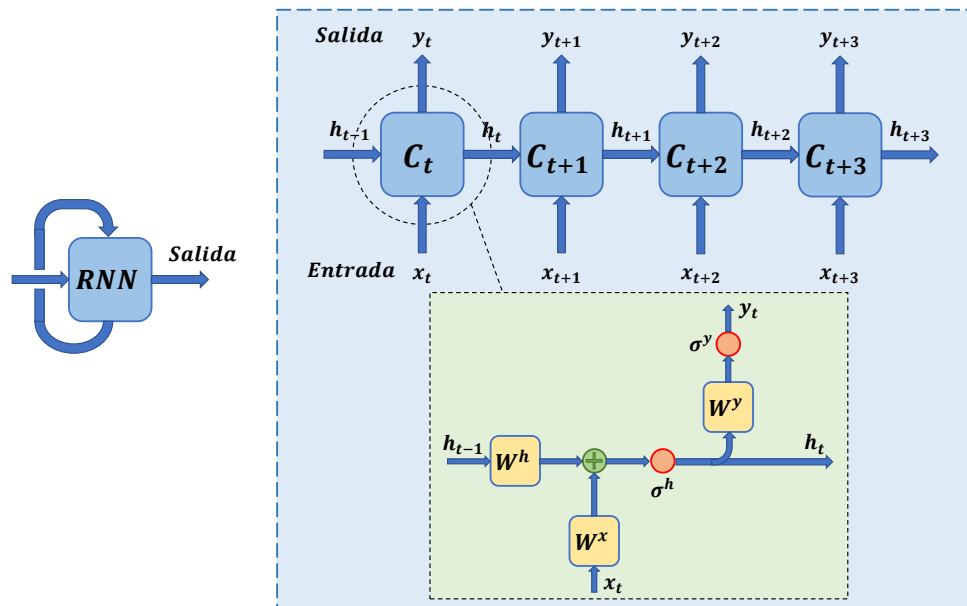


Figura 4.2: Contenido de una red neuronal RNN.

vector de memoria del instante inmediatamente anterior h_{t-1} . Mediante una serie de operaciones matemáticas sencillas dentro de la celda C_t , se obtiene un vector de salida y y el estado de memoria h_t actualizado. Los resultados generados a la salida de la red pueden calcularse de la siguiente manera:

$$h_t = \sigma^h(W^x \cdot x_t + W^h \cdot h_{t-1}) \quad (4.1)$$

$$y_t = \sigma^y(W^y \cdot h_t) \quad (4.2)$$

Los parámetros de las matrices W^h , W^x y W^y de las Ecuaciones (4.1) y (4.2) se calculan y optimizan durante el entrenamiento. Son matrices densas cuyos pesos se modifican a medida que avanza el entrenamiento de la RNN. A su vez, los parámetros σ^h y σ^y corresponden a funciones de activación no lineales.

4.1.2. Redes neuronales Long Short-Term Memory (LSTM)

Al igual que las redes RNN anteriormente mencionadas, las redes LSTM también son adecuadas para trabajar con series temporales y grandes volúmenes de datos,

aunque de forma más eficiente [145]. Cuentan con un sistema interno de comunicación entre celdas que funciona como memoria para almacenar las correlaciones temporales entre los datos.

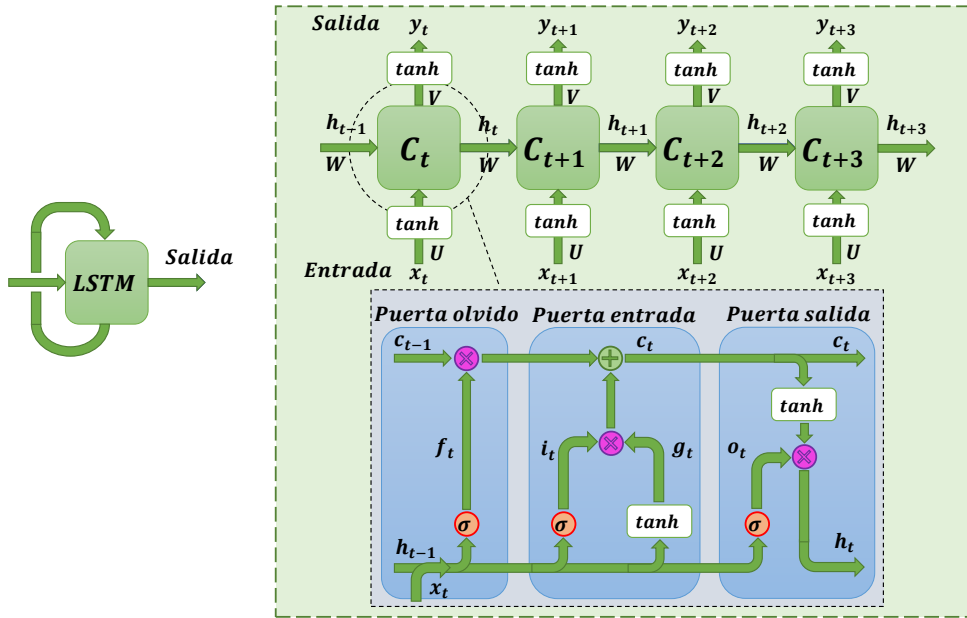


Figura 4.3: Contenido de una red neuronal LSTM.

El funcionamiento interno de las redes LSTM y la composición de sus celdas puede verse en la Figura 4.3. El vector de memoria h_t para un instante de tiempo dado t y el vector de salida de la red y_t se pueden calcular de la siguiente manera a partir del vector de memoria del instante anterior h_{t-1} y el vector de datos de entrada x_t :

$$h_t = \begin{cases} \tanh(Ux_t + b_t^h) & \text{si } t = 0 \\ \tanh(Ux_t + Wh_{t-1} + b_t^h) & \text{si } t > 0 \end{cases} \quad (4.3)$$

$$y_i = \tanh(Vh_i + b_i^h) \quad (4.4)$$

Los parámetros U , V y W de las ecuaciones (4.3) y (4.4) se calculan y optimizan durante el entrenamiento. Los parámetros b_t^h equivalen al sesgo. El parámetro U corresponde a la matriz de pesos que relaciona la capa de entrada con la capa oculta. El parámetro V corresponde a la matriz de pesos que relaciona la capa oculta con

la capa de salida. Por último, el parámetro W corresponde a la matriz de pesos que relaciona cada una de las capas ocultas de la red LSTM.

Como puede verse en la Figura 4.3, las células LSTM internas pueden dividirse en tres secciones funcionales:

1. Puerta del olvido: Del inglés *Forget gate*. Esta puerta es la primera que aparece dentro de una célula LSTM. Se encarga de decidir qué componentes ya no son necesarios en la memoria de la red, y mediante un cálculo matemático activando una función *sigmoid* lo elimina o lo deja permanecer en memoria.
2. Puerta de entrada: Del inglés *Input gate*. Esta puerta aparece después de la puerta del olvido. Se encarga de añadir nuevos componentes a la memoria de la red LSTM.
3. Puerta de salida: Del inglés *Output gate*. Es la última puerta de la célula LSTM. Se encarga de utilizar los componentes de la memoria para calcular la salida de la célula, que corresponderá a la entrada de la siguiente célula.

A continuación se muestran los principales cálculos que se realizan dentro de una célula LSTM [146]:

$$f_t = \sigma(U_f x_t + W_f h_{t-1} + b_f) \quad (4.5)$$

La Ecuación 4.5 indica cómo se calcula el valor de la puerta del olvido. Tanto el parámetro U_f como el parámetro W_f se calculan y modifican durante el entrenamiento de la red.

$$i_t = \sigma(U_i x_t + W_i h_{t-1} + b_i) \quad (4.6)$$

Del mismo modo, la Ecuación 4.6 indica el método para calcular el valor de la puerta de entrada. Los subíndices de las matrices de pesos indican qué valores correspondientes a la puerta de entrada se van a utilizar.

$$g_t = \tanh(U_g x_t + W_g h_{t-1} + b_g) \quad (4.7)$$

La Ecuación 4.7 indica cómo se calcula el valor de g_t . Este parámetro representa el valor que simboliza el estado candidato a formar parte de la memoria LSTM.

$$o_t = \sigma(U_o x_t + W_o h_{t-1} + b_o) \quad (4.8)$$

La Ecuación 4.8 muestra cómo se calcula el valor de la puerta de salida a partir de las matrices de pesos iniciales específicas de esa puerta.

$$c_t = f_t \cdot c_{t-1} + i_t \cdot g_t \quad (4.9)$$

La Ecuación 4.9 se utiliza para calcular el estado de la memoria LSTM, habiendo olvidado los valores indicados por la puerta de olvido y habiendo incluido los valores indicados por la puerta de entrada.

$$h_t = o_t \cdot \tanh(c_t) \quad (4.10)$$

Por último, la Ecuación 4.10 se utiliza para actualizar el valor de las capas ocultas de la red LSTM, cuya salida será la entrada de la siguiente celda.

Si se dispone de la secuencia anterior t_{-1} y posterior t_1 a un tiempo determinado t_1 , es posible mejorar el rendimiento de la red neuronal sustituyendo la arquitectura formada por capas LSTM convencionales por capas LSTM bidireccionales. Para entrenar este tipo de red, es necesario realizar pasadas en ambas direcciones con el conjunto de datos [147]. De esta forma, el modelo se entrena teniendo en cuenta secuencias no sólo anteriores al instante analizado, sino también posteriores. Una ventaja muy favorable de las redes LSTM bidireccionales es el hecho de que se necesita un buffer menor que el de las redes LSTM convencionales para obtener resultados similares, lo que se traduce en una buena respuesta a los sistemas causales, siempre que la latencia de salida requerida para el sistema sea corta [148]. Este tipo de redes es muy beneficioso para tareas como el reconocimiento de emociones en el habla o el modelado del ruido [149].

La Figura 4.4 muestra la arquitectura de una red neuronal LSTM bidireccional con T etapas. Por ejemplo, para un instante t , la capa hacia delante utiliza los datos de $t - 1$ para generar los datos del instante $t + 1$. Sin embargo, con respecto a la capa hacia atrás, los datos de $t + 1$ se utilizan para generar los datos del instante $t - 1$. Hay que añadir que ambos procesos comparten la misma capa de activación, independientemente de la dirección del flujo de información.

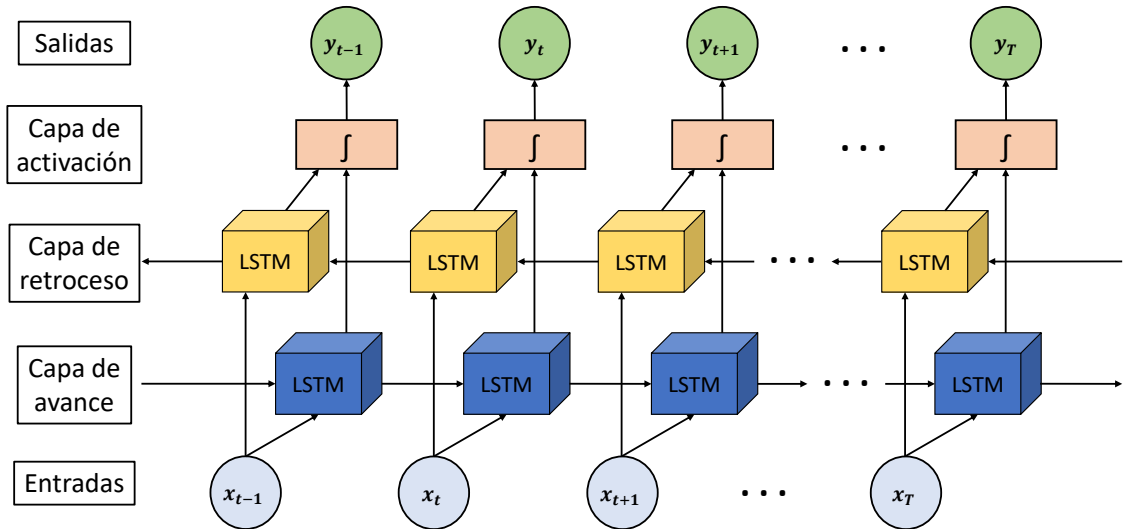


Figura 4.4: Arquitectura de una red neuronal LSTM bidireccional.

4.1.3. Redes neuronales Gated Recurrent Unit (GRU)

Las redes neuronales GRU son similares a las redes LSTM previamente explicadas, con la particularidad de que necesitan una menor carga computacional para entrenarlas y ejecutarlas. Esto es debido a que reducen el número de parámetros internos a modificar y ajustar durante el entrenamiento de la red, ya que el número total de puertas internas se reduce en una unidad [150]. Por lo tanto, la duración del entrenamiento se reduce enormemente, proporcionando resultados y prestaciones similares a las redes LSTM.

La Figura 4.5 muestra la disposición interna de una red GRU. La principal diferencia con las redes LSTM anteriormente mencionadas es la reducción de una de sus puertas internas. De esta forma, las redes GRU sintetizan la antigua puerta de entrada y olvido de las LSTM en una sola puerta de actualización [151]. Los modelos GRU se basan en las siguientes ecuaciones:

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \quad (4.11)$$

La Ecuación 4.11 muestra cómo calcular z_t , que corresponde con el valor de la

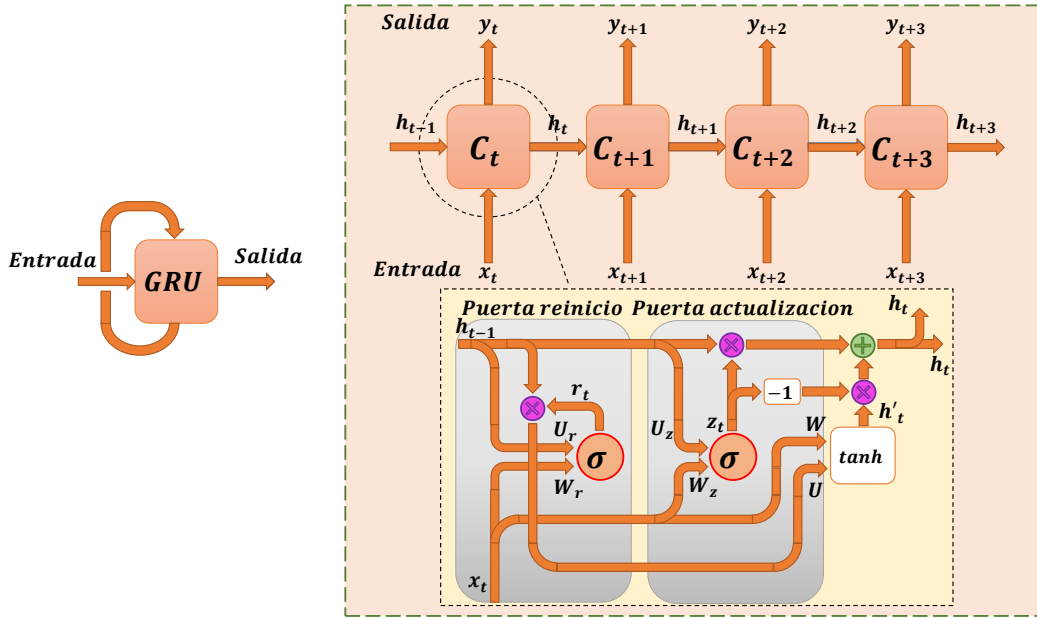


Figura 4.5: Contenido de una red neuronal GRU.

puerta de actualización para un tiempo dado t . Esta puerta decide qué información de eventos pasados se mantiene en la memoria, y la cantidad de información, con el fin de enviarla a las células posteriores.

$$r_t = \sigma(W_r X_t + U_r h_{t-1} + b_r) \quad (4.12)$$

La Ecuación 4.12 indica el método para calcular r_t , un parámetro que corresponde al valor de la puerta de reinicio. Esta puerta decide qué estados anteriores contribuyen a actualizar los estados ocultos de la red GRU. Por ejemplo, si el valor es 0, la red olvida el estado anterior.

$$h'_t = \tanh(W_h x_t + r_t \cdot U_h h_{t-1}) + b_h \quad (4.13)$$

La Ecuación 4.13 muestra cómo calcular el parámetro h'_t , que representa el candidato para el nuevo estado oculto de la red neuronal GRU.

$$h_t = (1 - z_t) \cdot h_{t-1} + z_t \cdot h'_t \quad (4.14)$$

Por último, en la Ecuación 4.14, se calcula el valor verdadero del nuevo estado oculto h_t . Es importante interpretar en la ecuación que este estado oculto depende del estado oculto anterior h_{t-1} . Además, las matrices W_z , W_r , W , U_z , U_r , U y los parámetros de sesgo b_z , b_r y b_h se modifican y redistribuyen durante el entrenamiento.

4.1.4. Redes neuronales convolucionales (CNN)

Las redes neuronales CNN se corresponden con un tipo de arquitectura de redes neuronales profundas especialmente diseñadas para el procesamiento eficiente de datos estructurados. Normalmente, este tipo de redes se llevan utilizando en los últimos años para el procesamiento de imágenes y señales de audio. Las redes CNN surgieron como una solución para abordar los desafíos asociados con el reconocimiento de patrones en datos de alta dimensionalidad, como la presencia de características locales y la aleatoriedad debida a traslaciones y deformaciones [152]. Por esta razón, este tipo de redes están especialmente diseñadas para tratar los píxeles que conforman las imágenes.

Las redes CNN consisten en capas que aplican filtros convolucionales para extraer características relevantes de los datos de entrada. Estas capas están seguidas por capas de submuestreo o reducción (también conocidas como capas de pooling) que reducen la dimensionalidad espacial de las características extraídas, preservando su información más relevante. Uno de los avances clave que permitió el éxito de las CNN fue la utilización de funciones de activación no lineales, como la función ReLU (Rectified Linear Unit), que introduce no linealidades en la red. Además, las CNN suelen incluir capas completamente conectadas al final de la arquitectura, las cuales integran las características extraídas por las capas convolucionales para realizar la clasificación final [153].

En la Figura 4.6 se pueden observar los componentes que conforman una red neuronal CNN. En la imagen se puede apreciar cómo las entradas de datos al modelo pasan por diferentes capas de convolución que extraen las características principales de las series de datos. A continuación, estas características pasan a través de capas de reducción para reducir la dimensionalidad. Por último, los datos pasan normalmente a través de capas de neuronas totalmente conectadas que actúan como capas de clasificación para dar lugar a la salida deseada. En el caso que atañe a la presente

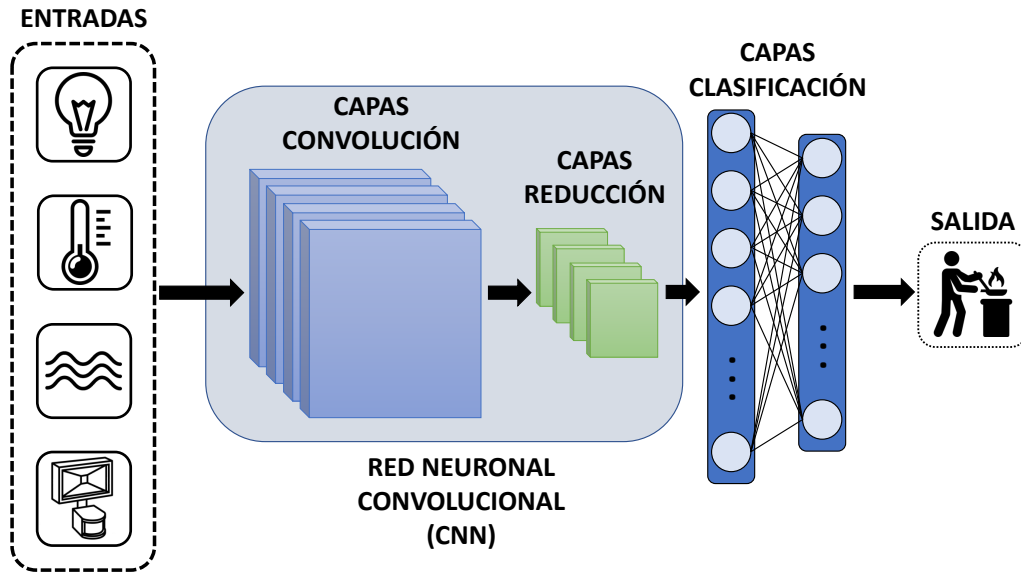


Figura 4.6: Contenido de una red neuronal CNN.

tesis, la salida deseada es la ADL que desarrollan los usuarios. Las redes CNN normalmente encadenan varias etapas de capas de convolución y capas de reducción de forma secuencial.

$$S(i, j) := (X * K)(i, j) = \sum_m \sum_n X(m, n)K(i - m, j - n) \quad (4.15)$$

La Ecuación 4.15 muestra el funcionamiento de la operación de convolución. La salida S viene expresada en función de la combinación del conjunto de datos de entrada X y el filtro de convolución K para una posición espacial de la salida (i, j) . La posición (m, n) se corresponde con la posición espacial de la entrada X .

$$ReLU(X) = \max(0, X) \quad (4.16)$$

La Ecuación 4.16 indica el funcionamiento de la función de activación ReLU para introducir no linealidades en la red.

$$S(i, j) = \max_{(m,n) \in R} X(m, n) \quad (4.17)$$

Por último, la Ecuación 4.17 muestra el funcionamiento de las capas de reducción previas a la etapa de clasificación. El término R se corresponde con la región de pooling escogida para aplicar la reducción.

4.1.5. Modelos transformadores

El método de redes neuronales Transformers representa una innovación crucial en el ámbito del aprendizaje automático y el procesamiento del lenguaje natural (NLP, del inglés *Natural Language Processing*) [154]. Su arquitectura se basa en el uso de modelos de atención y permite el procesamiento simultáneo de secuencias completas de entrada, superando así la limitación de los modelos recurrentes que procesan secuencialmente. Este enfoque revolucionario de atención múltiple permite al modelo enfocarse en partes específicas de la entrada, capturando relaciones complejas entre palabras y “tokens” [155]. Además, los Transformers son altamente paralelizables y eficientes, lo que facilita el procesamiento rápido de grandes volúmenes de datos y los hace escalables para aplicaciones de NLP de gran escala como la traducción automática y la generación de texto. Su capacidad para capturar dependencias a largo plazo y su robustez frente al ruido en los datos de entrada son características especialmente destacables [156]. Además, el enfoque de transferencia de aprendizaje permite a los Transformers pre-entrenados adaptarse fácilmente a tareas específicas con conjuntos de datos más pequeños. Los Transformers son una herramienta poderosa y versátil que ha demostrado ser efectiva en una amplia gama de aplicaciones de aprendizaje automático y NLP [157].

Cada cabeza de atención, del inglés *head attention*, se encarga de mapear una consulta (*query*) con un conjunto de pares clave-valor (*key-value*). A partir de una query como entrada de una capa de atención (Q), se obtiene una salida calculada como la suma ponderada de los valores obtenidos (V) [158].

En la Figura 4.7 se puede observar cómo se compone un modelo transformador. En el esquema se puede comprobar que es necesario separar las entradas de las salidas a la hora de realizar el entrenamiento del modelo. Las entradas pasan a través de un Encoder con cabezas de atención, las cuales se encargan de extraer la información con mayor relevancia del conjunto de salidas. En cambio, las salidas pasan a través de un decoder, compuesto a su vez de más capas de atención. Sin embargo, la primera

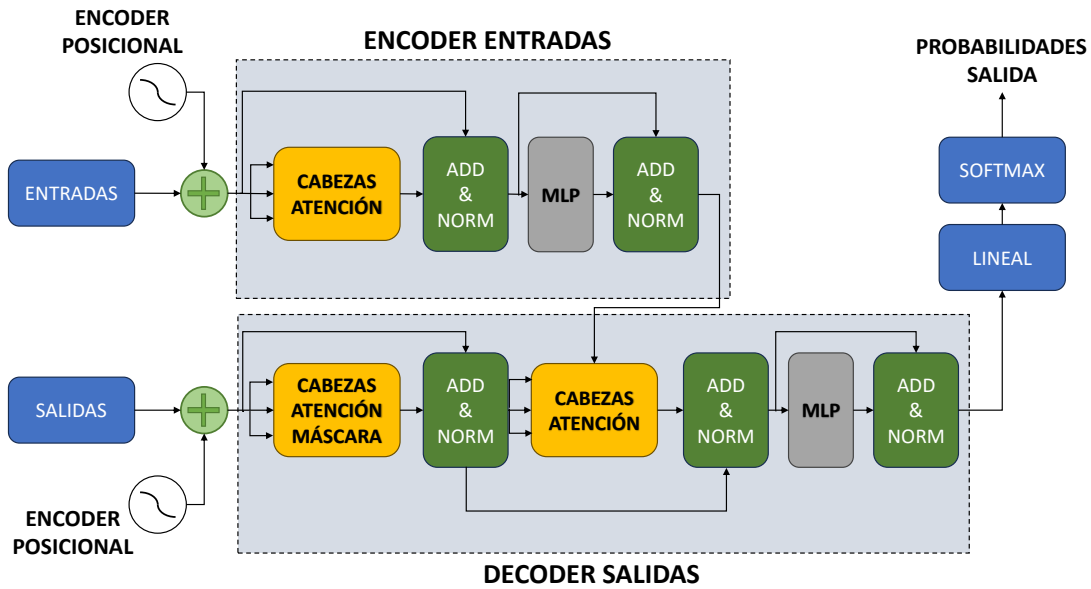


Figura 4.7: Esquema de funcionamiento de un modelo transformador.

etapa del decoder está compuesta por cabezas de atención con máscaras, las cuales eliminan la parte superior de la matriz de atención para no adelantar información futura. El encoder posicional se encarga de asignar un valor al orden en el que los datos entran al modelo. Las capas “Add & Norm” se encargan de unir las salidas de las capas de atención y normalizar sus vectores de probabilidad. Las capas MLP son capas de neuronas densas que se encargan de procesar las salidas de los encoder y decoder.

$$Z(Q, K, V) = \text{Softmax} \left[\frac{QK^T}{\sqrt{d_K}} \right] V \quad (4.18)$$

La Ecuación 4.18 modela el comportamiento de las capas de atención. La matriz Q se corresponde con los valores de las queries, la matriz K se corresponde con los valores de las keys y la matriz V con los valores obtenidos de la suma ponderada de Q y K . La capa softmax se encarga de acotar la salida a un rango $[0,1]$, amplificando las puntuaciones de salida altas del modelo y minimizando las más bajas. El parámetro d_K se corresponde con un constante que actúa como un factor de calentamiento opcional [159].

Cuando se utilizan múltiples cabezas (multi-head attention), se repite el proceso anterior para distintos mapeos de Q , K y V . Esto se consigue variando la transformación lineal de entrada con diferentes pesos. Si se supone que se cuenta con n cabezas, se tienen Q_1, \dots, Q_n queries, K_1, \dots, K_n keys y V_1, \dots, V_n valores. Cada cabeza trabaja con su tripleta (Q_i, K_i, V_i) , donde existe un conjunto de pesos del mapeo lineal diferente para cada una. Es decir, si la entrada de un sensor a lo largo del tiempo es X , las queries, keys y valores son: $Q_i = XW_{Q_i}$; $K_i = XW_{K_i}$ y $V_i = XW_{V_i}$. La salida de las capas de atención se concatena y se conecta finalmente con la capa de clasificación, normalmente una capa totalmente conectada. Al encadenar múltiples capas del transformador, cada capa aprende una representación progresivamente más abstracta y compleja de los datos. Esto puede ayudar al modelo a capturar patrones más sutiles y contextos más largos en los datos de entrada.

4.2. Reconocimiento de actividades mediante aprendizaje no supervisado

Los métodos de aprendizaje no supervisado se corresponden con un conjunto de algoritmos que, a diferencia de los métodos de aprendizaje supervisado, no tienen la necesidad de un etiquetado manual de las salidas esperadas durante la etapa de entrenamiento [160]. Este conjunto de métodos sólo necesitan datos de entrada y, mediante distintos algoritmos de clasificación y agrupación, o mediante la incorporación del conocimiento del experto, el sistema es capaz de generar un conjunto de salidas [161]. Un ejemplo de métodos no supervisados podrían ser los modelos HMM [99]. Sin embargo, la etapa de aplicación en los métodos de aprendizaje no supervisado es computacionalmente más compleja que la aplicación de métodos supervisados (independientemente del entrenamiento) [162]. El principal problema de los métodos supervisados es la necesidad de tener una salida concreta para un conjunto de entradas, y esto no suele ocurrir en casos reales en los que se desea un aprendizaje en tiempo real [163]. Además, en los HMM se dispone de un control más preciso sobre el funcionamiento del sistema en comparación con los métodos basados en redes neuronales, en los que el conocimiento se difunde por todo el sistema [164]. Esta ventaja facilita la adaptación de los HMM a un mayor número de usuarios. Los

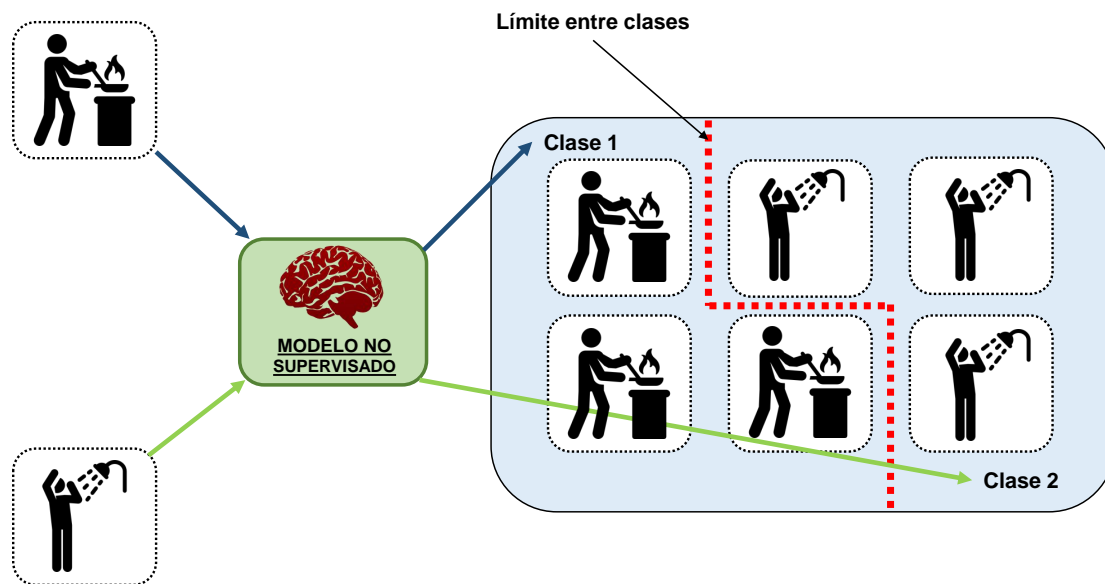


Figura 4.8: Esquema de funcionamiento de un método de aprendizaje no supervisado.

HMM también son ventajosos frente a los modelos basados en reglas [165] o árboles de decisión [166], que son métodos más estrictos y menos flexibles.

En la Figura 4.8 se puede visualizar un esquema del funcionamiento de un modelo basado en un método de aprendizaje no supervisado. Se va a retomar el ejemplo de detección de dos tipos diferentes de ADL's para aclarar el funcionamiento de los métodos no supervisados. En este caso, las actividades son cocinar y ducharse. Como se puede ver en la imagen, el objetivo de un método no supervisado es crear agrupaciones en función de la similitud que tengan las características de los datos de entrada. En este caso, el sistema sólo es capaz de separar las actividades de cocinar y de ducharse, aunque no sepa realmente que es lo que está analizando. Se puede observar que en ningún momento se le está pasando al modelo la información sobre qué ADL está analizando.

A lo largo del desarrollo de la tesis se han estudiado diferentes métodos de aprendizaje no supervisado para su posterior aplicación con datos reales. Los métodos de aprendizaje no supervisado analizados son los siguientes:

1. Modelo ocultos de Markov (HMM): Los HMMs son métodos estadísticos cuyo uso y aplicación está aumentando mucho en las últimas décadas, tal y como se evidencia en el elevado número de publicaciones que se pueden encontrar en la literatura. Estos métodos se utilizan mucho en áreas como el reconocimiento del habla, musicología o reconocimiento de gestos [167].
2. Filtros Bayesianos: Estos filtros son herramientas probabilísticas cuyo objetivo se basa en el cálculo de la probabilidad en función de una cadena de eventos pasados. El abanico de aplicaciones que se le pueden dar a este método es muy amplio, ya que forma parte de sistemas muy robustos como el GPS [168].
3. Algoritmos de propagación y retro propagación: Estos algoritmos están basados en el funcionamiento de los filtros Bayesianos, y se utilizan para calcular la probabilidad de cada estado en un instante t en función de una secuencia de observaciones conocida [169].
4. Algoritmo de Viterbi: Este algoritmo se utiliza acompañado de un HMM, cuyo objetivo es encontrar la secuencia de estados más probable que explique una determinada secuencia de observaciones [170].

En los próximos apartados se detallan los principios matemáticos de cada uno de los métodos mencionados.

4.2.1. Modelos ocultos de Markov (HMM)

Según [171], los HMMs se pueden dividir en dos niveles jerárquicos:

1. Estados ocultos S : Se corresponden con las variables del sistema que se desconocen y que se desean reconocer. Se pueden representar de la siguiente forma:

$$S = \{s_1, s_2, s_3, \dots, s_N\} \quad (4.19)$$

En la Ecuación 4.19, N corresponde al número total de estados ocultos del sistema y cada s_i se refiere, en el caso de la investigación que atañe a esta tesis, a cada una de las actividades analizadas.

2. Observaciones O : Secuencia de hechos observables (y siempre medibles) de los que se puede extraer información del entorno en el que se encuentra el sistema. En el caso que ocupa a esta tesis, esta secuencia de observaciones se corresponde con la información proporcionada por la tecnología con la que está equipado el hogar. La secuencia de observaciones puede representarse del siguiente modo:

$$O = \{o_1, o_2, o_3, \dots, o_T\} \quad (4.20)$$

En la Ecuación 4.20, cada observación o_i está relacionada con el momento en que se produce. T corresponde al número total de observaciones que hay que analizar. La secuencia de observaciones O incluye todas las mediciones posibles obtenidas del entorno en el que se encuentra el modelo. Así, el conjunto total de posibles observaciones V puede representarse de la siguiente manera:

$$V = \{v_1, v_2, v_3, \dots, v_M\} \quad (4.21)$$

En la Ecuación 4.21, M representa el número total de señales diferentes que entran en el sistema y cada v_i representa cada uno de los sensores.

Una vez definidos los estados ocultos S que incluye el modelo HMM y las observaciones O que desencadenan los cambios de estado, es necesario introducir las matrices de probabilidad que parametrizan el modelo:

1. Matriz de transición de estados A : Se corresponde con la matriz de probabilidad de las transiciones entre los distintos estados ocultos del HMM previamente definido. Por ello, es una matriz $N \times N$ (Ecuación 4.22). Además, la suma de todas las probabilidades de transición en la misma fila es igual a uno (Ecuación 4.23). La probabilidad de permanecer en el mismo estado corresponde a los valores de la diagonal, mientras que las probabilidades de pasar de un estado oculto a otro corresponden a las probabilidades restantes.

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1N} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2N} \\ a_{31} & a_{32} & a_{33} & \dots & a_{3N} \\ \dots & \dots & \dots & \dots & \dots \\ a_{N1} & a_{N2} & a_{N3} & \dots & a_{NN} \end{bmatrix} \quad (4.22)$$

$$\sum_{j=1}^N a_{ij} = 1, \forall i \in [1, N] \quad (4.23)$$

2. Matriz de emisión B : Se corresponde con la matriz de probabilidad de cambio a un determinado estado oculto en función de la observación concreta de entrada al sistema. Por ello, es una matriz cuyas dimensiones son $N \times M$ (Ecuación 4.24).

$$B = \begin{bmatrix} b_1(1) & b_1(2) & b_1(3) & \dots & b_1(M) \\ b_2(1) & b_2(2) & b_2(3) & \dots & b_2(M) \\ b_3(1) & b_3(2) & b_3(3) & \dots & b_3(M) \\ \dots & \dots & \dots & \dots & \dots \\ b_N(1) & b_N(2) & b_N(3) & \dots & b_N(M) \end{bmatrix} \quad (4.24)$$

3. Vector de probabilidad inicial: Al comienzo de la ejecución de un HMM, los diferentes estados ocultos deben comenzar con una probabilidad de salida (π_i). Esta probabilidad viene dada por la Ecuación 4.25.

$$\pi = \{\pi_1, \pi_2, \pi_3, \dots, \pi_N\} \quad (4.25)$$

4.2.2. Filtros Bayesianos

El uso de filtros Bayesianos se remonta al siglo XVIII, cuyo descubridor fue el clérigo y matemático Thomas Bayes. El algoritmo Naive Bayes es uno de los algoritmos de ML más comunes para realizar tareas de clasificación de clases debido a su sencillez y rapidez [172]. Este algoritmo tiene como base el Teorema de Bayes, cuyo fundamento se explica a continuación [173]:

$$P(A|B) = \frac{|A \cap B|}{|B|} = \frac{\frac{|A \cap B|}{|\Omega|}}{\frac{|B|}{|\Omega|}} = \frac{P(A \cap B)}{P(B)} \quad (4.26)$$

$$P(B|A) = \frac{|B \cap A|}{|A|} = \frac{\frac{|B \cap A|}{|\Omega|}}{\frac{|A|}{|\Omega|}} = \frac{P(A \cap B)}{P(A)} \quad (4.27)$$

En las Ecuaciones 4.26 y 4.27, A y B se corresponden con los estados analizados por el teorema de Bayes. Ω se corresponde con el espacio muestral que, según la teoría de conjuntos, se puede denominar conjunto universal $\Omega = \{1, 2, 3, 4, 5, 6\}$. Es necesario asumir que $|A| \neq 0$ y que $|B| \neq 0$. De estas ecuaciones se puede realizar las siguientes deducciones (Ecuaciones 4.28 y 4.29).

$$P(A \cap B) = P(A|B) P(B) = P(B|A) P(A) \quad (4.28)$$

$$P(A|B) = \frac{P(B|A) P(A)}{P(B)} \quad (4.29)$$

La Ecuación 4.29 se corresponde con la fórmula más simple del teorema de Bayes. Si el espacio muestral Ω puede dividirse en un número finito de estados mutuamente excluyentes $A = \{A_1, A_2, \dots, A_n\}$ y si B es un evento en el que $P(B) > 0$ (subconjunto de la unión de todos los A_i), la fórmula del teorema de Bayes generalizada es la siguiente:

$$P(A_i|B) = \frac{P(B|A_i) P(A_i)}{\sum_{j=1}^n P(B|A_j) P(A_j)} \quad (4.30)$$

A su vez, la Ecuación 4.30 se puede reescribir de la siguiente manera:

$$P(A|B) = \frac{P(B|A) P(A)}{P(B|A) P(A) + P(B|A^c) P(A^c)} \quad (4.31)$$

En la Ecuación 4.31, los parámetros A^c se corresponden con los valores de las probabilidades del complemento de A . De estas ecuaciones, la letra A se corresponde con la hipótesis que se está tomando y la letra B se corresponde con los valores de los datos observables.

El teorema de Bayes expuesto sirve para obtener la probabilidad de que suceda

una hipótesis en función de un conjunto de datos observables:

$$P(\text{hipotesis}|\text{datos}) = \frac{P(\text{datos}|\text{hipotesis}) P(\text{hipotesis})}{P(\text{datos})} \quad (4.32)$$

En la Ecuación 4.32, el parámetro $P(\text{datos}|\text{hipotesis})$ se corresponde con la probabilidad de observar la secuencia de datos teniendo en cuenta de que la hipótesis es verdadera. $P(\text{hipotesis})$ se corresponde con la probabilidad con la que puede suceder la hipótesis y $P(\text{datos})$ es la probabilidad de observar la secuencia de datos, independientemente de si sucede o no la hipótesis.

Puede suceder que un conjunto de datos puede surgir bajo dos hipótesis H_1 y H_2 , en el que $P(H_1) = 1 - P(H_2)$. Por lo tanto, las probabilidades de las hipótesis H_1 y H_2 son las siguientes:

$$P(H_1|C) = \frac{P(C|H_1) P(H_1)}{P(C|H_1) P(H_1) + P(C|H_2) P(H_2)} \quad (4.33)$$

$$P(H_2|C) = \frac{P(C|H_2) P(H_2)}{P(C|H_1) P(H_1) + P(C|H_2) P(H_2)} \quad (4.34)$$

El parámetro C de las Ecuaciones 4.33 y 4.34 se corresponde con el conjunto de datos observables.

El teorema de Bayes es la base del funcionamiento del clasificador Naive Bayes [174]. Este tipo de clasificador supone que el efecto del valor de un atributo en una clase determinada es independiente de los valores del resto de atributos. Esta suposición se conoce como independencia condicional de clases [175]. Asumiendo que el conjunto de datos contiene un total de n instancias x_i , donde $i = 1, \dots, n$, cada una de las instancias pertenecen solamente a una clase $y \in y_1, \dots, y_c$.

4.2.3. Algoritmos de propagación y retro propagación

Los algoritmos de propagación y retro propagación están basado en los filtros Bayesianos tradicionales [176]. Estos algoritmos se utilizan para conocer la secuencia de estados ocultos S a medida pasa el tiempo y el modelo reciba un elevado conjunto de observaciones. En la Figura 4.9 se puede apreciar un resumen del caso de uso de un modelo cuya máquina de estados consta de 3 estados ocultos S (Figura 4.9a) y

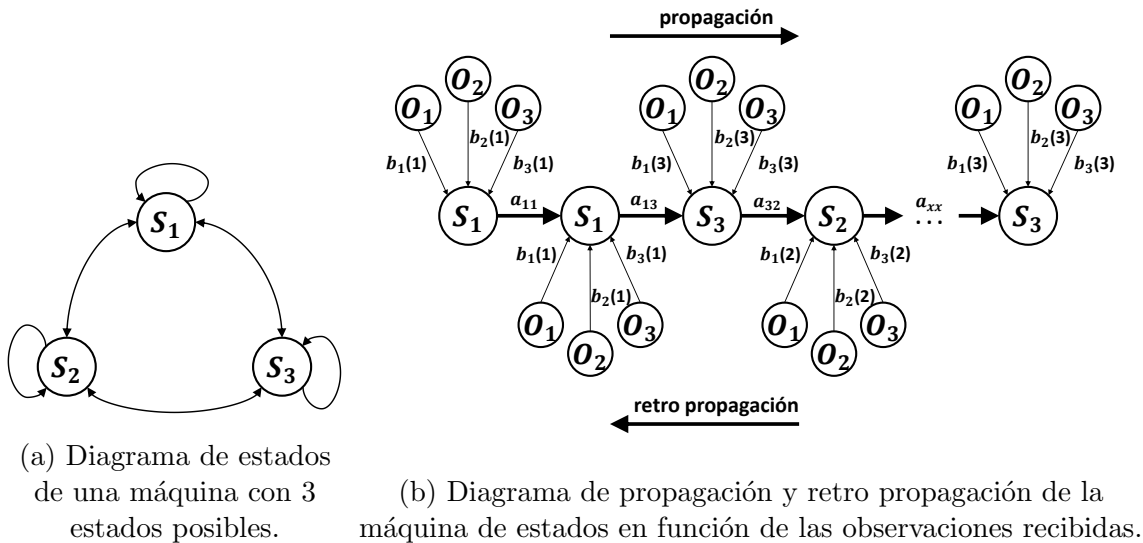


Figura 4.9: Caso de uso de una máquina de 3 estados genérica e implementación del algoritmo de propagación y retro propagación.

una posible secuencia en función del paso de tiempo t y la recepción de observaciones O (Figura 4.9b).

El algoritmo de propagación (o algoritmo directo) utiliza la probabilidad calculada en el paso de tiempo actual t con el fin de obtener la probabilidad del siguiente paso de tiempo $t + 1$. Este algoritmo necesita como entradas: un HMM que defina el sistema a simular y la secuencia de observaciones O que van a ir provocando las transiciones entre estados ocultos S . Cada etapa de propagación tiene lugar cuando se recoge una nueva observación, y en cada etapa de propagación tienen lugar dos sucesos: una primera dispersión de la certidumbre entre todos los estados ocultos S y una etapa de actualización en la que intervienen las matrices A y B en función del estado oculto actual y la observación recogida [177]. Para comenzar a iterar en el algoritmo, es necesario contar con la distribución inicial de probabilidades π (ver Algoritmo 1).

