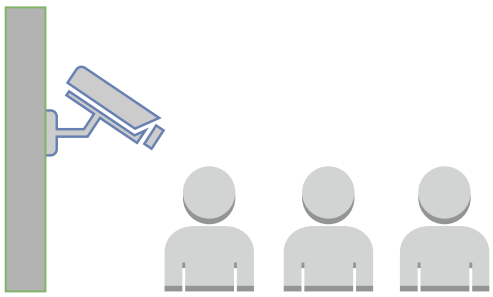
Universidad del País Vasco
Euskal Herriko Unibertsitatea
BILBOKO INGENIARITZA ESKOLA
ESCUELA DE INGENIERÍA DE BILBAO

MÁSTER UNIVERSITARIO EN INGENIERÍA DE TELECOMUNICACIÓN

TRABAJO FIN DE MÁSTER

DESARROLLO Y ANÁLISIS DE FIABILIDAD DE UN SISTEMA DE RECONOCIMIENTO FACIAL BASADO EN TÉCNICAS DE DEEP LEARNING Y FEDERATED LEARNING



Estudiante: Galván Rilo, Ander

Directora: Higuero Aperribai, Maria Victoria

Codirectora: Astorga Burgo, Jasone

Curso: 2023-2024

Fecha: Bilbao, 06, junio, 2024

Resumen

En los últimos tiempos, el reconocimiento facial es una aplicación que ha experimentado un notable avance gracias a la investigación y desarrollo de algoritmos de *Deep Learning*. Este tipo de identificación biométrica se implementa en diversas áreas, desde el desbloqueo de teléfonos hasta la vigilancia y control de espacios públicos. En estos casos, donde la manipulación de información facial es crítica y su compartición no siempre es posible, el aprendizaje federado emerge como una solución para desarrollar sistemas de reconocimiento facial salvaguardando la privacidad de las personas. Asimismo, el empleo de técnicas como *transfer learning* y *fine-tuning* permite transferir conocimientos de tareas generales a otras más específicas, lo que conduce a mejoras en la precisión del sistema de reconocimiento facial implementado.

En este contexto, se ha realizado este TFM (Trabajo de Fin de Máster), en el que, en primer lugar, se ha desarrollado un sistema de reconocimiento facial basado en *Deep Learning* y aprendizaje federado con el objetivo de identificar rostros de forma que se proporcione privacidad de diversos conjuntos de datos faciales.

De forma adicional, se han analizado los resultados de la aplicación de 2 tipos de *fine-tuning* a la implementación desarrollada. Esto ha dado lugar a la realización de un análisis de fiabilidad con el propósito de evaluar la necesidad de personalización de modelos preentrenados para tareas específicas.

Palabras clave: reconocimiento facial, aprendizaje federado, inteligencia artificial, *deep learning*, *transfer learning*, *fine-tuning*.

Laburpena

Azkenaldian, aurpegi-errekonozimendua aurrerapen nabarmena izan duen aplikazioa da, *Deep Learning* algoritmoen ikerketari eta garapenari esker. Identifikazio biometriko mota hau hainbat arlotan ezartzen da, telefonoak desblokeatzetik hasi eta espazio publikoak zaindu eta kontrolatzeraino. Kasu hauetan, non aurpegiarekin lotutako informazioren manipulazioa kritikoa den eta haren partekatzea beti posible ez den, ikaskuntza federatua aurpegi-errekonozimendu sistemak garatzeko irtenbide gisa agertzen da, pertsonen pribatutasuna bermatuz. Era berean, *transfer learning* eta *fine-tuning* teknikek zeregin orokorretatik zehatzagoak diren beste zeregin batzuetara ezagutzak transferitzea ahalbidetzen dute, aurpegi-errekonozimendu sistemen zehaztasuna hobetuz.

Testuinguru honetan, MAL (Master Amaierako Lana) hau burutu da, non lehenik eta behin, *Deep Learning* eta ikaskuntza federatuan oinarritutako aurpegi-errekonozimendu sistema bat garatu den, helburuak izanik aurpegiak zehaztasunez identifikatzea eta aurpegiak dituzten hainbat datu-multzoren pribatutasuna bermatzea.

Gainera, garatutako inplementazioari 2 *fine-tuning* mota aplikatzearen emaitzak aztertu dira. Horren ondorioz, fidagarritasun-azterketa bat egin da, aurrez entrenatutako modeloak zeregin espezifikoetara pertsonalizatzeko beharra ebaluatzeko.

Gako-hitzak: aurpegi-errekonozimendua, ikasketa federatua, adimen artifiziala, *deep learning*, *transfer learning*, *fine-tuning*.

Abstract

In recent times, facial recognition is an application that has experienced significant progress thanks to the research and development of Deep Learning algorithms. This type of biometric identification is implemented in various areas, from phone unlocking to surveillance and control of public spaces. In these cases, where the manipulation of facial information is critical and its sharing is not always possible, federated learning emerges as a solution to develop facial recognition systems while safeguarding the privacy of individuals. Also, the use of techniques such as transfer learning and fine-tuning allows the transfer of knowledge from general tasks to more specific ones, leading to improvements in the accuracy of the implemented facial recognition system.

In this context, this Master's Thesis has been carried out, in which, first of all, a face recognition system based on Deep Learning and federated learning has been developed with the aim of identifying faces in a way that provides privacy of various facial datasets.

Additionally, the results of the application of 2 types of fine-tuning to the developed implementation have been analyzed. This has entailed the performance of a reliability analysis with the purpose of evaluating the need for customization of pre-trained models to specific tasks.

Keywords: facial recognition, federated learning, artificial intelligence, deep learning, transfer learning, fine-tuning.

Índice

Lista de Figuras **6**

Lista de Tablas **8**

Lista de Acrónimos **10**

1. Introducción **11**

2. Contexto **13**

2.1. Inteligencia Artificial 13

2.2. Transfer Learning y Fine-Tuning 19

2.3. Aprendizaje Federado 20

2.4. Biometría 25

3. Objetivos y Alcance **27**

4. Beneficios **29**

4.1. Beneficios Técnicos 29

4.2. Beneficios Económicos 29

4.3. Beneficios Sociales 30

5. Estado del Arte **31**

6. Análisis de Alternativas **33**

6.1. Entrenamiento del Modelo 33

6.2. Librería de Deep Learning 35

6.3. Métrica de Comparación de Similitud 37

7. Análisis de Riesgos **40**

7.1. Definición de Riesgos 40
7.2. Comparación de Riesgos 42
7.3. Plan de Contingencia 42

8. Descripción de la Solución **44**
8.1. Diseño del Sistema 44
8.2. Implementación del Sistema 52

9. Evaluación de la Solución Propuesta **60**
9.1. Diseño del Plan de Pruebas 60
9.2. Métricas de Evaluación 62
9.3. Análisis de los Resultados 62

10. Descripción de Tareas **73**
10.1. Recursos Humanos y Materiales 73
10.2. Definición de los Paquetes de Trabajo y Tareas 74
10.3. Hitos del Proyecto 77
10.4. Diagrama de Gantt 78

11. Resumen de Costes **79**
11.1. Horas Internas 79
11.2. Amortizaciones 79
11.3. Gastos 79
11.4. Subcontrataciones 80
11.5. Coste Total del Trabajo 80

12. Conclusiones **81**

Referencias **82**

Lista de Figuras

1. Evolución de la inversión empresarial en Inteligencia Artificial a nivel global [2]. 11
2. Evolución global en el número de publicaciones sobre Inteligencia Artificial [3]. 12
3. Relación entre Inteligencia Artificial, *Machine Learning* y *Deep Learning* [10]. 14
4. Taxonomía de los tipos de aprendizaje de Machine Learning. 15
5. Arquitectura conceptual de una red CNN. 16
6. Arquitectura de un modelo CNN para el reconocimiento del abecedario de signos [13]. 17
7. Matriz de confusión para un problema de clasificación binaria. 18
8. Comparación entre el aprendizaje tradicional y aprendizaje a través de *transfer learning* [14]. 19
9. Comparación entre transfer learning y fine-tuning. 20
10. Arquitectura de un sistema de aprendizaje federado con comunicación centralizada. 21
11. Clasificación de los tipos de aprendizaje federado. 23
12. Evolución de la exactitud de un modelo en escenarios IID y Non-IID [19]. . . 24
13. Arquitectura de un sistema de reconocimiento facial [25]. 26
14. Comparación de alternativas de entrenamiento del modelo. 35
15. Comparación de alternativas de librería de *Deep Learning*. 37
16. Comparación de alternativas de métricas de similitud. 39
17. Matriz de probabilidad-impacto. 42
18. Diseño general del sistema de reconocimiento facial. 45
19. Arquitectura general del modelo InceptionResnetV1. 53
20. Cantidad de imágenes de reentrenamiento y prueba de cada delincuente. 55
21. Distribución de imágenes de delincuentes de cada cliente. 57
22. Esquema del diseño del plan de pruebas. 61

23. Valores de TNR de los diversos escenarios. 66
24. Valores de similitudes media de los delincuentes de los diversos escenarios. 66
25. Mapa de calor de similitudes de individuos delincuentes. 67
26. Mapa de calor de similitudes de individuos no delincuentes. 71

Lista de Tablas

1. Comparación de alternativas de entrenamiento del modelo. 34
2. Comparación de alternativas de librería de *Deep Learning*. 36
3. Comparación de alternativas de métricas de similitud. 38
4. Versiones de las herramientas software utilizadas. 54
5. Congelación de módulos durante el reentrenamiento parcial. 60
6. Comparación de los subescenarios del escenario E0. 63
7. Comparación de los subescenarios del escenario E1. 64
8. Comparación de los subescenarios del escenario E2. 64
9. Comparación de los subescenarios del escenario E3. 64
10. Comparación de los subescenarios del escenario E4. 64
11. Comparación de los subescenarios del escenario E5. 65
12. Comparación de los subescenarios del escenario E6. 65
13. Recursos humanos. 73
14. Recursos físicos. 73
15. Tareas del paquete de trabajo de definición del trabajo. 74
16. Tareas del paquete de trabajo de diseño del sistema de reconocimiento facial. 75
17. Tareas del paquete de trabajo de implementación del sistema de reconocimiento facial. 76
18. Tareas del paquete de trabajo de diseño, realización y evaluación del plan de pruebas. 77
19. Tareas del paquete de trabajo de gestión del trabajo. 77
20. Hitos del proyecto. 78
21. Horas internas del trabajo. 79
22. Amortizaciones del trabajo. 79
23. Gastos del trabajo. 80

24. Subcontrataciones del trabajo. 80
25. Resumen de costes del trabajo. 80

Lista de Acrónimos

AIoT Artificial Internet of Things

ANN Artificial Neural Networks

AUC Area Under Curve

CNN Convolutional Neural Network

CPD Centro de Procesamiento de Datos

CPU Central Processing Unit

DL Deep Learning

DFC Decoupled Feature Customization

FL Federated Learning

GDPR General Data Protection Regulation

GPU Graphics Processing Unit

IA Inteligencia Artificial

IID Identically and Independently Distributed

ML Machine Learning

MTCNN Multi-Task Cascaded Convolutional Neural Network

Non-IID Non-Identically and Independently Distributed

ROC Receiver Operating Characteristic

TFM Trabajo de Fin de Máster

TNR True Negative Rate

TPR True Positive Rate

1. Introducción

Durante los últimos años, la IA (Inteligencia Artificial) ha emergido como un elemento novedoso en la sociedad, pese a que su investigación y desarrollo se remonta al siglo pasado. Desde un punto de vista general, la Inteligencia Artificial es una aplicación tecnológica que se implementa mediante algoritmos o programas informáticos para ejecutar acciones imitando al proceso de razonamiento humano. Desde el reconocimiento de voz en asistentes personales como Alexa, hasta la conducción autónoma presentada por Tesla, la Inteligencia Artificial abarca numerosos ámbitos debido a su gran potencial [1].

Como muestra de este potencial, y del crecimiento que está experimentando esta tecnología, a continuación, en la Figura 1 se puede apreciar cómo la inversión empresarial en Inteligencia Artificial ha aumentado con los últimos años a nivel global.