El algoritmo de retro propagación (o algoritmo inverso) fluye en la línea temporal inversa que el algoritmo de propagación previamente expuesto. Este algoritmo utiliza la probabilidad calculada en el paso de tiempo $t + 1$ para obtener la probabilidad del instante actual t . Cada etapa de retro propagación tiene lugar cada vez que llega una

Algorithm 1 Algoritmo de propagación

Entradas: HMM y secuencia de observaciones $O = o_1 \dots o_T$

Salidas: $\delta_t(s_i) = P(X_t = s_j, O) \forall s_i$

```
1: if  $t = 1$  then
2:    $\delta_1(s_i) = b_i(o_1) \cdot \pi_i, i \in [1, n]$ 
3: else
4:   for  $k = 2$  to  $t$  do
5:     for  $j = 1$  to  $n$  do
6:        $\delta_k(s_j) = b_j(o_k) \sum_{i=1}^n (a_{ij} \cdot \delta_{k-1}(s_i))$ 
7:     end for
8:   end for
9: end if
10: return  $\delta_t(s_i), i \in [1, n]$ 
```

nueva observación al modelo. Hay que tener en cuenta que, si el algoritmo utiliza un total de n observaciones posteriores para calcular la probabilidad en t (primera iteración en $t+n$), siempre existirá un desfase de n observaciones a la hora de calcular la probabilidad en t [176]. Para comenzar a iterar en el algoritmo, es necesario contar con las matrices A y B de probabilidades, así como un vector inicial equiprobable $\beta_t(s_i)$ que normalmente tiene valor 1 (ver Algoritmo 2).

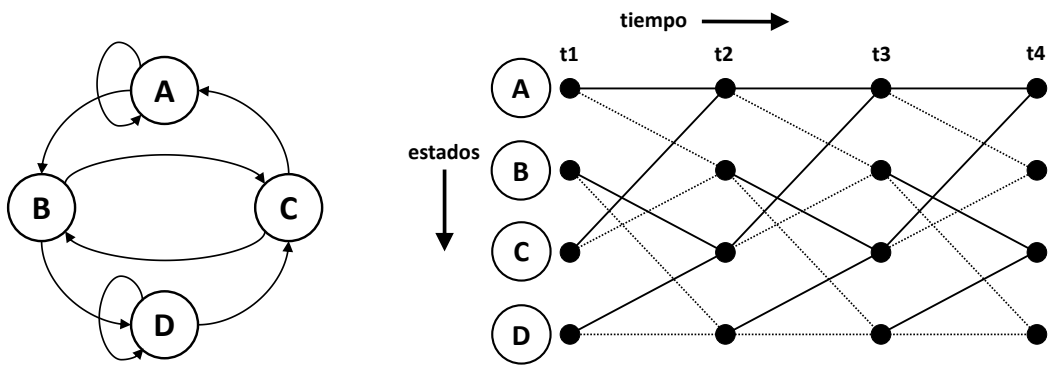
Algorithm 2 Algoritmo de retro propagación

Entradas: HMM y secuencia de observaciones $O = o_1 \dots o_T$

$k \in [1, t]$

Salidas: $\beta_k(s_i) = P(o_{k+1}, \dots, o_t | X_k = s_i) \forall s_i$

```
1: if  $t = k$  then
2:    $\beta_t(s_i) = 1, i \in [1, n]$ 
3: else
4:   for  $r = t - 1$  to  $k$  do
5:     for  $j = 1$  to  $n$  do
6:        $\beta_r(s_j) = \sum_{i=1}^n b_i(o_{r+1}) \beta_{r+1}(s_i) a_{ji}$ 
7:     end for
8:   end for
9: end if
10: return  $\beta_k(s_i), i \in [1, n]$ 
```



(a) Diagrama de estados de una máquina con 4 estados posibles.

(b) Diagrama de transiciones de la máquina de estados a lo largo del tiempo.

Figura 4.10: Caso de uso de una máquina de 4 estados genérica e implementación del algoritmo de Viterbi.

4.2.4. Algoritmo de Viterbi

El algoritmo de Viterbi surgió en el año 1967 como desarrollo por parte de Andrew James Viterbi [178]. Se considera que el algoritmo de Viterbi tiene como objetivo minimizar la probabilidad del error al comparar y calcular el conjunto de probabilidades de las posibles transiciones entre estados del modelo examinado. De esta forma, obteniendo el conjunto de transiciones con el mínimo error, se obtiene la secuencia de transiciones más probable. Este algoritmo se utiliza en un amplio campo de aplicaciones en la actualidad [179]. Por ejemplo, se utiliza en áreas como el reconocimiento de voz o interpretación de palabras manuscritas [180].

En la Figura 4.10 se puede observar un caso de uso típico a la hora de aplicar el algoritmo de Viterbi. El modelo se puede representar como una máquina de estados [181] en la que cada nodo representa un estado particular y cada flecha corresponde con cada una de las posibles transiciones entre estados. Estas transiciones sólo se pueden producir en intervalos de tiempo discretos consecutivos.

Antes de definir el algoritmo de Viterbi, es necesario conocer el significado de los siguientes símbolos:

1. t - Índice de tiempo discreto.
2. N - Número total de estados en la máquina de estados finitos.

3. x_n - El enésimo estado de la máquina de estados finitos.
4. o_t - Símbolo de observación en el tiempo t , que puede ser uno de M diferentes símbolos.
5. sp_{nt} - Ruta de supervivencia que termina en el tiempo t , en el enésimo estado de la máquina de estados finitos. Consiste en una lista ordenada de x_n visitados por esta ruta desde el tiempo $t = 0$ hasta el tiempo t .
6. T - Longitud de truncamiento del algoritmo de Viterbi. Es decir, el momento en que el algoritmo debe tomar una decisión sobre cuál sp_{nt} es el más probable.
7. π_n - Métrica de estado inicial para el enésimo estado en $t = 0$. Se define como la probabilidad de que el enésimo estado sea el inicio más probable, es decir, $P(x_n \text{ en } t = 0)$.
8. a_{nm} - Métrica de transición para la transición desde el estado x_m en el tiempo $t - 1$ al estado x_n en el tiempo t . Se define como la probabilidad de que, dado que el estado x_m ocurre en el tiempo $t - 1$, el estado x_n ocurrirá en el tiempo t , es decir, $P(x_n \text{ en } t | x_m \text{ en } t - 1)$.
9. b_n - Métrica de observación en el tiempo t , para el estado x_n . Se define como la probabilidad de que el símbolo de observación o_t ocurra en el tiempo t , dado que estamos en el estado x_n en el tiempo t , es decir, $P(o_t | x_n \text{ en } t)$.
10. Γ_{nt} - Métrica de la ruta de supervivencia de sp_{nt} . Se define como el producto de las métricas (π_n , a_{nm} y b_n) para cada transición en la enésima ruta de supervivencia, desde el tiempo $t = 0$ hasta el tiempo t .

Las ecuaciones de las métricas del modelo a_{nm} , b_n y π_n pueden derivarse matemáticamente si se trata de una aplicación conocida. Sin embargo, si no se cuenta con estas métricas, se pueden obtener mediante la aplicación de diferentes algoritmos de reestimación, como puede ser el caso del algoritmo Baum-Welch [182] [183]. Este algoritmo obtiene de forma iterativa las probabilidades óptimas de las métricas a_{nm} , b_n y π_n a partir del entrenamiento del modelo conociendo una serie prolongada de entradas reales.

Algorithm 3 Algoritmo de Viterbi

```
1: - Etapa 1: Inicialización.
2:  $t = 0$ ;
3: for  $n = 1$  to  $N$  do
4:    $\Gamma_{n0} = \ln \pi_n$ ;
5:    $sp_{n0} = [x_n]$ ;
6: end for
7: - Etapa 2: Cálculo.
8: for  $t = 1$  to  $T$  do
9:   for  $n = 1$  to  $N$  do
10:    for  $m = 1$  to  $N$  do
11:       $\Gamma_{nt} = \text{máx}[\Gamma_{mt-1} + \ln a_{nm} + \ln b_n]$ ;
12:    end for
13:     $sp_{nt} = \text{Append}[x_n, sp_{mt}] \mid \Gamma_{mt-1} + \ln a_{nm} + \ln b_n = \Gamma_{nt}$ ;
14:  end for
15: end for
16: - Etapa 3: Decisión.
17: if  $t = T$  then
18:   for  $n = 1$  to  $N$  do
19:      $\Gamma_T = \text{máx}[\Gamma_{nt}]$ ;
20:   end for
21:    $sp_T = sp_{nt} \mid \Gamma_{nt} = \Gamma_T$ ;
22: end if
```

Se puede observar que el algoritmo de Viterbi (ver Algoritmo 3) se encuentra dividido en 3 etapas: la etapa de inicialización donde se establecen las condiciones iniciales del modelo, la etapa de cálculo donde se obtienen de forma iterativa las probabilidades de todas las transiciones posibles y la etapa de decisión, en la que se escoge el conjunto de transiciones óptimas.

4.3. Modelos propuestos para el reconocimiento de actividades

En esta Sección se van a detallar los modelos realizados en la presente tesis doctoral para detectar las principales ADL's realizadas por las personas mayores en sus casas. Se va a tratar desde el caso más simple en el que la vivienda sólo cuenta con un usuario, hasta el caso más complejo donde la vivienda cuenta con múltiples usuarios de forma simultánea. Dentro de esta Sección se explican los modelos HAR realizados mediante técnicas de aprendizaje supervisado y los modelos HAR realizados mediante técnicas de aprendizaje no supervisado.

4.3.1. Modelo basado en redes LSTM para viviendas con un usuario

En primer lugar, para desarrollar los algoritmos de reconocimiento de ADL's mediante la aplicación de métodos de aprendizaje supervisado, se ha empleado la información recogida en una base de datos pública. La base de datos pública escogida para construir el algoritmo es Milan, que pertenece al grupo de bases de datos CASAS. El proyecto CASAS presenta una arquitectura simple que facilita el despliegue de sensores dentro de un hogar. Este proyecto de hogar inteligente es un proyecto de investigación elaborado en la Universidad de Washington, cuyo objetivo es realizar una recopilación masiva de datos mediante el análisis de las señales proporcionadas por sensores no intrusivos para poder interactuar con una serie de actuadores mediante la implementación de un conjunto de reglas simples [184]. Se ha escogido la variante Milan para trabajar en los algoritmos debido a que contiene un gran número de actividades etiquetadas (un total de 2310 actividades realizadas) [185]

Tipo de Sensores	Tiempo	Actividades	Actividades (Instancias)
Movimiento (M) Puerta (D) y Temperatura (T)	Oct. 16, 2009 - Ene. 6, 2010	15	Levantarse al baño (89), Trabajar (54), Tareas domésticas (23), Actividad comedor (22), Medicación nocturna (19), Leer (314), Actividad cocina (554), Uso baño invitados (330), Salir de casa (214), Uso baño principal (306), Dormir (96), Actividad dormitorio principal (117), Meditar (17), Medicación matutina (41), Ver TV (114)

Tabla 4.1: Principales características de la base de datos Milan (CASAS).

y sólo reside una persona. Una de las ventajas que proporciona esta base de datos es que sus medidas presentan cierto ruido debido a que la vivienda cuenta con una mascota. Este hecho provoca que sea necesario tener en cuenta las perturbaciones que puede provocar la mascota en los sensores. Otro hecho que provoca ruido en las medidas son las visitas recurrentes de personas ajenas a la vivienda. Las principales características de la variante Milan se resumen en la Tabla 4.1.

La base de datos Milan está compuesta por un total de 28 sensores de presencia, 3 sensores de contacto colocados en puertas (uno de ellos situado en el cajón de las medicinas) y 2 sensores de temperatura. Cada uno de los sensores tiene un código de identificación: M para los sensores de presencia, C para los sensores de contacto y T para los sensores de temperatura. Para la elaboración de la base de datos, se han

recogido todos los eventos que se han generado en la vivienda durante casi 3 meses. Esto supone que la recogida de los eventos es asíncrona, cada vez que se genera un evento se registra en la base de datos. Cada registro está compuesto por los siguientes campos: el instante de tiempo en el que ha ocurrido el evento, el sensor afectado, el nuevo estado del sensor y el campo de actividad. Este último campo se encarga de registrar si en ese instante de tiempo se ha iniciado o ha finalizado una de las ADL's contempladas en el estudio. El formato de la base de datos Milan se puede observar en la Tabla 4.2 con un ejemplo.

Tiempo t_i	Sensor	Estado	ADL
2009-11-19 08:47:38.000019	M017	ON	Inicio uso baño principal
2009-11-19 08:47:38.000021	M015	OFF	
2009-11-19 08:47:40.000041	M011	OFF	
2009-11-19 08:47:40.000089	M018	ON	
.	.	.	
.	.	.	
.	.	.	
2009-11-19 08:49:02.000086	M018	ON	
2009-11-19 08:49:08.000076	M017	ON	Final uso baño principal

Tabla 4.2: Ejemplo de registro de la base de datos Milan durante la realización de una ADL.

Tal y como se puede ver en el ejemplo de registro de la Tabla 4.2, sólo aparecen registrados los inicios y finales de cada ADL, por lo que es importante tener constancia de los periodos de tiempo en los que el usuario está libre o está realizando alguna ADL.

Para elaborar el algoritmo de reconocimiento de ADL, se ha escogido utilizar un método de aprendizaje supervisado basado en redes neuronales, ya que al utilizar la base de datos Milan, se cuenta en todo momento con las salidas etiquetadas del modelo. El objetivo final es obtener la ADL realizada cada vez que se excede un intervalo de tiempo T . Para poder alcanzar este objetivo, es necesario entrenar el modelo en condiciones similares a las buscadas en el resultado final. Por esta razón, hay que convertir la base de datos Milan, que es asíncrona, a un conjunto de datos síncrono muestreado a un tiempo fijo T . En primer lugar, hay que transformar los estados de los sensores de Milan (ON, OFF, OPEN, CLOSE ...) en un valor

numérico que pueda ser tratado por el modelo DL. Para los sensores de puertas, se toma un valor binario para representar la apertura o el cierre. Del mismo modo, para los sensores de presencia, se asigna también un valor binario para la detección o la ausencia de movimiento. Para los sensores de temperatura, se ha realizado una normalización entre 0 y 1 para que el modelo esté ajustado en ese rango, siendo 0 el valor mínimo de temperatura registrado y 1 el valor máximo. La base de datos cuenta con un total de 15 actividades distintas etiquetadas. Sin embargo, existen períodos de tiempo en los que el usuario no está realizando ninguna actividad y existen registros de los sensores. Por esta razón, se ha incorporado una nueva actividad ficticia para identificar estos periodos ociosos. Esta actividad número 16 recibe el nombre de “Otros”.

La hora a la que se registra el evento del sensor puede ser de mucha utilidad para diferenciar entre las distintas ADL’s que componen la base de datos. Por ejemplo, la actividad de dormir es más probable que suceda durante la noche, mientras que la actividad de comer es más probable que suceda durante el día. Para poder incorporar la hora al modelo, es necesario realizar una transformación que varíe de 0 a 1. Para que los valores de tiempo sean equidistantes cuantitativamente, se ha separado la hora en una pareja de valores seno - coseno. Se han utilizado las siguientes expresiones para calcular las parejas seno-coseno:

$$TimeX = \cos\left(\frac{2\pi\left(h + \frac{min}{60}\right)}{24}\right) \quad (4.35)$$

$$TimeY = \sin\left(\frac{2\pi\left(h + \frac{min}{60}\right)}{24}\right) \quad (4.36)$$

La Ecuación 4.35 sirve para calcular el valor del coseno de la hora, mientras que la Ecuación 4.36 se utiliza para calcular el valor del seno. Estas ecuaciones proporcionan unos valores cuyo rango es $[-1,1]$. Por lo tanto, tal y como ocurre con los valores de temperatura, es recomendable normalizar para que el rango sea $[0,1]$.

Para mejorar la calidad del entrenamiento y obtener el máximo rendimiento de los datos de Milan, se ha implementado 3 etapas de procesamiento que se pueden observar en la Figura 4.11. A continuación se van a detallar cada una de las etapas y su principio de funcionamiento:

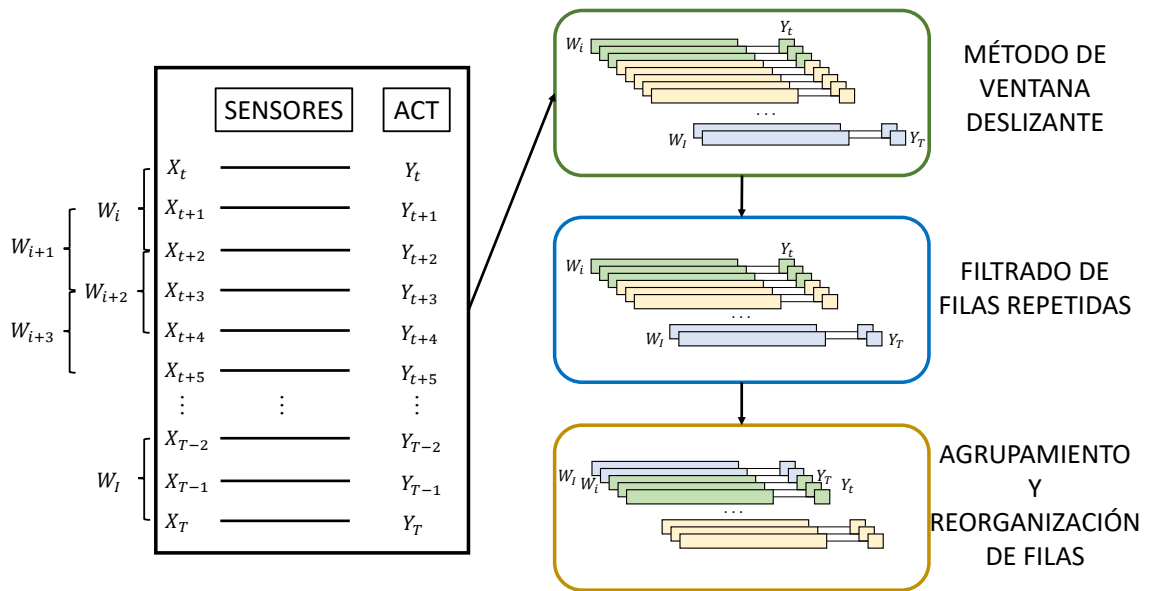


Figura 4.11: Etapas de procesamiento y tratamiento del conjunto de datos de entrada para algoritmos de un sólo residente.

1. Método de ventana deslizante: Para mantener la característica temporal de los datos, se ha empleado una técnica conocida como ventana deslizante, en los que cada entrada del modelo está compuesta por varios registros consecutivos de sensores para generar como resultado una cierta salida. Para un instante T , en la que se busca una salida del modelo Y_T , la entrada del modelo X es un vector con el siguiente formato $(X_{T-W-1}, \dots, X_{T-1}, X_T)$, en el que W se corresponde con el tamaño de la ventana temporal.
2. Filtrado de filas repetidas: Se ha utilizado esta etapa para equilibrar el número de categorías del modelo. Por ejemplo, existen un mayor número de registros de dormir que de comer. Si el entrenamiento no se realiza de forma balanceada y la diferencia de registros es muy grande, el modelo tiende a generar como salida la categoría con mayor número de registros durante el entrenamiento. Por esta razón, si una actividad excede un número de registros determinado, se mantiene un porcentaje de registros durante el inicio de la actividad y otro durante el final (que son los momentos con mayor información relativa a la

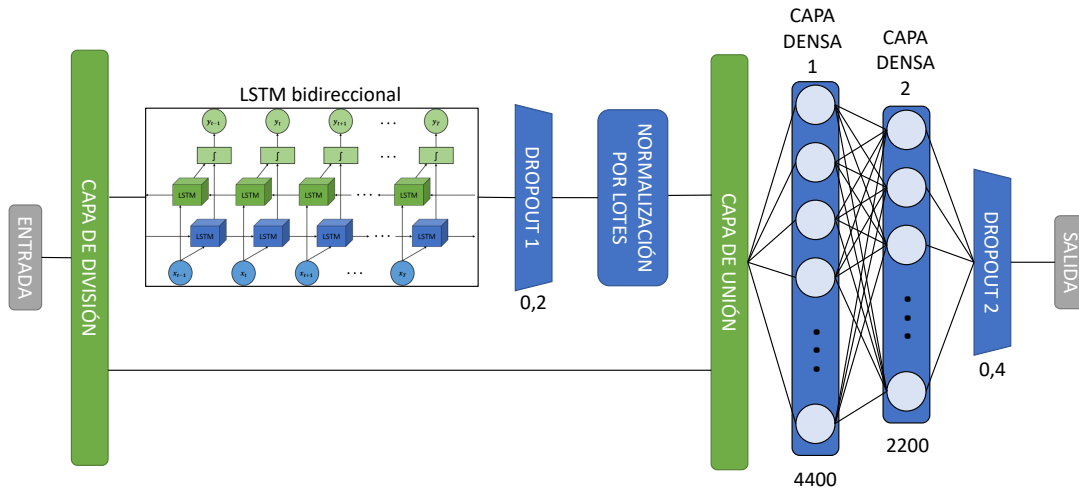


Figura 4.12: Arquitectura de red neuronal basada en capas LSTM bidireccionales para el reconocimiento de ADL's en viviendas con un residente.

ADL), y los registros intermedios se eliminan.

3. Agrupamiento y reorganización de filas: Para evitar el sobreentrenamiento, se ha realizado una agrupación de bloques de ventanas deslizantes (para mantener el orden temporal) y se han organizado de forma aleatoria para que la secuencia de los datos de entrenamiento no sea tan lineal.

Una vez que se cuenta con la base de datos tratada con los métodos comentados anteriormente, se puede definir el modelo de red neuronal. Se ha escogido un modelo basado en capas LSTM bidireccionales, cuyo funcionamiento se ha tratado en la Sección 4.1.2. En la Figura 4.12 se puede observar la arquitectura que se ha escogido para elaborar el modelo. A continuación, se van a detallar cada uno de los componentes y su función dentro de la arquitectura:

1. Capa de división: Se trata de la primera capa del modelo, y está directamente conectada con los datos de entrada. Su función es separar los datos de los sensores (que van por el camino superior) de las dos componentes seno-coseno de

la hora (que van por el camino inferior). Esto se debe a que la información con relación temporal intrínseca sólo se corresponde a los datos de los sensores, ya que se entiende que la hora ya se encuentra relacionada en el tiempo y no necesita pasar por las capas LSTM bidireccionales (ya se encuentran equidistantes en el tiempo).

2. LSTM bidireccional: Es el bloque principal del modelo de red neuronal y el que le proporciona la correlación temporal. Para mejorar el comportamiento de las capas LSTM bidireccionales y evitar el sobreentrenamiento, se ha empleado el método de regularización L2 [186]. Se utiliza la siguiente función de coste para calcular el valor de regularización:

$$C = C_o + \frac{\lambda}{2n} \sum_{\omega} \omega^2 \quad (4.37)$$

En la Ecuación 4.37, el parámetro C_o es el valor correspondiente al coste inicial, λ es el término que cuantifica el peso de la regularización, n es el tamaño del conjunto de datos de entrenamiento y ω se corresponde con todos los parámetros internos del modelo.

3. Dropout: Se conoce como dropout a las capas de “abandono”. Estas capas se utilizan para apagar y encender de forma temporal y aleatoria las conexiones entre las neuronas del modelo durante la etapa de entrenamiento. Este fenómeno se utiliza para evitar el sobreentrenamiento y, aunque hace que el modelo aprenda más lentamente, proporciona mejores resultados. El valor que acompaña a las capas de dropout indica el porcentaje de neuronas que apaga de forma temporal. Se puede observar que la arquitectura propuesta cuenta con dos capas de dropout, una al salir de las capas LSTM bidireccionales y otra antes de generar la salida del modelo.
4. Normalización por lotes: Esta capa se utiliza para contrarrestar el fenómeno de cambio de covarianza interna [187]. Esto ocurre debido al desequilibrio entre los parámetros internos del modelo entre las diferentes capas durante la etapa de entrenamiento. Por esta razón, esta capa agrupa los datos en lotes internos del mismo tamaño para balancear el entrenamiento.

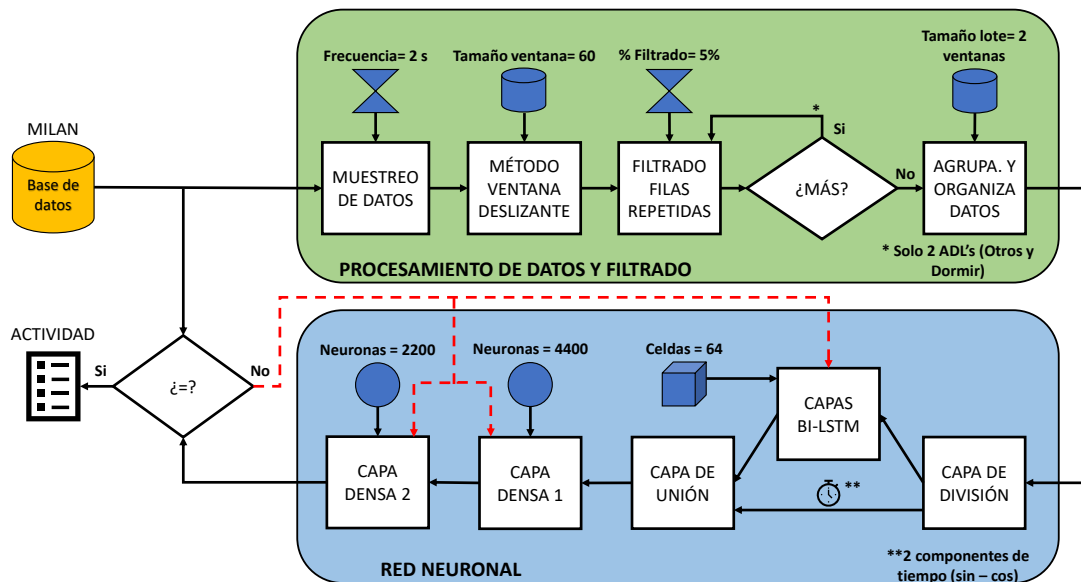


Figura 4.13: Diagrama general del sistema de monitorización de viviendas de un sólo residente: Procesamiento de datos, filtrado y red neuronal.

5. Capa de unión: Una vez que los datos de los sensores se han tratado por las LSTM bidireccionales y se han ajustado con el resto de capas internas, se vuelve a unir con las dos componentes de la hora para pasar a la etapa final del modelo.
6. Capas densas: Esta es la etapa final de la arquitectura de red neuronal. Son capas de neuronas en las que cada neurona se conecta con todas las demás, y se encargan de generar la salida del modelo. Se ha realizado una distribución de $(2/3)N$ para la primera capa y $(1/3)N$ para la segunda capa, donde N es el número total de neuronas (6600).

Por último, en la Figura 4.13 se puede observar un esquema general del sistema de detección de ADL's mediante métodos de aprendizaje supervisado para viviendas con un usuario, en el que aparecen todos los componentes que intervienen en su funcionamiento. En la parte superior de la figura se puede ver la etapa de procesamiento de datos y filtrado con los parámetros que se han escogido para el modelo, mientras que en la parte inferior se ve el modelo de red neuronal previamente explicado. Du-

rante la etapa de entrenamiento, cada vez que el modelo completo genera una salida, ésta se compara con la ADL real etiquetada en Milan. Si no coincide, el sistema reorganiza los parámetros internos del modelo para minimizar el error y mejorar los resultados.

4.3.2. Modelo basado en redes CNN, LSTM y GRU para viviendas con múltiples usuarios

Para realizar un algoritmo de reconocimiento de ADL's en viviendas con múltiples residentes es necesario contar con un sistema de posicionamiento de los usuarios dentro de la casa. Esto se debe a que, observando únicamente las señales de los sensores no intrusivos, no se puede inferir la persona que está llevando a cabo la actividad. Es importante saber con certeza el usuario que está en una situación peligrosa si, por ejemplo, uno de los usuarios de la casa se está medicando en exceso, o si lleva varios días sin comer. De forma paralela, el otro usuario puede estar realizando o dejando de realizar las mismas ADL's y el sistema tiene que ser capaz de atribuir la actividad a uno u otro. Se ha utilizado la base de datos SDHAR-HOME explicada en el Capítulo 3 y desarrollada en la presente tesis. Esta base de datos cuenta con información sobre dos usuarios viviendo de forma simultánea en una vivienda con una mascota y que reciben visitas esporádicas y aleatorias, lo que supone un beneficio a la hora de contar con una situación real y, a su vez, un reto que tratar. La base de datos tiene una duración de 2 meses de registros con un total de 17 ADL's etiquetadas junto con la actividad "Otros" para el tiempo sin actividad.

Del mismo modo que ocurre con los algoritmos para un residente, debido al volumen de datos, es necesario realizar un tratamiento previo antes de aplicar técnicas de aprendizaje profundo. Para poder utilizar los datos (que se encuentran registrados de forma asíncrona debido a la naturaleza estocástica de las señales), hay que desarrollar un algoritmo de muestreo que separe los registros en intervalos de tiempos específicos T . Esta característica permite que el sistema funcione en tiempo real, ya que se puede reducir la frecuencia de muestreo tanto como requiera la aplicación. Para cada uno de los usuarios, hay que elaborar un sistema de reconocimiento independiente para que puedan funcionar por separado, generando las salidas inherentes a cada usuario. El sistema de monitorización de la vivienda está compuesto por 3

subsistemas: la red de sensores no intrusivos, el sistema de localización de los usuarios mediante balizas Bluetooth y la información de las pulseras de actividad. Se entiende que la información proporcionada por las balizas y las pulseras de actividad es propia de cada usuario. Sin embargo, la información de la red de sensores no intrusivos es común a ambos. Por lo tanto, ambos sistemas de reconocimiento de ADL's se apoyan sobre la información de los sensores, y usan la información ligada a los usuarios para distribuir las actividades.

Con respecto a la información proporcionada por los sensores no intrusivos, es necesario realizar un tratamiento de señales similar al utilizado con la base de datos CASAS. Los valores de los sensores de presencia (verdadero y falso) y la información de los sensores de contacto (abierto y cerrado) hay que transformarlos en datos binarios (0 y 1). Del mismo modo, para los sensores que proporcionan un valor numérico (sensores de temperatura, iluminación, humedad o consumo eléctrico), se ha realizado una normalización en el rango $[0,1]$ para no desbalancear los valores de las entradas al modelo de predicción. Otro aspecto importante a tener en cuenta con la información de los sensores no intrusivos es la naturaleza de la señal y su significado. Por ejemplo, para detectar la ADL de poner la lavadora, sólo es necesario conocer el incremento de intensidad en el momento del encendido. El valor del consumo en vatios depende de la lavadora, el programa de lavado o la vida del dispositivo, por lo que no es información vital para el sistema. Por esta razón, se ha implementado un retenedor de orden cero para mantener el valor de la señal a 1 durante un cierto tiempo hasta que vuelve a valer 0 para obtener únicamente el momento de encendido del dispositivo.

En cuanto a las ADL's, del mismo modo que sucede para los algoritmos para un único residente, la hora a la que se realiza la predicción aporta mucho valor al sistema. Esto se debe a que algunas ADL's se realizan con mayor probabilidad en determinados horarios (dormir por la noche o comer al mediodía). Por esta razón, se ha implementado un algoritmo que descompone la hora en parejas seno-coseno equidistantes (ver Ecuaciones 4.35 y 4.36). Una vez separada la hora en dos componentes seno-coseno, se implementa una normalización en el rango $[0,1]$ para que los valores de las entradas estén balanceados.

A la hora de realizar el entrenamiento del modelo supervisado, es necesario que

los datos de entrada que van a formar parte del entrenamiento de la red se encuentren lo suficientemente balanceados como para no provocar un sobreentrenamiento. Del mismo modo, existen actividades en la base de datos que se realizan un mayor número de veces que otras. Por esta razón, se han aplicado las siguientes técnicas para dividir los datos correctamente en los grupos de entrenamiento, test y validación:

1. Sobremuestreo: Tal y como se ha comentado anteriormente, existen actividades que se repiten más que otras. Si se realiza un entrenamiento con demasiadas muestras de la actividad A y muy pocas muestras de la actividad B, el sistema tiende a generar como salida la actividad A porque la ha visto muchas más ocasiones. Para corregir esto, se ha elaborado un algoritmo que repite las secuencias completas de ejecución de las actividades con menos apariciones para que el conjunto de datos de entrenamiento esté balanceado.
2. Compartir datos: A pesar de utilizar la técnica del sobremuestreo, hay actividades que todavía no proporcionan buenos resultados. Esto se debe a que, aunque los datos de entrenamiento estén balanceados, muchas de las secuencias son idénticas debido a esta repetición. Para poder contar con una mayor variedad de casos (es decir, distintas formas de realizar una misma ADL), se ha elaborado un algoritmo que se encarga de compartir ADL's entre ambos usuarios, para que los conjuntos de entrenamiento de ambos casos tengan mayor variabilidad de secuencias. Por ejemplo, el usuario 2 ofrece malos resultados para las siguientes actividades: tareas del hogar, cocinar, cuidado de mascota y leer. Por lo tanto, el algoritmo obtiene diferentes ventanas temporales del usuario 1 realizando esas ADL's para aumentar el conjunto de datos de entrenamiento del usuario 2. Antes de aplicar el algoritmo, la eficacia en el reconocimiento de estas actividades no superaba el 50%. Sin embargo, después de aplicar el algoritmo, todas las ADL's superaron el 80% de precisión, mejorando de esta manera el rendimiento del sistema de monitorización.

En la Figura 4.14 se puede observar un diagrama donde aparece el proceso de distribución aleatoria del conjunto de datos en subconjuntos de entrenamiento, test y validación. De la base de datos completa, el conjunto de test se aparta para probar el rendimiento del sistema ante todas las ADL's posibles. Los conjuntos de datos

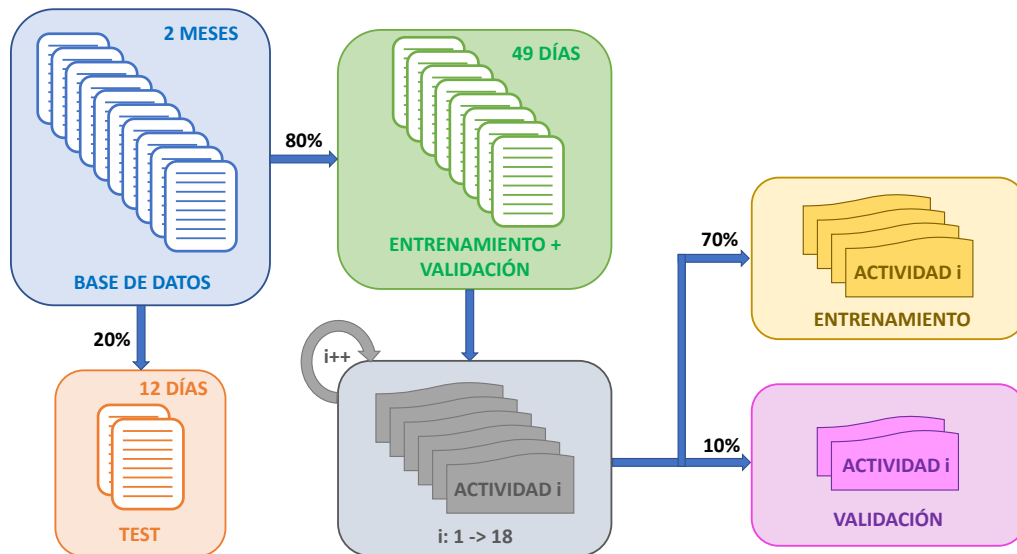


Figura 4.14: Proceso de distribución aleatoria de base de datos SDHAR-HOME en conjuntos de datos de entrenamiento, test y validación.

de entrenamiento y validación se separan y se utilizan para que el modelo realice el aprendizaje de forma realista. De todos estos datos, se separa para cada actividad las ventanas temporales contiguas correspondientes a esa ADL. Un hecho a tener en cuenta es que cada actividad tiene una duración determinada, por lo que el número de ventanas temporales es variable incluso para la misma ADL. Para cada bloque de actividad completo, se realiza una redistribución aleatoria de sus eventos para evitar el sobreentrenamiento y mejorar las prestaciones del modelo. Finalmente, para cada bloque de actividad, se separa una parte al conjunto de datos del entrenamiento y otra parte al conjunto de datos de validación. De esta manera, para cada ejecución de un bloque de actividad completa, parte se reparte a entrenamiento y otra parte a validación, dejando los datos perfectamente balanceados.

En la Figura 4.15 se puede observar un esquema de la arquitectura de red neuronal desarrollada para reconocer las ADL's de cada usuario. La arquitectura de red es la misma para ambos usuarios, pero cada usuario cuenta con su propio modelo entrenado con sus datos correspondientes para obtener un reparto de pesos personalizado. La capa de análisis de series temporales se corresponde con la implementación

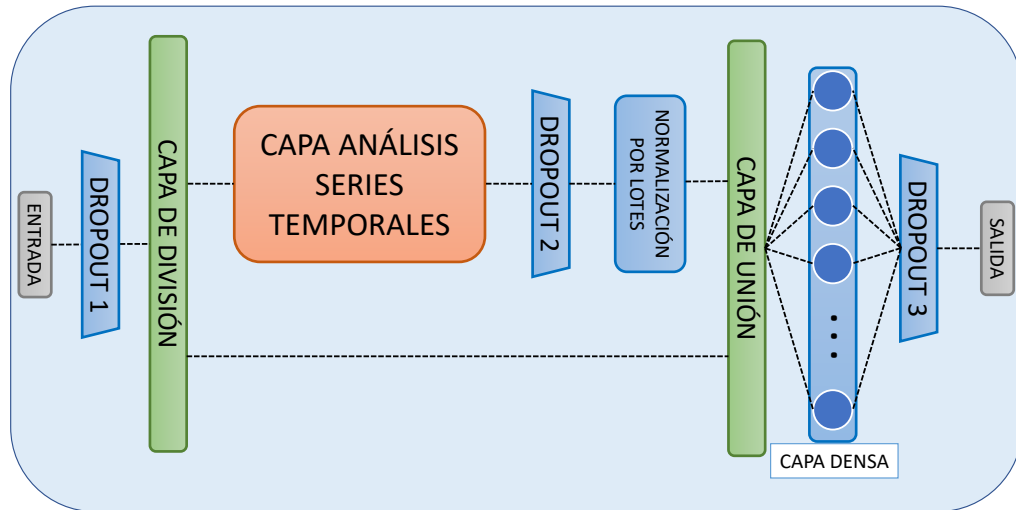


Figura 4.15: Arquitectura de red neuronal basada en capas de análisis de series temporales para el reconocimiento de ADL's en viviendas con múltiples residentes.

de los modelos de red estudiados en la Sección 4.1: RNN, LSTM y GRU. De esta manera, se puede comprobar qué tipo de red funciona mejor para cada usuario. Se han empleado esta vez 3 capas de dropout debido al alto sobreentrenamiento que presenta este modelo. Del mismo modo que para los algoritmos en viviendas de un único residente, se han utilizado capas de normalización por lotes, de división y unión para mejorar las prestaciones del modelo. También se han separado la pareja de seno-coseno correspondiente a la hora del resto de datos de los sensores, ya que el análisis de la característica temporal sólo interesa para este último grupo.

4.3.3. Modelo CVV-SV para viviendas con múltiples usuarios

Se ha propuesto realizar un modelo basado en redes neuronales CNN junto con una técnica conocida como votación por validación cruzada, del inglés *Cross Validation Voting* (CVV) [188]. En el esquema del modelo de convolución presentado en la Figura 4.16, se emplea un modelo simple de convolución 2D. Cada fila de entrada de

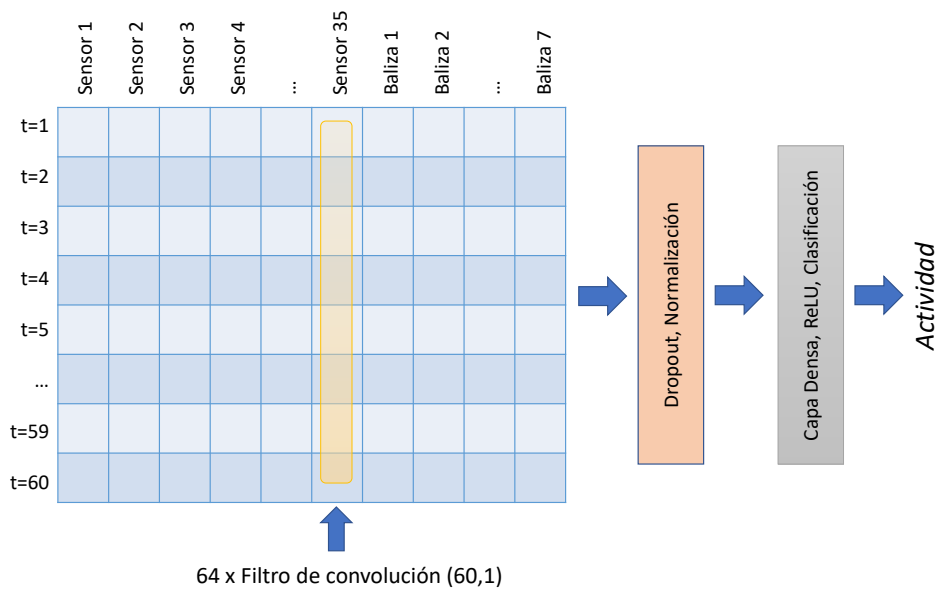


Figura 4.16: Esquema del modelo de convolución propuesto para realizar HAR utilizando la base de datos SDHAR-HOME.

SDHAR contiene la hora del día y una ventana temporal de 60 instantes de tiempo que incluye la información de 35 sensores y 7 balizas. Cada ventana representa un intervalo de 120 segundos. La separación entre la hora y los datos de los sensores se realiza mediante una bifurcación de la conexión, dirigiendo la hora hacia las capas densas del modelo. Se aplica un dropout del 0.2 tanto a los datos de los sensores como a las balizas, seguido por la aplicación de 100 filtros de convolución 2D con una dimensión de 60x1. Estos filtros se aplican a nivel de sensor o baliza a lo largo de los 60 instantes de tiempo. SDHAR considera muestras de 60 instantes de tiempo separados por 2 segundos para reconocer las actividades (2 minutos para cada actividad). Aunque podría contemplarse la aplicación de filtros de convolución temporales y multisensor, se considera que la disparidad entre los datos de los sensores y las balizas hace que carezca de sentido convolucionar datos de naturaleza diversa. Además, este enfoque repercute en que el procesamiento sea más costoso. Posterior a la capa de convolución, se aplica otro dropout del 0.2 y se normalizan los datos antes de su concatenación con los datos de la hora. Los datos de la hora no experimentan ningún tipo de dropout, dado que son datos fijos sin perturbación. Finalmente, a través de

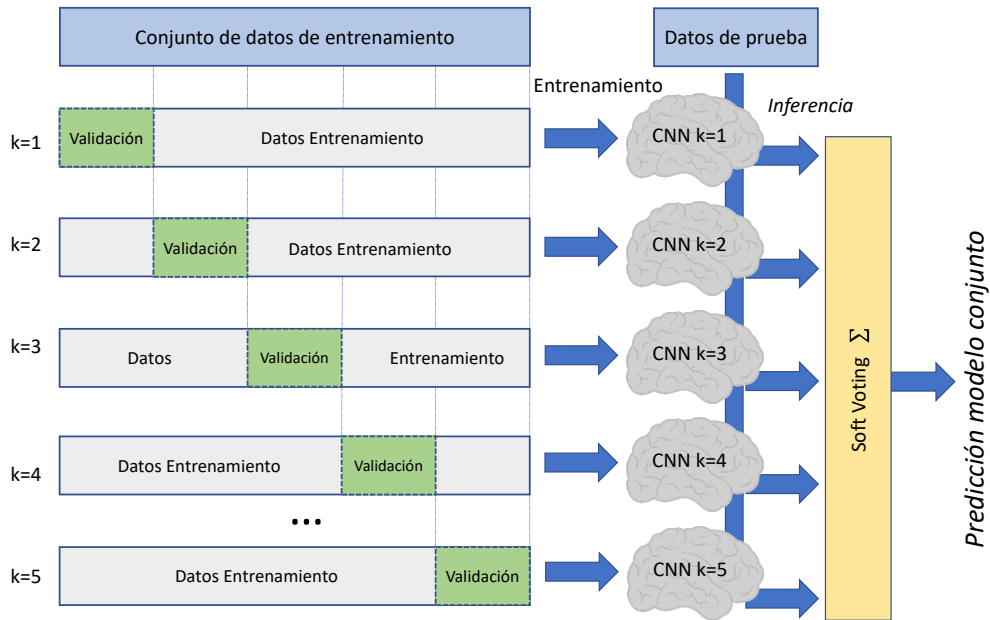


Figura 4.17: Esquema del modelo CVV propuesto para clasificar las ADL's del sistema HAR combinado con CNN.

una capa densa de 128 neuronas, una activación ReLU y una capa de clasificación con activación softmax, se obtiene la actividad inferida. Los 100 filtros se aplican sobre una matriz de tamaño 60x42 (60 instantes de tiempo por (35 sensores + 7 balizas)). Algunos enfoques alternativos sugieren el empleo de múltiples capas de convolución entrelazadas con capas de pooling. Sin embargo, en el modelo planteado, se propone una idea más simple que convoluciona toda la secuencia temporal y contribuye a mejorar los resultados en comparación con el modelo base basado en GRU y LSTM bidireccional.