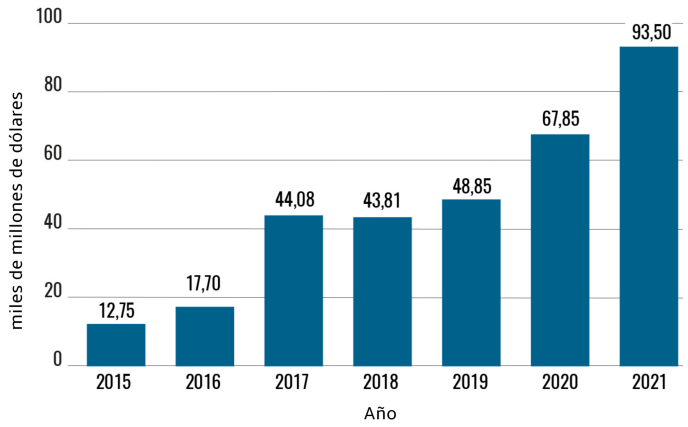


Figura 1: Evolución de la inversión empresarial en Inteligencia Artificial a nivel global [2].

El reconocimiento facial, ámbito muy ligado a la Inteligencia Artificial, es una aplicación muy presente que se implementa en escenarios en los que se tiene como objetivo identificar a una persona a través de su rostro. Un claro ejemplo de cómo se implementa junto a la Inteligencia Artificial es el FaceID que utiliza Apple en los iPhone. En este sistema se captura el rostro del usuario a través de la cámara del dispositivo y se compara con las imágenes previamente almacenadas en el móvil. Si el usuario es considerado como la persona propietaria, se le otorga el acceso al mismo.

Por otro lado, pese a que la Inteligencia Artificial es una herramienta con un gran potencial, existen situaciones en las que la información se considera confidencial o simplemente está distribuida al no poder recogerse toda en una ubicación central. Tal es el caso del ámbito médico, donde los datos de los pacientes están protegidos por leyes de privacidad. En dichos casos, el aprendizaje federado emerge como una solución muy eficaz, permitiendo desarrollar sistemas que aprenden de diversas fuentes

de información sin que sea necesario compartir los datos. Fundamentalmente, el modelo aprende de forma local e independiente, y luego se realiza una agregación de todos los conocimientos locales para obtener el modelo global. De este modo, la información local se mantiene confidencial y se preserva la privacidad.

La cantidad de información disponible, el tiempo de entrenamiento y los recursos computacionales necesarios para desarrollar un modelo de reconocimiento facial son varios de los aspectos críticos de los sistemas que implementan la Inteligencia Artificial. Por esta razón, a medida que la Inteligencia Artificial evoluciona, emergen nuevas técnicas para contrarrestar los mencionados inconvenientes. Dos claros ejemplos son el *transfer learning* y *fine-tuning*. Estas técnicas permiten transferir los conocimientos de una tarea a otra nueva, generalmente, mejorando la precisión y la fiabilidad de los modelos. En consecuencia, dichos modelos pueden ser reentrenados o adaptados a escenarios concretos, lo que facilita su aplicación en los mismos.

A continuación, en la Figura 2 se presenta la tendencia positiva de la publicación de artículos relacionados con la Inteligencia Artificial, lo cual muestra el incremento en la cantidad de trabajos publicados en este campo, que es paralelo al incremento de avances e innovaciones en este escenario.

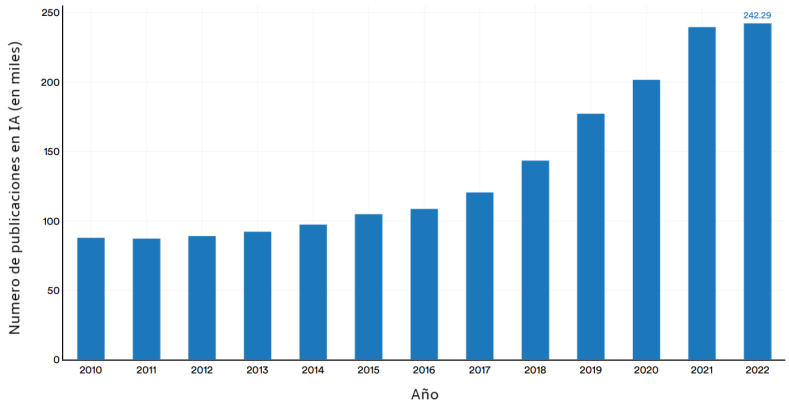


Figura 2: Evolución global en el número de publicaciones sobre Inteligencia Artificial [3].

En este contexto, ha surgido la motivación para el desarrollo de este TFM, en el marco de los trabajos de investigación del grupo I2T (Grupo de Investigación e Ingeniería Telemática) de esta universidad (UPV/EHU). En concreto, este trabajo está relacionado con la participación de I2T en el proyecto estatal "Biosurveillance through Artificial Intelligence (AI) in the post-COVID era: fundamental rights in the face of biometric technologies (AI-BioSurv)"subvencionado por el Ministerio de Ciencia e Innovación (número de subvención TED2021-129975B-C22).

En resumen, el presente trabajo contribuye a la investigación del reconocimiento facial a través de la Inteligencia Artificial. Concretamente, se estudia y se analiza la efectividad del reentrenamiento de un modelo que ha sido previamente entrenado con una cantidad masiva de imágenes. Además, se realiza un análisis de fiabilidad para la detección de individuos pertenecientes a distintas categorías. El presente trabajo se orienta hacia la identificación y clasificación de individuos en 2 categorías (como podrían ser delincuentes y no delincuentes, o personas con una enfermedad o sin ella), lo que permite su adaptación a diversos escenarios.

2. Contexto

Para describir el trabajo realizado, se considera de interés describir los distintos conceptos relacionados con las tecnologías que constituyen la base para el desarrollo del sistema. Por consiguiente, en este segundo apartado se explica en detalle cada uno de ellos.

2.1. Inteligencia Artificial

La Inteligencia Artificial es una ciencia o tecnología que transforma los sistemas informáticos en entidades inteligentes, dotándolos de la capacidad de analizar y procesar información de manera similar a los seres humanos. En otras palabras, se enfoca en estudiar y analizar cómo el cerebro humano razona, decide y deduce, para luego replicar estas habilidades en los sistemas informáticos previamente mencionados. Esto les permite realizar tareas que, anteriormente, solamente las personas eran capaces de llevar a cabo [4].

En cuanto a sus aplicaciones, la Inteligencia Artificial está integrada en numerosos sectores, como es el caso de la industria, fabricación, logística, finanzas y sanidad. Por ejemplo, en el ámbito de la salud, cada vez es más frecuente encontrar sistemas informáticos capaces de identificar a aquellas personas con mayor propensión a padecer ciertas enfermedades, así como predecir qué tratamiento resulta más efectivo para cada caso particular [5]. Asimismo, es muy útil para detectar posibles fallos de fabricación en ciertos productos que requieren de una precisión alta [6], así como para predecir las rutas óptimas en términos de logística [7]. Actualmente, el uso de la Inteligencia Artificial es muy amplio y se espera que su adopción continúe creciendo de manera exponencial [8].

2.1.1. Inteligencia Artificial, Machine Learning y Deep Learning

La Inteligencia Artificial, al ser un ámbito tan amplio y diverso, resulta fundamental distinguir entre varias técnicas de aprendizaje que la conforman. Dos de las más destacadas son el aprendizaje automático (ML, Machine Learning) y el aprendizaje profundo (DL, Deep Learning). Desde un punto de vista general, Taye [9] menciona que si la Inteligencia Artificial es comparada con un cerebro, el *Machine Learning* representa el proceso mediante el cual la Inteligencia Artificial adquiere nuevas capacidades cognitivas, mientras que el *Deep Learning* destaca como el sistema de autoaprendizaje más eficaz en la actualidad.

En la Figura 3 se puede apreciar cómo la Inteligencia Artificial es un campo amplio que engloba técnicas de *Machine Learning*, siendo este último un subconjunto de la Inteligencia Artificial. Además, el *Deep Learning* se encuentra dentro del *Machine*

Learning, lo que significa que todas las técnicas de *Deep Learning* son consideradas como técnicas de *Machine Learning*, aunque no todas las técnicas de *Machine Learning* corresponden al *Deep Learning* [10].

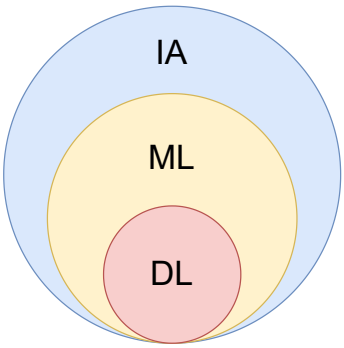


Figura 3: Relación entre Inteligencia Artificial, *Machine Learning* y *Deep Learning* [10].

2.1.1.1. Machine Learning

Profundizando en conceptos técnicos, el *Machine Learning* aprende a establecer relaciones entre las entradas y las salidas de los sistemas utilizando representaciones específicas, también conocidas como características, las cuales se diseñan manualmente para cada tarea [9]. Es decir, se enfoca en automatizar la tarea de construcción de modelos para realizar tareas cognitivas, así como predecir resultados con la mayor precisión posible (por ejemplo, la predicción del clima). Esto se consigue aplicando técnicas que aprenden iterativamente a partir de datos de entrenamiento del problema, lo que permite a los sistemas informáticos encontrar patrones complejos sin ser programados explícitamente. De esta manera, contribuye a la toma de decisiones más fiables y repetibles. Por ello, las técnicas de *Machine Learning* se incluyen en diversos ámbitos, por ejemplo, en detección de fraudes, en reconocimiento de voz, etc [11].

Según la manera en que un modelo aprende a tomar decisiones con los datos disponibles, se pueden identificar 4 tipos de aprendizajes de *Machine Learning*: aprendizaje supervisado, aprendizaje no supervisado, aprendizaje semisupervisado y aprendizaje por refuerzo. A continuación, se define cada uno de ellos, así como algunas de sus aplicaciones más destacadas [9].

- Aprendizaje supervisado. Se utilizan datos que están etiquetados, es decir, cada entrada de datos está asociado a una etiqueta. El modelo se centra en afinar las predicciones para reducir la diferencia entre sus predicciones y los resultados reales. Cuando la salida del modelo consiste en un valor discreto (por ejemplo, clase perro o clase gato), se trata de un problema de clasificación. En cambio, si la salida es un valor continuo (por ejemplo, el precio de una vivienda), se trata de un problema de regresión. Un ámbito donde se emplea este tipo de aprendizaje es el comercio, donde es fundamental anticipar precios.
- Aprendizaje no supervisado. Al contrario que en el aprendizaje supervisado, los datos no están etiquetados. El objetivo es entender, en profundidad, la estructura o distribución de estos últimos, extraer características, reconocer patrones y categorizar o etiquetarlos. Al no disponer de los resultados reales, no se puede determinar una salida correcta o incorrecta del modelo. Principalmente, se pueden diferenciar en 2 tipos de problemas: *clustering* (división de datos en diferentes grupos donde

los elementos del mismo grupo son similares entre sí) y asociación (identificación de relaciones entre las variables de los datos). Cabe resaltar que suele utilizarse en campos como la publicidad digital y el marketing, donde en función de los datos obtenidos de cada persona, se adaptan los servicios.

- Aprendizaje semisupervisado. Se sitúa en un punto intermedio entre el aprendizaje supervisado y no supervisado. Es el que más se emplea en problemas reales, debido a que solo una minoría de los datos están etiquetados, así como la mayoría no. El propósito de este tipo de aprendizaje es utilizar los datos etiquetados para luego clasificar o evaluar en los datos no etiquetados. Clasificación y *clustering* son los 2 tipos de problemas más comunes en los que se usa este tipo de aprendizaje. Generalmente, se emplea mucho en ámbitos como la salud y reconocimiento de voz.
- Aprendizaje por refuerzo. Se trata de un aprendizaje que se adapta al entorno del problema. En vez de seguir instrucciones específicas, los agentes aprenden de forma autónoma. Se basa en la idea de prueba y error, donde los agentes son recompensados al tomar decisiones correctas y penalizados por las erróneas. Los problemas más típicos son de clasificación y control, y es usual emplear este tipo de aprendizaje en entornos donde los datos son escasos o inconsistentes.