Los métodos convencionales de aprendizaje supervisado optan por seleccionar conjuntos aleatorios de datos para entrenar múltiples modelos, cada uno con muestras diversas. Esta estrategia tiende a obviar la inclusión de conjuntos de validación, una preocupación creciente en el ámbito de las redes neuronales, donde las técnicas de early-stopping permiten detener el entrenamiento antes de que la pérdida de validación se dispare. El método CVV, en cambio, hace uso de todos los datos disponibles para entrenar cada modelo, variando únicamente la selección de los elementos de validación. Este enfoque garantiza que no se desperdicien muestras durante el

entrenamiento de los modelos individuales. Además, puede aplicarse a una variedad de clasificadores, como redes neuronales, SVM [189] y métodos de boosting [190]. En la Figura 4.17 se muestra la técnica empleada para implementar el enfoque CVV, que implica la aleatorización previa de los datos y la selección de k conjuntos de validación distintos. Los datos restantes de cada conjunto se utilizan para el entrenamiento. Luego, se entrenan los k modelos idénticos con cada par de conjuntos de entrenamiento y validación. Finalmente, la predicción se realiza a través de técnicas de votación aplicadas a los modelos seleccionados. Este esquema de distribución de datos en diferentes conjuntos de validación, similar a la validación cruzada, facilita el entrenamiento del modelo con diversos conjuntos de entrenamiento, mejorando así su capacidad para generalizar a diferentes escenarios. Una vez entrenados los modelos con cada par de conjuntos de entrenamiento y validación, se combinan utilizando técnicas de votación clásicas. En este caso, se ha optado por un método de votación suave.

Sea τ el conjunto de todas las muestras del conjunto de datos de entrenamiento completo. Se denota por T_i y V_i los conjuntos de entrenamiento y validación correspondientes al slot i . Estos conjuntos deben cumplir con las Ecuaciones 4.38 a 4.43.

$$\tau = \bigcup_{i=1}^k T_i \quad (4.38)$$

$$\bigcap_{i=1}^k T_i = \emptyset \quad (4.39)$$

$$\tau = \bigcup_{i=1}^k V_i \quad (4.40)$$

$$\bigcap_{i=1}^k V_i = \emptyset \quad (4.41)$$

$$[\tau = T_i \cup V_i] \forall i \in k \quad (4.42)$$

$$[T_i \cap V_i = \emptyset] \forall i \in k \quad (4.43)$$

Por lo tanto, los elementos del slot i son aquellos utilizados por todos los otros slots en validación, como se muestra en la Ecuación 4.44.

$$\left[T_i = \bigcup_{j=1}^k V_{j:j \neq i} \right] \forall i \in k \quad (4.44)$$

Por último, se procede a entrenar k modelos de idéntica naturaleza, cada uno utilizando su respectivo par de slots de entrenamiento y validación. Posteriormente, se realiza la predicción mediante técnicas de votación suave aplicadas a los modelos seleccionados. Este enfoque permite al modelo generalizar eficazmente ante diversas situaciones. Sea k el número de clasificadores, cada uno asociado con su correspondiente slot de validación. Para una muestra de entrada, x , el vector de probabilidades de salida dado por el clasificador i se representa como $\overrightarrow{p_i(x)}$. Este vector está compuesto por las probabilidades, $p_{ij}(x)$, que indica la probabilidad de que una muestra x pertenezca a la clase j según el clasificador i . \overrightarrow{c} representa el vector de etiquetas, una por cada clase posible: $\{c_1, c_2, \dots, c_N\}$. En la Ecuación 4.45, el voto suave se obtiene acumulando las probabilidades de salida de cada clase j . w_i es un peso asociado a cada clasificador i , y en este caso, $\frac{1}{k}$. La función *argmax* devuelve la posición de la clase con mayor probabilidad acumulada.

$$S(x) = c_{\text{argmax} \sum_{i=1}^k w_i \cdot p_{ij}(x)} \quad (4.45)$$

4.3.4. Modelo transformador para viviendas con múltiples usuarios

Se ha propuesto también un modelo de red neuronal transformers utilizando los datos de SDHAR-HOME (Figura 4.18).

Si se considera que cada valor del sensor y/o posición se codifica de manera específica, y que la entrada del modelo es una matriz de valores de los sensores y datos de posicionamiento, el primer paso del modelo transformador es recodificar la secuencia de estos valores a un espacio de queries (Q), keys (K) y values (V). Las consultas

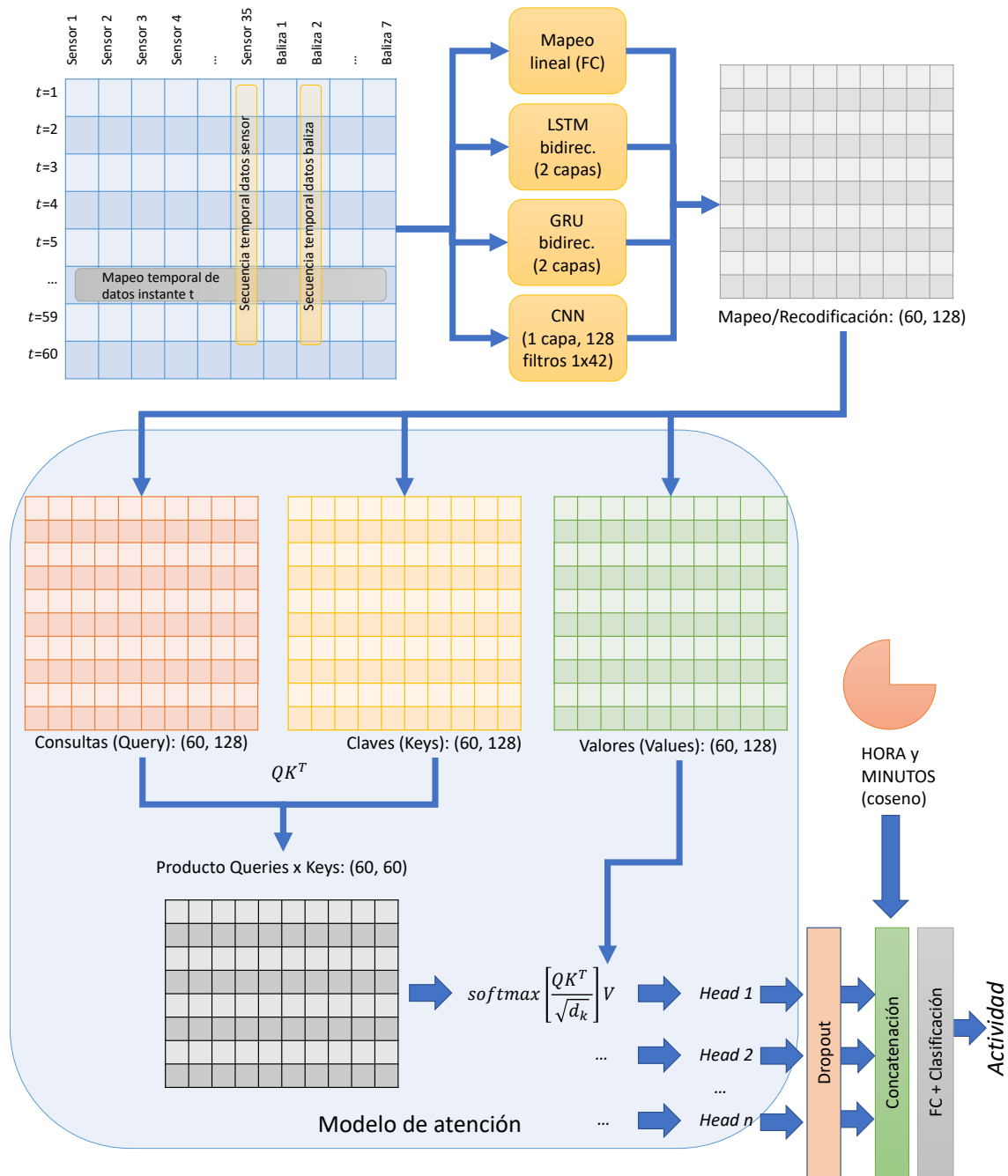


Figura 4.18: Arquitectura propuesta mediante la aplicación de modelos de redes neuronales transformer basados en capas de múltiples cabezas de atención.

se obtienen típicamente mediante una transformación lineal de los datos de entrada. Se han explorado cuatro configuraciones distintas, incluyendo además del lineal, el mapeo mediante recurrencia LSTM y GRU, así como el uso de CNN. Por ejemplo, en el procesamiento del lenguaje natural, las consultas pueden ser representaciones de palabras proyectadas a un espacio específico, mientras que en la detección de actividades a partir de datos de sensores, las consultas pueden ser características extraídas de los datos de sensores en cada instante de tiempo [191].

Se genera un vector de “embedding” en cada instante de tiempo que representa la dimensión de la representación de cada sensor/dato de posicionamiento en la secuencia. Este espacio de “embedding” se corresponde con la posición en la que los datos de cada sensor son proyectados antes de pasar por el modelo.

Cada conjunto de valores de los sensores y datos de posicionamiento en un instante de tiempo se convierte en un nuevo vector de entrada para la capa de atención, equivalente a las queries (Q), keys (K) y values (V). Luego, el mecanismo de atención considera cada valor en la secuencia como una representación de los elementos de Q , K y V . Se realiza una multiplicación matricial entre las queries y las keys para medir la similitud, seguida de una normalización mediante una función softmax para obtener los pesos de atención. Estos pesos se utilizan para combinar linealmente los valores correspondientes a las keys, generando la salida final de la capa de atención.

Cuando se utiliza atención multi-cabeza, el proceso se repite para diferentes mapeos de Q , K y V , variando la transformación lineal de entrada con diferentes pesos. La salida de las capas de atención se concatena y se conecta con la capa de clasificación, generalmente una capa completamente conectada (FC). Al encadenar múltiples capas del transformador, cada capa aprende representaciones progresivamente más complejas de los datos, permitiendo capturar patrones más sutiles y contextos más largos en los datos de entrada.

4.3.5. Modelo HMM combinatorio para viviendas con múltiples usuarios

Se ha diseñado un método de aprendizaje no supervisado para no tener que necesitar los datos etiquetados para elaborar los algoritmos de reconocimiento de ADL's. Por esta razón, se ha elaborado un modelo basado en HMM's para cada usuario

que se ejecutan en paralelo y funcionan también en tiempo real. Estos modelos no necesitan ningún tipo de muestreo, ya que a medida que el sistema va recopilando las señales de los sensores, van entrando al modelo para realizar predicciones de actividades. Un aspecto muy importante a tener en cuenta para utilizar HMM's es la naturaleza de las señales de los sensores. Por esta razón, las señales de los sensores no intrusivos se han categorizado dentro de dos grupos:

1. Sensores de evento: Estos sensores se encargan de proporcionar información sobre los distintos eventos directos que ocurren dentro de la vivienda. Es decir, se corresponden con las señales inherentes a acciones. Por ejemplo, un evento puede ser la apertura del frigorífico, vibración en la cama o movimiento en el dormitorio. Por lo tanto, los sensores que pertenecen a este grupo son: presencia, vibración y contacto.
2. Sensores ambientales: Estos sensores proporcionan información de las condiciones de la vivienda en un momento dado. Por ejemplo, si la televisión esta encendida o si la temperatura de la cocina ha aumentado por encima de un cierto valor. Estos hechos no son causa directa de que un usuario está realizando una acción, ya que la televisión puede estar encendida pero nadie la está viendo. Esta información es necesario que vaya acompañada de una señal proporcionada por un sensor de evento. En el ejemplo comentado, además del consumo, es necesario detectar presencia en el salón o vibración en el sofá para deducir que el usuario está viendo la televisión. Los sensores que pertenecen a este grupo son los siguientes: temperatura, humedad, iluminación y consumo.

De esta división en dos subgrupos se puede deducir que, dentro de los HMM's, los sensores que influyen en la matriz de emisión B son los sensores de evento. Sin embargo, para poder parametrizar la influencia de los sensores ambientales, se ha creado una matriz denominada Matriz Ambiental C :

$$C = \begin{bmatrix} c_1(1) & c_1(2) & c_1(3) & \dots & c_1(E) \\ c_2(1) & c_2(2) & c_2(3) & \dots & c_2(E) \\ c_3(1) & c_3(2) & c_3(3) & \dots & c_3(E) \\ \dots & \dots & \dots & \dots & \dots \\ c_N(1) & c_N(2) & c_N(3) & \dots & c_N(E) \end{bmatrix} \quad (4.46)$$

La estructura de la matriz C se puede ver en la Ecuación 4.46. Se trata de una matriz de dimensiones $N \times E$, donde N se corresponde con el número total de estados ocultos (ADL's) del HMM y E es el número total de sensores ambientales. De esta manera se encuentra distribuido el peso de influencia de cada sensor ambiental a cada ADL.

Con respecto a la información proporcionada por el sistema de localización de usuarios, se puede deducir que existe ruido en las señales que puede provocar fallos en el sistema. Esto se debe a la precisión que se puede alcanzar mediante las señales BLE, ya que son sensibles a paredes, obstáculos en movimiento o posible ruido electromagnético provocado por los electrodomésticos de la vivienda. Esto puede provocar que el sistema informe sobre transiciones entre habitaciones sin que el usuario realice ningún movimiento. De este comportamiento se pueden extraer las siguientes conclusiones: es imposible realizar la transición entre dos habitaciones que no estén conectadas entre sí y es imposible moverse entre estancias sin que los sensores PIR detecten presencia. Por todo esto, se ha desarrollado otro HMM independiente al de las ADL's que se encarga de filtrar las transiciones entre estancias para cada usuario. En este HMM, los estados ocultos se corresponden con cada una de las 7 estancias que componen la vivienda de SDHAR-HOME.

En la Figura 4.19 se puede observar un esquema de la morfología de la vivienda de SHDAR-HOME y sus posibles transiciones. Se puede ver que es imposible realizar una transición entre estancias sin pasar antes por el pasillo. Este hecho va a condicionar los valores de la matriz de transición A del HMM de localizaciones, ya que los valores de las transiciones imposibles son cercanos a 0 (por ejemplo, pasar del estudio a la cocina sin pasar por el pasillo). Con respecto a la matriz de emisión B , las observaciones se corresponden con la información de los sensores PIR de la instalación y los valores proporcionados por las balizas BLE.

En la Figura 4.20 se puede observar la estructura del HMM para la localización de usuarios en interiores. La Figura 4.20a muestra las posibles transiciones entre las diferentes estancias de la vivienda (relaciones de matriz A), mientras que la Figura 4.20b indica la relación entre la cadena de estados ocultos con un ejemplo de posibles observaciones del sistema. Los nodos correspondientes a los sensores PIR se marcan en amarillo, mientras que los nodos correspondientes a la información de las balizas se

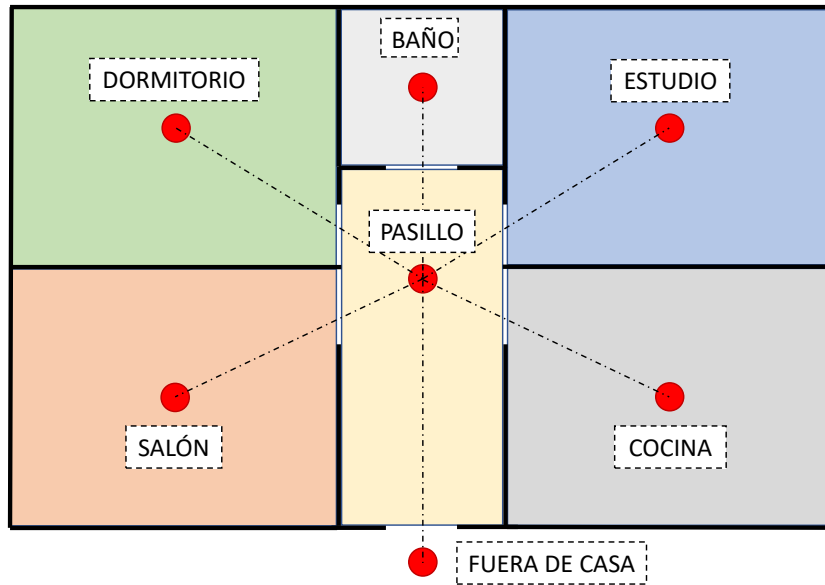
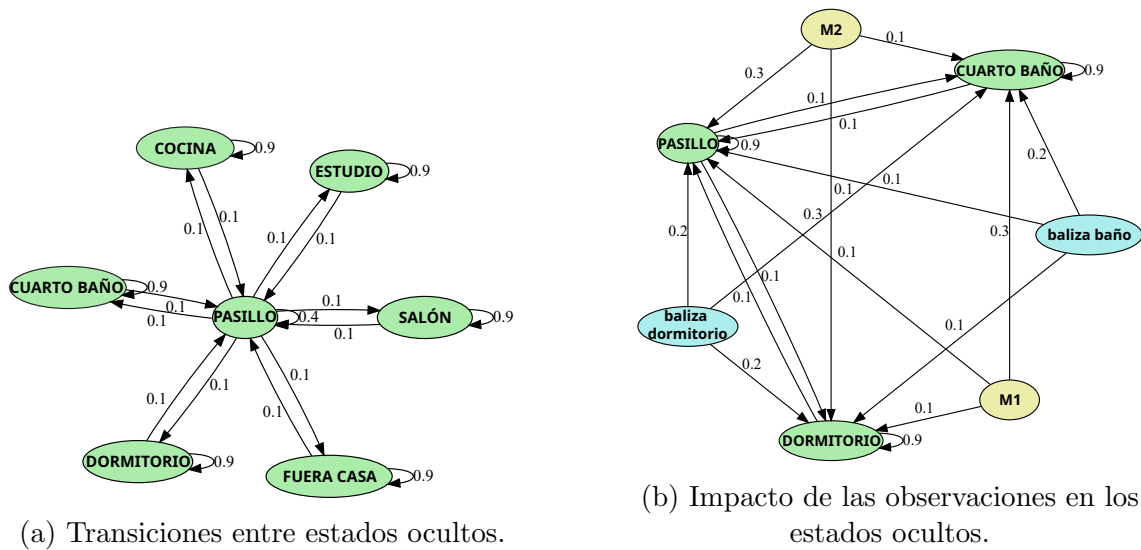


Figura 4.19: Esquema de la morfología de la vivienda de SDHAR-HOME y las posibles transiciones entre estancias.



(a) Transiciones entre estados ocultos.

(b) Impacto de las observaciones en los estados ocultos.

Figura 4.20: Algoritmo basado en HMM para la localización en interiores.

muestra en color azul y los estados ocultos (estancias de la vivienda) están marcadas en verde.

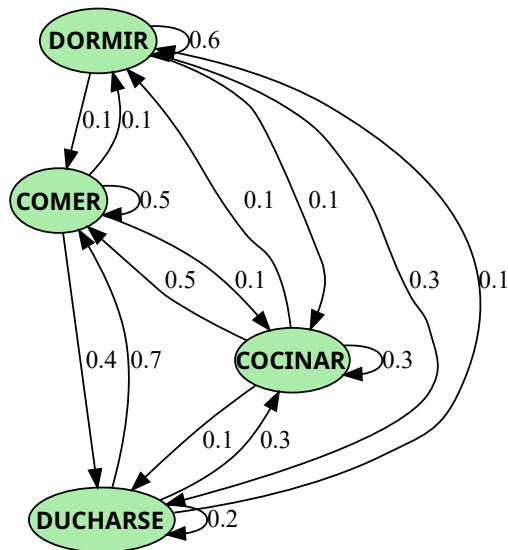
Una vez que se considera que la información de la posición de los usuarios en la vivienda está filtrada y es precisa, se puede definir una nueva matriz que relacione cada una de las estancias con la posible actividad que se puede llevar a cabo. Por ejemplo, es muy poco probable que una persona esté comiendo en el baño, y la probabilidad es mayor si se encuentra en la cocina. Por esta razón, se ha creado la matriz M :

$$M = \begin{bmatrix} m_1(1) & m_1(2) & m_1(3) & \dots & m_1(R) \\ m_2(1) & m_2(2) & m_2(3) & \dots & m_2(R) \\ m_3(1) & m_3(2) & m_3(3) & \dots & m_3(R) \\ \dots & \dots & \dots & \dots & \dots \\ m_N(1) & m_N(2) & m_N(3) & \dots & m_N(R) \end{bmatrix} \quad (4.47)$$

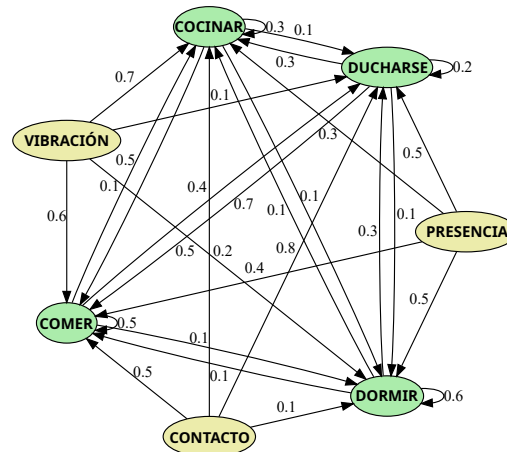
En la Ecuación 4.47 se puede observar el formato de la matriz M . Esta matriz tiene unas dimensiones de $N \times R$, donde R es el número de posibles estancias de la vivienda mientras que N es el número de estados ocultos del HMM, en este caso, las ADL's.

En la Figura 4.21 se puede observar la estructura del HMM encargado de realizar el reconocimiento de las ADL's en viviendas con múltiples usuarios. Es necesario recalcar que cada usuario tiene su modelo propio y personalizado. En la Figura 4.21a se muestra un ejemplo de las posibles ADL's que el sistema es capaz de detectar y la relación entre ellas (matriz A), mientras que la Figura 4.21b muestra la relación de las ADL's con las señales proporcionadas por la red de sensores no intrusivos. En este caso, se muestra en color verde las posibles ADL's a detectar y en color amarillo los nodos correspondientes a los sensores.

Los algoritmos correspondientes a los métodos de aprendizaje no supervisado (explicado en la Sección 4.2) escogidos para realizar las predicciones están basados en los algoritmos de propagación y retro propagación comentados en la Sección 4.2.3. Se ha realizado esta elección debido a que son algoritmos en los que se pueden implementar nuevas funcionalidades. Por ejemplo, para el HMM encargado de detectar la posición de los usuarios de la casa, se han empleado algoritmos de propagación (ver



(a) Transiciones entre estados ocultos.



(b) Impacto de las observaciones en los estados ocultos.

Figura 4.21: Algoritmo basado en HMM para la detección de ADL's en viviendas con múltiples usuarios.

Algoritmo 1). Sin embargo, para el HMM encargado de detectar las ADL's, se ha utilizado un algoritmo de retro propagación pero se han implementado una serie de cambios al algoritmo de propagación para poder incorporar los sensores ambientales (ver Algoritmo 4).

Se ha llevado a cabo un estudio para comprobar las ADL's que se realizan de forma simultánea por ambos usuarios. De este estudio se puede deducir que existen ADL's cuya probabilidad de realización de forma conjunta es más alta. Por ejemplo, se ha podido ver que si el usuario 1 se encuentra durmiendo, es muy probable que el usuario 2 también esté durmiendo. Del mismo modo ocurre con la actividad de comer, ya que ambos usuarios comen juntos. Por esta razón, se ha implementado una nueva matriz R que relaciona las ADL's de ambos usuarios y refuerza el modelo (ver Ecuación 4.48).

Algorithm 4 Algoritmo de propagación con sensores ambientales

Entradas: HMM, secuencia de observaciones $O = o_1 \dots o_T$ y secuencia de sensores ambientales $E = e_1 \dots e_T$

Salidas: $\alpha_t(s_i) = P(X_t = s_j, O, E) \forall s_i$

```
1: if  $t = 1$  then
2:    $\alpha_1(s_i) = b_i(o_1) \cdot \pi_i, i \in [1, n]$ 
3: else
4:   for  $k = 2$  to  $t$  do
5:     for  $j = 1$  to  $n$  do
6:        $\alpha_k(s_j) = b_j(o_k) \sum_{i=1}^n (a_{ij} \cdot \alpha_{k-1}(s_i))$ 
7:       if  $e_k \neq 0$  then
8:          $\alpha_k(s_j) = c_j(e_k) \cdot \alpha_k(s_j)$ 
9:       end if
10:    end for
11:  end for
12: end if
13: return  $\alpha_t(s_i), i \in [1, n]$ 
```

$$R = \begin{bmatrix} r_1(1) & r_1(2) & r_1(3) & \dots & r_1(N_2) \\ r_2(1) & r_2(2) & r_2(3) & \dots & r_2(N_2) \\ r_3(1) & r_3(2) & r_3(3) & \dots & r_3(N_2) \\ \dots & \dots & \dots & \dots & \dots \\ r_{N_1}(1) & r_{N_1}(2) & r_{N_1}(3) & \dots & r_{N_1}(N_2) \end{bmatrix} \quad (4.48)$$

$$\gamma(U_1) = R \cdot (\alpha_i(U_2) \cdot \beta_i(U_2) \cdot \delta_i(U_2)) \quad (4.49)$$

La Ecuación 4.49 muestra cómo se puede calcular el parámetro γ , el cuál se utiliza para recoger la salida del otro usuario con el fin de aplicar este valor en el motor de inferencia del usuario analizado.

Finalmente, una vez que se conocen todos los parámetros del modelo (α , β , δ y γ) para cada usuario, se puede obtener el vector de probabilidades de realización de ADL's para un instante t :

$$ACT(s_i) = \alpha(s_i) \cdot \beta(s_i) \cdot \gamma(s_i) \cdot \delta(s_i) \quad (4.50)$$

En la Ecuación 4.50 se puede observar el funcionamiento del proceso o motor

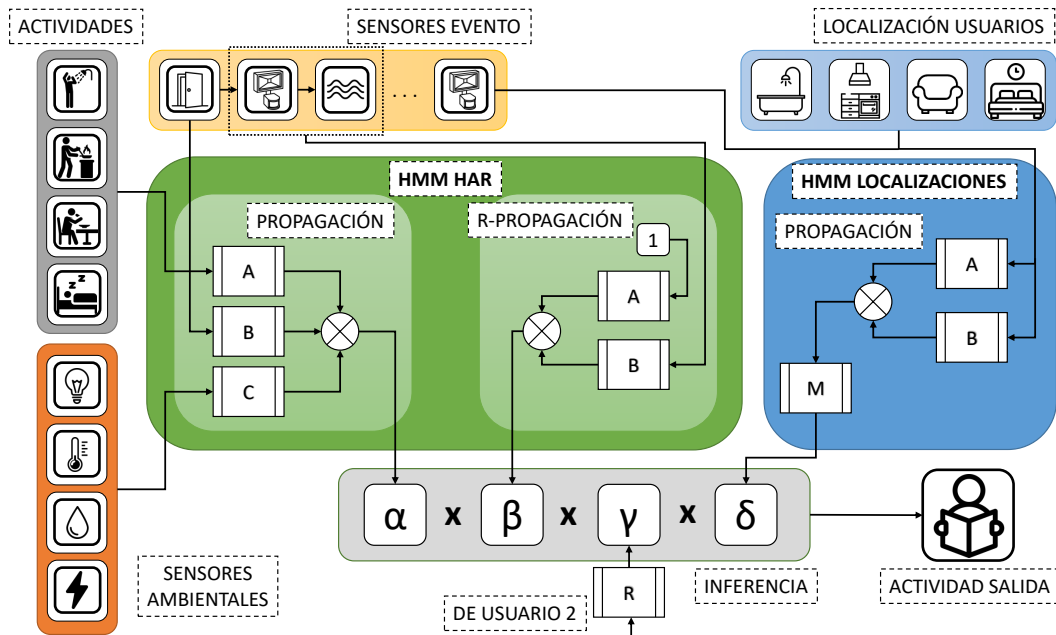


Figura 4.22: Arquitectura de modelo de aprendizaje no supervisado basado en HMM's para el reconocimiento de ADL's en viviendas con múltiples residentes.

de inferencia. El resultado $ACT(s_i)$ se corresponde con un vector de probabilidades, cuya ADL más probable se corresponde con la componente del vector más alta. Como es lógico, todas las probabilidades del vector suman 1.

4.3.6. Modelo HMM jerárquico para viviendas con múltiples usuarios

En la Figura 4.22 se puede observar un esquema general de la arquitectura generada para realizar HAR en viviendas con múltiples residentes utilizando métodos de aprendizaje no supervisado. En el diagrama se pueden ver los dos HMM's implementados: en la parte de la izquierda (HMM HAR) se encuentra el modelo encargado de detectar las ADL's, mientras que en la parte derecha (HMM LOCALIZACIONES) se encuentra el modelo encargado de localizar a los usuarios en la vivienda. Se puede ver que HMM HAR está compuesto de algoritmos de propagación y retro propagación, mientras que HMM LOCALIZACIONES sólo tiene etapa de propagación. Cada uno de los HMM cuenta con sus propias matrices internas. Las entradas del algoritmo de

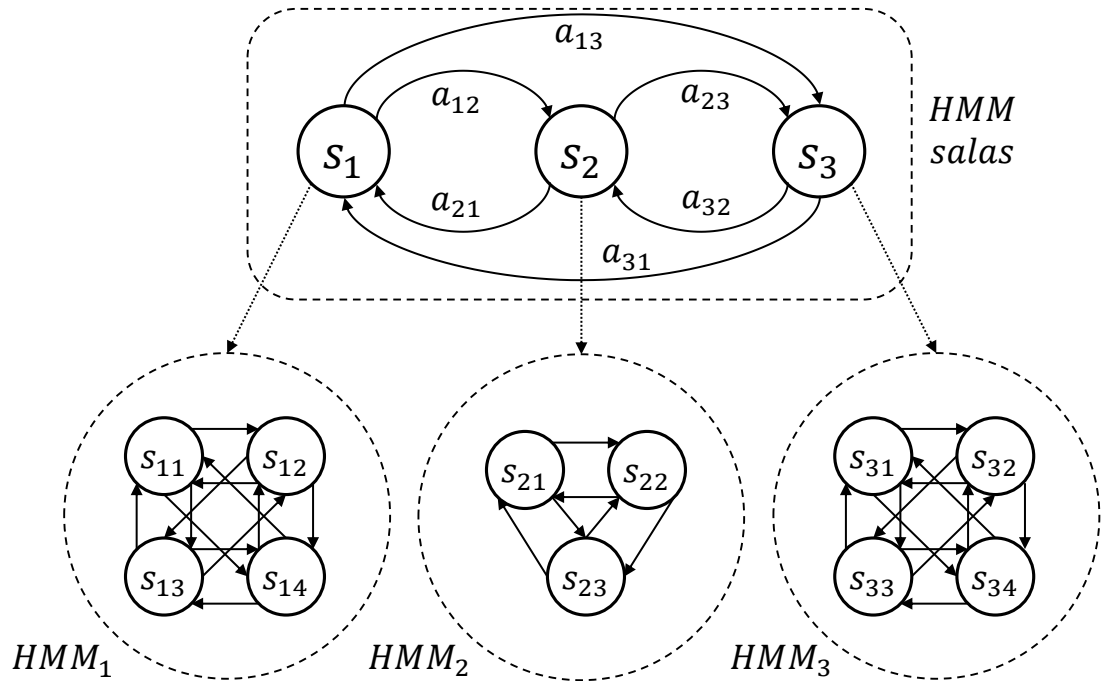


Figura 4.23: Modelo de HMMs jerárquico empleado en los despliegues del sistema de reconocimiento de actividades en hogares reales.

propagación de HMM HAR son: estado de las ADL's en el instante $t - 1$, señal proporcionada por sensor de evento y estado de sensores ambientales. Sin embargo, en la etapa de retro propagación, las entradas son las señales de los sensores de eventos en los instantes $t + 1$ y $t + 2$. Las entradas del algoritmo de propagación de HMM LOCALIZACIONES son: la información de las balizas BLE y los eventos de los PIR (todo ello acompañado de la posición del usuario en el instante $t - 1$). Todas las salidas se multiplican en el bloque de inferencia para obtener la ADL más probable en el instante t .

Para realizar los despliegues del sistema de reconocimiento de actividades en hogares reales, es necesario tener en cuenta que la topología de la vivienda y su número de estancias son parámetros muy variables. Además, los usuarios pueden rechazar el uso de las pulseras de actividad y sus hábitos de mantenimiento, como la necesidad de carga de los dispositivos. Estas desventajas pueden ocasionar que los despliegues de los sistemas de localización basados en balizas en las viviendas sean un problema. Por esta razón, se ha elaborado un modelo jerárquico basado en HMMs

con dos etapas secuenciales, tal y como se puede ver en la Figura 4.23. Estas etapas secuenciales son las siguientes:

- HMM salas: Este modelo se corresponde con la parte superior del diagrama. En este modelo, cada uno de los estados ocultos s_i se corresponde con las posibles salas que componen la vivienda. Para esta parte del modelo, las observaciones o_k se corresponden con las señales proporcionadas por los diferentes sensores de evento con los que cuenta la instalación (PIR, contacto y vibración). En este caso, no se tienen en cuenta los sensores ambientales, ya que no pueden desencadenar un cambio de estancia por ellos mismos. Por ejemplo, puede estar la televisión encendida, pero el usuario puede estar en cualquier parte de la casa. En cambio, si se detecta la apertura de una puerta, se sabe con seguridad que el usuario se encuentra en dicha estancia. Los parámetros a_{ij} se corresponden con los valores de las matrices de transición del modelo.
- HMM actividades: Este modelo se corresponde con la parte inferior del diagrama. Los estados ocultos s_{ii} se corresponden con las diferentes ADL's que el usuario puede realizar en la sala s_i en la que se encuentra. En este caso, las observaciones o_{ik} se corresponden con las señales proporcionadas por los sensores de eventos que se encuentran en esa estancia. A la hora de realizar la detección de actividad, las señales de los sensores ambientales se integran en el Algoritmo 4 del mismo modo que se explicó para el modelo HMM anterior.

Capítulo 5

Ecosistema de inteligencia ambiental para el reconocimiento de actividades de personas mayores

En el presente capítulo se aborda el ecosistema de inteligencia ambiental planteado para el cuidado de personas mayores en sus domicilios, así como el método de aplicación del sistema de reconocimiento de actividades desarrollado en la presente tesis. En primer lugar, se analizan distintas soluciones aplicadas al cuidado de personas mayores mediante robótica social (Sección 5.1). A continuación, en la Sección 5.2, se detalla la arquitectura elaborada para recoger, tratar y utilizar la información de la vivienda en la que reside el usuario. Por último, en la Sección 5.3, se exponen las diferentes herramientas y soluciones elaboradas para prestar servicios a los residentes.

5.1. Robótica social aplicada al cuidado de personas mayores

Integrar la robótica en el cuidado de personas mayores responde a una necesidad creciente en España y en gran parte de Europa. A medida que la población envejece, surgen desafíos significativos en la atención y el bienestar de las personas mayores. En primer lugar, el aumento de la longevidad conlleva una mayor incidencia de enfermedades crónicas y discapacidades relacionadas con la edad, lo que puede dificultar la realización de actividades diarias y aumentar la dependencia de la asistencia externa. La robótica ofrece soluciones prácticas para ayudar a las personas mayores a mantener su autonomía y calidad de vida, al proporcionar apoyo en tareas cotidianas, supervisar la salud y fomentar un estilo de vida activo [192]. Además, los cambios demográficos también están generando una creciente demanda de cuidadores, lo que puede llevar a una escasez de recursos humanos capacitados en el cuidado de personas mayores. Los robots pueden complementar el trabajo de los cuidadores humanos, brindando asistencia constante y reduciendo la carga de trabajo, lo que permite una distribución más eficiente de los recursos disponibles [193].

Otro aspecto importante es el bienestar emocional y social de las personas mayores. El aislamiento social y la soledad son problemas comunes entre esta población, especialmente en comunidades donde la movilidad puede estar limitada. Los robots sociales ofrecen compañía, interacción y entretenimiento, ayudando a mitigar la sensación de soledad y mejorar el bienestar emocional al mantener conexiones sociales significativas [194].

La aplicación de la robótica social en el cuidado de personas mayores abarca diversas áreas, cada una diseñada para abordar necesidades específicas y mejorar la calidad de vida de este grupo demográfico. Los robots de asistencia personal, como el reconocido robot Paro, ofrecen compañía terapéutica y ayuda en tareas diarias, que van desde recordatorios de medicamentos hasta asistencia en la movilidad [195]. Por otro lado, los robots de telepresencia permiten la interacción remota, facilitando la comunicación con familiares, amigos o personal médico, lo que ha demostrado reducir los sentimientos de soledad y mejorar el bienestar emocional [196]. El robot Pepper se ha utilizado en diversos estudios para comprobar su compatibilidad con las personas



Figura 5.1: Robot social Pepper interactuando con usuarios de la residencia Lacort en Viana de Cega (Valladolid).

mayores debido a su apariencia y funcionalidad [197]. En [7], los autores han probado la compatibilidad de este robot en un entorno real, concretamente en una residencia de mayores durante un largo periodo de tiempo, obteniendo como resultado la utilidad del robot para mantener activa las habilidades físicas y cognitivas de los usuarios de la residencia. Además, en este proyecto, los autores también proponen utilizar el robot para combatir la soledad, implementando funcionalidades centradas en el ocio y entretenimiento [198]. En la Figura 5.1 se pueden observar una serie de imágenes en las que el robot social Pepper interactúa con los usuarios de la residencia. Además, existen robots diseñados específicamente para el entrenamiento físico, guiando a las personas mayores a través de ejercicios y proporcionando retroalimentación sobre la técnica, promoviendo así un estilo de vida activo y saludable [199]. Los robots de compañía y entretenimiento ofrecen características como conversación, reproducción de música y juegos, contribuyendo a la reducción del aislamiento social y a la promoción del bienestar emocional [200]. Por otra parte, los robots de monitoreo de salud supervisan constantemente el bienestar físico, detectando caídas, midiendo signos vitales y proporcionando alertas tempranas en caso de emergencias médicas [201]. Finalmente, los robots de aprendizaje y adaptación utilizan algoritmos de inteligencia artificial para personalizar las interacciones y servicios de acuerdo con las necesidades y preferencias individuales de las personas mayores, contribuyendo así a una experiencia de cuidado más personalizada y efectiva [202].

Estos avances en la robótica social están transformando la provisión de cuidados para personas mayores, ofreciendo soluciones innovadoras para promover su bienestar físico, emocional y social [203]. Se ha demostrado que la interacción humano-robot puede impactar positivamente en la función cognitiva, el bienestar emocional y la

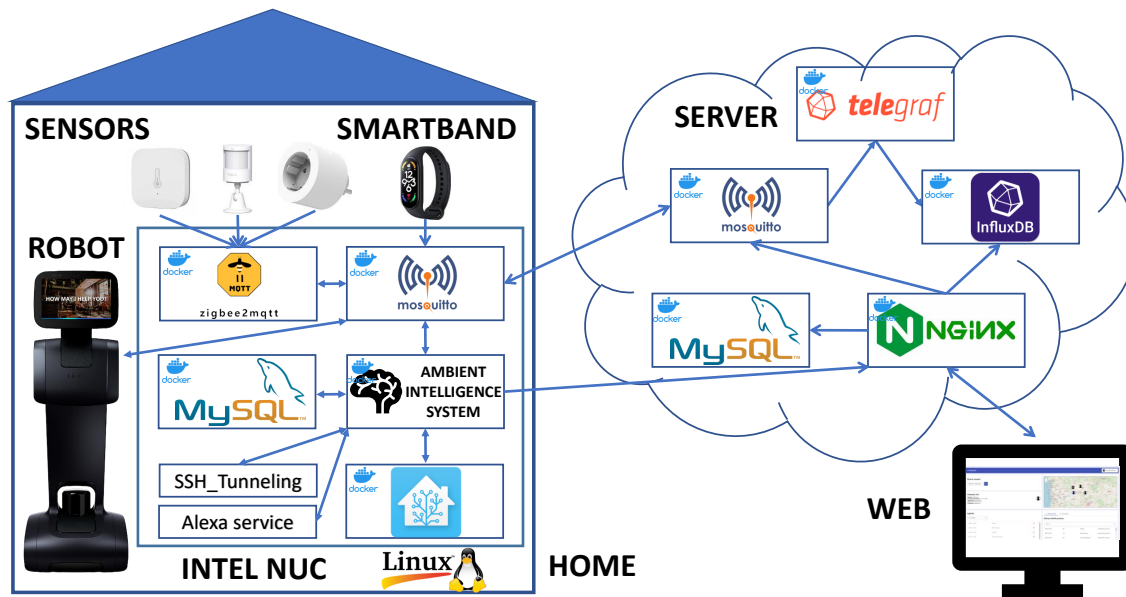


Figura 5.2: Diagrama de la arquitectura propuesta para el ecosistema de inteligencia ambiental para el cuidado de personas mayores.

calidad de vida general entre las personas mayores [204]. Al aprovechar los robots sociales junto con sistemas avanzados de reconocimiento de actividad, los cuidadores pueden proporcionar un cuidado más personalizado y adaptado a las necesidades y preferencias únicas de cada individuo [205].

5.2. Arquitectura propuesta para el ecosistema de inteligencia ambiental

Se ha propuesto una arquitectura genérica para desplegar el ecosistema de inteligencia ambiental en los domicilios de las personas mayores que viven solas. El diagrama de la arquitectura propuesta se puede comprobar en la Figura 5.2. A continuación, se van a detallar cada uno de los componentes de la arquitectura, así como su función dentro del sistema:

- Red de sensores no intrusivos: Dispositivos encargados de recopilar la información y señales necesarias para que el módulo HAR detecte con precisión las

ADL's realizadas por el usuario. Estos sensores se han analizado en el Capítulo 3.

- Pulseras de actividad: Este es el dispositivo encargado de recopilar información relacionada con el usuario. Este dispositivo permite el monitoreo en tiempo real de la salud del usuario, proporcionando detalles sobre patrones de sueño, conteo de pasos y parámetros médicos como la frecuencia cardíaca o los niveles de oxígeno en la sangre. Del mismo modo que ocurre con la red de sensores no intrusivos, estos dispositivos se han analizado en el Capítulo 3.
- Robot social: Se ha desarrollado una solución basada en un robot social para apoyar a los usuarios y mejorar su seguridad al recopilar información más allá de las capacidades de los sensores no intrusivos. Se han desarrollado las siguientes funcionalidades: entablar diálogos, hacer videollamadas con cuidadores y/o familiares, buscar al usuario por toda la casa en caso de emergencia, proporcionar juegos y ejercicios para la estimulación cognitiva y ofrecer información sobre noticias o el tiempo. Estas características tienen como objetivo aliviar la sensación de soledad del usuario. El robot cuenta con capacidades de navegación dentro de la vivienda, reconocimiento facial, seguimiento de usuarios y la capacidad de entablar conversaciones fluidas con el usuario. Se ha desarrollado una aplicación para el robot, que incorpora varios servicios como juegos de memoria y adivinanzas, ejercicios mentales y físicos, videoconferencias con cuidadores o familiares e interacción con ChatGPT proporcionando información adicional sobre cualquier tema que el usuario consulte. La aplicación cuenta con una interfaz amigable para el usuario con caracteres grandes diseñados para personas mayores. También incluye una cara virtual que interactúa con los usuarios respondiendo a una frase de despertar personalizada. Esta interfaz permite a los residentes interactuar con el robot tanto a través de la pantalla táctil como mediante comandos de voz.
- Concentrador de señales: El dispositivo “siguiente unidad de computación” (NUC, del inglés *Next Unit of Computing*) que supervisa los componentes del sistema de inteligencia ambiental es responsable de coordinar las comunicaciones entre varios elementos. No solo recopila señales de sensores, sino que también realiza

la detección de ADL y anomalías. Además, gestiona tareas como almacenar y actualizar configuraciones del sistema. Este componente está conectado a un servidor externo fuera de la residencia.

- **Servidor:** El servidor en la nube es responsable de almacenar datos históricos del sistema de inteligencia ambiental, incluyendo ADL's, anomalías y señales de sensores, entre otra información. Además, el servidor se comunica con el NUC para enviar comandos y recuperar información de usuario en tiempo real para verificación de cuidadores o familiares. El servidor también alberga una aplicación web para interactuar con los diferentes elementos de la solución.

Los componentes descritos son sistemas heterogéneos, y lograr una coordinación sin problemas entre ellos plantea una tarea desafiante. Todas las aplicaciones dentro del sistema desarrollado se han implementado como contenedores Docker interconectados e independientes. Dentro del NUC, se ha implementado un módulo Zigbee2MQTT para recopilar y analizar señales Zigbee de sensores no intrusivos, convirtiéndolas en mensajes MQTT a través de un dispositivo hardware ConBee II. Estos mensajes, junto con la información emitida por los smartbands, son recopilados por un broker Mosquitto. Este broker está conectado al broker Mosquitto del servidor a través de una comunicación de puente, replicando todas las señales de la casa dentro del servidor. El robot social se comunica con el sistema a través de mensajes MQTT bidireccionales. La información del sistema es procesada por el módulo de inteligencia ambiental, que consiste en el sistema HAR, detector de anomalías y módulo cognitivo responsable de ejecutar acciones, como dirigir al robot a áreas específicas o activar alarmas y notificaciones. El NUC también incorpora una base de datos MySQL para almacenar temporalmente información del sistema y su configuración. Se incluye Home Assistant para el manejo de los dispositivos, flujos de comunicación y automatizaciones locales. Además de los contenedores Docker, se ha empleado otro conjunto de servicios, como la integración con dispositivos Alexa y un servicio de Tunelización SSH . Para que los sistemas externos se comuniquen con el sistema local, sería necesario un puerto de acceso abierto en el router doméstico. Sin embargo, esto plantea un desafío de replicabilidad debido a las variaciones en los tipos de routers y las posibles dificultades para abrir puertos de inmediato, especialmente con proveedores de Internet que utilizan CG-NAT para compartir direcciones

IP entre múltiples clientes. El servicio de Tunelización SSH facilita la comunicación bidireccional desde el NUC local hasta el servidor. En consecuencia, solo es necesario abrir el puerto en el servidor (donde se tiene control total), lo que permite que cualquier dispositivo se conecte al servidor como pasarela. Este enfoque proporciona un medio seguro y controlado de interacción con el sistema. El servidor también incorpora contenedores Docker que alojan complementos como Telegraf. El agente Telegraf se suscribe a mensajes MQTT del broker Mosquitto, almacenando la información en una base de datos InfluxDB. Esta base de datos almacena registros históricos de la solución. Una base de datos MySQL es responsable de almacenar configuraciones y parámetros de la solución. Finalmente, el complemento NGINX aloja la aplicación web para interactuar con el sistema.

Existen arquitecturas que también están orientadas a la integración de ecosistemas de inteligencia ambientales. En [206], los autores proponen una arquitectura multi-agente aplicada a un sistema de inteligencia ambiental orientada a la combinación de servicios independientes. Una de las ventajas que ofrece esta arquitectura es que se puede comunicar de forma transparente entre diferentes equipos hardware y software, asegurando el envío y la recepción de los mensajes. Los autores han denominado a esta arquitectura como FUSION@ (Flexible and User Services Oriented Multi-agent Architecture). La arquitectura permite alojar servicios generados mediante diferentes lenguajes de programación, como C/C++ o JavaScript, y el protocolo de comunicación que utiliza de forma predeterminada es el SOAP (protocolo simple de acceso a objetos, del inglés *Simple Object Access Protocol*). Sin embargo, una de las debilidades que presenta la arquitectura es que no permite la integración del lenguaje Python, el cual se está extendiendo cada vez más desde los últimos años. Otra debilidad con la que cuenta la arquitectura es que no permite la recepción de mensajes MQTT, protocolo cada vez más utilizado para el envío de datos de sensores en el ámbito del IoT. En [207], los autores proponen otra arquitectura para implementar una red de sensores en la que un teléfono móvil actúa como un nodo central que hospeda las aplicaciones del sistema y se conecta con los sensores. Los autores han denominado a la arquitectura con el nombre de MIMOSA. Esta arquitectura permite la comunicación Bluetooth y Wibree con diferentes elementos hardware, y se comunica con un servidor alojado en la nube mediante protocolo GPRS (servi-

cio general de paquetes vía radio, del inglés *General Packet Radio Service*) para interactuar con servicios remotos. Sin embargo, una de las debilidades de esta arquitectura es que no permite la comunicación con los robots sociales actuales, ya que no todos cuentan con protocolo de comunicación Bluetooth. Además, este método de comunicación exige cercanía con el dispositivo, por lo que limita en gran medida su aplicación. Por último, en [208], los autores han generado una arquitectura conocida como HYDRA. Esta arquitectura permite el desarrollo de servicios genéricos independientes basados en protocolos abiertos. Del mismo modo que FUSION@, esta arquitectura implementa el protocolo de comunicación SOAP entre los diferentes servicios que componen la solución. La arquitectura HYDRA cuenta con las siguientes capas independientes: aplicación, sistema operativo, comunicación y aplicación. Permite la integración de señales Zigbee, Bluetooth y WiFi, lo que supone una ventaja para la integración de dispositivos IoT modernos, como los sensores empleados en la presente tesis. Además, los autores especifican que está especialmente diseñada para la integración en soluciones domóticas, o soluciones relacionadas con el ámbito de la salud y la agricultura. Sin embargo, la principal desventaja que presenta HYDRA es su integración en hogares que cuenten con CG-NAT, lo que dificulta su despliegue en multitud de hogares en la región española.

5.3. Soluciones elaboradas para la interacción con el usuario final

En la Figura 5.3 se puede visualizar los tres módulos que componen el sistema de inteligencia ambiental que aportan funcionalidad a la solución: el módulo HAR, el módulo de métricas y anomalías y el sistema cognitivo.

Se ha desarrollado un módulo para detectar métricas y anomalías basado en los eventos recopilados por la red de sensores no intrusivos y las ADL's detectadas por el sistema HAR. En situaciones de emergencia, como la necesidad de atención médica urgente, se activa la incidencia "Emergencia". Esta métrica monitoriza el tiempo transcurrido desde el último evento recogido por el ecosistema de inteligencia ambiental y la sala en la que se encuentra el usuario, verificando la situación con el robot y notificando a la familia o cuidador para que tomen medidas inmediatas.

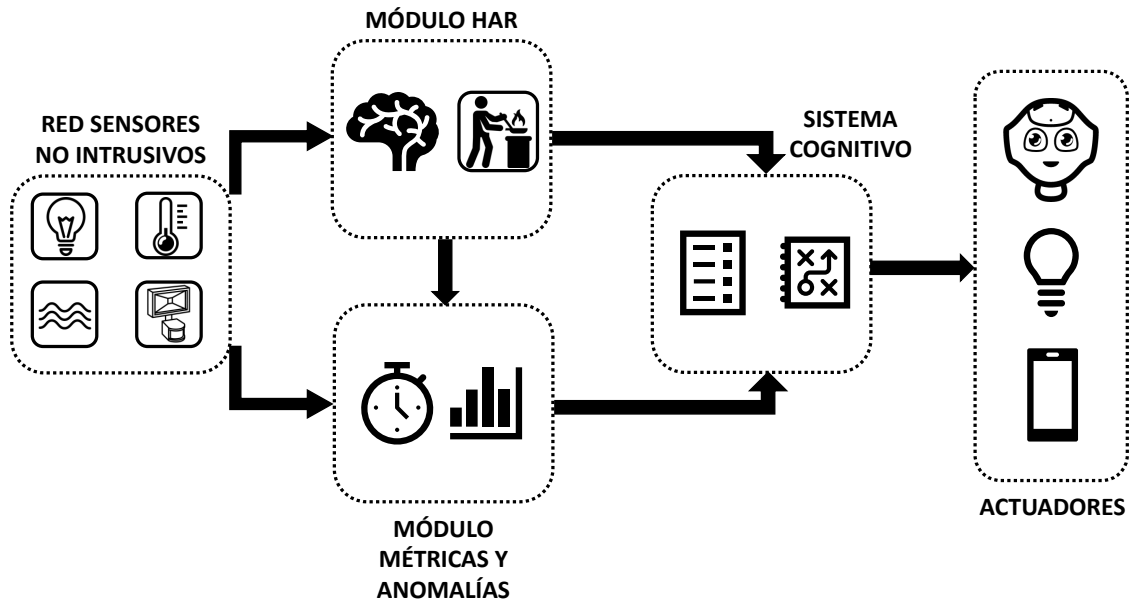


Figura 5.3: Componentes del sistema de inteligencia ambiental: módulo HAR, módulo de métricas y anomalías y sistema cognitivo.

Cuando el usuario sale de casa, se activa la incidencia “Fuera de Casa” y se registra la duración de su ausencia. Si esta salida ocurre durante la noche, por ejemplo, se comunica de inmediato a la familia o cuidador. La incidencia “Inactividad Prolongada” detecta períodos prolongados de inactividad del usuario junto con su última ubicación conocida. Tras detectarlo, el robot verifica la situación y alerta a la familia o cuidador, garantizando el bienestar del usuario y promoviendo la acción necesaria. Para el cumplimiento de las pautas de medicación, la incidencia “No Toma Medicación” registra el tiempo desde la última toma de medicinas. Recordatorios del robot o de Alexa instan al usuario a seguir su horario de medicación, asegurando así un tratamiento consistente. Cualquier deambulación nocturna anormal o interacciones con sensores de presencia activan la incidencia “Deambulación Nocturna”, que notifica de inmediato a la familia o cuidador para asegurar la seguridad y el bienestar del usuario. La televisión se monitoriza a través de la incidencia “Viendo TV”, que registra los tiempos de visualización durante el día. Después de períodos prolongados, el robot recuerda al usuario que se ponga de pie y se estire, promoviendo la actividad física. La incidencia “Sueño” monitoriza la duración del sueño, la hora de acostarse

y despertarse. Cada mañana, el robot saluda al usuario, registra la calidad del sueño y alerta a la familia o cuidador de cualquier anomalía. Finalmente, la incidencia “Comer” registra la frecuencia de las comidas, la duración y las horas de inicio. El robot promueve hábitos alimenticios saludables y las notificaciones aseguran que la familia o cuidador esté informado de cualquier preocupación nutricional. Todas las incidencias y métricas se encuentran recogidas en la Tabla 5.1.

Este módulo también recopila registros históricos de las ADL realizadas por el usuario, lo que permite establecer estadísticas basadas en el período de tiempo elegido. Para la familia/cuidador, es importante conocer los patrones de sueño del usuario a lo largo de la semana, cuántas veces ha comido en un día específico, o evaluar el número de interacciones con ejercicios propuestos por el robot durante un mes.

El sistema cognitivo se encarga de recoger las ADL detectadas por el módulo HAR y las métricas y anomalías descritas anteriormente. Con esta información, este módulo establece una serie de reglas para actuar con el ambiente. Entre las posibles acciones, se incluyen: interacciones con el robot social, intervenciones con el hogar (por ejemplo, encender o apagar luces, subir o bajar persianas, etc), y notificar a la aplicación web la información que el usuario solicite.

La arquitectura generada en la presente tesis doctoral está diseñada para generar alarmas e interacciones con el usuario a través del robot social. Para ello, se ha generado una aplicación en el robot que ofrece diferentes funcionalidades al usuario tal y como se puede ver en la Figura 5.4. Las funcionalidades que se han implementado en la aplicación son las siguientes:

- **Utilidades:** Permite al usuario realizar videollamadas a sus familiares y/o cuidadores, añadir eventos a su calendario personal con el fin de generar recordatorios personalizados y añadir artículos a su lista de la compra. La aplicación es capaz de comunicar los recordatorios y los artículos de la lista de la compra mediante correos o notificaciones via app.
- **Ocio:** Esta funcionalidad permite al usuario interactuar con juegos de orientación, cálculo, memoria o lenguaje para estimular sus habilidades cognitivas. También permite seleccionar diferentes tipos de música para su entretenimiento. Gracias a la aplicación web alojada en el servidor, los familiares y/o cuida-

Incidencia	Métricas	Acción
Emergencia	Tiempo, ubicación de la vivienda	Verificación del robot, notificación a la familia/cuidador
Fuera de casa	Hora de salida, período de tiempo fuera de casa	Notificación a la familia/cuidador si la hora de salida es de noche
Inactividad prolongada	Período de inactividad, última ubicación del usuario	Verificación del robot, notificación a la familia/cuidador
No tomar medicación	Tiempo desde la última toma	El robot o Alexa recuerda tomar la medicación
Deambulación nocturna	Número de deambulaciones nocturnas/interacciones con sensor de presencia	Notificación a la familia/cuidador
Ver televisión	Tiempo viendo televisión (mañana, tarde, noche)	El robot recuerda caminar para estirar las piernas después de un período de visualización
Sueño	Número de horas de sueño, hora de irse a dormir, hora de levantarse	El robot dice buenos días, registra la calidad del sueño y notificación a la familia/cuidador
Comer	Número de comidas, duración y hora de inicio de la comida	El robot recomienda una correcta alimentación y notifica a la familia/cuidador

Tabla 5.1: Resumen de incidencias, métricas y acciones asociadas del ecosistema de inteligencia ambiental.

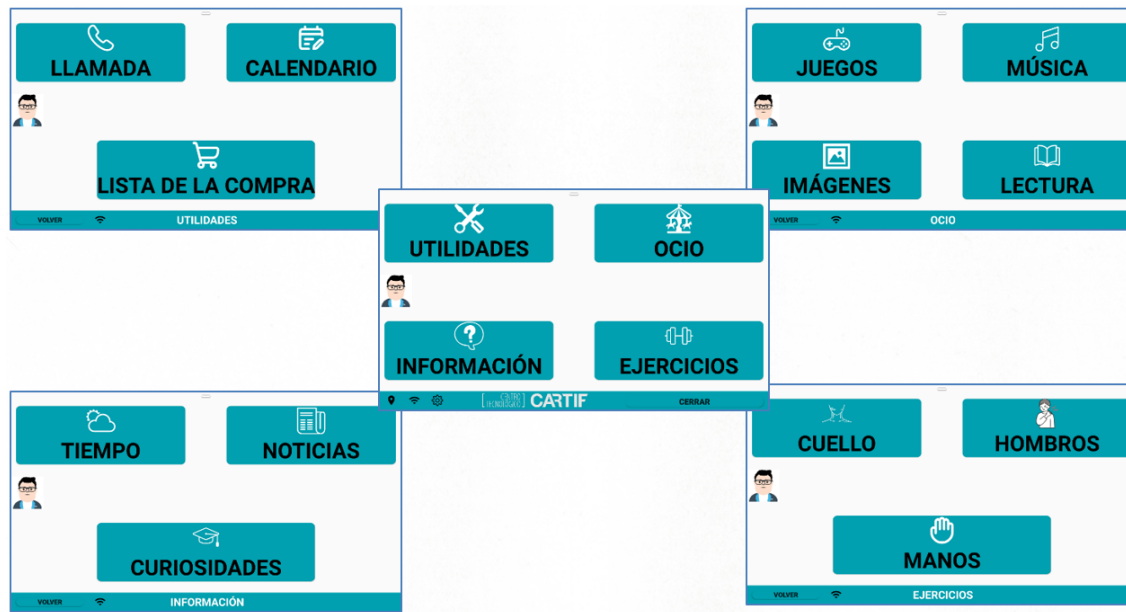


Figura 5.4: Funcionalidades implementadas en el robot social: información, utilidades, ocio y ejercicios.

dores pueden colgar imágenes junto con un texto explicativo para mostrar al usuario, y lecturas para su entretenimiento.

- Información: Permite al usuario preguntar por el tiempo, tanto actual como diferido, de la localización que quiera. Del mismo modo, puede preguntar por el tipo de noticias que quiera, ya sean locales, nacionales, de deportes, etc. Por último, el robot también ofrece curiosidades de interés para el usuario.
- Ejercicios: Esta funcionalidad reproduce vídeos en los que se pueden visualizar diferentes ejercicios de cuello, hombros y manos con el fin de que el usuario los repita. De esta manera, es posible mantener al usuario activo físicamente.

Todas las interacciones que se pueden hacer con el robot social se pueden realizar mediante comandos de voz o pulsando en la pantalla táctil, con el fin de mejorar la usabilidad en función de las capacidades específicas de los usuarios. La aplicación del robot social se ha realizado en Android, y es totalmente genérica para no depender de un modelo específico de robot. Además, también se permite la interacción con Alexa para poder ejecutar comandos de voz.

Capítulo 6

Experimentación y discusión

En este capítulo se detallan los distintos experimentos realizados para comprobar el rendimiento y eficacia de los sistemas de detección de actividades explicados en el Capítulo 4. En la Sección 6.1 se detallan los experimentos realizados mediante algoritmos de aprendizaje supervisado para realizar HAR mediante la aplicación de redes neuronales. En la Sección 6.2 se explican los experimentos realizados para comprobar la eficiencia de los algoritmos de aprendizaje no supervisado desarrollados en la presente tesis. En la Sección 6.3 se pueden observar los diferentes experimentos y resultados obtenidos durante las pruebas realizadas con el ecosistema de inteligencia ambiental y la incorporación de la robótica social. Por último, en la Sección 6.4 se comentan y se discuten los resultados obtenidos durante la realización de los experimentos.

6.1. Experimentos de métodos de aprendizaje supervisado

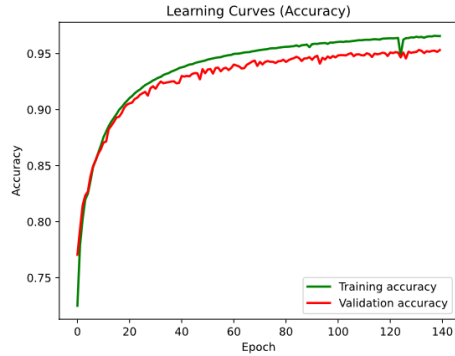
En esta Sección se van a explicar los experimentos realizados con los modelos basados en métodos de aprendizaje supervisado propuestos en las Secciones 4.3.1 a 4.3.4.

Para probar el funcionamiento del método de aprendizaje supervisado basado en redes neuronales LSTM en viviendas con un único usuario explicado en la Sección 4.3.1, se ha realizado un entrenamiento utilizando la siguiente disposición de los datos

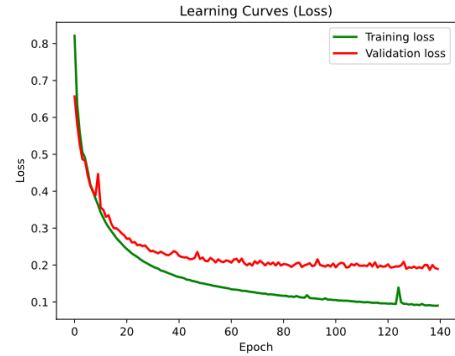
de Milan (CASAS): el 70 % del total de los datos se destina para entrenamiento, el 20 % para test y el 10 % para validación.

Primero, se van a detallar los valores de los parámetros que se han escogido para el modelo. Con respecto al tiempo de muestreo de la base de datos Milan, se ha escogido un total de 2 segundos, ya que según las pruebas realizadas, es el tiempo máximo suficiente para no perder ningún evento dentro de la vivienda. Para la ventana deslizante, se han utilizado un total de 60 registros para componer una única ventana. Este valor es adecuado ya que, teniendo en cuenta que el tiempo de muestreo es de 2 segundos, una ventana temporal contiene información correspondiente a 2 minutos de registros. Se entiende que este tiempo es suficiente para no saturar la red neuronal. Se ha realizado además un filtrado de las siguientes ADL's debido a su elevada duración: Otros y Dormir. Para el filtrado, se ha mantenido únicamente el 5 % correspondiente al inicio de la actividad y el 5 % del final. De esta manera, al realizar el filtrado, se ha reducido en gran medida su número de registros, permitiendo que el resto de ADL's se encuentren balanceadas en el modelo. Por último, para el módulo de agrupamiento y reorganización de filas, se han mantenido en lotes las parejas de dos ventanas deslizantes para mantener el orden secuencial de las actividades.

Con respecto a los hiperparámetros del modelo, se han utilizado un total de 64 celdas para las capas LSTM bidireccionales. Este número de celdas es suficiente para que el modelo proporcione buenos resultados y no provoque un entrenamiento demasiado lento. Se ha utilizado un valor de regularizador L2 muy pequeño ($1e-6$) ya que se recomienda utilizar un valor lo más cercano a cero posible para mejorar las prestaciones de las capas LSTM. Si se utilizase un valor demasiado alto, el regularizador tiene un efecto demasiado grande en el modelo y provoca que pierda convergencia. Se han escogido unos valores de dropout de 0.2 para la capa de dropout 1 y 0.4 para la capa de dropout 2. Son valores pequeños para que el modelo no se vuelva demasiado lento. Se ha comprobado de forma experimental que el modelo funciona mejor si las capas de dropout cercanas a los valores puros de entrada tienen un valor pequeño, mientras que cuanto más se acerquen a los valores de salida, mayor puede ser el porcentaje de apagado de neuronas. Por último, para las capas densas, se ha escogido una distribución de 2 capas densas de neuronas con distribución $(2/3)N$ y $(1/3)N$, siendo N el número total de neuronas (6600). Generalmente, este tipo de



(a) Precisión del modelo.



(b) Error del modelo.

Figura 6.1: Evolución de las curvas de aprendizaje del modelo de red neuronal basado en capas LSTM bidireccionales.

distribución provoca que el modelo sea más lento, pero ofrece mejores resultados en las épocas más avanzadas del entrenamiento.

El entrenamiento se ha realizado en un equipo Intel(R) Core(TM) i9-10900K CPU@3.70 GHz/128 Gb con dos GPUs RTX3090. El entrenamiento ha realizado un total de 140 épocas con un tamaño de lotes de datos de 256 y ha tenido una duración de 630 minutos.

Para obtener el rendimiento y la robustez del sistema, se han utilizado las siguientes métricas [209]:

$$Precision = \frac{TP}{TP + FP} \quad (6.1)$$

$$Recall = \frac{TP}{TP + FN} \quad (6.2)$$

$$F1Score = \frac{2}{\frac{1}{Recall} + \frac{1}{Precision}} \quad (6.3)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (6.4)$$

De las Ecuaciones 6.1-6.4 se puede observar el método de cálculo de las métricas que indican el rendimiento del modelo. TP se corresponde con los verdaderos positivos, FP son los falsos positivos. Del mismo modo, TN son los verdaderos negativos y FN los falsos negativos.

Actividad	Precisión	Recall	F1-score
Levantarse al baño	0.93	0.92	0.93
Tareas domésticas	0.95	0.97	0.96
Trabajar	0.98	0.98	0.98
Actividad comedor	0.97	0.96	0.96
Medicación nocturna	0.90	0.94	0.92
Uso baño invitados	0.93	0.97	0.95
Actividad cocina	0.97	0.97	0.97
Salir de casa	0.93	0.96	0.94
Uso baño principal	0.96	0.95	0.96
Meditar	0.95	0.99	0.97
Ver televisión	0.97	0.97	0.97
Dormir	0.90	0.93	0.91
Leer	0.97	0.97	0.97
Medicación matutina	0.92	0.91	0.91
Actividad dormitorio principal	0.96	0.94	0.95
Otros	0.90	0.88	0.89
Precisión total			0.95
Macro avg	0.94	0.95	0.95
Weighted avg	0.95	0.95	0.95

Tabla 6.1: Resultados del modelo supervisado en viviendas con un usuario.

En la Figura 6.1 se puede observar la evolución de las curvas de aprendizaje del modelo de red neuronal. La Figura 6.1a muestra el incremento de la precisión a medida que avanzan las épocas del entrenamiento, mientras que la Figura 6.1b muestra cómo disminuye el error. De estas gráficas se puede deducir que el error se estabiliza sobre la época 140. También se puede deducir que el modelo no tiene sobreentrenamiento debido a que las curvas de entrenamiento y validación están muy próximas en todas las épocas. Esto es beneficioso para el modelo, ya que significa que se comporta de manera correcta para datos que no conoce y que no hayan entrado durante la fase de entrenamiento.

En la Tabla 6.1 se pueden observar los valores de Precisión, Recall y F1-Score para cada una de las ADL's de la base de datos Milan. El resultado global para el test es del 0.9542 de precisión y un 0.184 para el error. Estos valores indican que el modelo es muy eficaz ante estas ADL's.

En la Figura 6.2 se puede observar la dispersión de las predicciones para cada

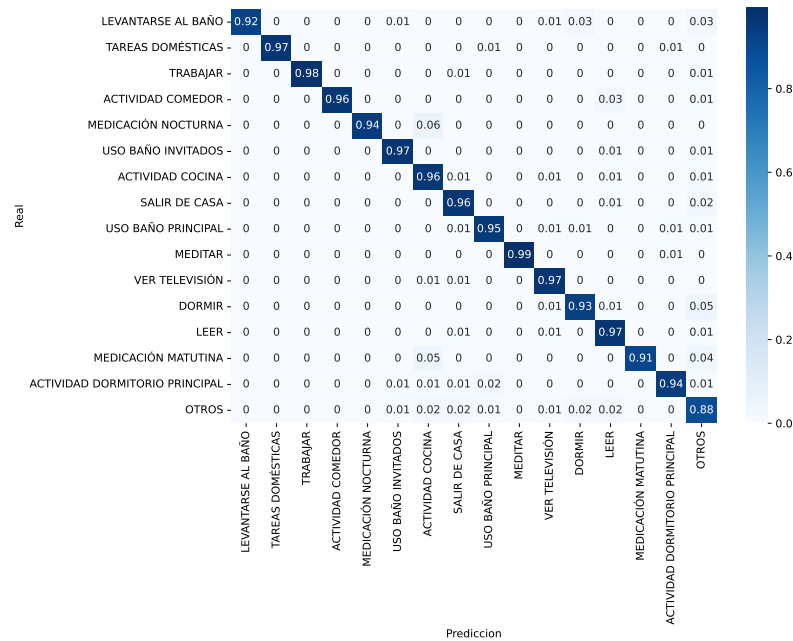


Figura 6.2: Matriz de confusión para el modelo de red neuronal LSTM en viviendas con un usuario.

una de las ADL's con respecto al resto de ADL's de la base de datos Milan. De esta matriz de confusión, se puede observar que todas las probabilidades más elevadas se encuentran distribuidas a lo largo de la diagonal de la matriz, lo que es un indicador de que el modelo es preciso y no genera muchas predicciones erróneas.

En un primer momento, se optó por utilizar capas LSTM simples en lugar de capas LSTM bidireccionales. Sin embargo, los resultados del modelo fueron peores ya que no se alcanzó ni el 85% de precisión. Esto se debe a que el modelo, al utilizar las capas bidireccionales, es capaz de aprender patrones de activación de los sensores en contra del flujo temporal. Esto supone un mayor abanico de posibles combinaciones de sensores y secuencias para realizar una misma ADL, lo que repercute en la elaboración de un conjunto de datos de entrenamiento más variado.

Para conseguir que el sistema tenga un rendimiento tan elevado y proporcione buenos resultados, se ha realizado un ajuste del modelo de forma progresiva y gradual. En primer lugar, el objetivo fue la reducción del sobreentrenamiento, ya que en las primeras etapas del modelo, la precisión del sistema para el conjunto de datos de entrenamiento era muy superior en comparación con el conjunto de datos de

validación. Para reducir este sobreentrenamiento, se probaron distintos valores para las capas de dropout. Por ejemplo, al probar con un dropout del 0.2, el sobreentrenamiento era demasiado grande. Sin embargo, para un valor de dropout de 0.6, el sistema ofrecía resultados similares que para un valor de 0.4 en términos de sobreentrenamiento, aunque se generaba un entrenamiento más lento. Por esta razón, se escogió que el valor óptimo era 0.4. Con respecto al tamaño de la ventana deslizante, se comenzó realizando pruebas con un tamaño de ventana menor (del orden de 30 muestras para analizar un total de 1 minuto de eventos). Sin embargo, para valores de tamaño de ventana pequeños, el modelo reconocía peor las actividades largas. Del mismo modo, si el tamaño de ventana era muy grande (120 muestras para analizar un total de 4 minutos), se reconocían peor las actividades cortas. Por todo esto, se llegó a la conclusión que el tamaño óptimo de ventana era de 60 muestras para que el reconocimiento estuviese balanceado. En cuanto al número de neuronas óptimo en las capas densas y las celdas en las capas LSTM, se comenzó probando con valores más pequeños. Para estos parámetros, los resultados eran mucho más bajos. A medida que se iban aumentando tanto las neuronas como las celdas, los resultados mejoraban. Se escogió utilizar 6600 neuronas y 64 celdas LSTM porque a partir de esos valores, el modelo no mejoraba más sus resultados y el entrenamiento se volvía demasiado lento.

Con el fin de comprobar que el valor global de 95.42% de precisión para esta base de datos (Milan) era un buen resultado, se ha realizado una búsqueda de trabajos similares que utilicen esta misma base de datos con métodos de aprendizaje supervisado. El trabajo realizado por D.Liciotti et al. en [210] utilizaba también la base de datos Milan utilizando redes neuronales basadas en capas LSTM bidireccionales. Sin embargo, los autores realizaron un entrenamiento utilizando un tamaño de ventana deslizante variable, coincidiendo con la duración de la ADL a detectar. Este método no es aplicable para un caso real, ya que si se está realizando un reconocimiento de ADL's en tiempo real, no se conoce la duración de la actividad que se está detectando. Por esta razón, el método desarrollado en la presente tesis utiliza un tiempo de muestreo fijo, con el fin de reproducir fielmente las condiciones de un uso real. Además, con respecto al valor de precisión global, los autores D.Liciotti et al. alcanzan un valor del 94.12%. Por lo tanto, el método de aprendizaje supervisado

desarrollado en esta tesis supone un incremento del rendimiento del 1.3 % con respecto a otros trabajos en las mismas condiciones, además de permitir su funcionamiento en tiempo real.

Para probar el funcionamiento del método de aprendizaje supervisado basado en redes neuronales LSTM en viviendas con varios usuarios explicado en la Sección 4.3.2, se ha realizado un entrenamiento utilizando la siguiente disposición de los datos SDHAR-HOME desarrollada en la presente tesis y explicada en el Capítulo 3: el 70 % del total de los datos se destina para entrenamiento, el 20 % para test y el 10 % para validación.

Del mismo modo que se ha realizado para la experimentación en viviendas con un sólo residente, se va a comenzar explicando en primer lugar los hiperparámetros escogidos para el modelo. El muestreo de la base de datos se ha realizado a un tiempo fijo de 2 segundos, siguiendo las conclusiones obtenidas de la primera fase de experimentación con la base de datos Milan. Del mismo modo ocurre con el tamaño de la ventana, ya que se han utilizado un total de 60 muestras para elaborar una ventana temporal. Es decir, cada ventana posee información sobre los últimos 2 minutos de sensores. Con respecto al balanceo de los datos de entrenamiento del modelo, se han filtrado las siguientes ADL's: ver la televisión, salir de casa, dormir y otros. Esto se debe a que se corresponden con las ADL's con un mayor número de muestras recogidas y, de esta manera, el entrenamiento se ejecuta de forma equilibrada para todas las posibles ADL's de la base de datos. Esta fase de experimentación, tal y como se ha indicado en el análisis del método propuesto de la Sección ??, va a tener en cuenta tres tipos de capas de análisis temporal para cada usuario: RNN, LSTM y GRU. Estas capas cuentan con un total de 64 celdas para ambos usuarios, y un valor de (1e-6) para el regularizador L2. En cuanto al número de neuronas de la capa densa, se ha observado que el número óptimo de neuronas para esta arquitectura de modelo es 8000. En cuanto a los valores de dropout se ha escogido un valor de 0.6 para las tres capas. Se necesita mucho dropout en ese modelo porque el sobreentrenamiento era demasiado grande.

El entrenamiento se ha realizado en un equipo Intel(R) Core(TM) i9-10900K CPU@3.70 GHz/128 Gb con dos GPUs RTX3090. Se ha escogido un tamaño de lotes de datos de 256. En cuanto al número de épocas, se ha aplicado la técnica de

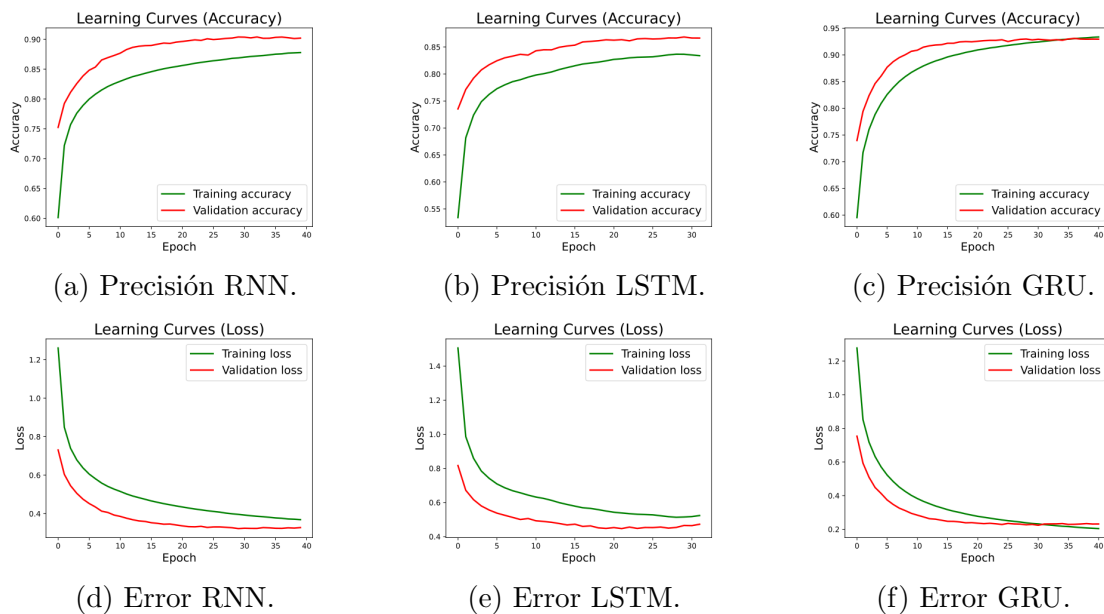


Figura 6.3: Evolución de las curvas de aprendizaje del usuario 1 mediante el método de aprendizaje supervisado para viviendas con varios residentes.

detención temprana. Esta técnica permite que el modelo se vaya entrenando de forma automática e iterativa hasta que el error en la evolución de la validación comienza a aumentar. Este hecho indica el momento de máximo rendimiento del sistema, ya que a partir de esas épocas, el modelo comienza a sobreentrenarse.

En la Figura 6.3 se pueden observar la evolución de las curvas de precisión y error del entrenamiento del modelo de red neuronal en función del tipo de capa para el usuario 1. El entrenamiento se ha realizado usando las capas RNN (a), LSTM (b) y GRU(c). Se pueden observar en las gráficas en número de épocas totales del entrenamiento. El entrenamiento de los modelos RNN, LSTM y GRU han tenido una duración de 40, 32 y 41 épocas respectivamente.

Del mismo modo, en la Figura 6.4 se puede ver la evolución de las curvas de aprendizaje durante el entrenamiento del modelo para el usuario 2. El entrenamiento se ha realizado utilizando capas RNN (a), LSTM (b) y GRU (c) manteniendo las mismas condiciones y parámetros para todos los entrenamientos de ambos usuarios. En este caso, las duraciones de los entrenamientos de los modelos RNN, LSTM y GRU han sido 132, 40 y 70 épocas respectivamente.

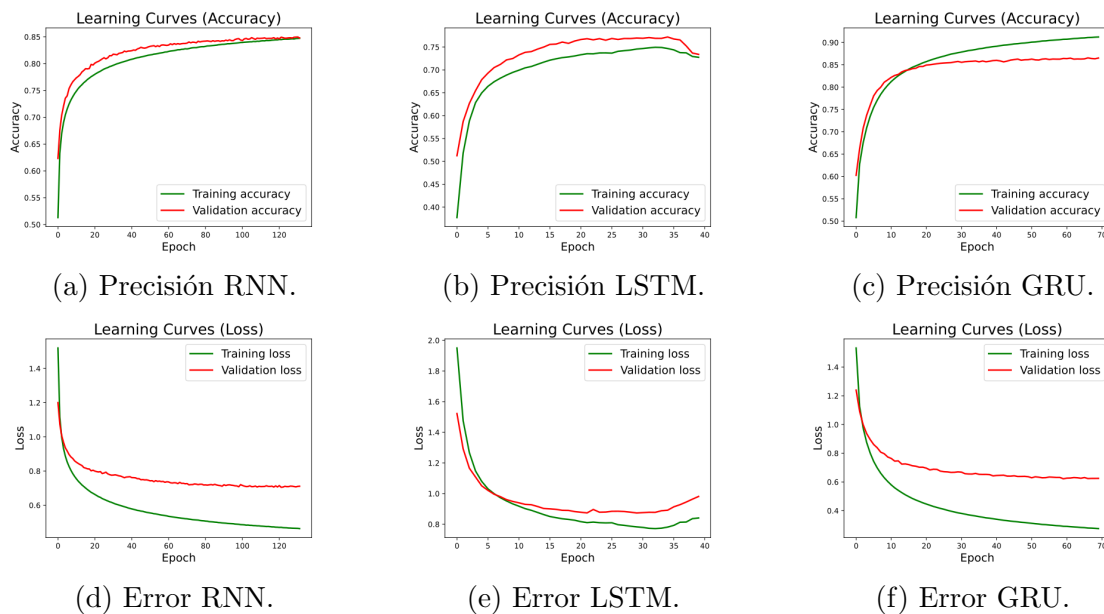


Figura 6.4: Evolución de las curvas de aprendizaje del usuario 2 mediante el metodo de aprendizaje supervisado para viviendas con varios residentes.

Para obtener los resultados globales del modelo, se ha implementado un algoritmo que considera como un resultado positivo si la ADL se detecta en un intervalo de tiempo anterior o posterior al momento de la predicción igual a 5 minutos. Esto se debe a que no debe considerarse como fallo que el modelo tarde unos pocos minutos en detectar una actividad, ya que es necesario esperar un poco a que la ventana deslizante se cargue con los eventos correspondientes a esa actividad. De esta manera, se puede despreciar y no tener en cuenta los periodos de transición entre ADL's consecutivas.

En la Tabla 6.2 aparece un resumen general de los resultados obtenidos durante el entrenamiento de los métodos de aprendizaje supervisado para viviendas con varios usuarios. En esta tabla aparecen los resultados separados para cada usuario y el tipo de modelo analizado. De la tabla se puede deducir que, para el usuario 1, el modelo ganador es la red GRU con un porcentaje de acierto del 90.91 %. En cambio, para el usuario 2, el modelo ganador es la red LSTM con un porcentaje de acierto del 88.29 %. De los resultados también se puede ver la diferencia entre el número de épocas necesarias para el entrenamiento del usuario 1 y 2. Los modelos del usuario 2

Usuario	Métricas	Modelo RNN	Modelo LSTM	Modelo GRU
Usuario 1	Precisión	89.59 %	89.63 %	90.91 %
	Épocas	40	32	41
	Duración entrenamiento	18 222 s	35 616 s	56 887 s
	Duración test	32 s	47 s	110 s
	Parámetros	1 515 410	2 508 050	2 177 554
Usuario 2	Precisión	86.26 %	88.29 %	86.21 %
	Épocas	132	40	70
	Duración entrenamiento	67 115 s	50 209 s	111 223 s
	Duración test	32 s	94 s	110 s
	Parámetros	1 515 410	2 508 050	2 177 554

Tabla 6.2: Resumen del entrenamiento de modelos de aprendizaje supervisado para viviendas con varios usuarios.

son mas lentos en alcanzar la convergencia. La diferencia de la precisión y rapidez del modelo entre ambos usuarios se puede deber a la precisión en el etiquetado durante la realización de la base de datos y la estabilidad del patrón de comportamiento de cada uno de los usuarios de la vivienda. Por todo esto, en los siguientes pasos de la experimentación, se va a hacer referencia a los modelos ganadores y se van a extraer los resultados para estos modelos. Es decir, el modelo GRU para el usuario 1 y el modelo LSTM para el usuario 2.

En la Tabla 6.3 se puede observar un resumen de todas las métricas obtenidas para cada actividad en función del usuario. El método de obtención de estas métricas se obtiene mediante la aplicación de las Ecuaciones 6.1-6.4. De esta tabla se puede deducir que para el usuario 1, las ADL's con mayor porcentaje de acierto son: vestirse, lavar la ropa, cuidados de mascota, leer, ver la televisión y trabajar. Se considera que son las que mejor porcentaje de acierto tienen porque alcanzar un valor del 100 % en la métrica de recall. Para el usuario 2, las ADL's con mejor resultado de recall son: tareas del hogar, cocinar, lavar los platos, vestirse, lavar la ropa, cuidados de mascota, ducharse y trabajar. Estas actividades dan como resultado de recall un 100 %, al igual que el usuario 1. Sin embargo, las ADL's con peor tasa de acierto para el usuario 1 es lavar los platos, con un 55 %. Para el usuario 2, las ADL's con

Actividad	Precisión	Recall	F1-Score
Actividad en el baño	0.87 - 0.82	0.92 - 0.99	0.89 - 0.90
Tareas del hogar	0.99 - 0.64	0.59 - 1.00	0.74 - 0.78
Cocinar	0.96 - 0.72	0.79 - 1.00	0.86 - 0.84
Lavar los platos	0.64 - 1.00	0.55 - 1.00	0.59 - 1.00
Vestirse	0.94 - 0.40	1.00 - 1.00	0.97 - 0.57
Comer	0.97 - 0.97	0.87 - 0.95	0.92 - 0.96
Lavar la ropa	1.00 - 0.49	1.00 - 1.00	1.00 - 0.66
Hacer comida simple	0.98 - 0.80	0.92 - 0.52	0.95 - 0.63
Salir de casa	0.94 - 1.00	0.96 - 0.91	0.95 - 0.95
Cuidados de mascota	0.97 - 0.88	1.00 - 1.00	0.98 - 0.93
Leer	0.91 - 0.58	1.00 - 0.54	0.95 - 0.56
Relajarse	0.42 - 0.82	0.91 - 0.84	0.58 - 0.83
Ducharse	0.85 - 0.98	0.86 - 1.00	0.85 - 0.99
Dormir	1.00 - 0.92	0.83 - 0.87	0.90 - 0.89
Tomar medicinas	1.00 - 0.76	0.93 - 0.94	0.97 - 0.84
Ver TV	0.94 - 0.69	1.00 - 0.94	0.97 - 0.80
Trabajar	0.99 - 0.95	1.00 - 1.00	1.00 - 0.97
Otros	0.53 - 0.70	0.93 - 0.75	0.68 - 0.72
Precisión total			0.91 - 0.88
Macro avg.	0.88 - 0.78	0.89 - 0.90	0.88 - 0.82
Weighted avg.	0.93 - 0.92	0.91 - 0.88	0.91 - 0.90

Tabla 6.3: Resultados del modelo supervisado en viviendas con varios usuarios (Usuario 1 - Usuario 2).

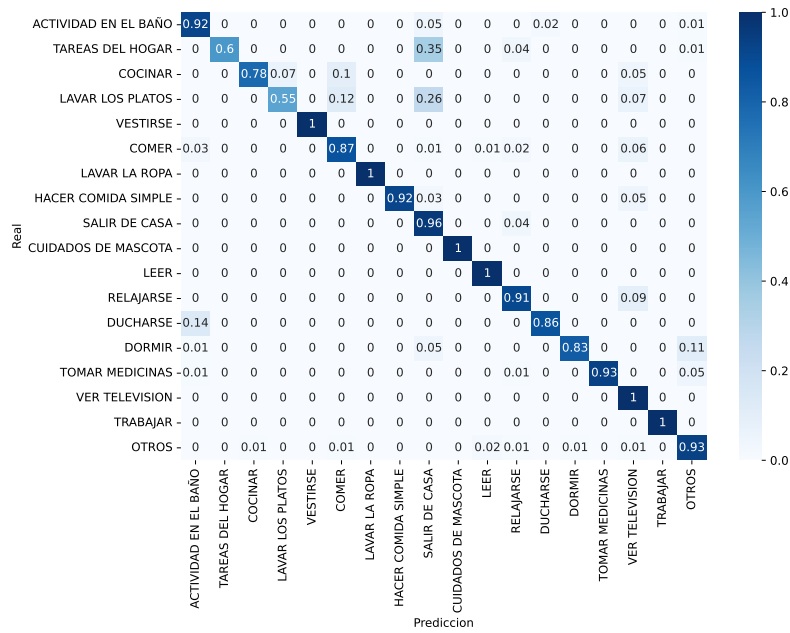


Figura 6.5: Matriz de confusión para el modelo de red neuronal GRU en viviendas con múltiples usuarios: usuario 1.

peor tasa de acierto es hacer comida simple, con un 52 %.