En la Figura 4 se resumen los tipos de aprendizajes de *Machine Learning* previamente explicados.

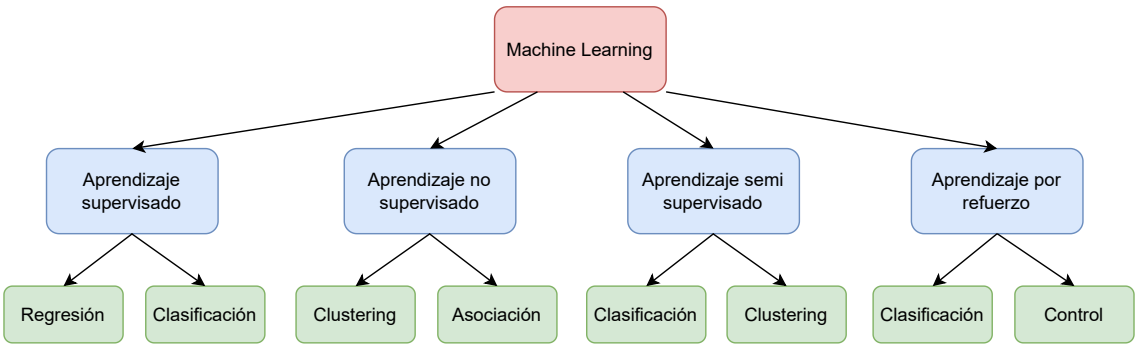


Figura 4: Taxonomía de los tipos de aprendizaje de Machine Learning.

2.1.1.2. Deep Learning

Dentro del mundo del *Machine Learning*, en la actualidad, el *Deep Learning* emerge como una técnica poderosa y efectiva que se basa en la simulación del cerebro humano y es capaz de detectar objetos y personas, reconocer voces, traducir textos, etc. Una de sus ventajas más significativas es la capacidad para aprender sin supervisión humana, a partir de datos no estructurados [12].

Respecto al funcionamiento, el *Deep Learning* está constituido por múltiples capas de algoritmos, conocidas como redes neuronales artificiales (ANN, Artificial Neural Networks). Cada una de ellas proporciona una interpretación diferente de los datos que se le han transmitido [10]. En el caso del reconocimiento facial, las capas inferiores, que constituyen las primeras etapas de procesamiento en una red neuronal, se encargan de identificar características simples, como bordes, esquinas y texturas básicas, mientras que las capas superiores, que representan etapas posteriores del procesamiento, son capaces de identificar características más complejas, como ojos, sonrisas, nariz, etc [12].

2.1.1.2.1 Redes Neuronales Convolucionales

Las redes neuronales convolucionales (CNN, Convolutional Neural Networks) es un tipo de red neuronal artificial especialmente diseñada para procesar datos organizados con forma matricial, como las imágenes.

En cuanto a su arquitectura, estas redes están compuestas por numerosas capas convolucionales seguidas de funciones de activación, seguidas a su vez de capas *pooling*, finalizando con capas totalmente conectadas [10]. Todas ellas están constituidas por neuronas que realizan cálculos sobre los datos de entrada y transmiten esta información a las capas siguientes. A continuación, se describe cada uno de estos tipos de capas [10].

- Capa convolucional. Está formada por filtros convolucionales, también conocidos como *kernels*. Cada filtro convolucional es una pequeña matriz de pesos que se desliza sobre la imagen de entrada y realiza una operación de convolución en un área local. Cada filtro aprende a detectar características específicas en la imagen. La salida de este tipo de capas es un mapa de características (*feature map*).
- Capa de función de activación. Se emplea después de las capas convolucionales para realizar el mapeo de la entrada a la salida de forma no lineal. Esto ayuda a la red a aprender representaciones más complejas de los datos.
- Capa *pooling*. Se encarga de reducir el tamaño de los mapas de características, manteniendo la mayor parte de la información. Esto ayuda a reducir la cantidad de parámetros y computaciones en la red, lo que se traduce en una red más eficiente.
- Capa totalmente conectada. Normalmente, estas capas se encuentran al final de la red CNN. Cada neurona que forma parte de esta capa está conectada a todas las neuronas de la capa anterior. Se utiliza como clasificador y su entrada tiene forma de vector, que se crea a partir de los mapas de características. En resumen, es utilizada en el caso de necesidad de interpretación de los resultados y toma de decisiones.

En la Figura 5 se puede apreciar una representación de una arquitectura conceptual de una red CNN.

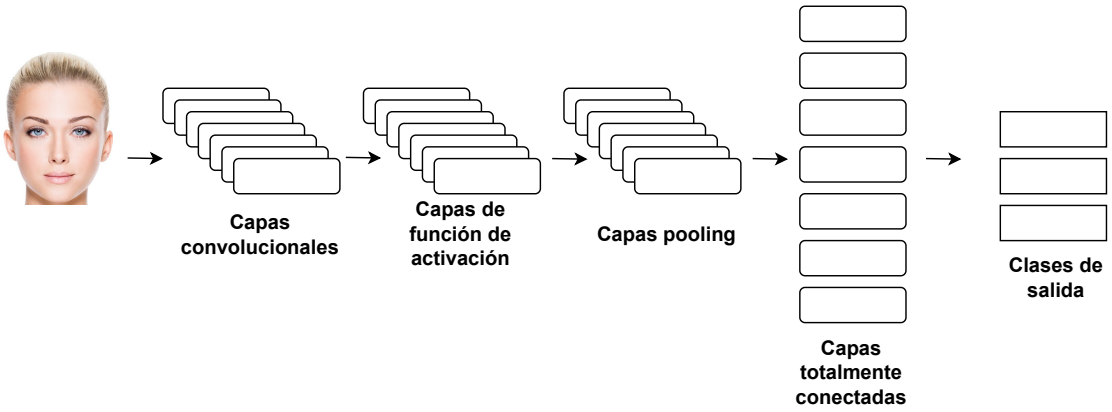


Figura 5: Arquitectura conceptual de una red CNN.

Sin embargo, las redes o modelos CNN realmente no siguen la arquitectura conceptual mostrada previamente. En la práctica, las capas convolucionales, función de activación y *pooling* se intercalan entre sí, principalmente para aumentar la profundidad y

complejidad del modelo. En la Figura 6 se puede ver un ejemplo de la arquitectura de un modelo CNN real.

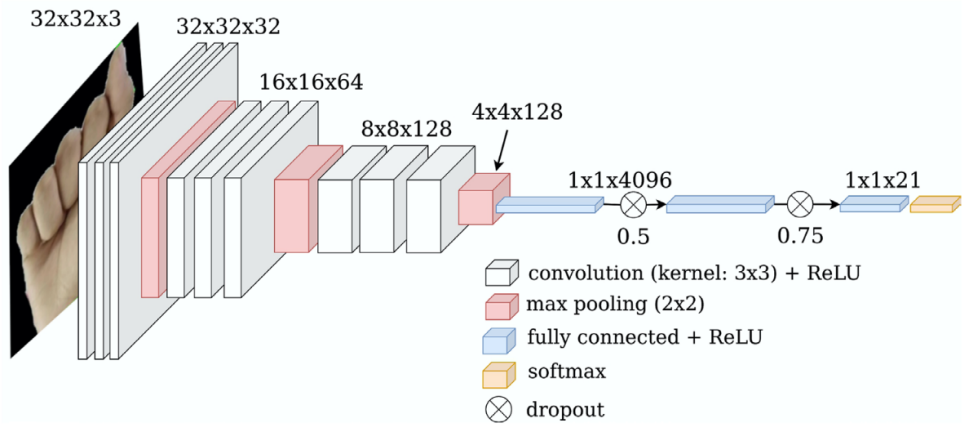


Figura 6: Arquitectura de un modelo CNN para el reconocimiento del abecedario de signos [13].

2.1.1.2.2 Entrenamiento y Evaluación de Redes Neuronales Convolucionales

El entrenamiento y la evaluación son las principales fases que conforman el ciclo de desarrollo de un modelo. En consecuencia, es importante ser consciente de qué se realiza y cómo se lleva a cabo cada una de ellas.

a) El entrenamiento del modelo se divide en 6 fases diferentes. Desde la tercera fase en adelante, estas se repiten varias veces, que se definen como épocas (*epochs*).

1. Elección de la arquitectura del modelo. Se estudia, analiza y decide cuál es la arquitectura óptima para resolver el problema, debido a que la complejidad del modelo influye en la capacidad de generalización del mismo. Un modelo complejo puede captar características más detalladas, pero también puede causar un sobreajuste respecto al conjunto de entrenamiento, lo que llevaría a una mala generalización en datos no vistos.
2. Inicialización de los parámetros del modelo. Se inicializan los parámetros del modelo de manera aleatoria o mediante alguna inicialización específica. La selección de la inicialización adecuada depende de diversos factores, tales como la arquitectura del modelo, los datos a utilizar, etc. Los parámetros del modelo están formados por pesos y sesgos. Los pesos son los valores numéricos que representan las conexiones entre las neuronas en diferentes capas de una red neuronal. En cambio, los sesgos son valores adicionales que se suman a la entrada de cada neurona antes de aplicar una función de activación. Estos sesgos permiten que la red neuronal aprenda representaciones más complejas y no lineales de los datos de entrada.
3. Introducción de los datos. Los datos se introducen en el modelo para obtener las predicciones. Para garantizar que los datos tienen el mismo formato antes de ser introducidos en el modelo, deben ser preprocesados de igual manera.
4. Cálculo de la función de pérdida. Se calcula la función de pérdida, que mide la diferencia entre las predicciones y los valores reales.
5. Propagación hacia atrás. Se calcula el gradiente de la función de pérdida con respecto a cada uno de los parámetros del modelo mediante retropropagación.

6. Actualización de los parámetros del modelo. Los parámetros del modelo se actualizan utilizando un algoritmo de optimización con el objetivo de minimizar la función de pérdida.

b) En cuanto a la evaluación del modelo, se emplea un conjunto de datos que el modelo no ha visto anteriormente. Además, para determinar la calidad del modelo, se hace uso de diversas métricas que tienen como propósito medir la fiabilidad del mismo. Estas métricas se basan en la matriz de confusión, que resume las predicciones correctas e incorrectas. En la Figura 7 se puede apreciar la matriz de confusión para un problema de clasificación binaria, como es el caso de este trabajo.

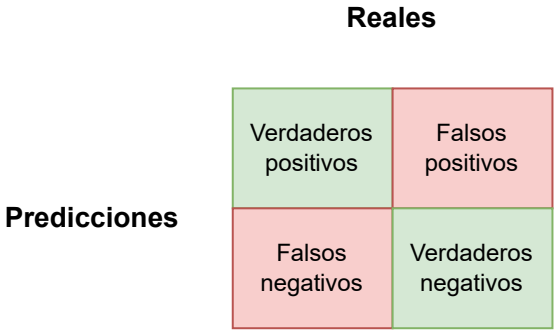


Figura 7: Matriz de confusión para un problema de clasificación binaria.

A continuación, se describen el significado de los términos que aparecen en la Figura 7.

- Verdaderos positivos. Predicciones correctas del modelo sobre la clase positiva.
- Verdaderos negativos. Predicciones correctas del modelo sobre la clase negativa.
- Falsos positivos. Predicciones incorrectas del modelo sobre la clase positiva (en realidad son clase negativa).
- Falsos negativos. Predicciones incorrectas del modelo sobre la clase negativa (en realidad son clase positiva).

Una vez entendidos los diferentes elementos que conforman la matriz de confusión, se definen las métricas más empleadas en el ámbito de la Inteligencia Artificial.

- Precisión (*Accuracy*). Indica la proporción de predicciones correctas sobre el total de predicciones realizadas.
- Precisión (*Precision*). Indica la proporción de verdaderos positivos sobre el total de predicciones positivas.
- Sensibilidad (*Recall* o *TPR, True Positive Rate*). Indica la proporción de verdaderos positivos sobre el total de positivos reales.
- Especificidad (*TNR, True Negative Rate*). Indica la proporción de verdaderos negativos sobre el total de negativos reales.
- F1. Media armónica de *precision* y *recall*. Es útil cuando se necesita un balance entre ambas métricas.

- ROC-AUC (*Receiver Operating Characteristic - Area Under Curve*). Indica la capacidad del modelo para distinguir las diversas clases.

La elección de las métricas a utilizar es una decisión fundamental en la fase de evaluación del modelo, debido a que la decisión debe ser tomada con base en los objetivos del trabajo y las características del mismo.

2.2. Transfer Learning y Fine-Tuning

En este segundo subapartado se procede a explicar las técnicas *transfer learning* y *fine-tuning*.

2.2.1. Transfer Learning

Tal y como se ha mencionado anteriormente, los modelos de *Deep Learning* requieren grandes cantidades de información para poder ser entrenados adecuadamente. Sin embargo, existen situaciones en las que la falta de información y/o recursos computacionales impide entrenar un modelo desde cero. En consecuencia, el *transfer learning* es un concepto del *Deep Learning* que está ganando mucha importancia en la actualidad.

El *transfer learning* es una técnica que consiste en transferir el conocimiento de un modelo diseñado para resolver una tarea a otro modelo que se utilizará para resolver otra tarea. En tales casos, ayuda a mejorar la fiabilidad del modelo y a reducir el tiempo de entrenamiento, siendo su uso muy beneficioso [14].

En la Figura 8 se muestra la comparación entre el aprendizaje tradicional y el aprendizaje a través de *transfer learning*, teniendo en cuenta que la cantidad de información del conjunto de datos *Dataset 2* es considerablemente menor que la de *Dataset 1*.

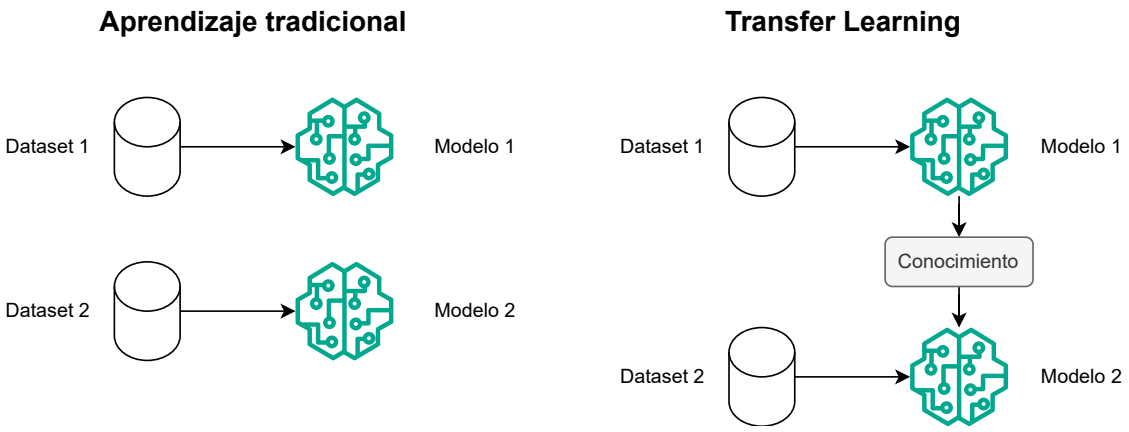


Figura 8: Comparación entre el aprendizaje tradicional y aprendizaje a través de *transfer learning* [14].

En cuanto a su funcionamiento, se congelan las capas encargadas de la extracción de características, mientras que se ajustan, añaden o modifican las últimas capas encargadas de la clasificación para la nueva tarea. Así, se mantiene la capacidad de las capas extractoras de características y se ajustan las capas finales para adaptarlas al nuevo conjunto de datos. El término *congelar* implica la prohibición de que capas previamente entrenadas adquieran nuevas habilidades durante el entrenamiento.

2.2.2. Fine-tuning

La técnica del *fine-tuning* es el enfoque más común dentro del *transfer learning*. A diferencia del *transfer learning* general, no solo se ajustan los parámetros de las capas finales, sino que también los de ciertas capas encargadas de la extracción de características. Cabe recordar que los parámetros son los valores ajustables que se modifican durante el entrenamiento para minimizar el error del modelo. Normalmente, se diferencian 2 prácticas diferentes de *fine-tuning* [15]:

- Ajuste de todos los parámetros del modelo. En este caso, todas las capas que contienen parámetros entrenables son ajustadas al nuevo conjunto de datos, después de haber inicializado los parámetros de estas con los preentrenados. Es importante destacar que si el nuevo conjunto de datos es relativamente pequeño y el modelo utilizado tiene una gran cantidad de parámetros entrenables, puede llevar a sobreajustarse.
- Ajuste de los parámetros de las capas finales. En vez de ajustar todos los parámetros, solo se ajustan los parámetros de las últimas capas, que son las que están más relacionadas con la tarea específica a resolver. El problema principal de este enfoque es que no está definido el número de capas que deben ser congeladas, es decir, sigue siendo una opción de diseño manual.

En la Figura 9 se muestra la comparación entre las técnicas de *transfer learning* y *fine-tuning*.

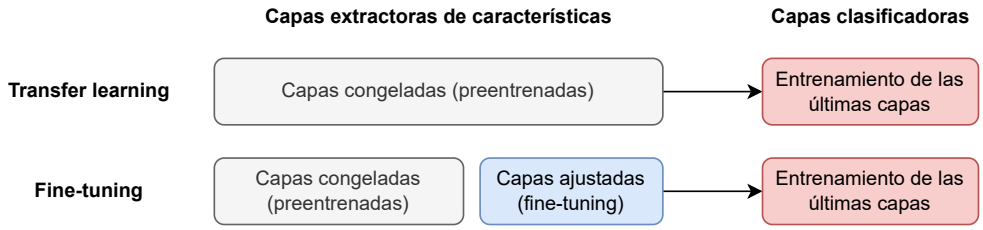


Figura 9: Comparación entre transfer learning y fine-tuning.

2.3. Aprendizaje Federado

Otro de los puntos importantes de este trabajo es el aprendizaje federado (FL, Federated Learning). En la mayoría de las situaciones del mundo real, la información está repartida entre diferentes entidades u organizaciones, de forma que se crean islas de información [16] [17]. Por ende, el aprendizaje federado permite a diferentes entidades u organizaciones entrenar modelos de Inteligencia Artificial sin la necesidad de tener que compartir su información local [17]. Dicho en otras palabras, permite entrenar un sistema de forma colaborativa sin que cada una de las entidades u organizaciones expongan su información. El objetivo es alcanzar una fiabilidad muy cercana a la que se conseguiría mediante un aprendizaje centralizado, donde la información se procesa de manera centralizada [16], pero ofreciendo privacidad a cada conjunto de datos.

Un ejemplo son los hospitales, en los cuales la información de los pacientes es confidencial y no se puede compartir, y un solo hospital puede no ser capaz de entrenar

un modelo de alta calidad que tenga una adecuada precisión para una tarea específica [17].

A su vez, es una tarea ardua integrar toda la información de manera centralizada, bien o porque el coste es muy alto o porque las leyes obligan a garantizar la privacidad [16]. De hecho, en este aspecto es importante señalar que en Europa en el año 2018 se publicó el Reglamento General de Protección de Datos de la Unión Europea (GDPR, General Data Protection Regulation) [18] con el objetivo de regular la compartición de información entre diferentes organizaciones [16] [17].

2.3.1. Componentes de un Sistema de Aprendizaje Federado

Un sistema de aprendizaje federado lo forman, principalmente, 3 componentes: los clientes (organizaciones u entidades), el servidor (gestor) y las bases de datos de los clientes.

- Los clientes son los dueños de los datos (organizaciones, entidades, móviles, etc.) que participan en el proceso de aprendizaje federado para contribuir a generar un modelo global [17].
- El servidor es un gestor de gran capacidad computacional que se encarga de gestionar el proceso de entrenamiento y las comunicaciones. Sus características más destacadas son la estabilidad y fiabilidad [17].
- Las bases de datos de los clientes son los almacenes de información que cada cliente posee, es decir, los datos privados que cada sistema local o cliente utiliza para contribuir al conocimiento del modelo global.

En la Figura 10 se puede apreciar la arquitectura general de un sistema de aprendizaje federado, representando los elementos que lo componen a través de una estructura de comunicación centralizada.

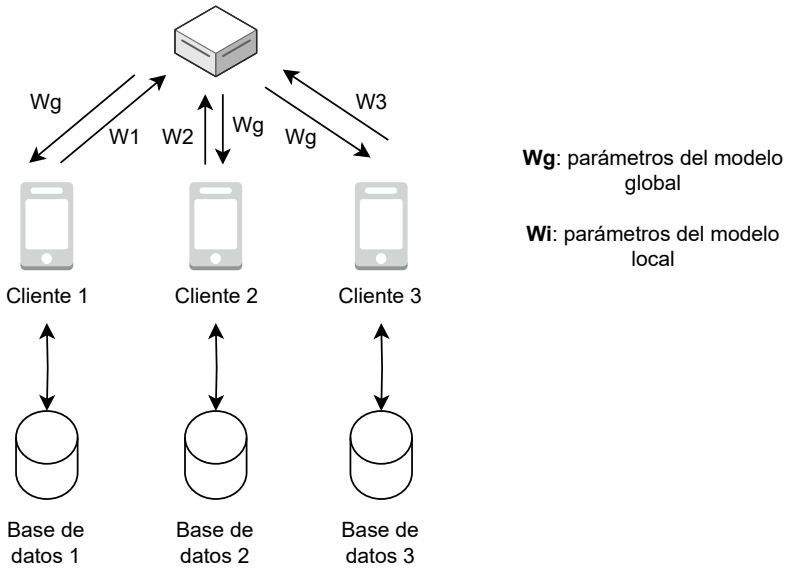


Figura 10: Arquitectura de un sistema de aprendizaje federado con comunicación centralizada.

2.3.2. Tipos de Aprendizaje Federado

El aprendizaje federado se puede clasificar en diferentes tipos según distintas características. En este caso, se clasifican según los criterios: partición de datos, arquitectura de comunicación y escalabilidad [17].

a) En lo que respecta a la partición de datos, es importante considerar el espacio de los conjuntos de datos y las características de dichos conjuntos. En consecuencia, se diferencian 3 posibles escenarios de aprendizaje federado.

   - Horizontal. Los clientes poseen diferentes conjuntos de datos con las mismas características. Este es el escenario más comúnmente implementado, y un ejemplo es la aplicación *OK Google*, donde cada usuario dice la misma frase con una voz diferente.
   - Vertical. Al contrario que el escenario anterior, los clientes comparten los conjuntos de datos, pero no las características de los mismos. Un ejemplo [16] es cuando un banco y una tienda de la misma ciudad colaboran en la creación de un modelo para predecir la compra de productos. Cada comercio y tienda almacena características diferentes de los ciudadanos, que serán los mismos o parecidos.
   - Híbrido. Es una combinación de los 2 escenarios previamente descritos. Un ejemplo es cuando diferentes hospitales quieren entrenar un modelo para el diagnóstico de una enfermedad, donde los pacientes son distintos y sus resultados médicos también.

b) En cuanto a la arquitectura de comunicación utilizada entre los elementos que forman el sistema de aprendizaje federado, se pueden diferenciar 2 tipos.