En la Figura 6.5 aparecen los resultados globales del método de aprendizaje supervisado para viviendas con múltiples residentes para el usuario 1 en formato de matriz de confusión. En esta tabla, los conjuntos de ADL's reales se distribuyen en filas, mientras que las predicciones de la red neuronal se distribuyen por columnas. Se puede observar que los valores con mayor peso se distribuyen en la diagonal de la matriz, lo que refleja un comportamiento preciso del modelo. En esta matriz se puede ver algún peso importante que se aleja de la diagonal. Por ejemplo puede ser el caso de la confusión entre las actividades: tareas del hogar y fuera de casa. Esto puede deberse a que el usuario 1 se quitaba la pulsera de actividad durante las tareas de limpieza de la casa para evitar daños en el dispositivo debidos a los productos de limpieza.

En la Figura 6.6 se pueden observar los resultados globales del método de aprendizaje supervisado para viviendas con múltiples residentes para el usuario 2 en formato de matriz de confusión. Del mismo modo que ocurre con el usuario 1, los porcentajes más elevados del modelo se distribuyen a lo largo de la diagonal de la matriz, lo

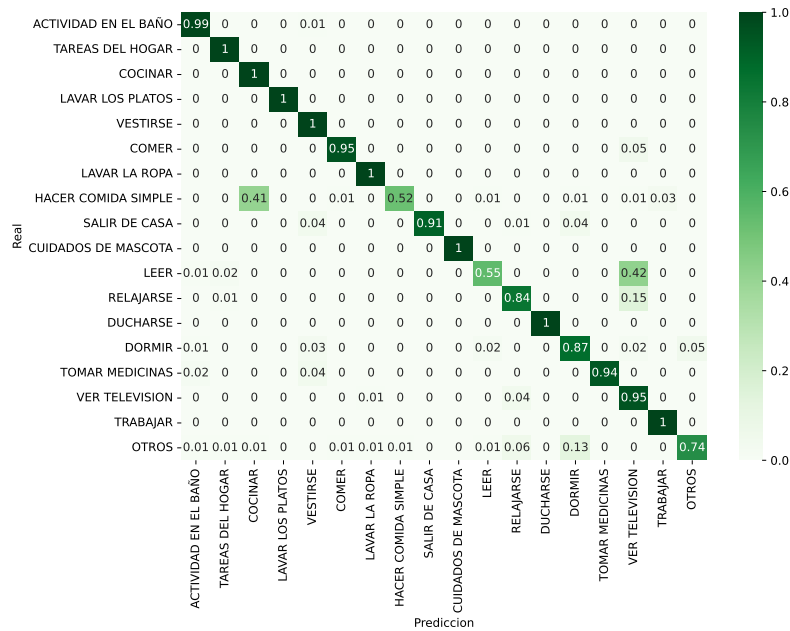
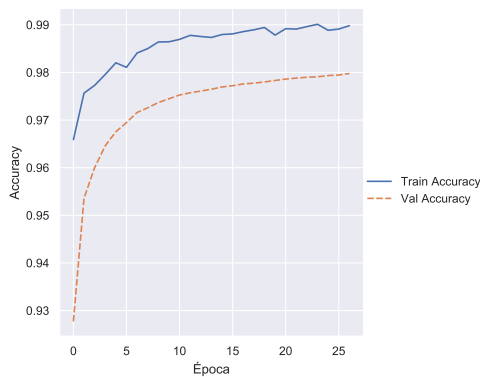


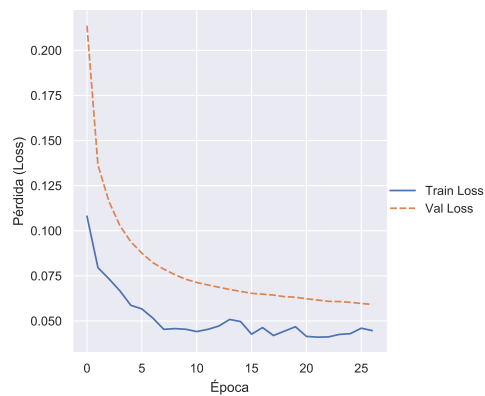
Figura 6.6: Matriz de confusión para el modelo de red neuronal LSTM en viviendas con múltiples usuarios: usuario 2.

que es un indicador de que el modelo es efectivo. Sin embargo, el sistema confunde algunas de las actividades. Por ejemplo, existe confusión entre leer y ver la televisión. Esto se debe a que el usuario 2 normalmente lee en el salón, estancia donde también se encuentra la televisión. Si el usuario 1 se encuentra viendo al mismo tiempo la televisión, el sistema puede confundirse. El modelo también confunde las actividades de cocinar y hacer comida simple. Esto se debe a que la naturaleza de las ADL's es muy similar y comparten muchas señales de los sensores, como puede ser el frigorífico o el microondas.

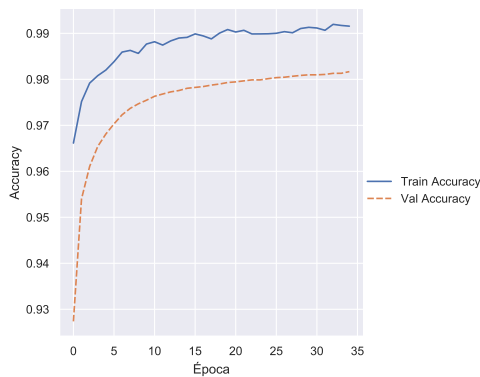
Se han realizado una secuencia de experimentos para comprobar el funcionamiento del modelo de red neuronal basado en CNN con CVV. Para ello, se ha utilizado la base de datos elaborada en la presente tesis (SDHAR-HOME). La base de datos SDHAR-HOME contiene datos de dos usuarios conviviendo de forma simultánea. Se ha utilizado la versión donde cada fila contiene una ventana de 60 instantes de tiempo, cada una con los valores de sus sensores y balizas, además de la hora. Cada fila incluye también la etiqueta de la ADL asociada a dicha secuencia temporal. En total, se cuenta con 2 033 753 filas de datos de entrenamiento, 225 973 filas de valida-



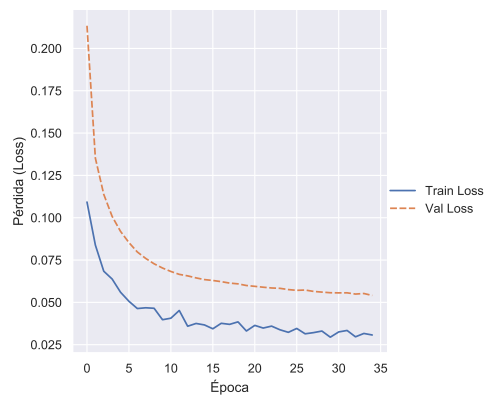
(a) Precisión $k=0$.



(b) Error $k=0$.



(c) Precisión $k=1$.



(d) Error $k=1$.

Figura 6.7: Entrenamientos de los dos primeros agrupamientos de k para CVV-SV utilizando *early-stopping* (Usuario 1).

ción y 518 400 filas de test. Como cada persona tiene sus hábitos, se han entrenado diferentes modelos para cada usuario. En ambos casos, se ha utilizado CVV con Soft Voting (SV) y $k = 5$. Para realizar el reparto de la nueva validación, se han unificado los datos de entrenamiento y validación y se han vuelto a repartir en k slots de acuerdo a CVV.

Se han realizado dos experimentos para evaluar el desempeño del modelo de convolución individual y su mejora mediante el método CVV. Para cada modelo de convolución, se ha empleado el optimizador Adam con un factor de aprendizaje de 0.001 y la función de pérdida de entropía cruzada categórica. En la Figura 6.7 se presentan las gráficas de entrenamiento de los dos primeros modelos del usuario 1 (se

Modelo	Test accuracy sin filtro (Usuario 1)	Test accuracy con filtro (Usuario 1)	Test accuracy sin filtro (Usuario 2)	Test accuracy con filtro (Usuario 2)
Convolutacional con filtro de (60,1) ($k = 0$)	0.8262	0.9579	0.6753	0.8914
Convolutacional con filtro de (60,1) ($k = 1$)	0.8537	0.9700	0.6646	0.8956
Convolutacional con filtro de (60,1) ($k = 2$)	0.8507	0.9653	0.6539	0.8930
Convolutacional con filtro de (60,1) ($k = 3$)	0.8367	0.9583	0.6813	0.9064
Convolutacional con filtro de (60,1) ($k = 4$)	0.8419	0.9601	0.6253	0.8701
Media Convolutacional con filtro de (60,1) (5 ejecuciones)	0.8418	0.9623	0.6601	0.8913
CVV-SV con $k = 5$ con filtros de (60,1)	0.8560	0.9576	0.6977	0.8701
CVV-SV con $k = 5$ con filtros de (60,1) (3 mejores modelos)	0.8607	0.9608	0.6848	0.8808

Tabla 6.4: Resultados obtenidos del entrenamiento del modelo basado en redes neuronales CNN con CVV.

han entrenado $k = 5$ modelos en total). Aunque las gráficas son bastante similares, cada modelo ha alcanzado su valor máximo de validación en épocas diferentes. En el caso del modelo CVV, el entrenamiento de los modelos se detiene mediante el early-stopping, utilizando un parámetro de paciencia establecido en 5. Este parámetro indica el número de épocas transcurridas sin reducción en la pérdida de validación. Los modelos CVV, tanto para el usuario 1 como para el usuario 2, han sido entrenados en un período de 70 minutos utilizando PyTorch y una GPU RTX3090.

Los resultados de la evaluación de los datos de prueba para cada uno de los usuarios se presentan en la Tabla 6.4. En esta Tabla, se muestran los resultados obtenidos mediante el modelo simple de convolución propuesto, que es equivalente a cualquiera de los modelos CVV individualmente, junto con los resultados obtenidos con CVV-SV. En los resultados del dataset SDHAR-HOME, se aplicó el filtro para determinar si una actividad detectada se había realizado dentro de un intervalo

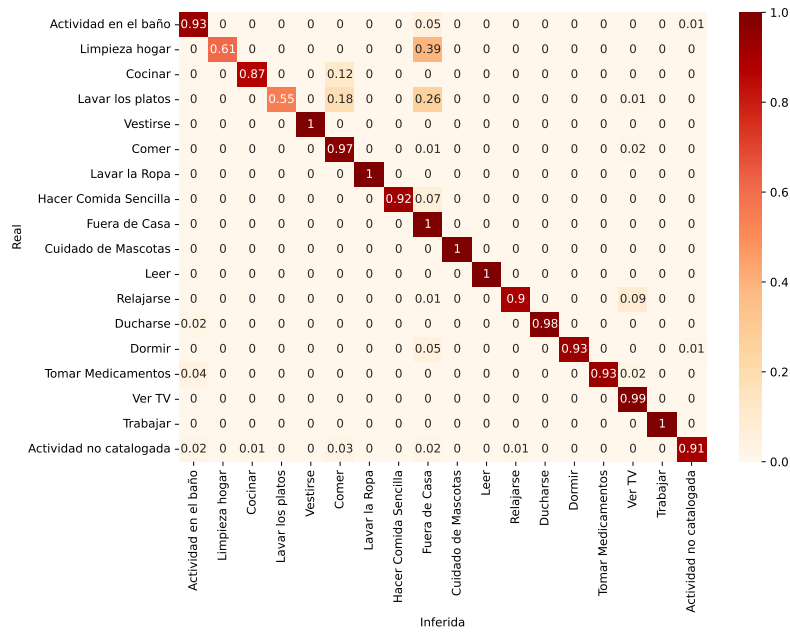


Figura 6.8: Matriz de confusión para el modelo CNN con CVV-SV en viviendas con múltiples usuarios: usuario 1.

temporal de 10 minutos. En los experimentos, se ha aplicado el mismo filtro para comparar los resultados en condiciones similares, aunque también se muestran las métricas de precisión sin el filtro. La tabla también presenta los resultados de los diversos modelos de convolución entrenados para cada usuario, así como un promedio de los resultados. A partir de estos datos, se puede observar cómo el modelo CVV-SV siempre mejora los resultados de los clasificadores en términos de precisión en el conjunto de prueba. La aplicación del filtro distorsiona ligeramente los resultados finales, y es evidente que algunos modelos individuales ofrecen incluso resultados bastante altos. Sin embargo, los modelos compuestos tienden a ser más estables en sus predicciones, ya que no dependen exclusivamente de un único clasificador. Los resultados del usuario 2 han sido ligeramente inferiores, posiblemente debido a la falta de información de las balizas en algunas de sus actividades. Esto puede haber sido causado por la carga insuficiente de la pulsera de actividad utilizada para recopilar dicha información o por su desactivación en ciertos momentos.

Las Figuras 6.8 y 6.9 muestran las matrices de confusión de las ADL's realizadas por el primer y el segundo usuario, respectivamente. La ADL no catalogada en el

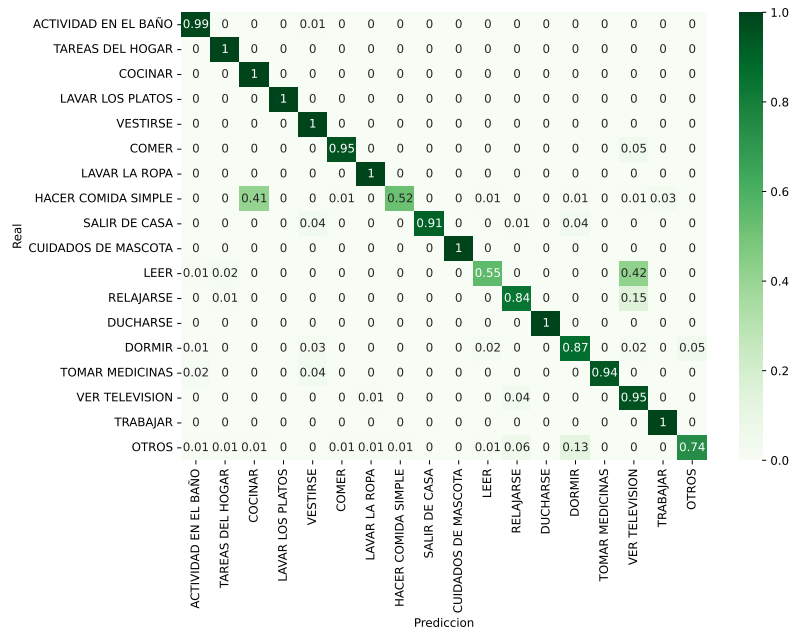


Figura 6.9: Matriz de confusión para el modelo CNN con CVV-SV en viviendas con múltiples usuarios: usuario 2.

dataset SDHAR-HOME corresponde a períodos en los que el usuario no ha realizado ninguna otra actividad. El modelo logra resolver correctamente el 96.08% de las detecciones del usuario 1 y el 88.08% del usuario 2. Algunos errores, como los ocurridos entre actividades en la misma ubicación, no representan un problema significativo, dado que estas actividades son de naturaleza similar. Por ejemplo, “Ducharse” y “Actividad en el baño” son actividades de higiene personal, mientras que “Lavar los platos” y “Comer” están relacionadas con la alimentación. Además, el modelo a veces confunde la “Limpieza en el hogar” con “Fuera de casa”, lo cual podría atribuirse a situaciones en las que el usuario no llevaba puesta la pulsera utilizada para registrar las ubicaciones de las balizas.

Para comprobar la validez y el rendimiento del modelo basado en redes neuronales transformer con capas de múltiples cabezas de atención, se han realizado 4 entrenamientos diferentes en paralelo con los siguientes tipos de mapeo: lineal, GRU bidireccional, LSTM bidireccional y CNN. Los modelos se han entrenado utilizando un optimizador de tipo Adam, cuya tasa de aprendizaje es igual a 0.001, con una función de pérdida de entropía cruzada categórica. Se ha utilizado además un me-

Modelo	Test accuracy sin filtro (Usuario 1)	Test accuracy con filtro (Usuario 1)	Test accuracy sin filtro (Usuario 2)	Test accuracy con filtro (Usuario 2)
Transformer con 3 capas y mapeo lineal	0.8323	0.9408	0.7128	0.8985
Transformer con 3 capas y mapeo GRU bidireccional (2 capas)	0.7865	0.9102	0.6891	0.8972
Transformer con 3 capas y mapeo LSTM bidireccional (2 capas)	0.7889	0.8965	0.7330	0.9155
Transformer con 3 capas y mapeo CNN (1 capa y 128 filtros)	0.7917	0.9104	0.6716	0.8696

Tabla 6.5: Resultados obtenidos del entrenamiento del modelo basado en redes neuronales transformer con 4 tipos de mapeo: lineal, GRU, LSTM y CNN.

canismo de “early stopping” con un periodo de 10 épocas. Por lo tanto, cuando el algoritmo detecta durante la fase de entrenamiento que la precisión del modelo no aumenta durante 10 épocas seguidas, el algoritmo detiene el entrenamiento y proporciona los resultados. Cada entrenamiento se ha completado en 2 horas, parando alrededor de la época 30. Todos los datos se han cargado en memoria en un equipo Intel(R) Core(TM) i9-10900K CPU@3.70 GHz/128 Gb con dos GPUs RTX3090.

En la Tabla 6.5 se pueden apreciar los resultados obtenidos durante el entrenamiento de los diferentes modelos transformers. Al igual que en los experimentos previos con la misma base de datos SDHAR-HOME, se ha implementado el algoritmo que considera como un resultado positivo si la ADL se detecta en un intervalo de tiempo anterior o posterior al momento de la predicción igual a 5 minutos. Este algoritmo actúa como filtro a la hora de determinar si la predicción es correcta. Se ha comprobado de forma experimental durante la realización de los entrenamientos que los mejores resultados se han obtenido al utilizar 3 capas en el modelo transformer. Para el caso del mapeo con GRU y LSTM bidireccional, se han empleado 2 capas

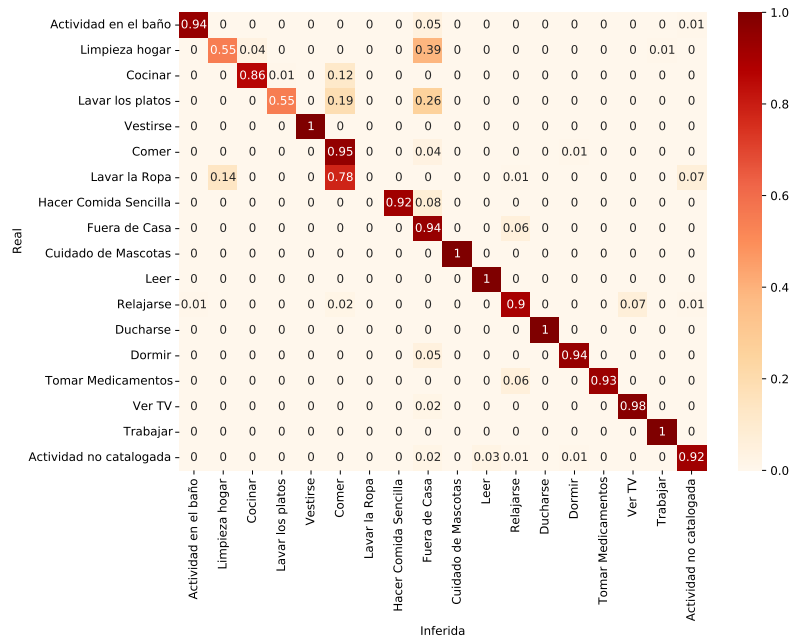


Figura 6.10: Matriz de confusión para el modelo transformer en viviendas con múltiples usuarios: usuario 1.

para ambos. En el caso del mapeo con CNN, se ha empleado 1 capa junto a 128 filtros de convolución.

Las Figuras 6.10 y 6.11 muestran las matrices de confusión de las actividades realizadas por la primera y segunda persona, evaluadas según el modelo transformer con 8 “heads” y mapeo lineal. La actividad no catalogada corresponde a períodos en los que el usuario no ha llevado a cabo ninguna otra actividad (es decir, la actividad “Otros”). El modelo con mapeo lineal logra una precisión del 94.08 % en las detecciones del usuario 1 y del 89.85 % en las del usuario 2. Para el usuario 2, el mapeo LSTM con transformador obtiene un 91.55 %. Cabe destacar que, en algunas ocasiones, el modelo no reconoce correctamente la actividad debido a que el usuario no llevaba puesta la pulsera utilizada para registrar su posición. Por otro lado, algunas actividades no se detectan correctamente en alguna de las personas, probablemente debido a que es una actividad desarrollada habitualmente por la otra persona (lavar la ropa, cuidado de la mascota, etc.).

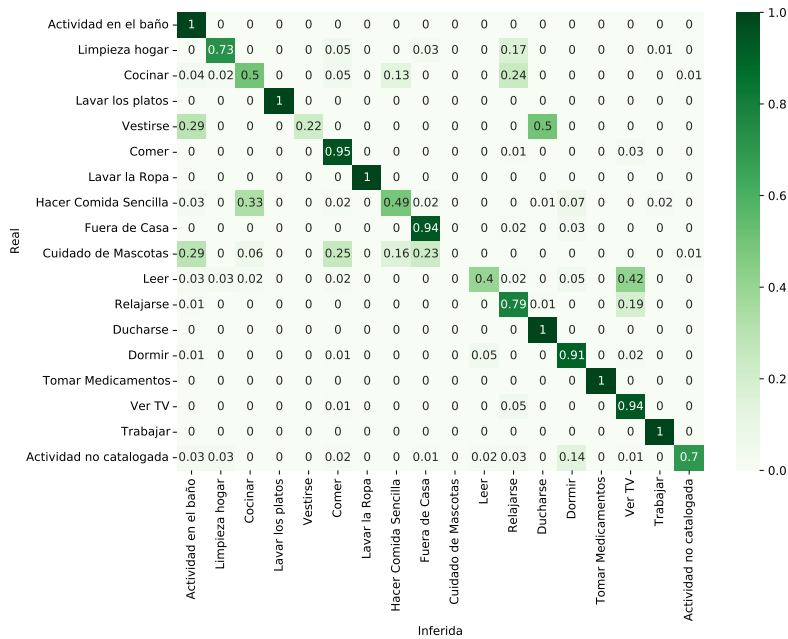


Figura 6.11: Matriz de confusión para el modelo transformer en viviendas con múltiples usuarios: usuario 2.

6.2. Experimentos de métodos de aprendizaje no supervisado

Con la finalidad de probar el funcionamiento del método de aprendizaje no supervisado basado en HMM explicado en la Sección 4.3.5, se ha utilizado la información de la vivienda de múltiples usuarios de la base de datos SDHAR-HOME, desarrollada en la presente tesis.

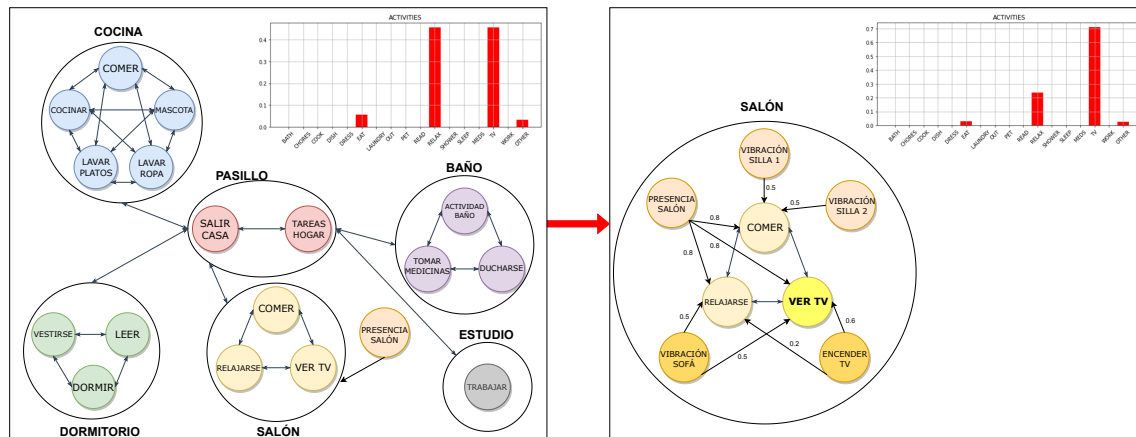
El método propuesto es capaz de generar una predicción de ADL cada vez que recibe una señal de un sensor de eventos. Sin embargo, la frecuencia de etiquetado de las ADL's reales es completamente diferente a la frecuencia de recogida de las señales de los sensores, ya que su naturaleza es totalmente asíncrona. Por esta razón, se ha desarrollado un algoritmo de muestreo para poder realizar una comparación segundo a segundo. Del mismo modo que se ha comentado en la sección anterior, se considera que la predicción es correcta si la ADL real se detecta en un intervalo de tiempo anterior o posterior al momento de la predicción igual a 5 minutos. Este

intervalo de tiempo es admisible para hacer frente a las transiciones entre actividades o a fallos propios del etiquetado, ya que a la hora de realizar la base de datos pueden existir desfases entre el etiquetado y el inicio de la actividad. Por ejemplo, existen ADL's con cierta inercia, como puede ser el caso de ducharse. La actividad se detecta gracias al incremento de humedad en el cuarto de baño. Sin embargo, el usuario ha comenzado a ducharse un rango de tiempo antes de que el sensor sea capaz de detectar la variación de humedad. Con la actividad de cocinar ocurre lo mismo. Esta ADL se detecta mediante el sensor de temperatura y humedad situado al lado de la vitrocerámica. Tiene que transcurrir cierto tiempo hasta que el sensor detecte la variación de temperatura y humedad de la estancia. Las actividades de cocinar y hacer comida simple se han agrupado para este método, ya que los sensores encargados de la detección son los mismos y no se pueden diferenciar mediante este método.

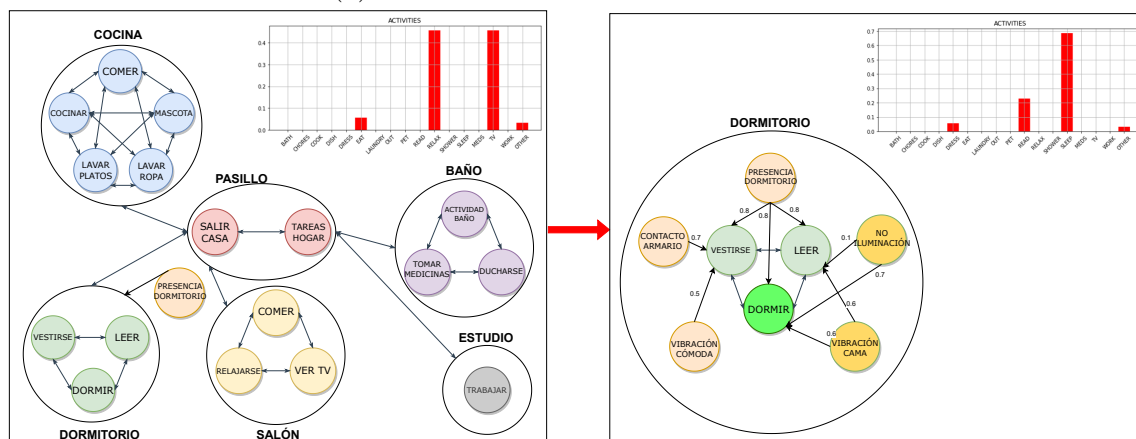
Todos los experimentos de este método se han realizado en un Intel(R) Core(TM) i7-10875H CPU@2.30 GHz/16 Gb con una GPU NVIDIA GeForce RTX2060. El tiempo total de procesamiento de la base de datos SDHAR-HOME completa ha sido de 70 segundos.

En la Figura 6.12 se pueden observar dos casos de uso en los que el sistema basado en métodos de aprendizaje no supervisado detecta 2 ADL's en función de la secuencia de señales recibidas por los sensores. Por ejemplo, en la Figura 6.12a se puede comprobar cómo se detecta la actividad de ver la televisión. En primer lugar, se detecta presencia en el salón mediante un sensor PIR. En ese momento, todas las ADL's posibles del salón igualan su probabilidad. En el instante posterior, se detecta vibración en el sofá y que la televisión está encendida. Gracias a estas dos señales, el sistema calcula las probabilidades y otorga la máxima probabilidad a la actividad de ver la televisión. En el caso de la Figura 6.12b, el caso de uso es la detección de la actividad de dormir. En primer lugar, el sistema detecta presencia en el dormitorio gracias al sensor PIR, por lo que todas las probabilidades de las actividades del dormitorio se igualan. En el instante posterior, se detecta vibración en la cama y que la luz está apagada. Por lo tanto, el sistema otorga la probabilidad más alta a la actividad de dormir.

En la Tabla 6.6 aparece un resumen de las duraciones medias de cada una de



(a) Proceso de detección ADL ver TV.



(b) Proceso de detección ADL dormir.

Figura 6.12: Ejemplos de detección de actividad mediante la implementación de métodos de aprendizaje no supervisado basado en las señales de los sensores.

	Act. baño	Tareas del hogar	Cocina	Lavar platos	Vestir.	Comer	Lavar ropa	Salir de casa
Usuario 1	505	1594	630	723	268	1649	224	24003
Usuario 2	914	1887	604	631	607	2213	832	27261
	Mascota	Leer	Relax	Ducha	Dormir	Medic.	Ver TV	Trabajo
Usuario 1	146	1997	1960	634	25268	103	4319	5202
Usuario 2	304	8924	3788	1602	28907	60	4262	4244

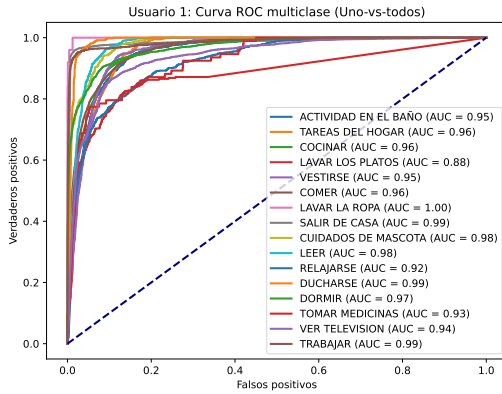
Tabla 6.6: Duración media de cada una de las ADL's de la base de datos SDHAR-HOME para cada usuario.

las ADL's de la base de datos SDHAR-HOME para cada uno de los usuarios de la vivienda en segundos (s). Se puede deducir de esta tabla que existe mucha variabilidad en las duraciones de cada una de las actividades. La etapa de la propagación del método es inmediata y funciona en tiempo real. Sin embargo, para ejecutar la etapa de retro propagación, es necesario contar con las señales posteriores al momento de la predicción para poder inferir la ADL. Debido a esta etapa de retro propagación, se ha calculado que el tiempo de respuesta medio del modelo se encuentra alrededor de los 6 minutos.

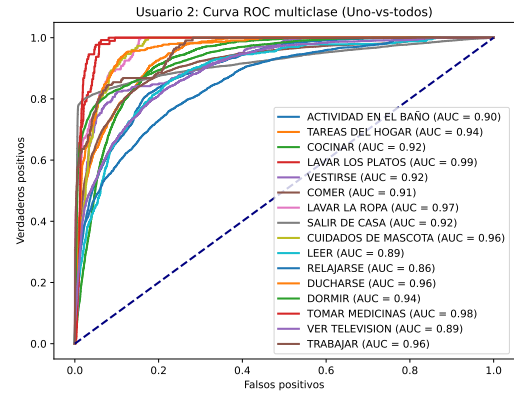
En la Tabla 6.7 se pueden observar los resultados obtenidos mediante la aplicación de los modelos no supervisados utilizando la base de datos SDHAR-HOME. En esta tabla aparecen las métricas de Precisión, Recall y F1-Score calculadas mediante la aplicación de las Ecuaciones 6.1-6.4. De esta tabla se puede deducir que la precisión del modelo para el usuario 1 es del 91.68%, mientras que para el usuario 2 es del 86.78%. La diferencia entre el rendimiento del modelo para cada uno de los usuarios puede darse debido a posibles fallos durante la etapa de etiquetado de ADL's a la hora de elaborar la base de datos. Los fallos también pueden deberse al ajuste de las matrices internas del modelo (A , B , C , M o R). Este ajuste se ha realizado de forma manual mediante el conocimiento experto. Por ejemplo, se sabe que el sensor de contacto situado en el cajón de las medicinas va a influir mucho en la ADL de

Actividad	Precisión	Recall	F1-score
Actividad en el baño	0.81 - 0.63	0.77 - 0.6	0.79 - 0.61
Tareas del hogar	0.28 - 0.25	0.65 - 0.57	0.39 - 0.35
Cocinar	0.68 - 0.37	0.83 - 0.6	0.75 - 0.46
Lavar los platos	0.24 - 0.15	0.71 - 0.79	0.36 - 0.25
Vestirse	0.12 - 0.19	0.78 - 0.61	0.21 - 0.29
Comer	0.91 - 0.65	0.82 - 0.64	0.86 - 0.64
Lavar la ropa	0.85 - 0.11	1.0 - 0.54	0.92 - 0.18
Salir de casa	0.99 - 0.99	0.97 - 0.91	0.98 - 0.95
Cuidados de mascota	0.82 - 0.11	0.86 - 0.55	0.84 - 0.18
Leer	0.64 - 0.58	0.54 - 0.67	0.59 - 0.62
Relajarse	0.42 - 0.76	0.69 - 0.63	0.52 - 0.69
Ducharse	0.87 - 0.47	0.88 - 0.73	0.87 - 0.57
Dormir	0.96 - 0.94	0.92 - 0.94	0.94 - 0.94
Tomar medicinas	0.09 - 0.09	0.83 - 0.96	0.16 - 0.16
Ver TV	0.89 - 0.79	0.82 - 0.8	0.85 - 0.79
Trabajar	0.99 - 0.57	0.94 - 0.77	0.96 - 0.66
Precisión total			0.92 - 0.87
Macro avg.	0.66 - 0.48	0.81 - 0.71	0.69 - 0.52
Weighted avg.	0.94 - 0.9	0.92 - 0.87	0.93 - 0.88

Tabla 6.7: Resultados del modelo no supervisado en viviendas con varios usuarios (Usuario 1 - Usuario 2).



(a) ROC y AUC para usuario 1.



(b) ROC y AUC para usuario 2.

Figura 6.13: Curvas ROC uno-vs-todos obtenidas de la aplicación de los métodos de aprendizaje no supervisado para la base de datos SDHAR-HOME.

toma de medicinas aumentando su probabilidad en la matriz B . Del mismo modo, se sabe que el sensor de contacto del frigorífico afecta mucho a la ADL de cocinar. Además, es posible que ambos usuarios se encuentren en la misma sala realizando actividades distintas. Por ejemplo, pueden encontrarse ambos en el salón, uno de ellos viendo la televisión y el otro usuario comiendo.

En la Figura 6.13 se pueden ver las curvas de característica operativa del receptor (ROC, del inglés *Receiver Operating Characteristic*) [211]. Estas curvas representan la sensibilidad del sistema ante falsas detecciones. La forma de las curvas ROC para ambos usuarios es adecuada, ya que indica que la tasa de falsos positivos es pequeña debido a que todas ellas se alejan de la diagonal. De forma adicional se indican los valores del área bajo las curvas ROC (AUC, del inglés *Area Under the Curve*) para cuantificar la capacidad de clasificación del sistema para cada ADL [212]. Se puede observar de esta figura que la capacidad de clasificación para todas las ADL's se encuentra por encima del 90 %.

La matriz de confusión para el usuario 1 se puede observar en la Figura 6.14. Todas las ADL's se distribuyen a lo largo de la diagonal de la matriz, lo que indica que el modelo es preciso. En cambio, hay parejas de ADL's que el sistema tiende a confundir. Un ejemplo es la actividad de dormir con leer. Esto se debe a que el usuario 1 normalmente lee en la cama, por lo que el modelo puede confundirse. Otra pareja que puede confundir es lavar los platos con comer. Puede deberse a que a

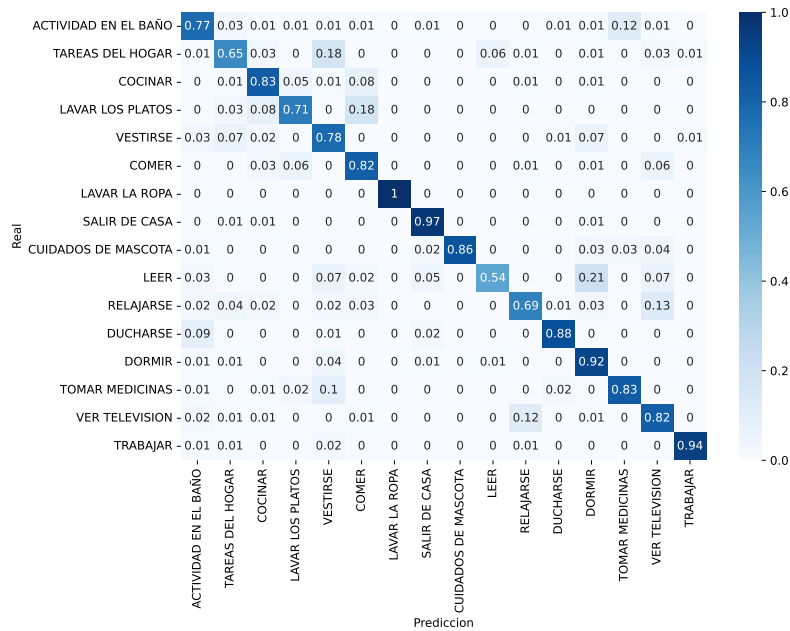


Figura 6.14: Matriz de confusión para el modelo de aprendizaje no supervisado: usuario 1.

veces el lavavajillas se queda abierto mientras el usuario come, lo que puede producir fallos en el sistema.

En la Figura 6.15 se puede observar la matriz de confusión para el usuario 2. También cuenta con todas las probabilidades más elevadas distribuidas a lo largo de la diagonal. En este caso, la actividad de dormir se confunde con cuidados de mascota. Esto se debe a que, normalmente, estos usuarios realizan la actividad de cuidados de mascota durante la noche. Si el sistema de localización falla, puede intercambiar las ADL's reconocidas entre los usuarios y provocar que el sistema falle.

6.3. Experimentos y resultados del ecosistema de inteligencia ambiental

Para evaluar la funcionalidad y la validez del ecosistema de inteligencia ambiental planteado en el Capítulo 5, se ha realizado un despliegue en una vivienda donde residen dos personas con su mascota. Se ha utilizado un robot social Temi V3 [213]

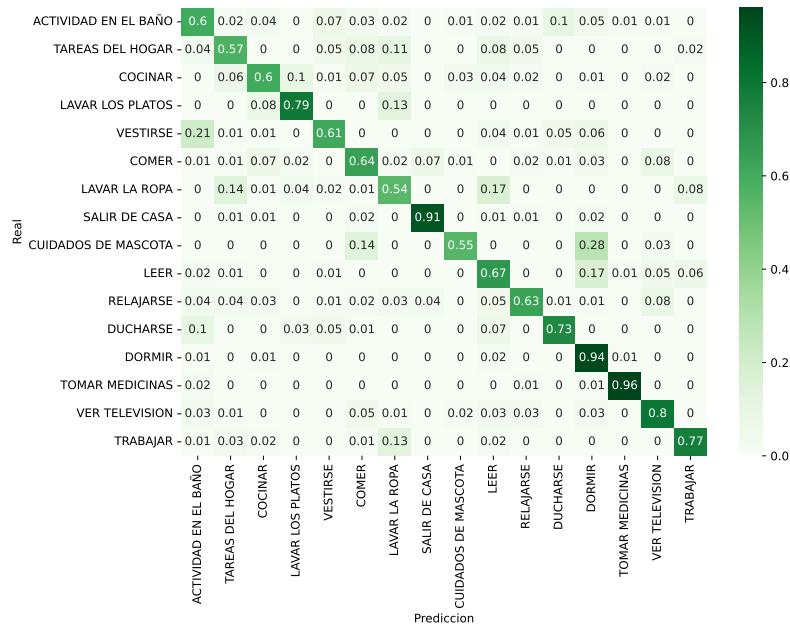


Figura 6.15: Matriz de confusión para el modelo de aprendizaje no supervisado: usuario 2.

para apoyar a los usuarios durante un total de 2 meses de duración. A continuación se van a exponer las tareas que el robot debe realizar en el día a día. El robot tiene la tarea de recordar al usuario que tome medicamentos durante el desayuno, el almuerzo y la cena. Por la mañana, el robot verifica el bienestar del usuario, proporciona actualizaciones meteorológicas, destaca eventos diarios y comparte las últimas noticias si así se desea. A las 12:00, el robot sugiere participar en juegos de memoria y ejercicios físicos centrados en movimientos de cuello, hombros y manos. Los usuarios pueden acceder a los servicios del robot, incluidos juegos, videollamadas y diálogos, en cualquier momento. Si el usuario ve la televisión continuamente durante más de una hora, el robot le anima a dar un paseo corto. Si el usuario sale de casa entre las 20:00 y las 8:00, el robot verifica las estancias y notifica al cuidador y/o familiar si no hay nadie presente. Presionar el botón de emergencia incita al robot a explorar la casa, comenzando desde la última actividad detectada, buscando al usuario y evaluando su bienestar. Lo mismo ocurre si se detecta la métrica de inactividad durante un periodo prolongado.

Se han propuesto los siguientes planes para el robot social: plan de emergencia,

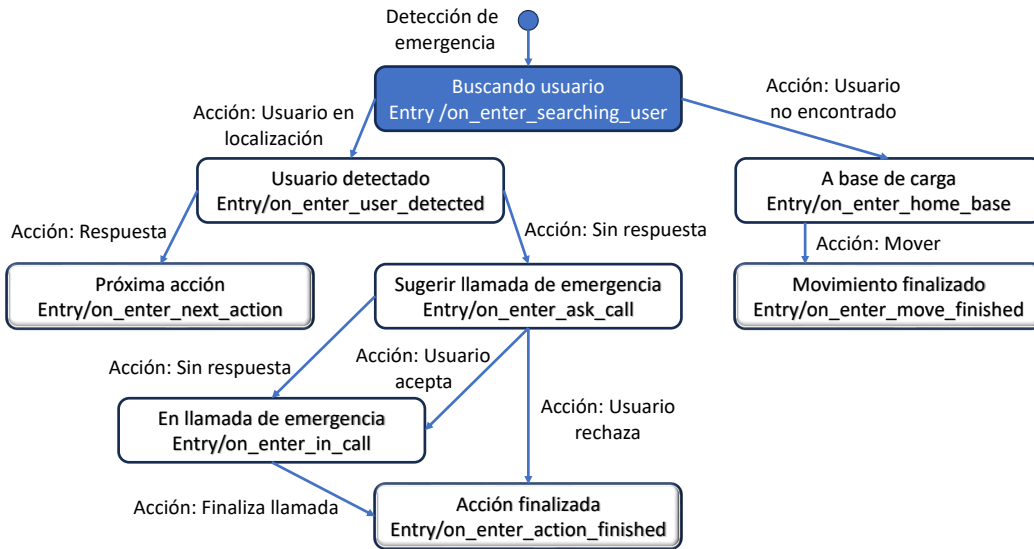


Figura 6.16: Diagrama de secuencia de actuación del robot social durante el plan de detección de emergencia.

inactividad prolongada, toma de medicinas, ver TV, despertar y comer.

En la Figura 6.16 se describe la secuencia de acciones a realizar cuando se detecta la anomalía “Emergencia”. Esta situación se detecta si el usuario presiona un botón de emergencia o si se detecta la métrica de inactividad excede los 30 minutos. Para iniciar el plan, el robot busca al usuario por la casa partiendo de la estancia en la que se ha detectado la última actividad. El robot es capaz de reconocer a la persona mediante un método de visión por computador basado en MediaPipe [214]. Si no detecta al usuario, el robot vuelve a la base de carga y se finaliza el plan. En cambio, si encuentra al usuario, el robot le pregunta si se encuentra bien. En caso afirmativo, el robot finaliza el plan. En cambio, si el usuario no responde, le sugiere realizar una llamada a su contacto de emergencia. Si el usuario no responde o acepta la llamada, el robot realiza una videoconferencia con el contacto que tenga registrado como contacto de emergencia. Si el usuario rechaza la llamada, el robot finaliza el plan.

En la Figura 6.17 se puede observar el plan de ejecución de los planes de inactividad prolongada y de despertar. Para el plan de despertar, cuando el sistema HAR

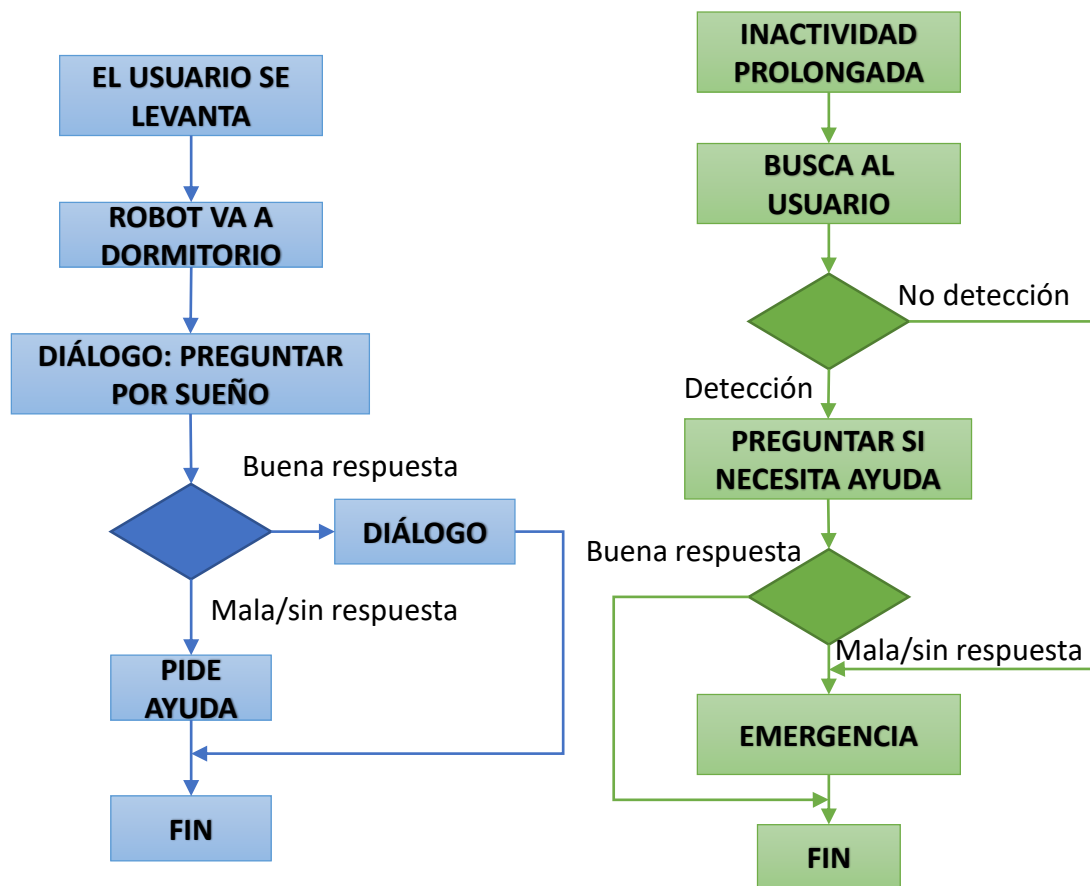


Figura 6.17: Diagrama de ejecución de planes: despertar e inactividad prolongada.

deja de detectar la ADL dormir, envía al robot al dormitorio. Una vez que llega a la localización, el robot pregunta por la calidad del sueño del usuario. Esta calidad del sueño se registra en la base de datos para poder obtener un histórico sobre el sueño del usuario. En función de la respuesta del usuario, el robot ofrece un diálogo personalizable y/o proporciona sugerencias de funciones del robot. Si el usuario no responde o manifiesta malestar, el robot sugiere pedir ayuda mediante videollamada al cuidador. Para el plan de inactividad prolongada, si el módulo HAR y el módulo de anomalías y métricas no detecta actividad durante una duración específica (en este caso, 30 minutos), el sistema envía al robot a buscar al usuario. Si el robot se desplaza por las habitaciones y no lo encuentra, o si manifiesta que necesita ayuda o no responde, se lanza la opción de notificar al cuidador o realizar una videollamada en el punto donde se encuentre el usuario.

Durante los 2 meses de pruebas, se han obtenido los siguientes resultados aproximados. Para el plan de emergencia, se simularon 10 situaciones de emergencia en diferentes momentos y lugares dentro de la residencia. El sistema detectó exitosamente el 90 % de las emergencias, brindando soporte al usuario a través del robot y la opción de videollamadas de emergencia. El margen de error ocurrió debido a que el robot no pudo encontrar al usuario mediante visión artificial. Respecto a la detección de salir de casa, se registraron todos los momentos en los que los usuarios salieron de la vivienda. En promedio, los usuarios salieron de casa dos veces al día durante el experimento de 2 meses. El sistema reconoció el 97 % de estas salidas. El error en la detección ocurrió debido a que la puerta abierta quedó abierta en varias ocasiones, causando confusión en el sistema. Para la métrica de períodos prolongados de inactividad, el sistema pudo detectar el 90 %. Los errores pueden atribuirse a registros de sensores de presencia causados por la mascota del usuario. La métrica para la toma de medicamentos logró una efectividad del 83 %. Para las deambulaciones nocturnas, se registraron un total de 20 noches en las que el usuario se levantó para ir al baño. El sistema detectó el 95 % de estas deambulaciones. El usuario ha visto la televisión todos los días durante 1 hora, y el sistema detectó exitosamente el 82 % de estos períodos, recomendando levantarse para estirar las piernas. Para la detección del sueño se ha considerado que los usuarios han dormido en casa todos los días que ha durado el experimento, y el sistema detectó con precisión el 92 % de estos

Incidencia	Situaciones	Precisión
Emergencia	10	90 %
Salir de casa	120	97 %
Inactividad	25	90 %
Medicinas	60	83 %
Deambulaciones	20	95 %
TV	60	82 %
Dormir	60	92 %
Comer	120	82 %

Tabla 6.8: Resultados del sistema de detección de anomalías y efectividad del sistema cognitivo.

períodos. Finalmente, para la detección de las comidas, se consideraron los períodos de comida y cena. El sistema detectó el 82 % de estos períodos. Para obtener una precisión general del sistema, se calculó la media aritmética de todos los casos de estudio y sus tasas de éxito. El sistema logró una precisión general del 88.32 %. La Tabla 6.8 proporciona un resumen de todos los resultados del sistema basados en los incidentes analizados.

En la Figura 6.18 se puede ver un caso real de asistencia del robot en el hogar. Se puede observar cómo el robot propone ejercicios físicos al usuario, juegos de memoria para estimular habilidades cognitivas y asistiendo durante la actividad de cocinar (el robot puede sugerir recetas o contestar preguntas del usuario).

En la Figura 6.19 se puede observar la apariencia de la aplicación web. La apariencia final de la aplicación web para acceder a información sobre el hogar del usuario monitorizado se caracteriza por presentar una interfaz limpia e intuitiva. La página de inicio presenta un diseño simple pero eficaz, con un menú de navegación que proporciona fácil acceso a diferentes secciones de la aplicación. El panel principal presenta un resumen de información clave, como el estado actual de varios sensores del hogar, actividades recientes y cualquier alerta o notificación. Los diferentes gráficos y diagramas proporcionan representaciones visuales de tendencias de datos a lo largo del tiempo, permitiendo a los cuidadores y/o familiares evaluar rápidamente la situación. La aplicación cuenta con mapas interactivos que proporcionan actualizaciones en tiempo real sobre la ubicación y movimiento del robot social dentro del



Figura 6.18: Robot social Temi V3 asistiendo al usuario en 3 situaciones diferentes: (a) proponiendo ejercicio físico, (b) proponiendo juegos de memoria y (c) asistiendo en la cocina.

hogar, mejorando la adquisición de información para cuidadores y familiares.

El sistema de inteligencia ambiental, junto con el módulo de detección de actividades basado en HMMs jerárquicos explicado en la Sección 4.3.6, se ha desplegado en 5 hogares reales de personas mayores durante una duración de 2 meses con el fin de comprobar la validez del mismo. En la Figura 6.20 se puede observar un diagrama de Gantt que recoge las actividades realizadas por el usuario de una de las viviendas analizadas durante un día del estudio y su secuencia de ejecución. En esta gráfica, el eje de ordenadas representa las diferentes ADL's que el usuario ha llevado a cabo durante el día analizado. El eje de abscisas representa cada una de las horas del día. En este caso, el usuario ha realizado las siguientes actividades: dormir, actividad en el baño, salir de casa, cocinar, comer y ver la televisión. De esta gráfica se pueden extraer métricas como el número de horas de sueño durante ese día o el tiempo que ha visto la televisión. También es útil para conocer el tiempo que el usuario ha estado fuera de casa y sus horarios, así como las veces que ha ido al cuarto de baño. Se puede ver que el sistema recoge el patrón de comportamiento que puede adoptar una persona mayor. Una de las debilidades del sistema que se puede apreciar en la gráfica es la confusión entre la actividad de comer y cocinar. Esto se debe a que ambas actividades comparten sensores, por lo que el sistema puede dudar entre escoger una actividad u otra.

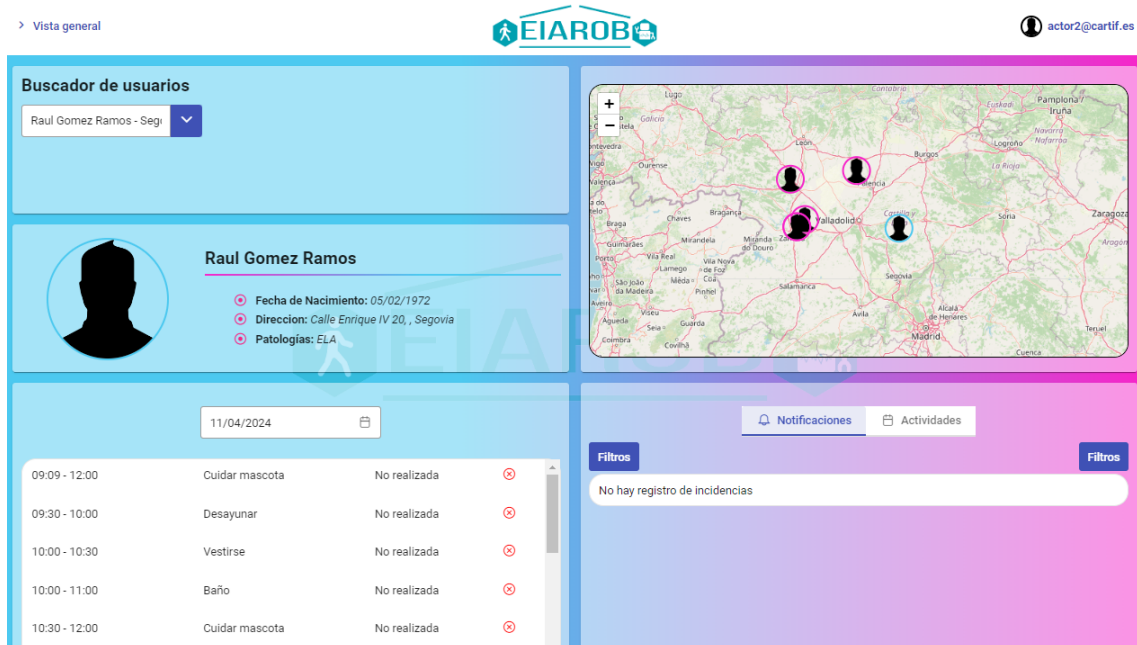


Figura 6.19: Aspecto general de la aplicación web desarrollada para acceder a la información de los usuarios de la vivienda monitorizada.

En las gráficas de la Figura 6.21, se pueden observar los patrones de comportamiento de los usuarios de las 5 viviendas analizadas. En estos diagramas se recoge en el eje de ordenadas el número de activaciones u ocurrencias de las ADL's posibles para esa vivienda durante los 2 meses del estudio. En el eje de abscisas aparecen las diferentes horas del día. Con todo esto, el diagrama aporta información sobre el patrón de comportamiento del usuario analizado. Cada vivienda cuenta con un número diferente de ADL's, ya que sólo se registraron las actividades que el usuario podía realizar de forma independiente. De estas gráficas se puede extraer como resultado que cada usuario tiene un patrón de comportamiento diferente en función de su secuencia de actividades a lo largo del día. Sin embargo, se puede extraer alguna semejanza. Por ejemplo, para todos los usuarios, la actividad de "Dormir" se lleva a cabo durante la noche, por lo que en todas las gráficas curva crece durante esas horas. Para actividades más espontáneas, como "Actividad en el baño", cada usuario las realiza de forma aleatoria. En la actividad de "Salir de casa", se puede observar que algunos usuarios siguen cierta pauta, lo que podría indicar que mantienen un horario determinado.

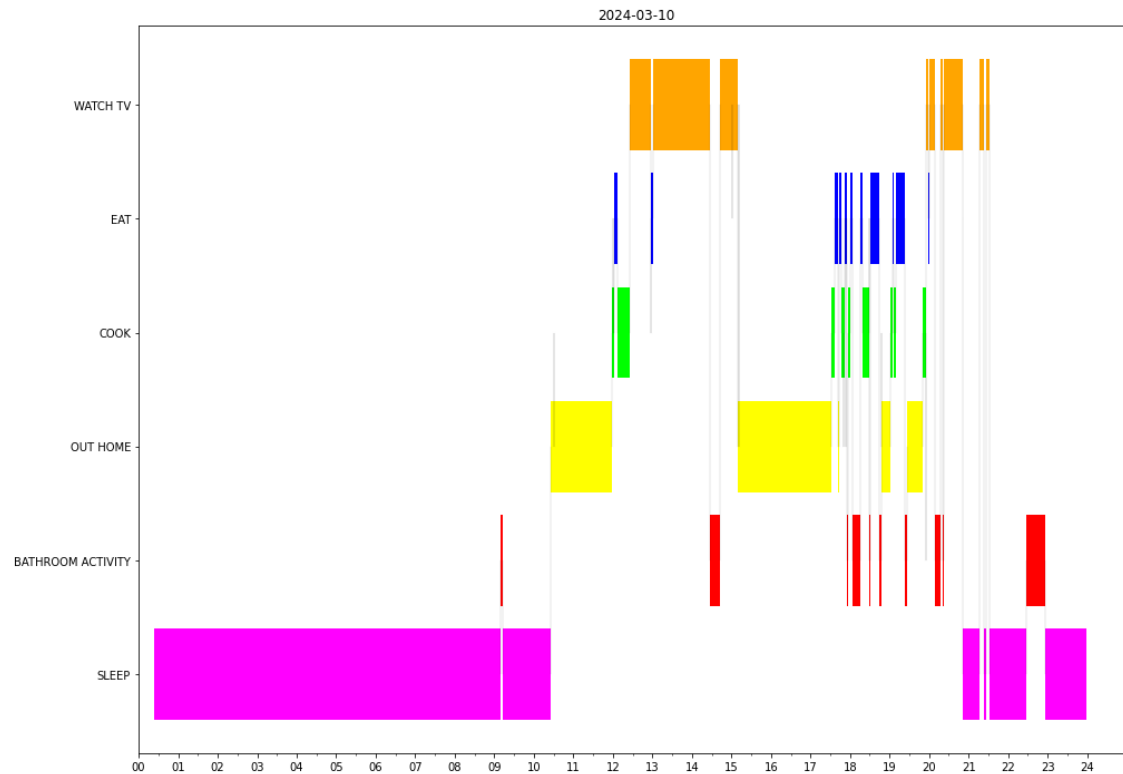
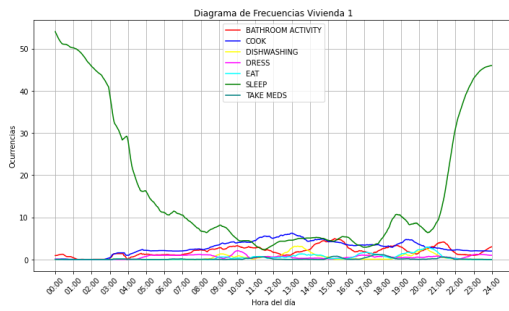
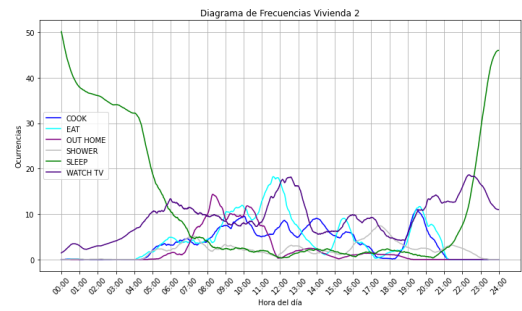


Figura 6.20: Diagrama de Gantt resultado del sistema de detección de actividades para una vivienda real.

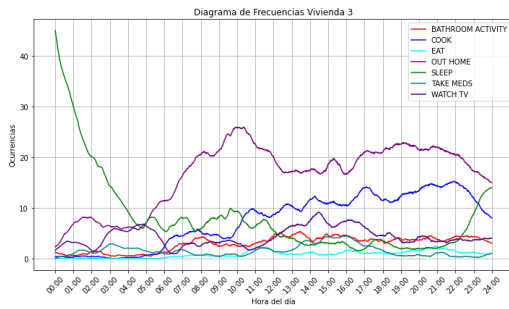
Se ha realizado un estudio por vivienda para analizar la evolución de la duración total de cada una de las actividades detectadas. Este estudio se ha elaborado para controlar la regularidad con la que los usuarios realizan las actividades y para saber el número de horas totales que necesitan para desempeñar una actividad. En las gráficas de la Figura 6.22 se puede observar la evolución de la duración de las actividades para cada vivienda. En el eje de abscisas aparecen numerados los días del estudio, y en el eje de ordenadas se encuentra el número total de horas que se han necesitado para cada actividad. De las gráficas se puede extraer que la actividad que más duración tiene es la de “Dormir”. Se puede apreciar también que muchas actividades tienen un patrón de duración muy estable, como puede ser el caso de “Actividad en el baño”. Por último, se puede observar también que otras actividades más arbitrarias, como es el caso de “Ver la televisión”, tienen un patrón de duración más irregular, ya que es una actividad que se puede activar en cualquier momento. En las gráficas de la Figura



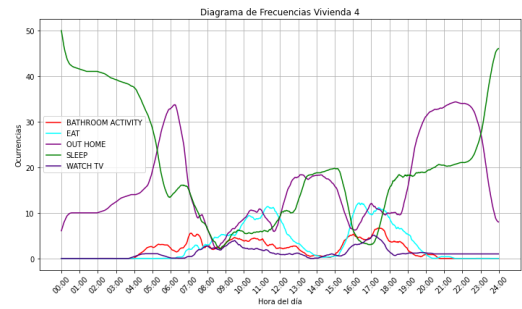
(a) Vivienda 1.



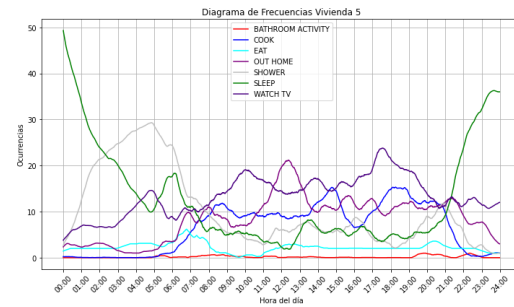
(b) Vivienda 2.



(c) Vivienda 3.

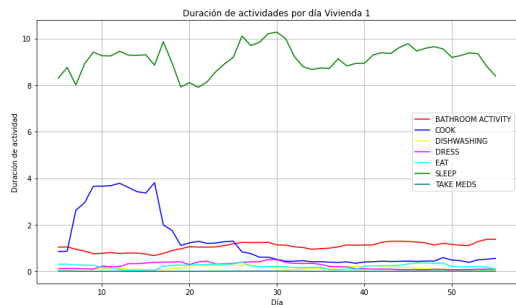


(d) Vivienda 4.

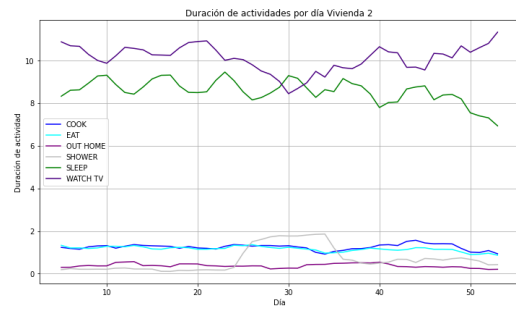


(e) Vivienda 5.

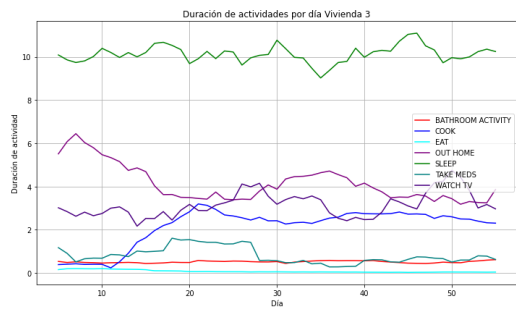
Figura 6.21: Diagramas de frecuencias de actividades para las 5 viviendas reales analizadas durante 2 meses.



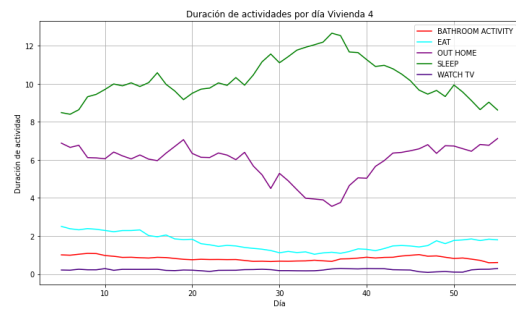
(a) Vivienda 1.



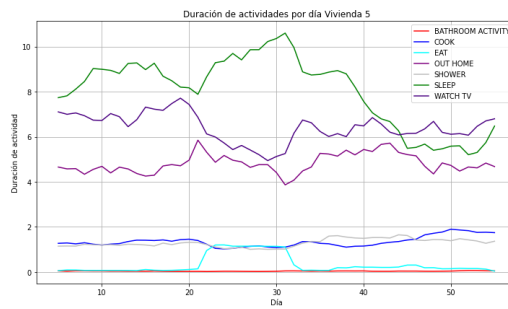
(b) Vivienda 2.



(c) Vivienda 3.



(d) Vivienda 4.



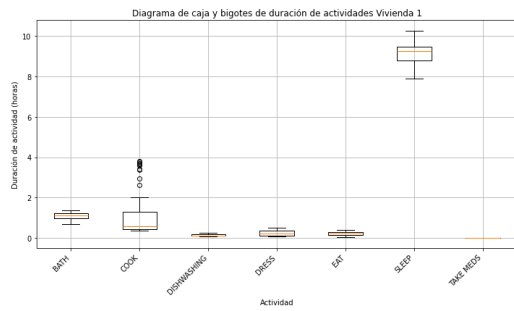
(e) Vivienda 5.

Figura 6.22: Gráficas de duración de actividades por día para las 5 viviendas reales analizadas durante 2 meses.

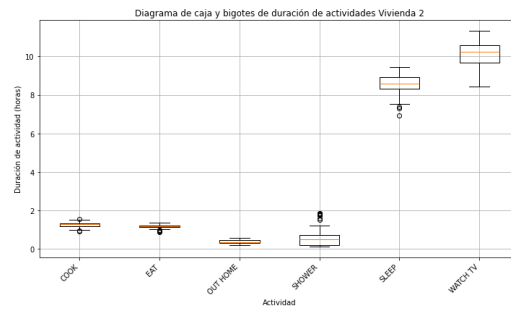
6.23 se pueden observar los diagramas de caja y bigotes de todas las actividades para cada vivienda. En estos diagramas se pueden apreciar 3 componentes diferentes: la caja, los bigotes y los puntos fuera de los bigotes. En la caja, se puede ver el rango intercuartílico de los datos, lo que corresponde a la duración (en horas) del 50 % total de los datos. El borde inferior de la caja corresponde al primer cuartil (Q1), que es el valor por debajo del cual se encuentra el 25 % de los datos. El borde superior de la caja se corresponde al tercer cuartil (Q3), que es el valor por debajo del cual se encuentra el 75 % de los datos. La línea que aparece en el medio de la caja se corresponde con la mediana, cuya labor es dividir el conjunto de datos en dos partes iguales. Con respecto a los bigotes del diagrama, se extienden desde los bordes de la caja hasta los valores extremos del conjunto de datos. Estos valores extremos son puntos fuera del rango intercuartílico, y se calculan como 1.5 veces el rango intercuartílico por encima del tercer cuartil Q3 o por debajo del primer cuartil Q1. Por último, los puntos fuera de los bigotes se consideran valores atípicos. De los diagramas se puede extraer que las actividades con las cajas más pequeñas se corresponden con las ADL's con duraciones más estables, como puede ser la actividad de "Comer". Del mismo modo, se pueden extraer las actividades con mayor desviación. Por ejemplo, en la vivienda 5, la ADL más variable es "Dormir".

En la Tabla 6.9 aparece un resumen de los resultados obtenidos para cada vivienda al calcular la media y la desviación típica para cada una de las ADL's detectadas. En la tabla se ha representado con un signo "—" las actividades que no se han detectado en cada vivienda debido a que los usuarios no podían realizar dicha actividad por si mismos sin necesitar una ayuda externa.

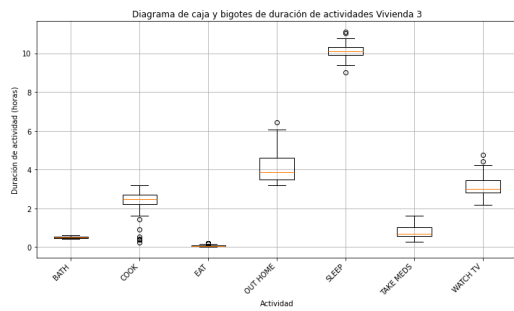
Para verificar la efectividad del método empleado para detectar las distintas ADL's posibles en las 5 viviendas reales, se ha realizado un etiquetado posterior mediante el conocimiento de un experto. Este etiquetado se ha realizado de forma manual, observando gráficamente la información proporcionada por los sensores de cada vivienda y la salida del modelo de predicción, tal y como se puede visualizar en la Figura 6.24. Los sensores de eventos aparecen en la parte superior, mientras que la información de los sensores ambientales se sitúa en la parte inferior. Por último, en la parte de abajo de la figura, se pueden observar las distintas predicciones por colores que ha realizado el modelo, junto con una leyenda explicativa para mejorar



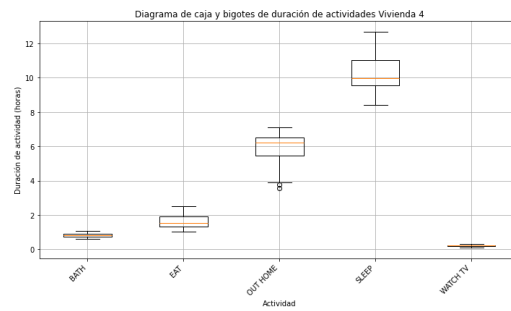
(a) Vivienda 1.



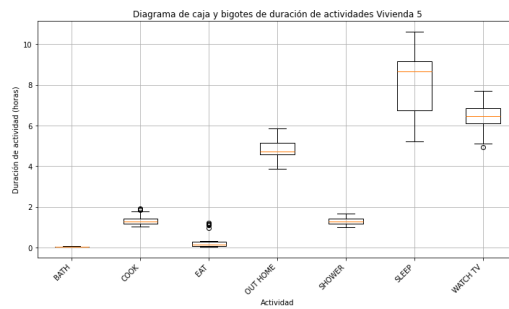
(b) Vivienda 2.



(c) Vivienda 3.



(d) Vivienda 4.



(e) Vivienda 5.

Figura 6.23: Diagramas de caja y bigotes de duración de actividades por día para las 5 viviendas reales analizadas durante 2 meses.

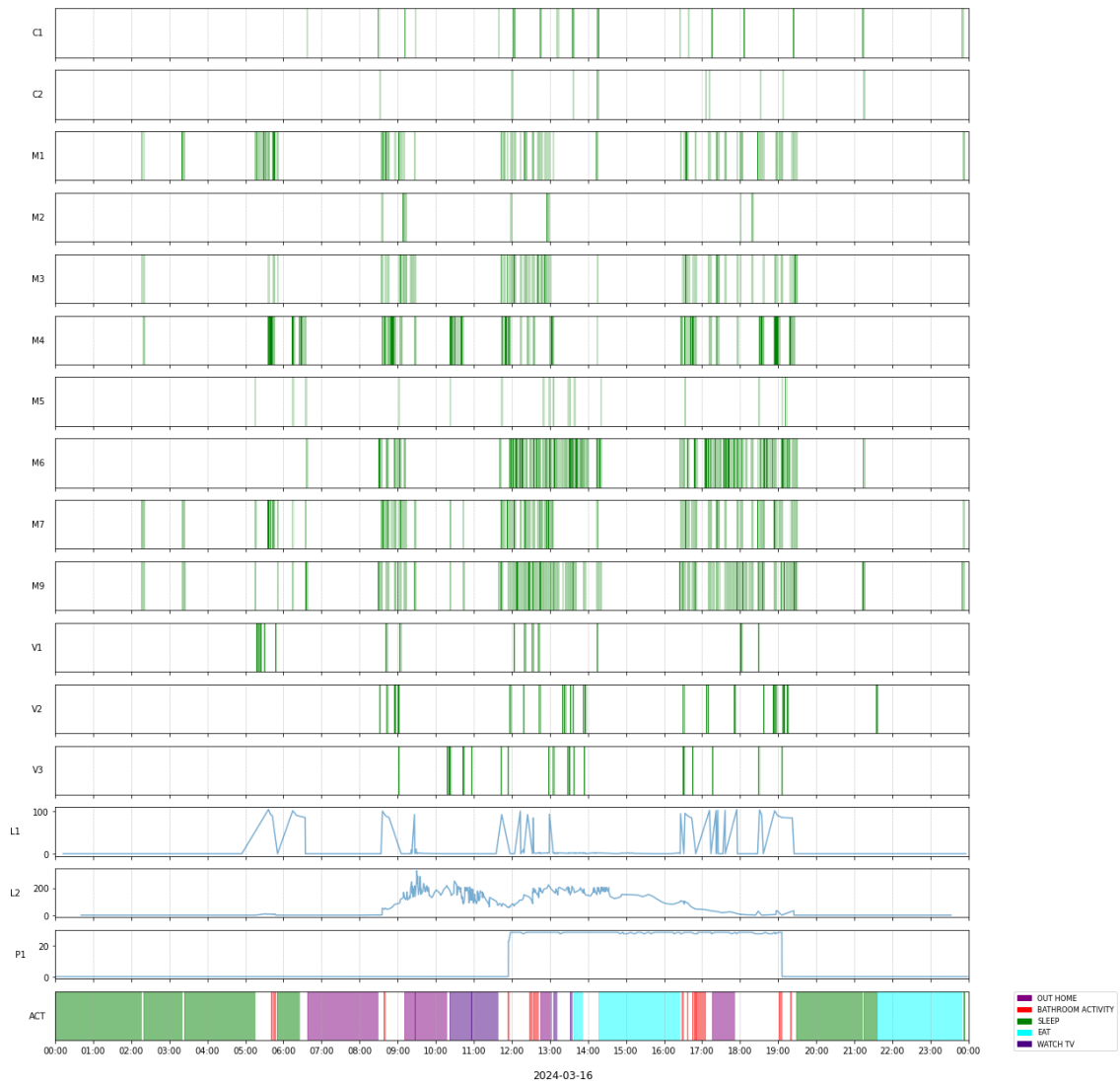


Figura 6.24: Resumen de información relativa a una vivienda durante un día completo para realizar el etiquetado manual: gráficas de sensores y salida del modelo.

Actividad	Vivienda 1 (h)	Vivienda 2 (h)	Vivienda 3 (h)	Vivienda 4 (h)	Vivienda 5 (h)
Actividad en el baño	1.07 ± 0.51	-	0.52 ± 0.20	1.15 ± 0.31	0.04 ± 0.05
Cocinar	1.64 ± 3.80	1.24 ± 0.62	1.96 ± 1.29	-	1.36 ± 0.49
Lavar los platos	0.14 ± 0.17	-	-	-	-
Vestirse	0.21 ± 0.32	-	-	-	-
Comer	0.21 ± 0.35	1.13 ± 0.43	0.08 ± 0.09	2.88 ± 0.94	0.28 ± 1.07
Salir de casa	-	0.34 ± 0.45	4.31 ± 1.84	18.12 ± 1.71	4.72 ± 2.25
Ducharse	-	0.59 ± 1.13	-	-	1.28 ± 0.56
Dormir	8.66 ± 3.45	8.44 ± 2.30	10.22 ± 2.01	1.22 ± 0.92	7.92 ± 2.85
Tomar medicinas	0.01 ± 0.02	-	0.84 ± 1.37	-	-
Ver televisión	-	10.26 ± 2.29	3.07 ± 2.12	0.17 ± 0.22	6.56 ± 2.10

Tabla 6.9: Media y desviación típica (en horas) de duración de actividades para las 5 viviendas reales analizadas durante 2 meses.

la legibilidad. De este modo, el experto ha realizado un etiquetado diario para cada vivienda teniendo en cuenta toda la información de la figura.

La fase de etiquetado se ha realizado durante un total de 50 días y siempre mediante el mismo experto. Además, el experto no se encuentra en la vivienda, hecho que aumenta la complejidad de la tarea. Una vez que se ha realizado el etiquetado manual, se han comparado estas etiquetas con las salidas del modelo. Los resultados obtenidos de esta comparación se pueden visualizar en la Tabla 6.10. En esta tabla se pueden observar las métricas de precision, recall y F1-Score para cada una de las viviendas. La columna “N^o” hace referencia al número de la vivienda, la columna “Métrica” indica el tipo de métrica a analizar y en la columna “Actividades” muestra las diferentes ADL’s para cada vivienda. En la tabla aparece con un símbolo “-” las ADL’s que no se analizan en esa vivienda en concreto, tal y como sucede en la Tabla 6.9. De estos resultados se puede extraer que el modelo es preciso ya que todas las métricas de F1-Score superan un porcentaje de acierto del 50% para todas las actividades. De forma adicional, se ha extraído un resultado de precisión o acierto

global para cada vivienda. Las viviendas tienen una precisión global del 87 %, 85 %, 84 %, 83 % y 84 % respectivamente.

Nº	Métrica	Actividades									
		Actividad en el baño	Cocinar	Lavar los platos	Vestirse	Comer	Salir de casa	Ducharse	Dormir	Tomar medicinas	Ver TV
1	Precision	0.97	0.63	0.57	0.91	0.84	-	-	0.93	0.71	-
	Recall	0.98	0.92	0.80	0.88	0.47	-	-	0.95	0.55	-
	F1-Score	0.97	0.74	0.66	0.89	0.60	-	-	0.94	0.63	-
2	Precision	-	0.59	-	-	0.88	0.75	0.67	0.80	-	0.89
	Recall	-	0.83	-	-	0.40	0.60	0.86	0.71	-	0.96
	F1-Score	-	0.69	-	-	0.55	0.67	0.75	0.75	-	0.92
3	Precision	0.84	1.00	-	-	0.83	0.60	-	0.94	0.54	0.93
	Recall	0.89	0.67	-	-	0.56	0.94	-	0.81	0.93	0.83
	F1-Score	0.86	0.80	-	-	0.67	0.73	-	0.87	0.68	0.88
4	Precision	0.93	-	-	-	0.92	0.54	-	0.92	-	0.94
	Recall	0.96	-	-	-	0.85	0.97	-	0.74	-	0.52
	F1-Score	0.94	-	-	-	0.88	0.70	-	0.82	-	0.67
5	Precision	1.00	1.00	-	-	1.00	0.97	0.57	0.89	-	0.82
	Recall	0.53	0.83	-	-	0.50	0.97	0.94	0.69	-	0.96
	F1-Score	0.69	0.91	-	-	0.67	0.97	0.71	0.78	-	0.89

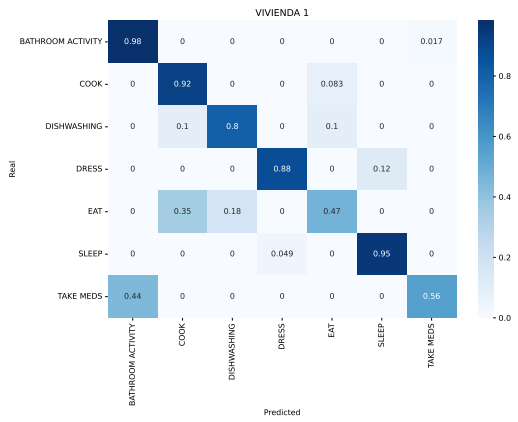
Tabla 6.10: Resultados de precision, recall y F1-Score de las actividades detectadas para las 5 viviendas reales.

Por último, se han obtenido las matrices de confusión de las ADL's para las 5 viviendas, tal y como se puede comprobar en la Figura 6.25. En estas matrices de confusión se puede observar que todas las ADL's con mayor porcentaje de acierto se encuentran distribuidas a lo largo de las diagonales de las matrices. Esto es un indicador fiable de que el modelo es preciso y se comporta de forma similar al etiquetado del experto. De estas matrices se pueden extraer otras conclusiones debido a los pesos que se encuentran alejados de la diagonal. Por ejemplo, se puede ver que el mayor porcentaje de fallos del modelo se debe a confusiones entre actividades realizadas dentro de la misma estancia. Esto se debe a que las actividades que se realizan dentro de las mismas estancias pueden compartir sensores entre sí o pueden

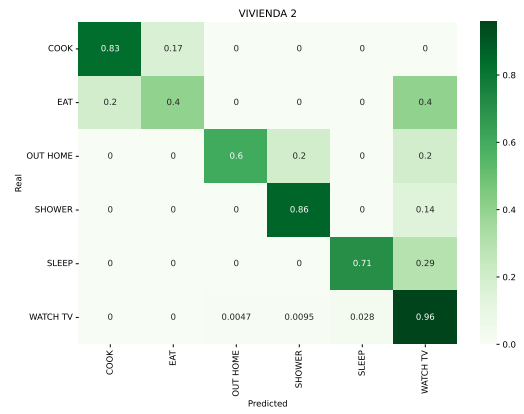
presentar similitudes durante su realización. Observando detenidamente las matrices se pueden extraer otras conclusiones. Por ejemplo, para la vivienda 1, el modelo confunde la actividad de comer con cocinar. Esta confusión se puede deber a que ambas actividades se realizan de manera próxima en el tiempo y, normalmente, de forma secuencial. Por lo tanto, debido al sensor de temperatura y humedad de la cocina, puede existir cierta inercia que produzca que el sistema detecte que el usuario sigue cocinando cuando realmente ya ha empezado a comer. Otro ejemplo que se puede ver en la vivienda 2 es la tendencia a que el modelo genere como salida la actividad de ver la televisión. Esto se debe a que el usuario de esa vivienda tiene encendida la televisión una media de 10 horas al día (tal y como se pudo ver en la Figura 6.22). Por último, en la vivienda 5, el modelo confunde la actividad de dormir con ver la televisión. Esto se debe a que el usuario cuenta con televisión en el dormitorio y tiende a dormir con la televisión encendida.

6.4. Discusión

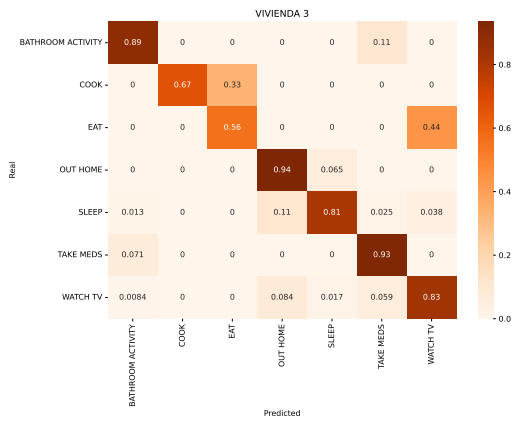
Una de las fortalezas del método de aprendizaje supervisado para viviendas con un único usuario es la capacidad de distinguir entre ADL's que se llevan a cabo en la misma estancia o que se encuentran muy próximas en el espacio. Puede ser el caso de la actividad dormir y actividad en el dormitorio principal. Ambas actividades se desarrollan en la misma habitación y se reconocen mediante los mismos sensores. Sin embargo, el modelo es capaz de diferenciar las ADL's debido al hábito y el patrón de comportamiento del usuario, así como por la hora a la que se ejecuta la actividad. Otra fortaleza se corresponde con la capacidad de diferenciar entre medicación matutina y medicación nocturna. Ambas actividades se realizan de la misma forma, pero debido a que el modelo cuenta con la hora del día, el sistema es capaz de diferenciar entre ambas. Esto es muy útil para diferenciar entre los hábitos de medicación del usuario, pudiendo generar una alerta en el caso de que esté cometiendo un fallo en cualquiera de ellos. En cuanto a las debilidades de este método, se puede ver que las dos actividades correspondientes a la medicación las confunde de forma ocasional con actividad en la cocina. Esto se debe a que las medicinas se guardan en un armario de la cocina. Además, se ha observado en la



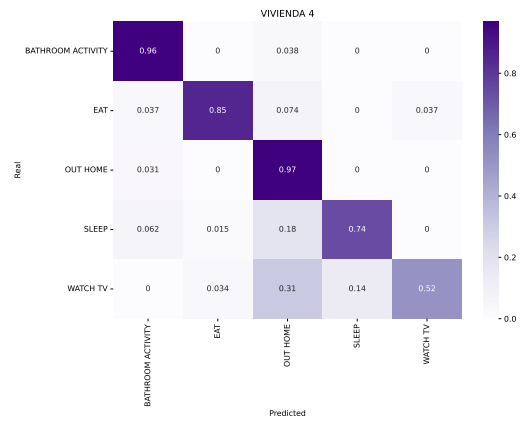
(a) Vivienda 1.



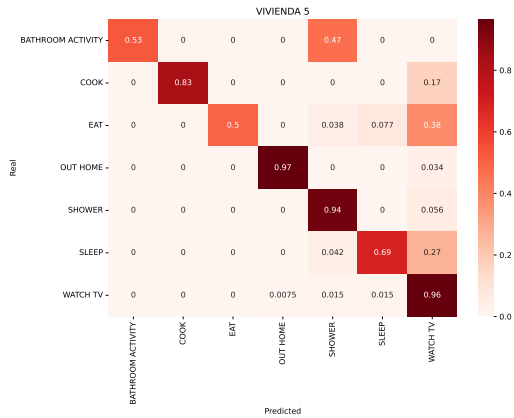
(b) Vivienda 2.



(c) Vivienda 3.



(d) Vivienda 4.



(e) Vivienda 5.

Figura 6.25: Matrices de confusión de las actividades detectadas para las 5 viviendas reales.

base de datos que, en ocasiones, está etiquetada la actividad de medicación y no ha habido interacción con el armario. Se puede deducir que el usuario a veces se deja el medicamento fuera del armario, o deja la puerta abierta. Otro aspecto a resaltar es la tendencia de confusión de la actividad otros. Esto se debe a que es una actividad en la que el usuario no está interactuando con nada de la vivienda. Por esta razón, este tiempo ocioso es difícil de cuantificar.

De los resultados obtenidos para la experimentación mediante los métodos de aprendizaje supervisado para viviendas con varios residentes, se puede extraer que la información de las posiciones de los usuarios en las viviendas es de vital importancia para inferir qué usuario está realizando la actividad. Del mismo modo, se obtiene que los modelos de redes neuronales basados en capas LSTM y GRU funcionan mejor que las RNN tradicionales para este caso particular. Además se extrae que es posible reconocer ADL's mediante la aplicación de redes neuronales, proporcionando resultados muy elevados. La técnica de detención temprana es muy útil para evitar el sobreentrenamiento de los modelos. De las curvas obtenidas de los entrenamientos para ambos usuarios se puede deducir que el método apenas tiene sobreentrenamiento, ya que en todos los casos las curvas correspondientes al conjunto de datos de entrenamiento y validación son muy próximas y siguen la misma tendencia. Se ha obtenido poco sobreentrenamiento gracias al efecto provocado por la técnica de detección temprana. Además, del número de épocas se puede deducir también los entrenamientos son más rápidos para los modelos basados en LSTM y GRU, ya que se corresponden con redes más avanzadas y optimizadas que las redes RNN convencionales. De los experimentos realizados con los modelos CNN con CVV-SV se ha podido extraer que el rendimiento del modelo es mejor y proporciona mejores resultados que mediante la aplicación de redes RNN, LSTM y GRU. Concretamente, para el usuario 1, el modelo convolucional simple con filtro de (60,1) y $k=1$ es el que mejores resultados proporciona (97%). En cambio, para el usuario 2, el método más favorable es el convolucional con filtro de (60,1) y $k=3$, ya que proporciona una precisión del 90.64%. Con respecto a las pruebas realizadas con los modelos transformers, se ha podido comprobar que se han obtenido mejores resultados utilizando mapeo lineal que mediante el mapeo con redes LSTM, CNN o GRU. Además, los modelos transformers también parecen ofrecer resultados superiores a

modelos exclusivamente basados en capas recurrentes, como GRU o LSTM.