   - Arquitectura centralizada. Cada cliente entrena los parámetros de su modelo de forma local, para posteriormente transmitírselos al servidor. Este se encarga de agregar los parámetros recibidos de los diferentes clientes y de retornar el resultado de la agregación realizada a los respectivos clientes. A día de hoy es la arquitectura más empleada, pero conlleva el peligro de tener la información de los modelos locales concentrada en un solo punto.
   - Arquitectura descentralizada. Las comunicaciones se realizan entre los clientes y cada uno puede actualizar los parámetros globales directamente. Sin embargo, la ecuanimidad y la sobrecarga de comunicación son 2 factores a tener en cuenta a la hora de diseñar un sistema de este tipo.

c) Para terminar con la clasificación de los distintos tipos de aprendizaje federado, es importante mencionar la escalabilidad. Se pueden diferenciar 2 escenarios basados en la escalabilidad del sistema: *cross-silo* y *cross-device*. La diferencia entre ambos es el número de usuarios que colaboran en el proceso del entrenamiento del modelo y la cantidad de datos que contiene cada cliente.

   - *Cross-silo*. El sistema de aprendizaje federado lo forman pocos clientes, que suelen ser organizaciones o centros de datos que contienen grandes cantidades de información, al igual que grandes recursos computacionales.
   - *Cross-device*. Los clientes son abundantes, pero contienen poca información para entrenar el modelo (los clientes suelen ser dispositivos móviles).

En la Figura 11 se puede observar la clasificación de los diferentes tipos de aprendizaje federado.

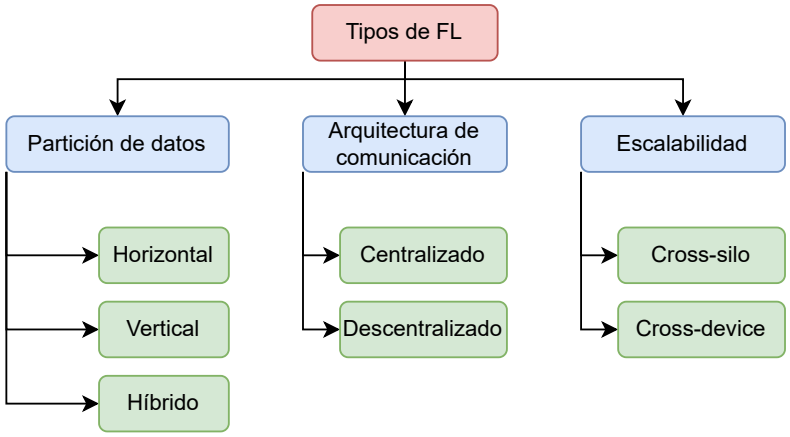


Figura 11: Clasificación de los tipos de aprendizaje federado.

2.3.3. Proceso de Entrenamiento

Dado que la arquitectura de comunicación centralizada es ampliamente empleada, a continuación se incluye un resumen de la descripción de las diversas etapas que componen el proceso de entrenamiento de un sistema de aprendizaje federado.

2.3.3.1. Configuración Inicial

- El servidor define la arquitectura del modelo e inicializa los parámetros del mismo, bien de forma aleatoria o bien a través de una inicialización específica.
- El servidor selecciona qué clientes participan en el entrenamiento.

2.3.3.2. Distribución y Entrenamiento del Modelo

- El servidor envía los parámetros del modelo global a los clientes seleccionados.
- Suponiendo que cada cliente almacena su propia información de manera local, cada uno entrena el modelo de forma local (ver apartado 2.1.1.2.2) y envía de vuelta al servidor los parámetros actualizados del modelo. Esto implica que los parámetros que el servidor recibe están ajustados a cada conjunto de datos de los clientes.

2.3.3.3. Agregación y Distribución del Modelo

- El servidor recibe los parámetros actualizados de los modelos de los clientes seleccionados y realiza la agregación de los mismos.
- Para finalizar, el servidor envía los parámetros del modelo global a los clientes.

2.3.3.4. Iteración

- Se repite este proceso durante múltiples rondas, hasta alcanzar un criterio de convergencia o hasta agotar el número predeterminado de rondas.

2.3.4. Distribución de Datos

Dentro de los 3 escenarios de partición de datos mencionados en el apartado anterior, es imprescindible tener en cuenta si los datos están distribuidos entre los clientes de manera idéntica e independiente (IID, Identically and Independently Distributed) o no (Non-IID, Non-Identically and Independently Distributed) [19].

- IID. Cada cliente atesora una distribución de datos similar, es decir, cada cliente posee una cantidad de datos similar de cada clase.
- Non-IID. Al contrario que IID, cada cliente posee una distribución de datos divergente, lo que implica que cada cliente dispone de diversas cantidades de información de cada clase.

En la Figura 12 se puede observar cómo en un entorno Non-IID, la precisión no alcanza los niveles que se logran en un entorno IID. Esto se debe a la variedad en las distribuciones de los datos, lo cual puede dificultar la convergencia del modelo global. Esta dificultad se traduce en la consecución de una precisión menor en el modelo global.

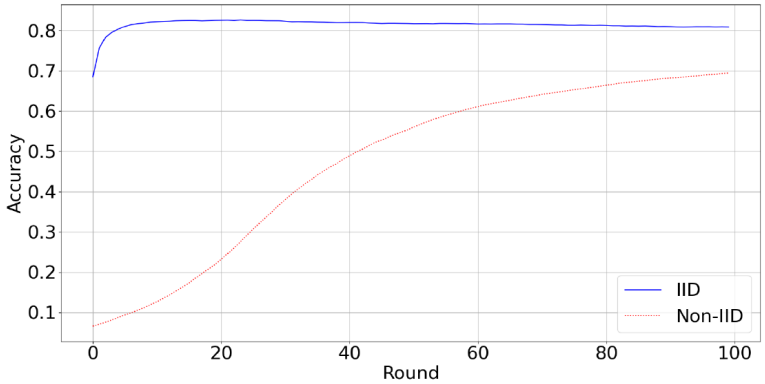


Figura 12: Evolución de la exactitud de un modelo en escenarios IID y Non-IID [19].

Asimismo, Zhao et al. [20] demuestran en su trabajo que la precisión en un sistema de aprendizaje federado se reduce hasta un 55 % para las redes neuronales entrenadas con una distribución de datos Non-IID muy sesgada. El mencionado concepto de sesgo implica que los datos no están distribuidos de manera uniforme entre las diferentes categorías.

2.3.5. Métodos de Agregación

El método de agregación empleado a la hora de consolidar los conocimientos obtenidos localmente por los clientes es uno de los aspectos críticos en el entrenamiento de sistemas que implementan aprendizaje federado. Pese a que en la actualidad existen diversos métodos, a continuación, se describen los más comunes.

- La agregación media, también conocida como *FedAvg*, es la técnica más común, ya que es fácil de implementar e interpretar. Sin embargo, es sensible a valores atípicos y no es eficiente en los casos en los que la información posee una distribución Non-IID. En cuanto al funcionamiento, básicamente realiza la media de los parámetros recibidos por cada cliente. Teniendo en cuenta que N es la cantidad de clientes que participan en el entrenamiento y w_i sus respectivos parámetros, la agregación se realiza como se muestra en la ecuación 2.1 [21].

$$w = (1/N) * \sum_{i=1}^{N} w_i \tag{2.1}$$

- Una variante de la agregación media es la agregación media recortada, también denominada *Trimmed Mean*. La diferencia radica en eliminar un porcentaje predefinido de los valores extremos de los parámetros de los modelos antes de calcular la media de los valores de los parámetros de los posibles modelos restantes. Esto ayuda a reducir el impacto de parámetros atípicos, así como parámetros maliciosos transmitidos por atacantes. La ecuación cambia ligeramente (ver ecuación 2.2), puesto que ahora se sustituye w_i por $clip(w_i, c)$ donde c se trata del umbral de recorte que limita los valores de w_i a un rango de $[-c, c]$ [21].

$$w = (1/N) * \sum_{i=1}^{N} clip(w_i, c) \tag{2.2}$$

- El método de agregación mediana, en lugar de calcular la media de determinados parámetros, los ordena y selecciona la mediana. Igual que la agregación media recortada, es menos sensible a valores atípicos, lo que lo hace más robusto frente ataques o parámetros atípicos. Respecto al funcionamiento, primero ordena los parámetros, w, en orden ascendente. Después, en función de si N es un valor impar o par, la mediana es el valor de la posición central o es el promedio de los dos valores centrales (ver ecuación 2.3).

$$w = \begin{cases} w\left(\frac{N+1}{2}\right) & \text{si N es impar} \\ \frac{w\left(\frac{N}{2}\right)+w\left(\frac{N+1}{2}\right)}{2} & \text{si N es par} \end{cases} \tag{2.3}$$

2.4. Biometría

La biometría es un enfoque que implica la identificación o verificación de las personas a través de sus características tanto físicas como comportamentales. Hoy en día, los sistemas biométricos emplean diversas técnicas para determinar la identidad de una persona, por ejemplo, el reconocimiento de huella dactilar, iris, retina, rostro, firma, entre otros [22], siendo vital su uso en los ámbitos bancarios, sanitarios y gubernamentales para proteger la información personal de los usuarios. Además, es una técnica fácil de implementar, escalable y personalizable, lo que la convierte en una herramienta poderosa para mejorar la seguridad y la privacidad en una variedad de aplicaciones [23].

2.4.1. Reconocimiento Facial

Pese a que existen diversas técnicas para llevar a cabo la identificación de individuos, el reconocimiento facial se ha convertido en una de las más populares en los últimos años. Hoy en día, tiene una amplia gama de aplicaciones; desde las cámaras de seguridad en aeropuertos y supermercados para detectar delincuentes hasta el acceso y la autenticación en los dispositivos móviles, como el FaceID de los iPhone [24].

Asimismo, uno de los mayores retos es la susceptibilidad del rostro a cambiar con el tiempo debido al envejecimiento o a factores externos, como las variaciones en la pose e iluminación, accesorios faciales, etc [24].

En la Figura 13 se muestra la arquitectura típica de un sistema de reconocimiento facial.

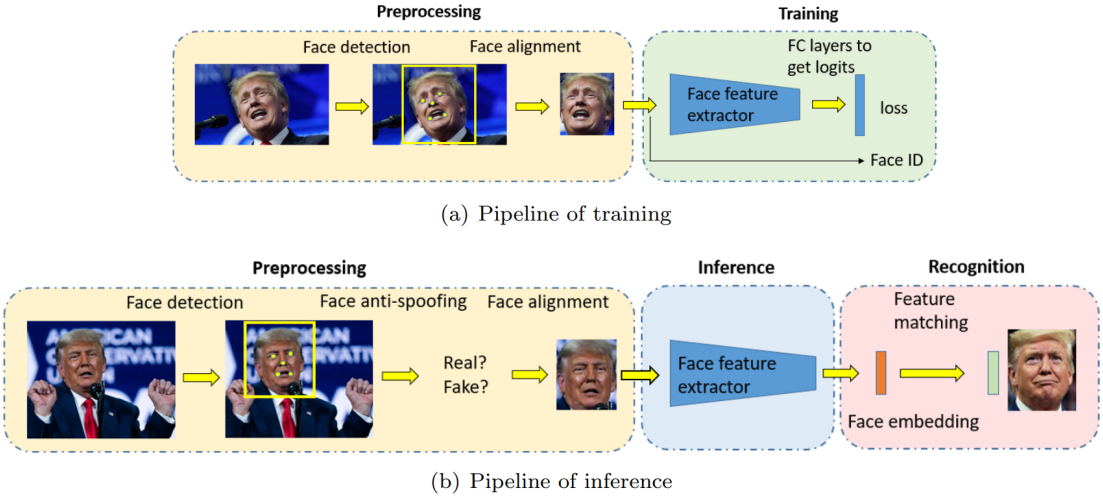


Figura 13: Arquitectura de un sistema de reconocimiento facial [25].

Por un lado, el sistema de entrenamiento de un modelo de este tipo consta de 2 fases: (1) el preprocesamiento de las imágenes faciales (2) el mismo entrenamiento del modelo. En cambio, el sistema de evaluación del modelo lo conforman 3: (1) el preprocesamiento de las imágenes faciales (2) la inferencia del modelo para obtener las características faciales (3) el reconocimiento facial. Es importante mencionar que la fase de preprocesamiento de los sistemas de entrenamiento y evaluación debe ser idéntica. Además, es habitual incorporar un mecanismo denominado *anti-spoofing* antes de la inferencia para prevenir ataques de suplantación [25]. Este verifica que la persona es real, asegurando que no sea una imagen o un vídeo. Para ello, emplea técnicas como análisis de movimiento, profundidad, etc.