En cuanto a los métodos de aprendizaje no supervisado, de ambas matrices de confusión se extrae que las actividades con mejor tasa de reconocimiento son dormir y salir de casa (ambas por encima del 88 % de precisión). Esto se debe a que son las ADL's que más se repiten en la generación de la base de datos y la que mayor número de registros tienen. Una de las debilidades del sistema, que se ha podido extraer de ambas matrices de confusión, es la dependencia del sistema de localización del usuario en la casa. Puede ser que el sistema realice el reconocimiento de la actividad de forma correcta, pero puede fallar a la hora de asignar la actividad a un usuario u otro. Por esta razón, se ha establecido una métrica general al modelo, la cual asigna como predicción correcta si el sistema ha sido capaz de reconocer por lo menos una de las ADL's, independientemente de la asignación al usuario. Por lo tanto, el sistema tiene una capacidad de reconocimiento del 95.51 %. De los resultados obtenidos, se puede deducir que es posible realizar un buen reconocimiento de actividades mediante la implementación de algoritmos de aprendizaje no supervisado, basados en HMMs. Este método desarrollado es fácilmente aplicable en un gran número de viviendas, ya que no es necesario realizar una labor exhaustiva de etiquetado de actividades para que el sistema aprenda a detectar las ADL's. Además, se ha podido comprobar que los resultados son muy próximos a los obtenidos mediante los métodos de aprendizaje supervisado aplicando redes neuronales.

Tanto para los modelos basados en técnicas de aprendizaje supervisado como para los modelos basados en técnicas de aprendizaje no supervisado, existe una dependencia evidente con la precisión y sensibilidad de los sensores. Durante la fase de experimentación se ha podido comprobar que, por ejemplo, los sensores de eventos de tipo vibración son bastante sensibles y pueden generar señales falsas si se detecta algún ruido fuerte. En cambio, los sensores de eventos de tipo contacto son muy fiables, ya que la apertura o el cierre de una puerta es una acción en la que hay que ejercer una determinada fuerza para generar la señal. También hay que tener en cuenta los posibles fallos en la comunicación entre los sensores y el concentrador central debido a las posibles interferencias que pueden ocurrir en el entorno. Todo esto repercute en la efectividad del sistema de detección de actividades y en la precisión que se puede alcanzar.

Modelo	Test accuracy (Usuario 1)	Test accuracy (Usuario 2)
Red neuronal RNN	0.8959	0.8626
Red neuronal LSTM	0.8963	0.8829
Red neuronal GRU	0.9091	0.8621
CNN con filtro de (60,1) (k=1)	0.9700	0.8956
CNN con filtro de (60,1) (k=3)	0.9583	0.9064
CVV-SV con k=5 con filtros de (60,1) (3 mejores modelos)	0.9608	0.8808
Transformer con 3 capas y mapeo lineal	0.9408	0.8985
Transformer con 3 capas y mapeo LSTM bidireccional (2 capas)	0.8965	0.9155
HMM	0.9168	0.8678

Tabla 6.11: Resumen de los diferentes métodos de reconocimiento de actividades empleados con la base de datos SDHAR-HOME.

En la Tabla 6.11 se recoge un resumen de los resultados obtenidos durante la fase de experimentación con la base de datos SDHAR-HOME desarrollada en la presente tesis. En la tabla aparecen los máximos resultados obtenidos (en test accuracy) para cada usuario en función del modelo utilizado, tanto para los métodos de aprendizaje supervisado como no supervisado. Como se puede comprobar, para el usuario 1, el modelo que mejor rendimiento ha proporcionado ha sido el CNN con filtro de (60,1) (k=1), dando un test accuracy del 97%. En cambio, para el usuario 2, el mejor modelo ha sido el transformer con 3 capas y mapeo LSTM bidireccional (2 capas), proporcionando un test accuracy del 91.55%. De estos resultados se puede extraer que los modelos más precisos para esta base de datos se alcanzan utilizando métodos de aprendizaje supervisado. Sin embargo, los resultados obtenidos mediante los métodos de aprendizaje no supervisado (HMM) también proporcionan un buen test accuracy, siendo cercano en ambos al 90%.

Para evaluar los resultados obtenidos de la aplicación del ecosistema de inteligencia ambiental, al final de los 2 meses de duración del estudio, se realizó una encuesta a los usuarios sobre la efectividad y funcionalidad de la solución. El usuario expresó una alta satisfacción con las tareas efectuadas por el sistema, indicando la mejora que ha experimentado su calidad de vida. El ecosistema de inteligencia ambiental ha

demostrado un impacto positivo en la calidad de vida del usuario al proporcionar apoyo y asistencia en actividades diarias, facilitando la comunicación con familiares y cuidadores, y alertando sobre situaciones de emergencia. La capacidad del robot social para interactuar de forma natural y ofrecer asistencia personalizada ha sido especialmente valorada por el usuario, lo que indica el potencial del sistema para mejorar la independencia y el bienestar de las personas mayores y aquellas con necesidades especiales. Actualmente, el sistema se encuentra implementado en 5 hogares piloto, y se espera que se despliegue en 100 hogares para finales del año 2024.

Durante el desarrollo de la tesis doctoral se ha podido comprobar que, tanto los modelos HAR realizados mediante métodos de aprendizaje supervisado como los modelos HAR realizados mediante métodos de aprendizaje no supervisado, ofrecen buenos resultados a la hora de detectar actividades realizadas por personas mayores en sus casas. Tal y como se ha visto en la Tabla 6.11, generalmente los modelos HAR realizados mediante métodos de aprendizaje supervisado ofrecen mejores resultados (como el modelo basado en CNN, ya que proporciona el mayor porcentaje de test accuracy). Esto se debe a que las técnicas actuales de aprendizaje automático están muy optimizadas para alcanzar tasas de éxito elevadas. Sin embargo, estos métodos de aprendizaje supervisado necesitan contar con una fase de entrenamiento extensa, en la que se cuente en todo momento con las salidas deseadas del modelo. En el caso que atañe a la presente tesis, es necesario contar con las actividades reales que realizan los usuarios durante la fase de entrenamiento para poder emplear estos métodos. Esto es una tarea difícil, ya que no se puede exigir al usuario que etiquete todas sus acciones durante su día. Por esta razón, en la mayoría de los casos, el modelo a utilizar es el basado en métodos de aprendizaje no supervisado, ya que no necesita contar con una etapa de entrenamiento.

A la hora de implementar el sistema de detección de actividades en una vivienda nueva, es necesario realizar una configuración previa de los sensores con los que cuenta la instalación. Esta configuración determina el número de sensores utilizados y su tipo. Si se opta por realizar un etiquetado manual de las actividades para utilizar un método de aprendizaje supervisado, el propio sistema aprende de forma automática y relaciona los eventos de los sensores con las actividades realizadas. En cambio, si se opta por utilizar un método de aprendizaje no supervisado, es necesario

indicar al sistema la relación de cada sensor con cada posible ADL para establecer una distribución de pesos inicial para el HMM. Esta distribución de pesos se puede ajustar en tiempo real si se aplican algoritmos de optimización, como puede ser el algoritmo de Baum-Welch.

Por último, se va a comparar los resultados de los modelos realizados en la presente tesis con los principales trabajos encontrados en la bibliografía relacionados con los métodos presentados en la tesis. En primer lugar, se van a analizar los modelos basados en métodos de aprendizaje supervisado. En [215], los autores desarrollaron un sistema capaz de analizar las principales articulaciones y movimientos de los usuarios mediante visión por computadora durante diferentes actividades diarias. Para construir el modelo de predicción, utilizaron una red neuronal Inception-ResNet. Además, evaluaron su enfoque con tres bases de datos públicas: UTD_MHAD, HDM05 y NTU RGB+D 60 (NTU60), obteniendo precisiones entre el 85.45 % y el 98.13 %. En general, los enfoques basados en visión por computadora pueden lograr altas tasas de reconocimiento, pero pueden comprometer significativamente la privacidad de los usuarios. Otros estudios, como el realizado por [216], también analizan secuencias de vídeo para obtener las posturas realizadas durante sesiones de gimnasia grabadas por ellos mismos, obteniendo una tasa de precisión superior al 98 %. Para lograr estos resultados, utilizaron un modelo basado en redes MLP, integrado con un modelo HMM no supervisado. En [217], los autores presentan un sistema capaz de analizar señales proporcionadas por sensores inerciales aplicando redes LSTM. Evaluaron la efectividad de su sistema con dos bases de datos de acceso abierto, WISDM y PAMAP2, obteniendo precisiones superiores al 97 %. En [142], los autores desarrollaron un modelo de clasificación robusto para realizar HAR analizando señales de sensores portátiles. Para ello, utilizaron redes CNN combinadas con capas LSTM bidireccionales (BiLSTM) para extraer las principales características de las señales antes de converger en las actividades resultantes. Aplicaron su modelo a tres bases de datos: UCI-HAR, WISDM y PAMAP2, obteniendo tasas de acierto entre el 94 % y el 96 %. Las mismas bases de datos se utilizaron en el trabajo [218], donde combinaron redes CNN con redes GRU, alcanzando una precisión del 97 %. Otros autores propusieron el uso de modelos supervisados SVM, como en [219], logrando detectar seis actividades (por ejemplo, caminar, subir escaleras o sentarse) con una precisión

del 94.66 %. En [220], se presenta el desarrollo de un modelo de reconocimiento de acciones basado en esqueletos mediante la aplicación de una red neuronal guiada por semántica (SGN), compuesta por la combinación de CNN con redes de convolución gráfica (GCN). El modelo se probó con tres bases de datos diferentes: NTU RGB+D 60 (NTU60), NTU RGB+D 120 (NTU120) y SYSU 3D Human-Object Interaction Dataset, obteniendo resultados que oscilan entre el 79.2 % y el 90.6 % de acierto. Otro trabajo que utiliza sensores ambientales es el desarrollado por [100], en el que crearon un algoritmo HAR basado en redes CNN para analizar las principales características del dominio del tiempo, combinado con una serie de capas densas. Utilizaron la base de datos CASAS (Cairo, Milan, Kyoto7, Kyoto8 y Kyoto11) para obtener una precisión entre el 86.68 % y el 97.08 %. Finalmente, el trabajo de [221] utilizó un método por modelos transformer. Los autores aplicaron este modelo a tres bases de datos: Penn-Action, NTU60 y NTU120, obteniendo resultados superiores al 90 %. Un resumen de todos los trabajos presentados, junto con los propuestos en la presente tesis, se encuentra en la Tabla 6.12.

En cuanto a los métodos de aprendizaje no supervisado, se ha realizado un estudio bibliográfico para obtener los resultados de investigación mas relevantes. El trabajo desarrollado por [222] propone el uso de métodos estadísticos para extraer las principales características de las señales generadas por los sensores de un smartphone de la base de datos WEKA. El método elegido es la aplicación de la transformada rápida de Fourier (FFT). En [223] los autores utilizan HMMs para realizar una regresión con respecto a la información de un conjunto de nueve acelerómetros. Los autores pudieron reconocer un total de 12 actividades con una tasa de acierto del 89 %. Otro sistema implementado sobre HMM es el elaborado por [224]. En este caso, los autores extrajeron previamente las características de las señales de entrada mediante un método micro-Doppler. El método se probó con una base de datos generada durante la experimentación, con cinco actividades y se obtuvo una precisión del 69 %. En [225], los autores proponen el uso de VAE para detectar actividades. Los autores evaluaron su método contra una base de datos no etiquetada conocida como HHAR, obteniendo una tasa de acierto del 87 %. El trabajo realizado por [226] presenta una comparación de tres métodos no supervisados: k-Means, modelos de mezcla gaussiana (GMM) y HMMs. Los autores desarrollaron su propia base de datos a partir de la

Trabajo	Método	Base de datos	Nº Actividades	Precisión
[215]	ResNet	UTD_MHAD HDM05 NTU RGB+D 60	27 130 60	98.13 % 90.67 % 85.45 %
[216]	MLP + HMM	Desarrollo propio	19	98.05 %
[217]	LSTM	WISDM PAMAP2	6 18	97.32 % 97.15 %
[142]	CNN +LSTM	UCI-HAR WISDM PAMAP2	6 6 18	96.37 % 96.05 % 94.29 %
[218]	CNN + GRU	UCI-HAR WISDM PAMAP2	6 6 18	96.20 % 97.21 % 95.27 %
[219]	SVM	Desarrollo propio	6	94.66 %
[220]	SGN	NTU RGB+D 60 NTU RGB+D 120 SYSU	60 120 12	89 % 79.20 % 90.60 %
[100]	CNN	Cairo Milan Kyoto7 Kyoto8 Kyoto11	13 15 13 12 25	91.99 % 95.35 % 86.68 % 97.08 % 90.27 %
[221]	Transformer	Penn-Action NTU RGB+D 60 NTU RGB+D 120	15 60 120	98.7 % 92 % 90.3 %
Tesis	BiLSTM	Milan	16	95.42 %
Tesis	RNN, LSTM y GRU	SDHAR- HOME	18	90.91 %
Tesis	CNN + CVV-SV	SDHAR- HOME	18	97.00 %
Tesis	Transformer	SDHAR- HOME	18	94.08 %

166 Tabla 6.12: Comparación de los principales métodos de reconocimiento de actividades mediante técnicas de aprendizaje supervisado.

información proporcionada por una serie de acelerómetros distribuidos en los cuerpos de los usuarios, resultando en una tasa de acierto de entre el 75 % y el 84 %. Compararon los resultados de estos métodos no supervisados con métodos supervisados, como SVM, obteniendo una tasa de acierto del 95.55 %. Esta comparación muestra que los métodos de aprendizaje supervisado suelen ser más precisos aunque menos replicables. Otro desarrollo que trabaja con los acelerómetros de sensores portátiles es el llevado a cabo por [227]. En este trabajo, los autores diseñaron un método llamado codificador reconstruido asistido por aprendizaje profundo no supervisado (UDR-RC). Este algoritmo se basa en métodos de extracción de características usando FFT. Los autores utilizaron la base de datos WISDM para analizar un total de seis actividades y obtuvieron una tasa de acierto del 97.28 %. En [228], los autores desarrollaron un algoritmo conocido como MaxGap, que se basa en la implicación de las señales de entrada al modelo en la generación de un resultado dado, y el algoritmo calcula un efecto positivo o negativo dependiendo del peso que tiene en el resultado. Para probar su veracidad, los autores compararon los resultados contra una base de datos experimental y un algoritmo HMM. Como resultado, obtuvieron una tasa de acierto para MaxGap del 91.4 % frente al 93.5 % para HMM. Esto refleja que el algoritmo converge bien, pero el HMM aún proporciona mejores resultados. Otros autores han optado por aplicar enfoques basados en reglas para reconocer actividades. En [229], los autores desarrollaron un método basado en reglas profundas (DRB). Los autores analizaron un total de 50 actividades utilizando 1000 imágenes de la base de datos UCF50. Varios autores han optado por utilizar mecanismos basados en árboles de decisión para realizar HAR. En [230], los autores utilizaron árboles de decisión para analizar la base de datos ADL obteniendo una precisión del 88.02 % para 8 actividades. Un resumen de todos los trabajos presentados, junto con los propuestos en la presente tesis, se encuentra en la Tabla 6.13.

De los resultados obtenidos en ambas tablas, se puede deducir la complejidad al comparar los métodos de reconocimiento de actividades para distintas bases de datos con distinto número de actividades. Aun así, se puede comprobar que los modelos desarrollados en la presente tesis son precisos debido a la complejidad de la base de datos SDHAR-HOME al contar con varios usuarios conviviendo de forma simultánea, con mascota y recepción de visitas frecuentes.

Trabajo	Método	Base de datos	Nº Activi- dades	Precisión
[222]	FFT	WEKA	5	79.98 %
[223]	HMM	Desarrollo propio	12	89 %
[224]	HMM	Desarrollo propio	5	69 %
[225]	VAE	HHAR	9	87 %
[226]	k.Means	Desarrollo propio	12	72.95 %
	GMM			75.60 %
	HMM			83.89 %
[227]	UDR-RC	WISDM	6	97.28 %
[228]	MaxGap	Desarrollo propio	17	91.40 %
	HMM			93.50 %
[229]	DRB	UCF50	50	82.00 %
[230]	Árboles de decisión	ADL	8	88.02 %
Tesis	HMM	SDHAR-HOME	18	91.68 %
Tesis	HMM	5 viviendas reales	7	87.00 %

Tabla 6.13: Comparación de los principales métodos de reconocimiento de actividades mediante técnicas de aprendizaje no supervisado.

Capítulo 7

Conclusiones y líneas futuras

En este capítulo se detallan las distintas conclusiones obtenidas durante el desarrollo de la tesis así como las posibles líneas futuras deducidas. En la Sección 7.1 se realiza un resumen de las principales conclusiones obtenidas de la tesis y sus resultados. En la Sección 7.2 se explican las posibles aplicaciones de los métodos propuestos en la tesis. Por último, en la Sección 7.3 se detallan las posibles líneas futuras que se pueden llevar a cabo a partir de los resultados obtenidos de la tesis.

7.1. Conclusiones

Esta tesis presenta un método de reconocimiento de las principales actividades de la vida diaria que las personas mayores pueden realizar en sus viviendas. Se presenta inicialmente un sistema basado en métodos de aprendizaje supervisado utilizando redes neuronales LSTM bidireccionales analizando la información de una base de datos pública conocida como Milan, del proyecto CASAS. Este método permite realizar un reconocimiento en tiempo real de actividades que pueden ser críticas para la salud de los usuarios, como puede ser la toma correcta de medicinas o sus hábitos de alimentación. Con este sistema se promueve la vida independiente de las personas mayores, brindando a este colectivo una mayor autonomía e independencia, permitiendo que puedan permanecer en sus hogares durante más tiempo de forma totalmente segura. El método propuesto es capaz de detectar un total de 16 actividades con una tasa de acierto del 95.42 % para esta base de datos, compuesta por un total de 33 sensores no

intrusivos durante 84 días completos de información. Para conseguir una eficacia tan elevada, la red neuronal ha utilizado datos tratados mediante las siguientes etapas: método de ventana deslizante, filtrado de filas repetidas y un módulo de agrupamiento y reorganización de dichas filas. Estas etapas suponen una mejora de la calidad de los datos tan sustancial que supone una mejora de las características del modelo que se ve reflejado en la diferencia de rendimiento con respecto a otros autores que utilizan la misma base de datos, como puede ser el trabajo realizado por D.Liciotii et al. en [210].

La tesis presenta el desarrollo de una base de datos propia (SDHAR-HOME) en la que se recogen un mayor número de actividades que la base de datos Milan para dos personas viviendo de forma simultánea en la vivienda. Esta base de datos recoge la información proporcionada por tres diferentes grupos tecnológicos: una red de sensores no intrusivos, un sistema de localización de los usuarios en la vivienda mediante balizas BLE y la información proporcionada por los sensores de pulseras de actividad. Esta base de datos se propuso ya que las principales bases de datos existentes sólo cuentan con la información de un usuario en la vivienda, hecho que no siempre sucede en las viviendas reales. Además, pocas bases de datos cuentan con un abanico tan amplio de actividades como SDHAR-HOME, ya que propone el análisis de actividades que pueden aportar mucha información sobre hábitos que pueden llegar a ser poco saludables, como ver la televisión un excesivo número de horas, higiene personal o tareas como lavar la ropa o los platos. Esta base de datos cuenta con 2 meses de información ininterrumpida. Además, los usuarios de la vivienda reciben visitas esporádicas y cuentan con una mascota que puede introducir ruido en los sensores, hecho que es beneficioso ya que reproduce las condiciones reales que podrían producirse en cualquier vivienda. De forma conjunta, la tesis plantea la aplicación de métodos de aprendizaje supervisados a SDHAR-HOME siguiendo la línea trabajada con la base de datos Milan. El sistema de predicción propuesto es capaz de alcanzar precisión en el reconocimiento del 90.91 % para el primer usuario y un 88.29 % para el segundo, lo que implica que el método elaborado tiene un buen rendimiento para la instalación propuesta. Además, este método cuenta con una amplia experimentación ya que se han realizado pruebas con distintos tipos de redes neuronales: RNN, LSTM y GRU. En la tesis se ha planteado un modelo basado en

redes neuronales convolucionales CNN en combinación con el método CVV-SV. Este modelo ofrece mejores resultados que los obtenidos mediante las redes RNN, LSTM y GRU para ambos usuarios. Se han probado diferentes filtros y configuraciones para alcanzar el mayor porcentaje de precisión posible. De forma adicional, se ha elaborado un modelo de predicción mediante la aplicación de redes neuronales transformers con capas de atención. Durante la fase de experimentación, se han probado 4 modelos distintos en función del tipo de mapeo: lineal, GRU bidireccional, LSTM bidireccional y CNN. De las pruebas se ha podido comprobar que el mapeo lineal ha tenido una mayor precisión para el usuario 1, y el mapeo con LSTM bidireccional ha dado mejores resultados para el usuario 2. De esta experimentación se llega a la conclusión de que para cada usuario le beneficia en mayor medida un tipo de red concreta. También se deduce que es importante realizar un correcto tratamiento de los datos antes de entrenar la red neuronal con el fin de mejorar sus prestaciones y evitar el sobreentrenamiento, de forma que sea capaz de predecir situaciones que nunca ha tenido en cuenta en la fase de entrenamiento.

De forma paralela, se ha trabajado en la elaboración de métodos de aprendizaje no supervisado, con el fin de facilitar el despliegue del sistema de monitorización en un mayor número de viviendas. Estos métodos tienen la ventaja de que no necesitan una fase de entrenamiento supervisada, evitando la fase de etiquetado de actividades previa. Para conseguir este objetivo, se ha elaborado un método de aprendizaje no supervisado basado en HMM y algoritmos de propagación y retro propagación, permitiendo el análisis de los distinta tecnología que contiene SDHAR-HOME: los sensores no intrusivos de tipo evento, de tipo ambiental y la posición de los usuarios. El método desarrollado se considera un modelo probabilístico, ya que va calculado la probabilidad de las distintas ADL's a medida que van llegando los eventos de los sensores. El método propuesto está compuesto por dos subsistemas: un HMM encargado de proporcionar de forma fiable la posición de cada usuario, mejorando las prestaciones del sistema de localización de la vivienda utilizando la información de las balizas y los sensores de presencia, y otro HMM encargado de proporcionar la ADL que está realizando cada usuario. Una ventaja que ofrece el método propuesto es que puede ampliarse para más usuarios que residan en la vivienda, ya que sólo habría que incluir un HMM más para ese usuario que vaya calculando sus probabili-

dades. El sistema propuesto es capaz de predecir las ADL's de SDHAR-HOME con una precisión del 91.68 % para el usuario 1 y un 86.42 % para el usuario 2. De estos resultados se llega la conclusión de que el método de aprendizaje no supervisado ofrece prestaciones similares que los modelos basados en redes neuronales anteriormente mencionados.

En la tesis se presenta el desarrollo de un ecosistema de inteligencia ambiental, que proporciona ayuda a los usuarios dentro de sus domicilios utilizando el sistema de reconocimiento de actividades desarrollado en la presente tesis. Este ecosistema combina la información recogida por la red de sensores no intrusivos junto con un robot social, el cual consigue acceder a aquella información a la que los sensores no son capaces de acceder. El ecosistema de inteligencia ambiental cuenta con la combinación de los siguientes componentes: el módulo HAR, el módulo de detección de métricas y anomalías y el sistema cognitivo. El robot social es capaz de proporcionar información al usuario sobre su entorno, sugiere ejercicios, juegos y ofrece la posibilidad de hacer videollamadas con familiares y/o cuidadores. La solución conjunta se encuentra instalada en 5 hogares piloto, y se espera alcanzar la cifra de 100 hogares al finalizar el año 2024.

7.2. Aplicaciones del método propuesto

Los sistemas de reconocimiento de actividades de personas mayores propuestos en la presente tesis son útiles para fomentar la vida independiente de este grupo de población, ya que mejoran su calidad de vida y su seguridad. Capturando los patrones de comportamiento de cada usuario de la vivienda, se pueden establecer alarmas ante situaciones indeseadas. Por ejemplo, si normalmente el usuario se levanta a las 9 de la mañana, y el sistema detecta que son las 12 y aún no se ha levantado, puede ser un indicador de que la persona está enferma. Otro ejemplo puede ser la detección de salidas a horas intempestivas de la vivienda, permitiendo de esta manera la generación de una señal de alarma al cuidador o familiar de la persona. Gracias al funcionamiento de los sistemas en tiempo real, permiten una actuación a tiempo ante una situación no deseada. Otros patrones de comportamiento son indicadores de situaciones de depresión y/o soledad, como puede ser el caso de deambulaciones

nocturnas por la vivienda o que el usuario vea la televisión en la madrugada. Detectar estas situaciones a tiempo permiten una intervención rápida para solventar el problema.

Otra aplicación que pueden tener estos sistemas es la detección de peligros en la vivienda sin tener relación con las actividades de los usuarios. Por ejemplo, gracias a la tecnología implementada, es posible detectar incendios en la cocina o inundaciones en el cuarto de baño. Aplicando los sensores adecuados, es posible detectar a tiempo otras situaciones peligrosas que no son visibles a los usuarios, como los incrementos de monóxido de carbono en la vivienda causados por malas combustiones o escapes de gas natural. Detectar estas situaciones a tiempo pueden permitir que el sistema avise a las autoridades locales para poder realizar una intervención temprana en la vivienda.

7.3. Trabajos futuros

Como posibles trabajos futuros, sería interesante contar con información más precisa de los usuarios dentro de las distintas estancias de la vivienda. Es decir, no sólo conseguir de forma precisa la estancia, sino la posición dentro de la misma. Esto permitiría que el sistema de monitorización repartiese mejor las actividades a los usuarios cuando estos se encuentran en la misma sala realizando tareas distintas. Una forma de conseguir esta localización podría darse mediante la implementación de la tecnología UWB emergente, ya que proporciona una mayor precisión que los sistemas basados en BLE.

Otra posible línea futura sería la elaboración de un sistema capaz de establecer acciones como respuesta del sistema de detección de actividades de forma automática y sin necesidad de un conocimiento experto. Utilizando la información de las actividades de los usuarios, sería interesante que un sistema fuese capaz de ir generando acciones, y mediante la recogida de la utilidad de esa acción para el usuario, ese sistema podría ir cambiando el plan de actuación hasta conseguir una personalización óptima para el usuario analizado. Este tipo de actuación personalizada se puede realizar mediante la aplicación de técnicas basadas en aprendizaje por refuerzo [231].

Se propone como línea de trabajo futura la mejora en la detección de patrones

de comportamiento anómalos para obtener de forma automática cualquier posible anomalía que pueda surgir dentro del hogar. En la presente tesis se ha obtenido un patrón de comportamiento para cada una de las 5 viviendas reales analizadas durante la fase de experimentación mediante el análisis de la duración media y desviación típica para cada una de las diferentes ADL's realizadas en cada vivienda. Sin embargo, se propone automatizar la obtención del patrón de comportamiento de cada usuario para generar una acción en consecuencia si se ha detectado una anomalía.

Por último, se propone como línea de trabajo futuro la implementación de la información proporcionada por las pulseras de actividad para poder analizar de forma paralela los patrones de ejercicio físico del usuario. Mediante el análisis de la información proporcionada por los acelerómetros y los giroscopios de las pulseras, se puede detectar el grado de sedentarismo del usuario, pudiendo inferir las horas que permanece sentado, tumbado o moviéndose por la casa. Además, la información de estos acelerómetros y giroscopios puede resultar de utilidad a la hora de detectar alguna ADL. Por ejemplo, la actividad de tareas del hogar requiere un mayor movimiento del usuario, mientras que durante la actividad de dormir, el usuario apenas va a realizar movimientos.

Glosario

- **ADL:** Actividades de la vida diaria, del inglés *Activity Daily Living*. Actividades comunes que desempeñan las personas en sus casas, como comer, dormir o ducharse.
- **BLE:** Bluetooth de baja energía, del inglés *Bluetooth Low Energy*. Tecnología de red inalámbrica de área personal que opera en 2.4 GHz con una tasa de transferencia de 1 Mbps en la capa física. Está diseñado para proporcionar un bajo consumo de energía.
- **CNN:** Red neuronal convolucional, del inglés *Convolutional Neural Network*. Redes compuestas por múltiples capas de filtros de convolución de una o más dimensiones, cuya fase de extracción de características se asemeja al proceso estimulante en las células de la corteza visual de los humanos.
- **CRF:** Campo aleatorio condicional, del inglés *Conditional Random Field*. Modelos discriminativos utilizados para segmentar y etiquetar secuencias de datos para realizar predicciones estructuradas.
- **CVV:** Votación por validación cruzada, del inglés *Cross Validation Voting*. Método de inferencia en el que se tienen en cuenta múltiples modelos entrenados en paralelo y se escoge el recorrido óptimo mediante votación.
- **DL:** Aprendizaje profundo, del inglés *Deep Learning*. Algoritmos de procesamiento de datos no estructurados que automatizan la extracción de características, suprimiendo la dependencia de expertos.
- **DNN:** Redes neuronales profundas, del inglés *Deep Neural Networks*. Red neuronales con múltiples capas ocultas entre las capas de entrada y salida, y que

puede modelar sistemas no lineales complejos.

- **GPRS:** Servicio general de paquetes vía radio, del inglés *General Packet Radio Service*. Servicio creado en la década de los 80 que permite fijar conexiones con Internet de forma inalámbrica. Este servicio está orientado a radio-enlaces que da mejor rendimiento a la conmutación de paquetes en dichos enlaces.
- **GPS:** Sistemas de Posicionamiento Global, del inglés *Global Positioning Systems*. Red de 24 satélites a 20200 km de altura sobre la Tierra con trayectorias sincronizadas.
- **GPU:** Unidad de Procesamiento Gráfico, del inglés *Graphics Processing Unit*. Procesador gráfico formado por un elevado número de pequeños núcleos especializados que proporciona un alto rendimiento al procesar tareas que pueden descomponerse en varios grupos en paralelo.
- **GRU:** Red neuronal de unidades recurrentes con compuertas, del inglés *Gated Recurrent Unit*.
- **HAR:** Reconocimiento de actividades humanas, del inglés *Human Activity Recognition*.
- **HMM:** Modelos ocultos de Markov, del inglés *Hidden Markov Models*. Modelos probabilísticos cuyos parámetros se obtienen a partir de efectos observables.
- **HTTP:** Protocolo de transferencia de hipertexto, del inglés *Hypertext Transfer Protocol*. Protocolo basado en el principio de cliente-servidor, proporciona conexiones únicas, cuenta con multiplexación y elimina información redundante.
- **JCR:** Informes de citas de revistas, del inglés *Journal Citation Reports*. Es un índice de citas inventado en el año 1963 por E.Garfield, con el fin de proporcionar una ayuda a la selección de revistas con mayor relevancia.
- **LSTM:** Red neuronal con memoria a largo o a corto plazo, del inglés *Long Short-Term Memory*.

- **ML:** Aprendizaje automático, del inglés *Machine Learning*. Se trata de una rama de la inteligencia artificial que se centra en el uso de algoritmos y el análisis de datos para construir modelos que se encarguen de una tarea o conjunto de tareas concretas.
- **MQTT:** Del inglés *Message Queuing Telemetry Transport*. Protocolo de mensajería basado en TCP/IP que utiliza la topología publicador/suscriptor especialmente utilizado en IoT debido a su sencillez y ligereza.
- **NFC:** Comunicación de campo cercano, del inglés *Near Field Communication*. Se trata de una tecnología inalámbrica que funciona en la banda de los 13.56 MHz con una tasa de transferencia de 424 kbit/s con un alcance reducido, ya que su funcionamiento se reduce a los 20 cm.
- **NLP:** Procesamiento del lenguaje natural, del inglés *Natural Language Processing*. El objetivo del procesamiento del lenguaje natural es que los equipos informáticos aprendan, generen e interpreten el lenguaje humano de forma natural, permitiendo generar aplicaciones de traducción automática, generación de texto o extracción de información.
- **NUC:** Siguiendo unidad de computación, del inglés *Next Unit of Computing*. se refiere a una categoría de computadoras compactas diseñadas por Intel. Estas máquinas suelen ser pequeñas en tamaño pero potentes en rendimiento, ofreciendo capacidades de computación comparables a las de computadoras de escritorio más grandes. Debido a su versatilidad y capacidad de integración en espacios reducidos, son utilizados en una gran variedad de aplicaciones.
- **PIR:** Pasivo infrarrojo, del inglés *Passive Infrared*. Se trata de una tecnología que es capaz de detectar las variaciones de radiación infrarroja del medio ambiente que cubre. Se utiliza como sensor para detectar presencia en determinadas zonas.
- **RFID:** Identificación por radiofrecuencia, del inglés *Radio Frequency Identification*. Se trata de una tecnología que es capaz de identificar objetos mediante el uso de ondas de radio.

- **RGB:** Rojo-Verde-Azul, del inglés *Red-Green-Blue*. Se trata de un modelo de generación de colores, ya que mediante la combinación binaria de diferentes intensidades de estos tres colores, es posible generar el abanico completo de colores visibles.
- **RNN:** Red neuronal recurrente, del inglés *Recurrent Neural Network*.
- **ROC:** Característica operativa del receptor, del inglés *Receiver Operating Characteristic*. Herramienta estadística utilizada para analizar la capacidad de clasificación de un determinado modelo.
- **SOAP:** Protocolo simple de acceso a objetos, del inglés *Simple Object Access Protocol*. Protocolo estándar creado por Dave Winer en 1998, el cual define que dos objetos funcionando en diferentes procesos pueden comunicarse mediante el intercambio de datos XML.
- **SVM:** Máquinas de vectores de soporte, del inglés *Support Vector Machines*. Se corresponden con un algoritmo de aprendizaje supervisado extensamente utilizado en problemas de regresión y clasificación. Un ejemplo de aplicación de estos algoritmos puede ser los sistemas de reconocimiento de voz, imágenes o sistemas de procesamiento del lenguaje natural.
- **UWB:** Tecnología de banda ultraancha, del inglés *Ultra-wideband*. Tecnología de comunicación de corto alcance que ofrece una elevada precisión a distancias menores de 10 cm. Funciona a altas frecuencias (mayor de 2.5 GHz) con bajo consumo de energía.
- **WLAN:** Red de área local inalámbrica, del inglés *Wireless Local Area Network*. Tecnología de radio inalámbrica con conexión punto a punto, sin interferencias con otros protocolos inalámbricos que se utilizan para la comunicación entre dispositivos dentro de una misma organización.

Bibliografía

- [1] Athanasios Lentzas y Dimitris Vrakas. Non-intrusive human activity recognition and abnormal behavior detection on elderly people: A review. *Artificial Intelligence Review*, 53(3):1975–2021, 2020.
- [2] Damien Bouchabou, Sao Mai Nguyen, Christophe Lohr, Benoit LeDuc, y Ioannis Kanellos. A survey of human activity recognition in smart homes based on iot sensors algorithms: Taxonomies, challenges, and opportunities with deep learning. *Sensors*, 21(18):6037, 2021.
- [3] Roshan Singh, Ankur Sonawane, y Rajeev Srivastava. Recent evolution of modern datasets for human activity recognition: A deep survey. *Multimedia Systems*, 26(2):83–106, 2020.
- [4] Aziz Khelalef, Fakhreddine Ababsa, y Nabil Benoudjit. An efficient human activity recognition technique based on deep learning. *Pattern Recognition and Image Analysis*, 29(4):702–715, 2019.
- [5] Zawar Hussain, Quan Z Sheng, y Wei Emma Zhang. A review and categorization of techniques on device-free human activity recognition. *Journal of Network and Computer Applications*, 167:102738, 2020.
- [6] Emiro De-La-Hoz-Franco, Paola Ariza-Colpas, Javier Medina Quero, y Macarena Espinilla. Sensor-based datasets for human activity recognition—a systematic review of literature. *IEEE Access*, 6:59192–59210, 2018.
- [7] Luis Cobo Hurtado, Pablo Francisco Viñas, Eduardo Zalama, Jaime Gómez-García-Bermejo, José María Delgado, y Beatriz Vielba García. Development

- and usability validation of a social robot platform for physical and cognitive stimulation in elder care facilities. In *Healthcare*, volumen 9, página 1067. MDPI, 2021.
- [8] Negar Golestani y Mahta Moghaddam. Human activity recognition using magnetic induction-based motion signals and deep recurrent neural networks. *Nature communications*, 11(1):1–11, 2020.
- [9] Chinmay Sawant. Human activity recognition with openpose and long short-term memory on real time images. *EasyChair Preprint*, (2297), 2020.
- [10] Minhyuk Jung y Seokho Chi. Human activity classification based on sound recognition and residual convolutional neural network. *Automation in Construction*, 114:103177, 2020.
- [11] Wesllen Sousa Lima, Eduardo Souto, Khalil El-Khatib, Roozbeh Jalali, y Joao Gama. Human activity recognition using inertial sensors in a smartphone: An overview. *Sensors*, 19(14):3213, 2019.
- [12] Macarena Espinilla, Javier Medina, y Chris Nugent. Ucaml cup. analyzing the uja human activity recognition dataset of activities of daily living. *Multidisciplinary Digital Publishing Institute Proceedings*, 2(19):1267, 2018.
- [13] Sakorn Mekruksavanich, Chanon Promsakon, y Anuchit Jitpattanakul. Location-based daily human activity recognition using hybrid deep learning network. In *2021 18th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, páginas 1–5. IEEE, 2021.
- [14] Encuesta europea de salud en españa. https://www.sanidad.gob.es/estadEstudios/estadisticas/EncuestaEuropea/EncuestaEuropea2020/EESE2020_inf_evol_princip_result.pdf, 2020 (accessed January 24, 2024).
- [15] Raúl Gómez Ramos, Jaime Duque Domingo, Eduardo Zalama, y Jaime Gómez-García-Bermejo. Daily human activity recognition using non-intrusive sensors. *Sensors*, 21(16), 2021. ISSN 1424-8220. doi: 10.3390/s21165270. URL <https://www.mdpi.com/1424-8220/21/16/5270>.

- [16] Raúl Gómez Ramos, Jaime Duque Domingo, Eduardo Zalama, Jaime Gómez-García-Bermejo, y Joaquín López. Sdhar-home: A sensor dataset for human activity recognition at home. *Sensors*, 22(21), 2022. ISSN 1424-8220. doi: 10.3390/s22218109. URL <https://www.mdpi.com/1424-8220/22/21/8109>.
- [17] Raúl Gómez-Ramos, Jaime Duque-Domingo, Eduardo Zalama, y Jaime Gómez-García-Bermejo. An unsupervised method to recognise human activity at home using non-intrusive sensors. *Electronics*, 12(23), 2023. ISSN 2079-9292. doi: 10.3390/electronics12234772. URL <https://www.mdpi.com/2079-9292/12/23/4772>.
- [18] M^a Tortosa Chuliá y Gerdt Sundström. El cohousing senior en españa: Cambios desde la economía social en los alojamientos y en la economía de los cuidados para personas mayores [senior cohousing in spain: Changes from the social economy in accommodation and in economy of care for older people]. *CIRIEC-España: Revista de Economía Pública, Social y Cooperativa*, (104):303–331, 2022.
- [19] Rosa María Flores Martínez, Sagrario Garay Villegas, et al. Soledad en el curso de vida de las mujeres mayores en méxico y españa. 2021.
- [20] Nazaret Martínez-Heredia y Esther Santaella Rodríguez. Uso de las tic para mejorar la atención en personas mayores. *Tecnología y educación en tiempos de cambio*, página 208.
- [21] Allah Bux, Plamen Angelov, y Zulfiqar Habib. Vision based human activity recognition: a review. In *Advances in Computational Intelligence Systems: Contributions Presented at the 16th UK Workshop on Computational Intelligence, September 7–9, 2016, Lancaster, UK*, páginas 341–371. Springer, 2017.
- [22] Ong Chin Ann y Lau Bee Theng. Human activity recognition: A review. In *2014 IEEE international conference on control system, computing and engineering (ICCSC 2014)*, páginas 389–393. IEEE, 2014.
- [23] Johannes Preis, Moritz Kessel, Martin Werner, y Claudia Linnhoff-Popien.

- Gait recognition with kinect. In *1st international workshop on kinect in pervasive computing*, volumen 14. New Castle, UK, 2012.
- [24] Kouros Khoshelham. Accuracy analysis of kinect depth data. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 38:133–138, 2012.
- [25] Tao Zhou, Deng-Ping Fan, Ming-Ming Cheng, Jianbing Shen, y Ling Shao. Rgb-d salient object detection: A survey. *Computational Visual Media*, 7:37–69, 2021.
- [26] Zhengyou Zhang. Microsoft kinect sensor and its effect. *IEEE multimedia*, 19(2):4–10, 2012.
- [27] Mi Zhang y Alexander A Sawchuk. Motion primitive-based human activity recognition using a bag-of-features approach. In *Proceedings of the 2nd ACM SIGHIT international health informatics symposium*, páginas 631–640, 2012.
- [28] L. Minh Dang, Kyungbok Min, Hanxiang Wang, Md. Jalil Piran, Cheol Hee Lee, y Hyeonjoon Moon. Sensor-based and vision-based human activity recognition: A comprehensive survey. *Pattern Recognition*, 108:107561, 2020. ISSN 0031-3203. doi: <https://doi.org/10.1016/j.patcog.2020.107561>. URL <https://www.sciencedirect.com/science/article/pii/S0031320320303642>.
- [29] L. Minh Dang, Md. Jalil Piran, Dongil Han, Kyungbok Min, y Hyeonjoon Moon. A survey on internet of things and cloud computing for healthcare. *Electronics*, 8(7), 2019. ISSN 2079-9292. doi: 10.3390/electronics8070768. URL <https://www.mdpi.com/2079-9292/8/7/768>.
- [30] Wei Wang, Alex X Liu, Muhammad Shahzad, Kang Ling, y Sanglu Lu. Device-free human activity recognition using commercial wifi devices. *IEEE Journal on Selected Areas in Communications*, 35(5):1118–1131, 2017.
- [31] Wenjie Ruan, Quan Z Sheng, Lina Yao, Xue Li, Nickolas JG Falkner, y Lei Yang. Device-free human localization and tracking with uhf passive rfid tags: A data-driven approach. *Journal of Network and Computer Applications*, 104: 78–96, 2018.

- [32] Nirmalya Roy, Archan Misra, y Diane Cook. Ambient and smartphone sensor assisted adl recognition in multi-inhabitant smart environments. *Journal of ambient intelligence and humanized computing*, 7:1–19, 2016.
- [33] Nabil Zerrouki, Fouzi Harrou, Ying Sun, y Amrane Houacine. Vision-based human action classification using adaptive boosting algorithm. *IEEE Sensors Journal*, 18(12):5115–5121, 2018. doi: 10.1109/JSEN.2018.2830743.
- [34] Juan C. Nunez, Raul Cabido, Juan J. Pantrigo, Antonio S. Montemayor, y Jose F. Velez. Convolutional neural networks and long short-term memory for skeleton-based human activity and hand gesture recognition. *Pattern Recognition*, 76:80–94, 2018. ISSN 0031-3203. doi: <https://doi.org/10.1016/j.patcog.2017.10.033>. URL <https://www.sciencedirect.com/science/article/pii/S0031320317304405>.
- [35] Stephen S Intille, Kent Larson, JS Beaudin, Jason Nawyn, E Munguia Tapia, y Pallavi Kaushik. A living laboratory for the design and evaluation of ubiquitous computing technologies. In *CHI'05 extended abstracts on Human factors in computing systems*, páginas 1941–1944, 2005.
- [36] Hande Alemdar, Halil Ertan, Ozlem Durmaz Incel, y Cem Ersoy. Aras human activity datasets in multiple homes with multiple residents. In *2013 7th International Conference on Pervasive Computing Technologies for Healthcare and Workshops*, páginas 232–235. IEEE, 2013.
- [37] Diane J Cook, Aaron S Crandall, Brian L Thomas, y Narayanan C Krishnan. Casas: A smart home in a box. *Computer*, 46(7):62–69, 2012.
- [38] Vijay Anant Athavale, Suresh Chand Gupta, Deepak Kumar, y Savita Savita. Human action recognition using cnn-svm model. *Advances in Science and Technology*, 105:282–290, 2021.
- [39] Talal Alshammari, Nasser Alshammari, Mohamed Sedky, y Chris Howard. Simadl: Simulated activities of daily living dataset. *Data*, 3(2):11, 2018.