Generalmente, la fase de preprocesamiento de las imágenes está formado por la detección del rostro y su alineamiento. En resumen, en la detección del rostro, este se trata como el objeto a detectar, y en la alineación del rostro, se ajusta su posición y ángulo para evitar problemas relacionados con las variaciones en la pose [25]. En la fase de entrenamiento, se entrena un modelo minimizando la función de pérdida. Posteriormente, en la fase de inferencia, se genera un vector de características de la imagen que se desea procesar. Este vector se utiliza para reconocer y vincular la imagen con la persona correspondiente en la fase de reconocimiento.

3. Objetivos y Alcance

En este tercer apartado, se definen tanto el objetivo principal como los objetivos parciales del trabajo que se plantean alcanzar mediante el desarrollo de este TFM. El objetivo principal de este trabajo es el **desarrollo y análisis de fiabilidad de un sistema de reconocimiento facial basado en técnicas de *Deep Learning* y aprendizaje federado**.

Para alcanzar el objetivo principal, se han definido los siguientes objetivos parciales, que de forma conjunta, constituyen el objetivo global de este trabajo.

Definición de un sistema de reconocimiento facial basado en técnicas de *Deep Learning* y aprendizaje federado

Este primer objetivo consiste en el diseño e implementación de un sistema de reconocimiento facial basado en la tecnología *Deep Learning*, en el que se busca la mayor fiabilidad posible en la identificación de rostros.

De forma adicional se ha planteado como requisito imprescindible de este proyecto, el soporte de privacidad por parte de este sistema, respecto a diversos conjuntos de personas a identificar, provenientes de distintas entidades, de forma que cada uno de estos conjuntos de imágenes no debe ser accedido de forma global por ninguna de las otras entidades, ni por el sistema central de reconocimiento. Por este motivo, se define la utilización del aprendizaje federado para garantizar la privacidad en este contexto.

Es importante señalar que se debe tener en cuenta que el soporte de privacidad por parte de este sistema no debe afectar a la fiabilidad del mismo en la identificación de personas.

El diseño y caracterización del sistema de reconocimiento facial debe partir de los estudios y desarrollos más innovadores en este campo, para lo que se define la necesidad de realizar un estudio del estado del arte sobre el reconocimiento facial basado en el *Deep Learning* y aprendizaje federado. Partiendo de las valoraciones de dichos desarrollos, en relación con los objetivos que se persiguen en este TFM, se pretende definir un sistema que ofrezca una adecuada fiabilidad en la identificación de personas.

Un reconocimiento facial fiable de personas, habilita la automatización de los procesos de identificación, en aquellos contextos en los que este proceso deba realizarse para un volumen grande de personas, o no se considere adecuada la identificación tradicional realizada por personas que llevan a cabo estos procesos de identificación contrastando documentos con la cara de personas. Estos escenarios pueden ser muy variados, como aeropuertos, hospitales, estadios deportivos, etc. La identificación automatizada de personas, basada en sistemas como el que se propone en este TFM, podría sustituir o complementar a los sistemas de identificación actuales.

De forma más concreta, en este trabajo se busca identificar personas que pertenezcan a un colectivo determinado. Pueden asociarse en distintos ámbitos a diferentes situaciones. Por ejemplo, en un escenario médico este sistema permitiría identificar a personas que padecen una determinada patología, o en un escenario de seguridad, podría identificar a delincuentes, etc.

Se trata de que el sistema pueda aprender a identificar a colectivos de esta categoría, como pueden ser los pacientes o los delincuentes, de forma que los conjuntos de imágenes de diversas entidades, que podrían ser en este caso, distintos centros médicos o fuerzas de seguridad, no se comparten con ninguna otra entidad.

En este caso, el objetivo del sistema sería proporcionar una fiabilidad adecuada a identificar a personas de esos colectivos, a la vez que detecta adecuadamente a personas que no pertenecen a dichos colectivos.

Definición de plan de pruebas para el análisis de la fiabilidad del modelo

Como segundo objetivo de este trabajo se plantea el diseño y despliegue de un plan de pruebas que permita analizar la fiabilidad del sistema definido en la detección e identificación de rostros.

Se debe definir el escenario, herramientas a utilizar, conjuntos de datos y tipo de pruebas a realizar, así como los criterios y parámetros o métricas de observación que permiten evaluar la fiabilidad del sistema.

Se plantea además la realización de pruebas bajo distintas condiciones de la plataforma con el objetivo de determinar la parametrización del sistema que optimiza la fiabilidad en el reconocimiento del desarrollo realizado.

Programación e implementación en plataforma de pruebas

Como último objetivo de este trabajo, se procede a implementar el sistema conforme al diseño establecido. Este proceso engloba la instalación, integración y programación de los elementos definidos en el diseño correspondiente al primer objetivo parcial.

Además, se llevan a cabo las diversas pruebas exhaustivas definidas en el segundo objetivo parcial. Posteriormente, se recopilan los resultados y se analizan para evaluar los valores obtenidos en cada caso.

4. Beneficios

Establecidos los objetivos, es momento de llevar a cabo un estudio de los beneficios que este trabajo conlleva. Los beneficios se clasifican en 3 categorías diferentes: técnicos, económicos y sociales.

4.1. Beneficios Técnicos

En lo que se refiere al ámbito del *Deep Learning*, este trabajo representa un avance adicional en el desarrollo de esta tecnología para su aplicación en escenarios de reconocimiento facial. Concretamente, el hecho de aplicar redes convolucionales para el procesamiento de imágenes faciales promueve la integración del reconocimiento facial junto al *Deep Learning*.

Contar con un análisis de fiabilidad que emplea el mecanismo *fine-tuning* contribuye al impulso y estudio del mismo. Esto se traduce en la mejora de la precisión de los sistemas de reconocimiento facial que posteriormente vayan a ser implementados en casos de uso reales.

El uso del aprendizaje federado para reentrenar un sistema de reconocimiento facial incentiva el estudio y la implementación de estos escenarios. Dicho en otras palabras, el hecho de garantizar la privacidad en este tipo de sistemas, abre la posibilidad de aplicarlo en escenarios de uso en los que sin privacidad no se considerarían una tecnología apropiada.

Cabe destacar que este trabajo representa un avance innovador en la automatización de tareas, permitiendo su ejecución de forma ágil y sin intervención humana. Además, la optimización de los parámetros constituye un progreso que puede servir como punto de partida para el desarrollo de otros sistemas con necesidades y objetivos similares.

Este trabajo constituye un paso más en el desarrollo y aplicación de este tipo de tecnologías en diversos ámbitos, especialmente en escenarios donde la garantía de la privacidad es un requerimiento, como puede ser en escenarios médicos o de seguridad ciudadana.

4.2. Beneficios Económicos

En cuanto a los beneficios económicos, cabe destacar que estos varían en función del caso de uso en el que se implementa el presente trabajo.

En términos generales, la automatización de la identificación de personas a través de

un sistema inteligente disminuye los costes asociados a la contratación de empleados adicionales. Esto significa que al implementar sistemas de este tipo, no se requiere la contratación de personal adicional para llevar a cabo el reconocimiento facial de forma manual.

De igual manera, la continua investigación de estas tecnologías conlleva el constante desarrollo de las mismas, lo que implica una mayor inversión por parte de entidades interesadas en dichas implementaciones.

4.3. Beneficios Sociales

En relación con los beneficios sociales, disponer de un sistema de reconocimiento facial basado en técnicas de *Deep Learning* y aprendizaje federado complementa los métodos tradicionales. Esto se traduce en la automatización del proceso de identificación de personas a la vez que se garantiza la privacidad de los mismos. De esta forma, las personas partícipes preservan su privacidad y al mismo tiempo, contribuyen al desarrollo de sistemas de reconocimiento facial más precisos. Esto lleva a una mayor satisfacción por parte de los usuarios en su utilización.

La implementación de este tipo de sistemas supone la agilización de los procesos de identificación en diferentes escenarios, tales como aeropuertos, eventos deportivos, etc. En estos escenarios los elevados tiempos de espera producidos por identificaciones manuales son comunes. Por consiguiente, este trabajo representa una mejora significativa en dichos procesos.

El aprendizaje federado en sistemas de reconocimiento facial desempeña un papel fundamental, puesto que garantiza el cumplimiento de la legislación en términos de privacidad de los datos. La implementación de este tipo de aprendizaje permite que los conjuntos de información de los diferentes clientes se mantengan confidenciales, minimizando los riesgos de violación de privacidad.

Para terminar con los beneficios, se considera importante destacar que este trabajo permite mejorar la seguridad ciudadana al posibilitar la identificación precisa y automatizada de individuos peligrosos. Así, los usuarios se encuentran en escenarios más seguros y protegidos.

5. Estado del Arte

Seguidamente, se lleva a cabo un estudio del estado del arte que engloba diversos trabajos que comparten algunos de los principales aspectos abordados en este trabajo, así como objetivos similares.

Divyansh Aggarwal et al. [26] desarrollaron un *framework* denominado FedFace que implica el entrenamiento de un sistema de reconocimiento facial mediante aprendizaje federado. Los resultados muestran una mejora en el rendimiento del sistema de reconocimiento facial preentrenado, CosFace [27], al utilizar datos faciales adicionales disponibles. Este trabajo comparte la mayoría de los conceptos con el presente estudio. Sin embargo, en el trabajo [26] se considera que cada cliente es un dispositivo móvil que contiene imágenes faciales pertenecientes únicamente al propietario del dispositivo, lo que aleja a este sistema de los objetivos que se persiguen con este TFM.

Youlong Ding et al. [28] desarrollaron un *framework* orientado a la industria para AIoT (Artificial Internet of Things) en el contexto de una aplicación de reconocimiento facial. En el mencionado trabajo, preentrenan el modelo con un conjunto de datos de acceso público, en el cual posteriormente se ajustan ligeramente los parámetros del modelo. Es decir, hacen uso de la técnica *fine-tuning*. Asimismo, proponen un método para proteger los gradientes compartidos entre los clientes y el servidor. Se trata de un trabajo que en gran parte se alinea con el presente estudio, pero al igual que el trabajo [26], no aborda el uso de *fine-tuning* parcial mediante congelamiento y ajuste de diversas capas, que se considera una parte muy significativa en este proyecto.

Lingyun Liu et al. [29] desarrollaron un *framework* para entrenar un modelo de reconocimiento facial en un entorno de aprendizaje federado donde cada cliente solo tiene acceso a una clase y los clientes no pueden compartir los vectores de características de clase entre sí. El objetivo del trabajo es asegurar que los vectores de características de clase de cada cliente están bien separados entre sí para mejorar la precisión del modelo. Yifan Niu et al. [30] desarrollaron un *framework* que garantiza una mayor privacidad en un sistema de reconocimiento facial en un escenario de aprendizaje federado. Para lograr esto, emplea una técnica de corrección de gradientes, así como un regularizador para aprender de manera efectiva representaciones faciales discriminatorias. Con base en los resultados obtenidos, se concluye que puede igualar el rendimiento de los métodos centralizados. Estos dos trabajos tienen objetivos muy similares y su investigación aporta un gran valor en la generación de representaciones faciales discriminatorias. No obstante, carecen de la implementación de las técnicas *transfer learning* y *fine-tuning*, que son parte de la base del presente trabajo.

Ioana Branescu et al. [31] diseñaron e implementaron un sistema de reconocimiento facial que funciona en un contexto distribuido. Cabe destacar que no se trata de un sistema que implementa aprendizaje federado. La razón principal es que la detección de rostros y la extracción de características se realiza localmente mediante el modelo

preentrenado FaceNet [32], pero dichas características se envían a un servidor central para realizar el entrenamiento del modelo global de manera centralizada. Este trabajo, pese a que utiliza el mismo modelo preentrenado para la extracción de características faciales que en el presente trabajo, tiene un enfoque diferente, puesto que no proporciona privacidad y no emplea la arquitectura usual de aprendizaje federado.