- [40] Tim Van Kasteren, Athanasios Noulas, Gwenn Englebienne, y Ben Kröse. Accurate activity recognition in a home setting. In *Proceedings of the 10th international conference on Ubiquitous computing*, páginas 1–9, 2008.
- [41] Paula Lago, Frédéric Lang, Claudia Roncancio, Claudia Jiménez-Guarín, Radu Mateescu, y Nicolas Bonnefond. The contextact@ a4h real-life dataset of daily-living activities: Activity recognition using model checking. In *Modeling and Using Context: 10th International and Interdisciplinary Conference, CONTEXT 2017, Paris, France, June 20-23, 2017, Proceedings 10*, páginas 175–188. Springer, 2017.
- [42] Jaime Duque Domingo, Jaime Gómez-García-Bermejo, y Eduardo Zalama. Improving human activity recognition integrating lstm with different data sources: Features, object detection and skeleton tracking. *IEEE Access*, 10:68213–68230, 2022.
- [43] L Minh Dang, Kyungbok Min, Hanxiang Wang, Md Jalil Piran, Cheol Hee Lee, y Hyeonjoon Moon. Sensor-based and vision-based human activity recognition: A comprehensive survey. *Pattern Recognition*, 108:107561, 2020.
- [44] Weiya Chen, Chenchen Yu, Chenyu Tu, Zehua Lyu, Jing Tang, Shiqi Ou, Yan Fu, y Zhidong Xue. A survey on hand pose estimation with wearable sensors and computer-vision-based methods. *Sensors*, 20(4):1074, 2020.
- [45] Wazha Mmereki, Rodrigo S Jamisola, Dimane Mpoeleng, y Tinao Petso. Yolov3-based human activity recognition as viewed from a moving high-altitude aerial camera. In *2021 7th International Conference on Automation, Robotics and Applications (ICARA)*, páginas 241–246. IEEE, 2021.
- [46] Shaohua Wan, Lianyong Qi, Xiaolong Xu, Chao Tong, y Zonghua Gu. Deep learning models for real-time human activity recognition with smartphones. *Mobile Networks and Applications*, 25:743–755, 2020.
- [47] Djamila Romaiassa Beddiar, Brahim Nini, Mohammad Sabokrou, y Abdenour Hadid. Vision-based human activity recognition: a survey. *Multimedia Tools and Applications*, 79(41-42):30509–30555, 2020.

- [48] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, y Gang Wang. Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, páginas 1010–1019, 2016.
- [49] Kishore K Reddy y Mubarak Shah. Recognizing 50 human action categories of web videos. *Machine vision and applications*, 24(5):971–981, 2013.
- [50] Jingen Liu, Jiebo Luo, y Mubarak Shah. Recognizing realistic actions from videos “in the wild”. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, páginas 1996–2003. IEEE, 2009.
- [51] Christian Schuldt, Ivan Laptev, y Barbara Caputo. Recognizing human actions: a local svm approach. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volumen 3, páginas 32–36. IEEE, 2004.
- [52] Moshe Blank, Lena Gorelick, Eli Shechtman, Michal Irani, y Ronen Basri. Actions as space-time shapes. In *Tenth IEEE International Conference on Computer Vision (ICCV’05) Volume 1*, volumen 2, páginas 1395–1402. IEEE, 2005.
- [53] Daniel Weinland, Edmond Boyer, y Remi Ronfard. Action recognition from arbitrary views using 3d exemplars. In *2007 IEEE 11th International Conference on Computer Vision*, páginas 1–7. IEEE, 2007.
- [54] Mikel D Rodriguez, Javed Ahmed, y Mubarak Shah. Action mach a spatio-temporal maximum average correlation height filter for action recognition. In *2008 IEEE conference on computer vision and pattern recognition*, páginas 1–8. IEEE, 2008.
- [55] Preksha Pareek y Ankit Thakkar. A survey on video-based human action recognition: recent updates, datasets, challenges, and applications. *Artificial Intelligence Review*, 54:2259–2322, 2021.
- [56] Graham Thomas, Rikke Gade, Thomas B Moeslund, Peter Carr, y Adrian Hilton. Computer vision for sports: Current applications and research topics. *Computer Vision and Image Understanding*, 159:3–18, 2017.

- [57] Amin Ullah, Khan Muhammad, Ijaz Ul Haq, y Sung Wook Baik. Action recognition using optimized deep autoencoder and cnn for surveillance data streams of non-stationary environments. *Future Generation Computer Systems*, 96: 386–397, 2019.
- [58] Jian Han Lim, Eng Yeow Teh, Ming Han Geh, y Chern Hong Lim. Automated classroom monitoring with connected visioning system. In *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, páginas 386–393. IEEE, 2017.
- [59] Peng Wang, Yuanzhouhan Cao, Chunhua Shen, Lingqiao Liu, y Heng Tao Shen. Temporal pyramid pooling-based convolutional neural network for action recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(12):2613–2622, 2016.
- [60] Shibo Zhang, Yaxuan Li, Shen Zhang, Farzad Shahabi, Stephen Xia, Yu Deng, y Nabil Alshurafa. Deep learning in human activity recognition with wearable sensors: A review on advances. *Sensors*, 22(4):1476, 2022.
- [61] Wearable technology database. <https://data.world/crowdfunder/wearable-technology-database>, 2022 (accessed January 15, 2024).
- [62] Rex Liu, Albara Ah Ramli, Huanle Zhang, Erik Henricson, y Xin Liu. An overview of human activity recognition using wearable sensors: Healthcare and artificial intelligence. In *International Conference on Internet of Things*, páginas 1–14. Springer, 2021.
- [63] Isah A Lawal y Sophia Bano. Deep human activity recognition with localisation of wearable sensors. *IEEE Access*, 8:155060–155070, 2020.
- [64] Mi Zhang y Alexander A Sawchuk. Usc-had: A daily activity dataset for ubiquitous activity recognition using wearable sensors. In *Proceedings of the 2012 ACM conference on ubiquitous computing*, páginas 1036–1043, 2012.
- [65] Emmanuel Munguia Tapia, Stephen S Intille, Louis Lopez, y Kent Larson. The design of a portable kit of wireless sensors for naturalistic data collection. In

- International conference on pervasive computing*, páginas 117–134. Springer, 2006.
- [66] Daniel Roggen, Alberto Calatroni, Mirco Rossi, Thomas Holleczeck, Kilian Förster, Gerhard Tröster, Paul Lukowicz, David Bannach, Gerald Pirkl, Alois Ferscha, et al. Collecting complex activity datasets in highly rich networked sensor environments. In *2010 Seventh international conference on networked sensing systems (INSS)*, páginas 233–240. IEEE, 2010.
- [67] Luca Arrotta, Claudio Bettini, y Gabriele Civitarese. The marble dataset: Multi-inhabitant activities of daily living combining wearable and environmental sensors data. In *International Conference on Mobile and Ubiquitous Systems: Computing, Networking, and Services*, páginas 451–468. Springer, 2021.
- [68] Olukunle Ojetola, Elena Gaura, y James Brusey. Data set for fall events and daily activities from inertial sensors. In *Proceedings of the 6th ACM multimedia systems conference*, páginas 243–248, 2015.
- [69] Marco Ruzzon, Alessandro Carfi, Takahiro Ishikawa, Fulvio Mastrogiovanni, y Toshiyuki Murakami. A multi-sensory dataset for the activities of daily living. *Data in brief*, 32:106122, 2020.
- [70] Hui Liu, Yale Hartmann, y Tanja Schultz. Csl-share: A multimodal wearable sensor-based human activity dataset, 2021.
- [71] Xiaomu Luo, Tong Liu, Baihua Shen, Liwen Gao, Xiaoyan Luo, et al. Human indoor localization based on ceiling mounted pir sensor nodes. In *2016 13th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, páginas 868–874. IEEE, 2016.
- [72] K Deepa, Nebojsa Bacanin, SS Askar, y Mohamed Abouhawwash. Elderly and visually impaired indoor activity monitoring based on wi-fi and deep hybrid convolutional neural network. *Scientific Reports*, 13(1):22470, 2023.
- [73] Lazzaro di Biase, Pasquale Maria Pecoraro, Giovanni Pecoraro, Maria Letizia Caminiti, y Vincenzo Di Lazzaro. Markerless radio frequency indoor monito-

- ring for telemedicine: gait analysis, indoor positioning, fall detection, tremor analysis, vital signs and sleep monitoring. *Sensors*, 22(21):8486, 2022.
- [74] Frank M. Clegg, Margaret Sears, Margaret Friesen, Theodora Scarato, Rob Metzinger, Cindy Russell, Alex Stadtner, y Anthony B. Miller. Building science and radiofrequency radiation: What makes smart and healthy buildings. *Building and Environment*, 176:106324, 2020. ISSN 0360-1323. doi: <https://doi.org/10.1016/j.buildenv.2019.106324>. URL <https://www.sciencedirect.com/science/article/pii/S0360132319305347>.
- [75] Luigi Bibbò, Riccardo Carotenuto, y Francesco Della Corte. An overview of indoor localization system for human activity recognition (har) in healthcare. *Sensors*, 22(21):8119, 2022.
- [76] F Serhan Daniş y Ali Taylan Cemgil. Model-based localization and tracking using bluetooth low-energy beacons. *Sensors*, 17(11):2484, 2017.
- [77] Priya Roy y Chandreyee Chowdhury. A survey on ubiquitous wifi-based indoor localization system for smartphone users from implementation perspectives. *CCF Transactions on Pervasive Computing and Interaction*, 4(3):298–318, 2022.
- [78] Yuri Alvarez y Fernando Las Heras. Zigbee-based sensor network for indoor location and tracking applications. *IEEE Latin America Transactions*, 14(7):3208–3214, 2016.
- [79] Stefania Monica y Federico Bergenti. Hybrid indoor localization using wifi and uwb technologies. *Electronics*, 8(3):334, 2019.
- [80] Jaime Duque Domingo, Jaime Gómez-García-Bermejo, Eduardo Zalama, Carlos Cerrada, y Enrique Valero. Integration of computer vision and wireless networks to provide indoor positioning. *Sensors*, 19(24):5495, 2019.
- [81] Aigerim Mussina y Sanzhar Aubakirov. Rssi based bluetooth low energy indoor positioning. In *2018 IEEE 12th International Conference on Application of Information and Communication Technologies (AICT)*, páginas 1–4. IEEE, 2018.

- [82] Chen Feng, Wain Sy Anthea Au, Shahrokh Valaee, y Zhenhui Tan. Received-signal-strength-based indoor positioning using compressive sensing. *IEEE Transactions on mobile computing*, 11(12):1983–1993, 2011.
- [83] Jun Ma, Xuansong Li, Xianping Tao, y Jian Lu. Cluster filtered knn: A wlan-based indoor positioning scheme. In *2008 International Symposium on a World of Wireless, Mobile and Multimedia Networks*, páginas 1–8. IEEE, 2008.
- [84] Tong Ji, Wenhao Li, Xiangcheng Zhu, y Mengshuang Liu. Survey on indoor fingerprint localization for ble. In *2022 IEEE 6th Information Technology and Mechatronics Engineering Conference (ITOEC)*, volumen 6, páginas 129–134. IEEE, 2022.
- [85] Xiaoyong Chai y Qiang Yang. Reducing the calibration effort for probabilistic indoor location estimation. *IEEE Transactions on Mobile Computing*, 6(6): 649–662, 2007.
- [86] Petros Spachos y Konstantinos N Plataniotis. Ble beacons for indoor positioning at an interactive iot-based smart museum. *IEEE Systems Journal*, 14(3): 3483–3493, 2020.
- [87] Raul Montoliu, Emilio Sansano, Arturo Gascó, Oscar Belmonte, y Antonio Caballer. Indoor positioning for monitoring older adults at home: Wi-fi and ble technologies in real scenarios. *Electronics*, 9(5):728, 2020.
- [88] Beakcheol Jang, Hyunjung Kim, y Jong wook Kim. Survey of landmark-based indoor positioning technologies. *Information Fusion*, 89:166–188, 2023.
- [89] Óscar Belmonte-Fernández, Adrian Puertas-Cabedo, Joaquín Torres-Sospedra, Raúl Montoliu-Colás, y Sergi Trilles-Oliver. An indoor positioning system based on wearables for ambient-assisted living. *Sensors*, 17(1):36, 2016.
- [90] Joaquín Torres, Óscar Belmonte, Raúl Montoliu, Sergio Trilles, y Andrea Calia. How feasible is wifi fingerprint-based indoor positioning for in-home monitoring? In *2016 12th International Conference on Intelligent Environments (IE)*, páginas 68–75. IEEE, 2016.

- [91] Alwin Poulouse y Dong Seog Han. Hybrid deep learning model based indoor positioning using wi-fi rssi heat maps for autonomous applications. *Electronics*, 10(1):2, 2020.
- [92] Mai Ibrahim, Marwan Torki, y Mustafa ElNainay. Cnn based indoor localization using rssi time-series. In *2018 IEEE symposium on computers and communications (ISCC)*, páginas 01044–01049. IEEE, 2018.
- [93] Kyeong Soo Kim, Sanghyuk Lee, y Kaizhu Huang. A scalable deep neural network architecture for multi-building and multi-floor indoor localization based on wi-fi fingerprinting. *Big Data Analytics*, 3:1–17, 2018.
- [94] Yan Wang, Xin Wang, Damla Arifoglu, Chenggang Lu, Abdelhamid Bouchachia, Yingrui Geng, y Ge Zheng. A survey on ambient sensor-based abnormal behaviour detection for elderly people in healthcare. *Electronics*, 12(7):1539, 2023.
- [95] Hossein Raeis, Mohammad Kazemi, y Shervin Shirmohammadi. Human activity recognition with device-free sensors for well-being assessment in smart homes. *IEEE Instrumentation & Measurement Magazine*, 24(6):46–57, 2021.
- [96] Ramesh Paudel, William Eberle, y Lawrence B Holder. Anomaly detection of elderly patient activities in smart homes using a graph-based approach. In *Proceedings of the 2018 International Conference on Data Science*, páginas 163–169, 2018.
- [97] Marek Novák, František Jakab, y Luis Lain. Anomaly detection in user daily patterns in smart-home environment. *J. Sel. Areas Health Inform*, 3(6):1–11, 2013.
- [98] Abdur Rahim Mohammad Forkan, Ibrahim Khalil, Zahir Tari, Sebti Foufou, y Abdelaziz Bouras. A context-aware approach for long-term behavioural change detection and abnormality prediction in ambient assisted living. *Pattern Recognition*, 48(3):628–641, 2015.
- [99] Narges Manouchehri y Nizar Bouguila. Human activity recognition with an hmm-based generative model. *Sensors*, 23(3):1390, 2023.

- [100] Yang Li, Guanci Yang, Zhidong Su, Shaobo Li, y Yang Wang. Human activity recognition based on multienvironment sensor data. *Information Fusion*, 91: 47–63, 2023.
- [101] Alexandros Konios, Matias Garcia-Constantino, Stavros-Richard Christopoulos, Mustafa Asan Mustafa, Idongesit Ekerete, Colin Shewell, Chris Nugent, y Gareth Morrison. Probabilistic analysis of temporal and sequential aspects of activities of daily living for abnormal behaviour detection. In *2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UI-C/ATC/CBDCOM/IOP/SCI)*, páginas 723–730. IEEE, 2019.
- [102] Fco Javier Ordóñez, Paula de Toledo, y Araceli Sanchis. Sensor-based bayesian detection of anomalous living patterns in a home setting. *Personal and Ubiquitous Computing*, 19:259–270, 2015.
- [103] Yonghua Yu, Changyang Li, Mainuddin Ahmad Jonas, Chuanhao Ma, Fay-sal Hossain Shezan, Siyuan Shen, Peng Gao, y Yuan Tian. Detecting abnormal behaviors in smart home. In *2019 IEEE 16th International Conference on Mobile Ad Hoc and Sensor Systems Workshops (MASSW)*, páginas 37–42. IEEE, 2019.
- [104] Yala Nawal, Mourad Oussalah, Belkacem Fergani, y Anthony Fleury. New incremental svm algorithms for human activity recognition in smart homes. *Journal of Ambient Intelligence and Humanized Computing*, 14(10):13433–13450, 2023.
- [105] Jens Lundström, Eric Järpe, y Antanas Verikas. Detecting and exploring deviating behaviour of smart home residents. *Expert Systems with Applications*, 55:429–440, 2016.
- [106] S Seth Long y Lawrence B Holder. Using graphs to improve activity prediction in smart environments based on motion sensor data. In *Toward Useful Services for Elderly and People with Disabilities: 9th International Conference on Smart*

- Homes and Health Telematics, ICOST 2011, Montreal, Canada, June 20-22, 2011. Proceedings 9*, páginas 57–64. Springer, 2011.
- [107] Aadel Howedi, Ahmad Lotfi, y Amir Pourabdollah. An entropy-based approach for anomaly detection in activities of daily living in the presence of a visitor. *Entropy*, 22(8):845, 2020.
- [108] Manan Jethanandani, Abhishek Sharma, Thinagaran Perumal, y Jieh-Ren Chang. Multi-label classification based ensemble learning for human activity recognition in smart home. *Internet of Things*, 12:100324, 2020.
- [109] Jérémie Saives, Clément Pianon, y Gregory Faraut. Activity discovery and detection of behavioral deviations of an inhabitant from binary sensors. *IEEE Transactions on Automation Science and Engineering*, 12(4):1211–1224, 2015.
- [110] Domus dataset. <https://domus.recherche.usherbrooke.ca/>, 2024 (accessed January 16, 2024).
- [111] Diane J Cook, Michael Youngblood, Edwin O Heierman, Karthik Gopalratnam, Sira Rao, Andrey Litvin, y Farhan Khawaja. Mavhome: An agent-based smart home. In *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, 2003.(PerCom 2003)*., páginas 521–524. IEEE, 2003.
- [112] Abubaker Elbayoudi, Ahmad Lotfi, Caroline Langensiepen, y Kofi Appiah. Modelling and simulation of activities of daily living representing an older adult’s behaviour. In *Proceedings of the 8th ACM International Conference on Pervasive Technologies Related to Assistive Environments*, páginas 1–8, 2015.
- [113] Wei-Yi Cheng, Alf Scotland, Florian Lipsmeier, Timothy Kilchenmann, Liping Jin, Jens Schjodt-Eriksen, Detlef Wolf, Yan-Ping Zhang-Schaerer, Ignacio Fernandez Garcia, Juliane Siebourg-Polster, et al. Human activity recognition from sensor-based large-scale continuous monitoring of parkinson’s disease patients. In *2017 IEEE/ACM International Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE)*, páginas 249–250. IEEE, 2017.

- [114] Peter F Edemekong, Deb Bomgaars, Suresh Sukumaran, y Shoshana B Levy. Activities of daily living. 2019.
- [115] Antonio Lazaro, Marti Boada, Ramon Villarino, y David Girbau. Study on the reading of energy-harvested implanted nfc tags using mobile phones. *IEEE Access*, 8:2200–2221, 2019.
- [116] Ayanle I Ali, Sibel Zorlu Partal, Salih Kepke, y Hakan P Partal. Zigbee and lora based wireless sensors for smart environment and iot applications. In *2019 1st Global Power, Energy and Communication Conference (GPECOM)*, páginas 19–23. IEEE, 2019.
- [117] Qahhar Muhammad Qadir, Tarik A Rashid, Nawzad K Al-Salihi, Birzo Ismael, Alexander A Kist, y Zhongwei Zhang. Low power wide area networks: A survey of enabling technologies, applications and interoperability needs. *IEEE Access*, 6:77454–77473, 2018.
- [118] Srikar Meka y Benedito Fonseca Jr. Improving route selections in zigbee wireless sensor networks. *Sensors*, 20(1):164, 2019.
- [119] Bashar Alathari, Mohammed Falih Kadhim, Salam Al-Khammasi, y Nabeel Saleh Ali. A framework implementation of surveillance tracking system based on pir motion sensors. *Indonesian Journal of Electrical Engineering and Computer Science*, 13(1):235–242, 2019.
- [120] Johannes K Becker, David Li, y David Starobinski. Tracking anonymized bluetooth devices. *Proc. Priv. Enhancing Technol.*, 2019(3):50–65, 2019.
- [121] Ana de la Casa Pérez, Pedro Ángel Latorre Román, Marcos Muñoz Jiménez, Manuel Lucena Zurita, José Alberto Laredo Aguilera, Juan Antonio Párraga Montilla, y José Carlos Cabrera Linares. Is the xiaomi mi band 4 an accuracy tool for measuring health-related parameters in adults and older people? an original validation study. *International Journal of Environmental Research and Public Health*, 19(3):1593, 2022.
- [122] Sourabh R Misal, SR Prajwal, HM Niveditha, HM Vinayaka, y S Veena. Indoor positioning system (ips) using esp32, mqtt and bluetooth. In *2020*

- fourth international conference on computing methodologies and communication (ICCMC)*, páginas 79–82. IEEE, 2020.
- [123] Marek Babiuch, Petr Foltýnek, y Pavel Smutný. Using the esp32 microcontroller for data processing. In *2019 20th International Carpathian Control Conference (ICCC)*, páginas 1–6. IEEE, 2019.
- [124] S Sophia, B Maruthi Shankar, K Akshya, AR C Arunachalam, VTY Avanthika, y S Deepak. Bluetooth low energy based indoor positioning system using esp32. In *2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA)*, páginas 1698–1702. IEEE, 2021.
- [125] Home — espresense. <https://espresense.com/>, 2023 (accessed December 29, 2023).
- [126] Gregory F Welch. Kalman filter. *Computer Vision: A Reference Guide*, páginas 1–3, 2020.
- [127] Fu Chen, Yujia Huo, Jianming Zhu, y Dan Fan. A review on the study on mqtt security challenge. In *2020 IEEE International Conference on Smart Cloud (SmartCloud)*, páginas 128–133. IEEE, 2020.
- [128] José Pino-Ortega, Carlos D Gómez-Carmona, y Markel Rico-González. Accuracy of xiaomi mi band 2.0, 3.0 and 4.0 to measure step count and distance for physical activity and healthcare in adults over 65 years. *Gait & Posture*, 87: 6–10, 2021.
- [129] Meng Shang, Walter De Raedt, Carolina Varon, y Bart Vanrumste. Are gyroscopes an added value in leave-one-subject-out activity recognition with imus? In *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, páginas 2399–2402. IEEE, 2022.
- [130] Ahmet Ali Süzen, Burhan Duman, y Betül Şen. Benchmark analysis of jetson tx2, jetson nano and raspberry pi using deep-cnn. In *2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, páginas 1–5. IEEE, 2020.

- [131] Sebastián Valladares, Mayerly Toscano, Rodrigo Tufiño, Paulina Morillo, y Diego Vallejo-Huanga. Performance evaluation of the nvidia jetson nano through a real-time machine learning application. In *Intelligent Human Systems Integration 2021: Proceedings of the 4th International Conference on Intelligent Human Systems Integration (IHSI 2021): Integrating People and Intelligent Systems, February 22-24, 2021, Palermo, Italy*, páginas 343–349. Springer, 2021.
- [132] Home assistant. <https://www.home-assistant.io/>, 2022 (accessed January 04, 2024).
- [133] T Maragatham, P Balasubramanie, y M Vivekanandhan. Iot based home automation system using raspberry pi 4. In *IOP Conference Series: Materials Science and Engineering*, volumen 1055, página 012081. IOP Publishing, 2021.
- [134] Syeda Noor Zehra Naqvi, Sofia Yfantidou, y Esteban Zimányi. Time series databases and influxdb. *Studienarbeit, Université Libre de Bruxelles*, 12, 2017.
- [135] Mohammad Nasar y Mohammad Abu Kausar. Suitability of influxdb database for iot applications. *International Journal of Innovative Technology and Exploring Engineering*, 8(10):1850–1857, 2019.
- [136] Mainak Chakraborty y Ajit Pratap Kundan. Grafana. In *Monitoring Cloud-Native Applications*, páginas 187–240. Springer, 2021.
- [137] Rishav Pramanik, Ritodeep Sikdar, y Ram Sarkar. Transformer-based deep reverse attention network for multi-sensory human activity recognition. *Engineering Applications of Artificial Intelligence*, 122:106150, 2023.
- [138] Cheng Xu, Duo Chai, Jie He, Xiaotong Zhang, y Shihong Duan. Innohar: A deep neural network for complex human activity recognition. *Ieee Access*, 7: 9893–9902, 2019.
- [139] Tianyuan Liu, Hangbin Zheng, Pai Zheng, Jinsong Bao, Junliang Wang, Xiaojia Liu, y Changqi Yang. An expert knowledge-empowered cnn approach for welding radiographic image recognition. *Advanced Engineering Informatics*, 56:101963, 2023. ISSN 1474-0346.

- [140] Mohamed Hibat-Allah, Martin Ganahl, Lauren E Hayward, Roger G Melko, y Juan Carrasquilla. Recurrent neural network wave functions. *Physical Review Research*, 2(2):023358, 2020.
- [141] Jaime Duque Domingo, Eduardo Zalama, y Jaime Gómez-García-Bermejo. Improving human activity recognition integrating lstm with different data sources: Features, object detection and skeleton tracking. *IEEE Access*, accepted on jun. 2022 for publication:in press.
- [142] Sravan Kumar Challa, Akhilesh Kumar, y Vijay Bhaskar Semwal. A multi-branch cnn-bilstm model for human activity recognition using wearable sensor data. *The Visual Computer*, 38(12):4095–4109, 2022.
- [143] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, y Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- [144] Tyler W Hughes, Ian AD Williamson, Momchil Minkov, y Shanhui Fan. Wave physics as an analog recurrent neural network. *Science advances*, 5(12):eaay6946, 2019.
- [145] Sakorn Mekruksavanich y Anuchit Jitpattanakul. Lstm networks using smartphone data for sensor-based human activity recognition in smart homes. *Sensors*, 21(5):1636, 2021.
- [146] Alex Sherstinsky. Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. *Physica D: Nonlinear Phenomena*, 404: 132306, 2020.
- [147] Zhiheng Huang, Wei Xu, y Kai Yu. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*, 2015.
- [148] Martin Wöllmer, Florian Eyben, Alex Graves, Björn Schuller, y Gerhard Rigoll. Bidirectional lstm networks for context-sensitive keyword detection in a cognitive virtual agent framework. *Cognitive Computation*, 2(3):180–190, 2010.

- [149] Sanjiban Sekhar Roy, Ali Ismail Awad, Lamesgen Adugnaw Amare, Mabrie Tesfaye Erkihun, y Mohd Anas. Multimodel phishing url detection using lstm, bidirectional lstm, and gru models. *Future Internet*, 14(11):340, 2022.
- [150] Rahul Dey y Fathi M Salem. Gate-variants of gated recurrent unit (gru) neural networks. In *2017 IEEE 60th international midwest symposium on circuits and systems (MWSCAS)*, páginas 1597–1600. IEEE, 2017.
- [151] JX Chen, DM Jiang, y YN Zhang. A hierarchical bidirectional gru model with attention for eeg-based emotion classification. *IEEE Access*, 7:118530–118540, 2019.
- [152] Saeid Raziani y Mehran Azimbagirad. Deep cnn hyperparameter optimization algorithms for sensor-based human activity recognition. *Neuroscience Informatics*, 2(3):100078, 2022.
- [153] Nafiul Rashid, Berken Utku Demirel, y Mohammad Abdullah Al Faruque. Ahar: Adaptive cnn for energy-efficient human activity recognition in low-power edge devices. *IEEE Internet of Things Journal*, 9(15):13041–13051, 2022.
- [154] Altaf Hussain, Tanveer Hussain, Waseem Ullah, y Sung Wook Baik. Vision transformer and deep sequence learning for human activity recognition in surveillance videos. *Computational Intelligence and Neuroscience*, 2022, 2022.
- [155] Iveta Dirgová Luptáková, Martin Kubovčík, y Jiří Pospíchal. Wearable sensor-based human activity recognition with transformer model. *Sensors*, 22(5):1911, 2022.
- [156] Jiewen Yang, Xingbo Dong, Liujun Liu, Chao Zhang, Jiajun Shen, y Dahai Yu. Recurring the transformer for video action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, páginas 14063–14073, 2022.
- [157] Mst Alema Khatun, Mohammad Abu Yousuf, Sabbir Ahmed, Md Zia Uddin, Salem A Alyami, Samer Al-Ashhab, Hanan F Akhdar, Asaduzzaman Khan,

- Akm Azad, y Mohammad Ali Moni. Deep cnn-lstm with self-attention model for human activity recognition using wearable sensor. *IEEE Journal of Translational Engineering in Health and Medicine*, 10:1–16, 2022.
- [158] Gabriela Augustinov, Muhammad Adeel Nisar, Frédéric Li, Amir Tabatabaei, Marcin Grzegorzec, Keywan Sohrabi, y Sebastian Fudickar. Transformer-based recognition of activities of daily living from wearable sensor data. In *Proceedings of the 7th international workshop on sensor-based activity recognition and artificial intelligence*, páginas 1–8, 2022.
- [159] Xiang Chen, Hao Li, Mingqiang Li, y Jinshan Pan. Learning a sparse transformer network for effective image deraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, páginas 5896–5905, 2023.
- [160] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, y Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33: 9912–9924, 2020.
- [161] Hui Zhang, Jason E Fritts, y Sally A Goldman. Image segmentation evaluation: A survey of unsupervised methods. *computer vision and image understanding*, 110(2):260–280, 2008.
- [162] Damien Bouchabou, Sao Mai Nguyen, Christophe Lohr, Benoit LeDuc, y Ioannis Kanellos. Using language model to bootstrap human activity recognition ambient sensors based in smart homes. *Electronics*, 10(20):2498, 2021.
- [163] Kaixuan Chen, Lina Yao, Dalin Zhang, Xianzhi Wang, Xiaojun Chang, y Feiping Nie. A semisupervised recurrent convolutional attention model for human activity recognition. *IEEE transactions on neural networks and learning systems*, 31(5):1747–1756, 2019.
- [164] Kareem Eltouny, Mohamed Gomaa, y Xiao Liang. Unsupervised learning methods for data-driven vibration-based structural health monitoring: A review. *Sensors*, 23(6):3290, 2023.

- [165] Ahmed Ahmim, Leandros Maglaras, Mohamed Amine Ferrag, Makhoul Derdour, y Helge Janicke. A novel hierarchical intrusion detection system based on decision tree and rules-based models. In *2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, páginas 228–233. IEEE, 2019.
- [166] Francesco Daghero, Daniele Jahier Pagliari, y Massimo Poncino. Two-stage human activity recognition on microcontrollers with decision trees and cnns. In *2022 17th Conference on Ph. D Research in Microelectronics and Electronics (PRIME)*, páginas 173–176. IEEE, 2022.
- [167] Bhavya Mor, Sunita Garhwal, y Ajay Kumar. A systematic review of hidden markov models and their applications. *Archives of computational methods in engineering*, 28:1429–1448, 2021.
- [168] Simo Särkkä y Lennart Svensson. *Bayesian filtering and smoothing*, volumen 17. Cambridge university press, 2023.
- [169] Kei Wakabayashi y Takao Miura. Forward-backward activation algorithm for hierarchical hidden markov models. *Advances in Neural Information Processing Systems*, 25, 2012.
- [170] Aleksandar Kavcic y Jose MF Moura. The viterbi algorithm and markov noise memory. *IEEE Transactions on information theory*, 46(1):291–301, 2000.
- [171] Timothy Chadza, Konstantinos G Kyriakopoulos, y Sangarapillai Lambotharan. Analysis of hidden markov model learning algorithms for the detection and prediction of multi-stage network attacks. *Future generation computer systems*, 108:636–649, 2020.
- [172] Priscila Valdiviezo-Diaz, Fernando Ortega, Eduardo Cobos, y Raúl Lara-Cabrera. A collaborative filtering approach based on naïve bayes classifier. *IEEE Access*, 7:108581–108592, 2019.
- [173] Daniel Berrar. Bayes theorem and naïve bayes classifier. *Encyclopedia of bioinformatics and computational biology. ABC of bioinformatics*, 403:412, 2018.

- [174] Kevin P Murphy et al. Naive bayes classifiers. *University of British Columbia*, 18(60):1–8, 2006.
- [175] K Ming Leung et al. Naive bayesian classifier. *Polytechnic University Department of Computer Science/Finance and Risk Engineering*, 2007:123–156, 2007.
- [176] Alexandre Bittar y Philip N Garner. Bayesian recurrent units and the forward-backward algorithm. *arXiv preprint arXiv:2207.10486*, 2022.
- [177] Shun-Zheng Yu y Hisashi Kobayashi. An efficient forward-backward algorithm for an explicit-duration hidden markov model. *IEEE signal processing letters*, 10(1):11–14, 2003.
- [178] G David Forney. The viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278, 1973.
- [179] Nir Shlezinger, Nariman Farsad, Yonina C Eldar, y Andrea J Goldsmith. Viterbinet: A deep learning based viterbi algorithm for symbol detection. *IEEE Transactions on Wireless Communications*, 19(5):3319–3331, 2020.
- [180] N. Seshadri y C.-E.W. Sundberg. List viterbi decoding algorithms with applications. *IEEE Transactions on Communications*, 42(234):313–323, 1994. doi: 10.1109/TCOMM.1994.577040.
- [181] Jean-Louis Colaço, Bruno Pagano, y Marc Pouzet. A conservative extension of synchronous data-flow with state machines. In *Proceedings of the 5th ACM international conference on Embedded software*, páginas 173–182, 2005.
- [182] Lawrence R Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.
- [183] Lawrence Rabiner y Biinghwang Juang. An introduction to hidden markov models. *iee assp magazine*, 3(1):4–16, 1986.
- [184] Hongqing Fang y Lei He. Bp neural network for human activity recognition in smart home. In *2012 International Conference on Computer Science and Service System*, páginas 1034–1037. IEEE, 2012.

- [185] Labiba Gillani Fahad y Syed Fahad Tahir. Activity recognition and anomaly detection in smart homes. *Neurocomputing*, 423:362–372, 2021.
- [186] Jian Wang y Zewen Cao. Chinese text sentiment analysis using lstm network based on l2 and nadam. In *2017 IEEE 17th International Conference on Communication Technology (ICCT)*, páginas 1891–1895. IEEE, 2017.
- [187] César Laurent, Gabriel Pereyra, Philémon Brakel, Ying Zhang, y Yoshua Bengio. Batch normalized recurrent neural networks. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, páginas 2657–2661. IEEE, 2016.
- [188] Jaime Duque Domingo, Roberto Medina Aparicio, y Luis Miguel González Rodrigo. Cross validation voting for improving cnn classification in grocery products. *IEEE Access*, 10:20913–20925, 2022.
- [189] R Anandha Praba y L Suganthi. Harnet: automatic recognition of human activity from mobile health data using cnn and transfer learning of lstm with svm. *Automatika*, 65(1):167–178, 2024.
- [190] Shuai Shao, Yu Guan, Bing Zhai, Paolo Missier, y Thomas Plötz. Convboost: Boosting convnets for sensor-based activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 7(2):1–21, 2023.
- [191] Hongsen Pang, Li Zheng, y Hongbin Fang. Cross-attention enhanced pyramid multi-scale networks for sensor-based human activity recognition. *IEEE Journal of Biomedical and Health Informatics*, 2024.
- [192] Joost Broekens, Marcel Heerink, Henk Rosendal, et al. Assistive social robots in elderly care: a review. *Gerontechnology*, 8(2):94–103, 2009.
- [193] Carlos A Cifuentes, Maria J Pinto, Nathalia Céspedes, y Marcela Múnera. Social robots in therapy and care. *Current Robotics Reports*, 1:59–74, 2020.
- [194] Suzanne Hutson, Soo Ling Lim, Peter J Bentley, Nadia Bianchi-Berthouze, y Ann Bowling. Investigating the suitability of social robots for the wellbeing

- of the elderly. In *Affective Computing and Intelligent Interaction: 4th International Conference, ACII 2011, Memphis, TN, USA, October 9–12, 2011, Proceedings, Part I 4*, páginas 578–587. Springer, 2011.
- [195] Takanori Shibata y Kazuyoshi Wada. Robot therapy: a new approach for mental healthcare of the elderly—a mini-review. *Gerontology*, 57(4):378–386, 2011.
- [196] Cory-Ann Smarr, Tracy L Mitzner, Jenay M Beer, Akanksha Prakash, Tiffany L Chen, Charles C Kemp, y Wendy A Rogers. Domestic robots for older adults: attitudes, preferences, and potential. *International journal of social robotics*, 6:229–247, 2014.
- [197] Lucile Bechade, Guillaume Dubuisson-Duplessis, Gabrielle Pittaro, Mélanie Garcia, y Laurence Devillers. Towards metrics of evaluation of pepper robot as a social companion for the elderly. In *Advanced Social Interaction with Agents: 8th International Workshop on Spoken Dialog Systems*, páginas 89–101. Springer, 2019.
- [198] Eduardo Zalama Casanova, Jaime Gómez García-Bermejo, Luis Cobo, José María Delgado Urrecho, Beatriz Vielba García, et al. Una nueva oportunidad de atención a las personas mayores mediante robótica social en las residencias lacort (uno mas). 2022.
- [199] Fraser Robinson y Goldie Nejat. An analysis of design recommendations for socially assistive robot helpers for effective human-robot interactions in senior care. *Journal of Rehabilitation and Assistive Technologies Engineering*, 9: 20556683221101389, 2022.
- [200] Pouyan Asgharian, Adina M Panchea, y François Ferland. A review on the use of mobile service robots in elderly care. *Robotics*, 11(6):127, 2022.
- [201] Chen Y. Sun Y. Yin X. Zhang S. Fan, Y. y W. Fan. A mobile robot with multi-sensors for eldercare in indoor environments. *Journal of Automatica Sinica*, 4 (2):378–390, 2017.

- [202] Elizabeth Broadbent, Rebecca Stafford, y Bruce MacDonald. Acceptance of healthcare robots for the older population: Review and future directions. *International journal of social robotics*, 1:319–330, 2009.
- [203] Luca Ragno, Alberto Borboni, Federica Vannetti, Cinzia Amici, y Nicoletta Cusano. Application of social robots in healthcare: Review on characteristics, requirements, technical solutions. *Sensors*, 23(15):6820, 2023.
- [204] Claudia Di Napoli, Giovanni Ercolano, y Silvia Rossi. Personalized home-care support for the elderly: a field experience with a social robot at home. *User Modeling and User-Adapted Interaction*, 33(2):405–440, 2023.
- [205] Samikshya Bhandari y Renu Baral Sapkota. The role of social robots to support elderly people’s mental health: a literature review. 2023.
- [206] Dante I Tapia, Juan A Fraile, Sara Rodríguez, Ricardo S Alonso, y Juan M Corchado. Integrating hardware agents into an enhanced multi-agent architecture for ambient intelligence systems. *Information Sciences*, 222:47–65, 2013.
- [207] Iiro Jantunen, Hannu Laine, Pertti Huuskonen, Dirk Trossen, y Vladimir Ermolov. Smart sensor architecture for mobile-terminal-centric ambient intelligence. *Sensors and Actuators A: Physical*, 142(1):352–360, 2008.
- [208] Markus Eisenhauer, Peter Rosengren, y Pablo Antolin. A development platform for integrating wireless devices and sensors into ambient intelligence systems. In *2009 6th IEEE annual communications society conference on sensor, mesh and ad hoc communications and networks workshops*, páginas 1–3. IEEE, 2009.
- [209] Ionuț Nica, Daniela Blană Alexandru, Simona Liliana Paramon Craciunescu, y Stefan Ionescu. Automated valuation modelling: Analysing mortgage behavioural life profile models using machine learning techniques. *Sustainability*, 13(9), 2021. ISSN 2071-1050. doi: 10.3390/su13095162. URL <https://www.mdpi.com/2071-1050/13/9/5162>.
- [210] Daniele Liciotti, Michele Bernardini, Luca Romeo, y Emanuele Frontoni. A sequential deep learning application for recognising human activities in smart

- homes. *Neurocomputing*, 396:501–513, 2020. ISSN 0925-2312. doi: <https://doi.org/10.1016/j.neucom.2018.10.104>. URL <https://www.sciencedirect.com/science/article/pii/S0925231219304862>.
- [211] Joakim Ekström, Jim Åkerrén Ögren, y Tobias Sjöblom. Exact probability distribution for the roc area under curve. *Cancers*, 15(6), 2023. ISSN 2072-6694. doi: 10.3390/cancers15061788. URL <https://www.mdpi.com/2072-6694/15/6/1788>.
- [212] Victoria Mingote, Antonio Miguel, Alfonso Ortega, y Eduardo Lleida. Optimization of the area under the roc curve using neural network supervectors for text-dependent speaker verification. *Computer Speech & Language*, 63:101078, 2020.
- [213] Janika Leoste, Kristel Marmor, y Mati Heidmets. Nonverbal behavior of service robots in social interactions—a survey on recent studies. 2024.
- [214] Jong-Wook Kim, Jin-Young Choi, Eun-Ju Ha, y Jae-Ho Choi. Human pose estimation using mediapipe pose and optimization method based on a humanoid model. *Applied sciences*, 13(4):2700, 2023.
- [215] Hritam Basak, Rohit Kundu, Pawan Kumar Singh, Muhammad Fazal Ijaz, Marcin Woźniak, y Ram Sarkar. A union of deep learning and swarm-based optimization for 3d human action recognition. *Scientific Reports*, 12(1):5494, 2022.
- [216] Jaime Duque Domingo, Jaime Gómez-García-Bermejo, y Eduardo Zalama. Visual recognition of gymnastic exercise sequences. application to supervision and robot learning by demonstration. *Robotics and Autonomous Systems*, 143: 103830, 2021.
- [217] Yong Li y Luping Wang. Human activity recognition based on residual network and bilstm. *Sensors*, 22(2):635, 2022.
- [218] Nidhi Dua, Shiva Nand Singh, y Vijay Bhaskar Semwal. Multi-input cnn-gru based human activity recognition using wearable sensors. *Computing*, 103: 1461–1478, 2021.

- [219] Luigi D’Arco, Haiying Wang, y Huiru Zheng. Assessing impact of sensors and feature selection in smart-insole-based human activity recognition. *Methods and Protocols*, 5(3):45, 2022.
- [220] Pengfei Zhang, Cuiling Lan, Wenjun Zeng, Junliang Xing, Jianru Xue, y Nanning Zheng. Semantics-guided neural networks for efficient skeleton-based human action recognition. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, páginas 1112–1121, 2020.
- [221] Dasom Ahn, Sangwon Kim, Hyunsu Hong, y Byoung Chul Ko. Star-transformer: A spatio-temporal cross attention transformer for human action recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, páginas 3330–3339, 2023.
- [222] Yongjin Kwon, Kyuchang Kang, y Changseok Bae. Unsupervised learning for human activity recognition using smartphone sensors. *Expert Systems with Applications*, 41(14):6067–6074, 2014.
- [223] Dorra Trabelsi, Samer Mohammed, Faicel Chamroukhi, Latifa Oukhellou, y Yacine Amirat. An unsupervised approach for automatic activity recognition based on hidden markov model regression. *IEEE Transactions on automation science and engineering*, 10(3):829–835, 2013.
- [224] Wenda Li, Yangdi Xu, Bo Tan, y Robert J Piechocki. Passive wireless sensing for unsupervised human activity recognition in healthcare. In *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, páginas 1528–1533. IEEE, 2017.
- [225] Lu Bai, Chris Yeung, Christos Efstratiou, y Moyra Chikomo. Motion2vector: Unsupervised learning in human activity recognition using wrist-sensing data. In *Adjunct proceedings of the 2019 ACM international joint conference on pervasive and ubiquitous computing and proceedings of the 2019 ACM international symposium on wearable computers*, páginas 537–542, 2019.
- [226] Ferhat Attal, Samer Mohammed, Mariam Dedabrishvili, Faicel Chamroukhi,

- Latifa Oukhellou, y Yacine Amirat. Physical human activity recognition using wearable sensors. *Sensors*, 15(12):31314–31338, 2015.
- [227] R Janarthanan, Srinath Doss, y S Baskar. Optimized unsupervised deep learning assisted reconstructed coder in the on-nodule wearable sensor for human activity recognition. *Measurement*, 164:108050, 2020.
- [228] Tao Gu, Shaxun Chen, Xianping Tao, y Jian Lu. An unsupervised approach to activity recognition and segmentation based on object-use fingerprints. *Data & Knowledge Engineering*, 69(6):533–544, 2010.
- [229] Allah Bux Sargano, Xiaowei Gu, Plamen Angelov, y Zulfiqar Habib. Human action recognition using deep rule-based classifier. *Multimedia Tools and Applications*, 79:30653–30667, 2020.
- [230] Veralia Gabriela Sánchez y Nils-Olav Skeie. Decision trees for human activity recognition in smart house environments. 2018.
- [231] Teerawat Kumrai, Joseph Korpela, Takuya Maekawa, Yen Yu, y Ryota Kanai. Human activity recognition with deep reinforcement learning using the camera of a mobile robot. In *2020 IEEE international conference on pervasive computing and communications (PerCom)*, páginas 1–10. IEEE, 2020.