Mohsen Heidari et al. [33] desarrollaron un sistema de reconocimiento facial basado en una arquitectura de red siamesa. Básicamente, una red siamesa consta de dos modelos similares en los que se introducen dos imágenes y se determina si pertenecen a la misma persona o no, basándose en un criterio de similitud. En dicho trabajo se hace uso del modelo VGG16 [34] preentrenado con la base de datos ImageNet [35], así como de la distancia euclidiana para calcular el nivel de similitud. Los resultados demuestran una precisión del 95,62 % en la base de datos LFW [36]. Jianming Zhang et al. [37] realizaron un trabajo similar al de [33]. En este caso, desarrollaron dos sistemas basados en una arquitectura de red siamesa, alcanzando la misma precisión con ambos sistemas. La diferencia radica en que el segundo sistema es más ligero, es decir, contiene una menor cantidad de parámetros. En dicho trabajo, la red siamesa minimiza la función de pérdida contrastiva [38] para ampliar la medida de similitud en los rostros de la misma persona y reducirla en los rostros de personas diferentes. Esta función de pérdida también está basada en la distancia euclidiana. Es importante destacar que los dos trabajos previamente descritos están fuertemente alineados entre sí y proporcionan un sistema de verificación de identidades adecuado. Sin embargo, no presentan el uso de aprendizaje federado, lo que hace que difieran ligeramente de los objetivos del presente trabajo.

Haipeng Zheng et al. [39] desarrollaron un *framework* para el reconocimiento facial mediante una arquitectura descentralizada de aprendizaje federado en la que se utiliza la tecnología Blockchain en sustitución del servidor centralizado. Los diferentes nodos que forman la red descentralizada evalúan la calidad de los modelos, aplicando un mecanismo de incentivos para estimular a los usuarios más fiables a participar en el entrenamiento. Pese a que el trabajo [39] es interesante porque sustituye el servidor central, que es uno de los puntos críticos del aprendizaje federado, por la tecnología Blockhain, este no sería aplicable en el tipo de escenario en el que se enfoca este TFM. Además, carece del uso de *transfer learning* y *fine-tuning*.

Suleman Khan et al. [40] desarrollaron un *framework* de reconocimiento facial utilizando el modelo preentrenado AlexNet [41]. Después de entrenarlo en cuatro clases diferentes, cada una con 1000 imágenes, los resultados muestran una precisión del 97,95 %. Maheen Zulfiqar et al. [42] desarrollaron un modelo de reconocimiento facial de 30 personas, en el que las imágenes de entrenamiento son aumentadas, incorporando imágenes con diferentes condiciones de iluminación y ruido. Según los resultados, se obtiene una precisión del 98.76 %. Estos dos últimos trabajos están principalmente relacionados con el reconocimiento facial y *transfer learning/fine-tuning*. Sin embargo, al igual que el trabajo [33], al no hacer uso del aprendizaje federado, no cubre los objetivos del presente trabajo. Además, la falta de un análisis de fiabilidad, como el que se proporciona en este trabajo, hace que difiera de los objetivos del mismo.

6. Análisis de Alternativas

Realizar un profundo análisis de alternativas que determine las opciones más apropiadas ayuda a maximizar el resultado del trabajo. En consecuencia, con base en los objetivos del trabajo, se ha tenido que realizar un estudio y selección de las alternativas más adecuadas para los siguientes 3 aspectos del proyecto: entrenamiento del modelo, librería de *Deep Learning* y métrica de comparación de similitud.

6.1. Entrenamiento del Modelo

A medida que la Inteligencia Artificial, específicamente el *Deep Learning*, avanza, cada vez existen más variantes de técnicas de aprendizaje de modelos para resolver problemas de reconocimiento facial. Por ello, a través de este primer análisis se desea determinar si, para este trabajo en concreto, es preferible entrenar un modelo desde cero o utilizar las técnicas *transfer learning* o *fine-tuning*.

- **Modelo entrenado desde cero**. Entrenar un modelo desde cero consiste en inicializar los parámetros del modelo (normalmente, aleatoriamente) y optimizarlos con base en el conjunto de datos de entrenamiento. Este proceso requiere de grandes recursos computacionales, específicamente, de GPU potentes. Además, en función del número de parámetros entrenables del modelo y del tamaño del conjunto de datos de entrenamiento, el tiempo de entrenamiento puede variar considerablemente. Esta alternativa requiere de una gran cantidad de datos de entrenamiento para evitar que el modelo se sobreajuste y en consecuencia, generalice correctamente frente a datos no vistos anteriormente. Asimismo, cabe destacar que entrenar un modelo desde cero proporciona un control completo, aumentando la flexibilidad y la personalización del mismo.
- ***Transfer learning***. La técnica de *transfer learning* consiste en utilizar un modelo preentrenado con un conjunto de datos grande para aplicarlo a nueva tarea en la que el conjunto de datos es menor. Con este fin, se congelan las capas encargadas de la extracción de características y se sustituyen las últimas capas clasificadoras por nuevas capas específicas para la nueva tarea (solo estas últimas capas son entrenadas). Al tener que entrenar solamente las últimas capas, los recursos computacionales requeridos son menores, y el tiempo de entrenamiento también disminuye. La cantidad de datos nuevos no tiene que ser tan grande como la utilizada para el entrenamiento previo del modelo. Sin embargo, este enfoque implica un menor control y personalización del modelo.
- ***Fine-tuning***. A diferencia del *transfer learning*, el *fine-tuning* implica el reentrenamiento de determinadas capas preentrenadas. Al tener que reentrenar capas pre-

entrenadas y entrenar las últimas, los recursos computaciones necesarios son mayores, y el tiempo de entrenamiento asciende ligeramente respecto a la técnica *transfer-learning*. Cabe destacar que el conjunto de datos empleado debe ser similar al conjunto de datos original en el que se entrenó el modelo. Al igual que la técnica de *transfer learning*, se tiene un menor control del modelo en comparación con el entrenamiento del modelo desde cero.

Los criterios de evaluación para la selección entre el entrenamiento del modelo desde cero o el uso de la técnica *transfer learning* o *fine-tuning* se definen a continuación.

- **CE1 - Recursos computacionales (20 %)**. Se refiere a la necesidad de capacidad de computación para realizar el procesamiento de imágenes y el entrenamiento del modelo.
- **CE2 - Tiempo de entrenamiento (20 %)**. Se refiere al tiempo que el modelo necesita para ser entrenado.
- **CE3 - Disponibilidad de datos (15 %)**. Se refiere a la cantidad de nuevos datos disponibles para entrenar el modelo.
- **CE4 - Control sobre el modelo (10 %)**. Se refiere a la facilidad de personalización del modelo, así como la modificación del mismo.
- **CE5 - Objetivo del modelo (35 %)**. Se refiere al objetivo específico por el que se entrena el modelo. Para este trabajo, el modelo debe ser entrenado para crear vectores de características discriminatorios, es decir, no debe ser entrenado para realizar la clasificación de individuos.

En la Tabla 1 se comparan las alternativas según los criterios de evaluación mencionados previamente.

	Desde cero	Transfer learning	Fine-tuning
CE1 (20 %)	3	10	8
CE2 (20 %)	3	9	7
CE3 (15 %)	2	9	3
CE4 (10 %)	9	4	4
CE5 (35 %)	8	3	9
Media ponderada	5,2	6,6	**7**

Tabla 1: Comparación de alternativas de entrenamiento del modelo.

Con base en los resultados presentados en la Tabla 1, se concluye que la alternativa más adecuada es el uso de la técnica ***fine-tuning***. La razón principal radica en que el objetivo del modelo para este trabajo es la creación de vectores de características discriminatorios, en lugar de llevar a cabo una tarea de clasificación. En consecuencia, la técnica de *transfer learning* queda descartada, puesto que esta se enfoca en el entrenamiento de las últimas capas clasificadoras. Sin embargo, es importante señalar que no se tiene la misma cantidad de datos nuevos que los utilizados para el entrenamiento previo del modelo, y que los recursos computacionales no son los más apropiados para llevar a cabo la técnica *fine-tuning*, puesto que no se dispone de GPU.

Respecto al entrenamiento del modelo desde cero, no se considera una opción viable debido a que la cantidad de datos disponibles es escasa, lo que podría ocasionar

sobreajuste. Además, los recursos computaciones disponibles no son los adecuados para llevar a cabo un entrenamiento del modelo desde cero.

En la Figura 14 se presenta la comparación entre las distintas alternativas.

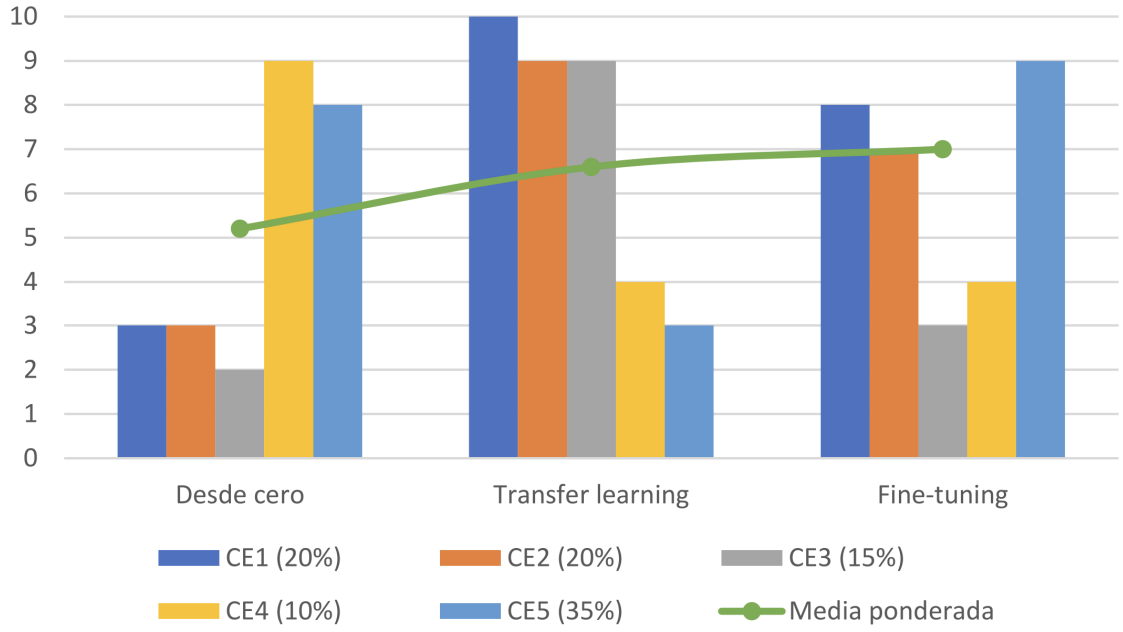


Figura 14: Comparación de alternativas de entrenamiento del modelo.

6.2. Librería de Deep Learning

En el ámbito del *Deep Learning*, la elección de la librería adecuada puede inferir en los resultados finales de manera significativa. En consecuencia, a continuación se realiza una comparación de las librerías de *Deep Learning* más utilizadas en la actualidad. Esta comparación se basa en el trabajo [43].

- **TensorFlow** [44]. Es un *framework* desarrollado por Google que se convirtió en una herramienta de código abierto en el año 2015. Aunque no se caracteriza por su fácil y rápido aprendizaje, entre las tres alternativas, es la segunda más popular, después de Keras. En cuanto a la velocidad de entrenamiento y ejecución de modelos, muestra un rendimiento alto y veloz, similar al de PyTorch.
- **PyTorch** [45]. Se trata del *framework* de *Deep Learning* más reciente de los tres, desarrollado por Facebook. Al igual que TensorFlow, se convirtió en herramienta de código abierto, en este caso en el año 2016. Pese a que es la menos popular de las tres, su aprendizaje es muy sencillo e intuitivo, y se caracteriza por su alto y rápido rendimiento.
- **Keras** [46]. Es una API de código abierto de alto nivel publicada en el año 2015. En el año 2017 fue adoptada por TensorFlow, aunque todavía es posible utilizarla independientemente de TensorFlow. Una de sus ventajas más destacadas es la documentación proporcionada, con ejemplos codificados, para que los usuarios comprendan rápidamente los conceptos. Sin embargo, no destaca por su rápido y alto rendimiento.

Seguidamente, se describen los criterios utilizados para llevar a cabo el análisis de alternativas.

- **CE1 - Código abierto (10 %)**. Se refiere a la disponibilidad gratuita del software y la capacidad de cualquier individuo para utilizarlo sin restricciones.
- **CE2 - Curva de aprendizaje (15 %)**. Se refiere a la facilidad y rapidez para aprender a utilizar la herramienta software.
- **CE3 - Popularidad (15 %)**. Se refiere a la frecuencia de implementación y uso en el ámbito académico.
- **CE4 - Eficiencia en velocidad y rendimiento (15 %)**. Se refiere a la rapidez y eficacia de entrenamiento y ejecución de código.
- **CE5 - Código de punto de partida (45 %)**. Se refiere a la existencia de código base y ejemplos prácticos que están disponibles para su uso.

En la Tabla 2 se puntúan las diversas alternativas con base en los criterios descritos anteriormente.

	TensorFlow	**PyTorch**	**Keras**
CE1 (10 %)	10	10	10
CE2 (15 %)	7	10	9
CE3 (15 %)	8	6	10
CE4 (15 %)	10	10	5
CE5 (45 %)	6	9	6
Media ponderada	7,45	**8,95**	7,3

Tabla 2: Comparación de alternativas de librería de *Deep Learning*.

Como se puede observar en la Tabla 2, la librería más apropiada a utilizar en este trabajo es **PyTorch**. Esta elección se fundamenta principalmente en el uso del código ya escrito y optimizado para PyTorch del trabajo [47], lo que facilita considerablemente el desarrollo del trabajo evitando la necesidad de iniciar el desarrollo del trabajo desde cero, y aprovechando las buenas prácticas implementadas en dicho trabajo. Además, su fácil y rápido aprendizaje favorecen su uso en el ámbito académico. Es importante destacar que, a pesar de que en la actualidad es la librería con menos renombre, su popularidad está aumentando entre los investigadores e investigadoras de Inteligencia Artificial.

Respecto a las librerías TensorFlow y Keras, se concluye que son librerías totalmente implementables en este trabajo. Sin embargo, el hecho de que la base del código esté programado con PyTorch supone una desventaja considerable para ambas alternativas.

A continuación, en la Figura 15 se muestra la comparación de las 3 alternativas.

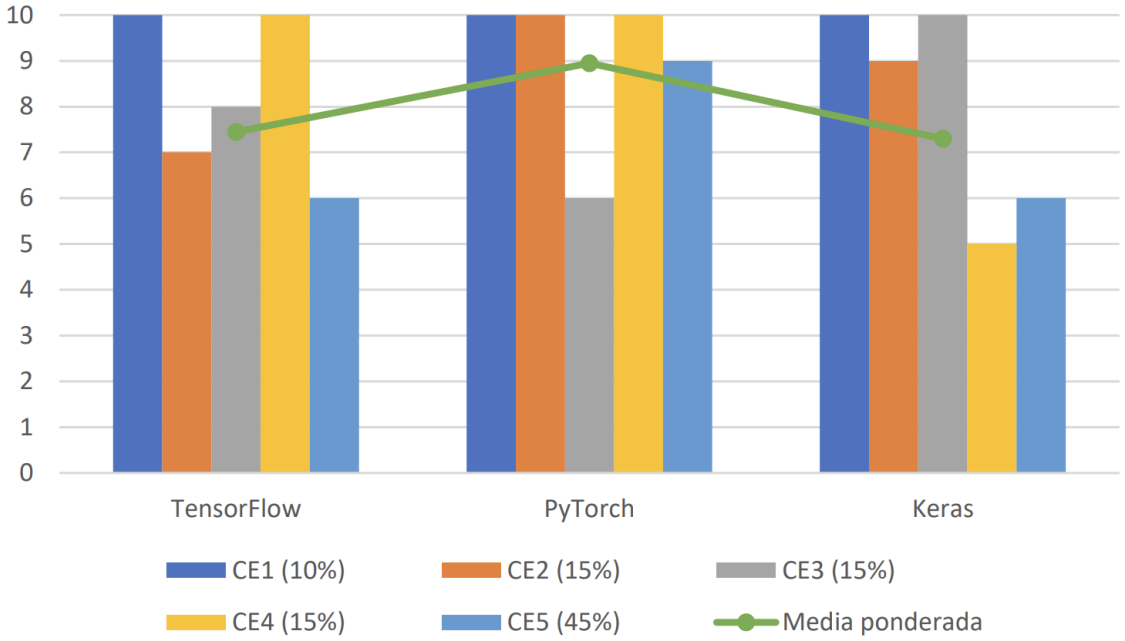


Figura 15: Comparación de alternativas de librería de *Deep Learning*.

6.3. Métrica de Comparación de Similitud

En este tercer análisis se comparan las diferentes métricas para evaluar la similitud entre vectores de características. Las alternativas posibles son la similitud del coseno, el punto de producto y la distancia euclidiana. Al igual que la comparativa anterior, esta comparativa se fundamenta en el trabajo [48].

- **Similitud del coseno**. Mide la similitud entre dos vectores mediante el coseno del ángulo formado por ambos, lo que hace que esta métrica sea invariante respecto a la magnitud de los vectores y adecuada para el análisis de datos de alta dimensión. Además, es fácil de interpretar y ofrece una similitud intuitiva. Sin embargo, al ignorar las diferencias de magnitud entre los vectores, puede no capturar adecuadamente las disparidades entre ellos, lo que podría afectar a la precisión de los resultados.
- **Punto de producto**. Calcula la suma de los productos de los elementos correspondientes de dos vectores, es decir, calcula la similitud basada en la proyección de un vector sobre otro. Pese a que su interpretabilidad no es tan intuitiva como la de la similitud del coseno, esta métrica es capaz de capturar tanto la dirección como la magnitud de los vectores, lo que proporciona resultados más precisos. No obstante, cabe destacar que es sensible a las diferencias de magnitud entre los vectores y puede volverse menos eficiente en dimensiones más altas debido al aumento en los cálculos requeridos.
- **Distancia euclidiana**. Determina la distancia directa entre dos puntos en un espacio multidimensional. Al igual que el punto de producto, esta métrica es capaz de capturar tanto la dirección como la magnitud de los vectores. Su compresión es factible y es sensible a la dimensionalidad. Sin embargo, su rendimiento se ve afectado por las diferencias de magnitud entre los vectores.

Para continuar, se describen los criterios en los que se fundamenta la comparativa que se presenta posteriormente.

- **CE1 - Robustez a la magnitud de los vectores (30 %)**. Se refiere a la capacidad de la métrica para manejar diversas magnitudes de vectores sin perjudicar el resultado final. Una métrica robusta debería reportar resultados firmes independientemente de la magnitud de los vectores.
- **CE2 - Balance entre interpretación y precisión (35 %)**. Se refiere a la simplicidad para interpretar los resultados de la métrica, así como a la precisión de la misma. Se busca una métrica comprensible y, a la vez, que proporcione resultados precisos.
- **CE3 - Robustez a la dimensión (35 %)**. Se refiere a la capacidad de la métrica para manejar espacios de alta dimensionalidad. Al igual que el criterio CE1, una métrica robusta debería mantener su rendimiento independientemente de la dimensionalidad de los vectores.

En la Tabla 3 se muestra la comparación entre las diferentes alternativas con base en los criterios definidos previamente.

	Similitud del coseno	**Punto de producto**	**Distancia euclidiana**
CE1 (30 %)	10	5	5
CE2 (35 %)	7	8,5	8,5
CE3 (35 %)	9	6	6
Media ponderada	**8,6**	6,575	6,575

Tabla 3: Comparación de alternativas de métricas de similitud.

Tras analizar los resultados de la Tabla 3, se deduce que la métrica más adecuada para este trabajo es la **similitud del coseno**. Su robustez tanto a la dimensión como a la magnitud de los vectores la hace destacar sobre las otras alternativas. Además, cabe destacar que su interpretación es más simple. Estas razones hacen que la similitud del coseno sea ampliamente empleada en diversas aplicaciones. Sin embargo, en cuanto a la precisión, puede verse afectada en ciertos casos.

Pese a que las alternativas restantes no son las más adecuadas para este trabajo, pueden ser útiles en otros escenarios.

En la Figura 16 se presenta la comparación gráfica de las alternativas.

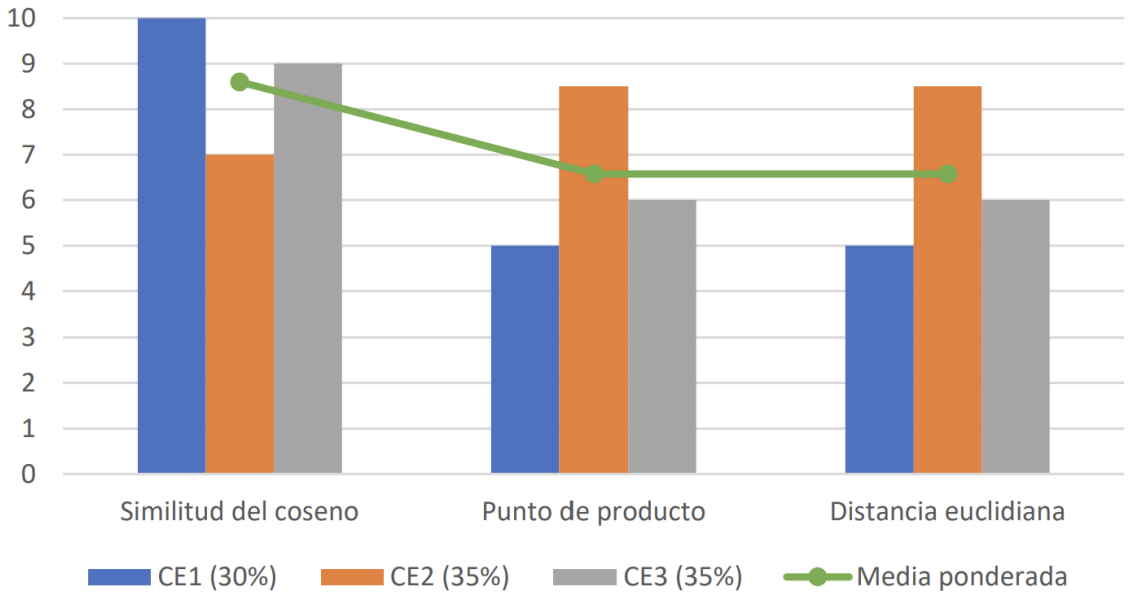


Figura 16: Comparación de alternativas de métricas de similitud.

7. Análisis de Riesgos

Todo trabajo de ingeniería conlleva determinados riesgos que pueden causar dificultades en su desarrollo. Por esta razón, el objetivo de este apartado es realizar un análisis de los riesgos inherentes a este trabajo para intentar minimizar su impacto al máximo posible.

Para ello, primero se definen los riesgos que pueden aparecer en el ciclo de vida del trabajo, así como la probabilidad de ocurrencia y el impacto que tendría cada riesgo en el resultado del trabajo. Posteriormente, se analizan dichos riesgos a través de la matriz de probabilidad-impacto. Esta es una herramienta que ayuda a gestionar los riesgos, priorizando aquellos que requieren más atención. Para terminar, se define un plan de contingencia donde se describen las medidas a seguir para cada riesgo identificado.

7.1. Definición de Riesgos

En este primer subapartado, se procede a identificar y describir los posibles riesgos que pueden aparecer durante el desarrollo del trabajo.

- **R1 - Inoperatividad del servidor**. Tanto el procesamiento de imágenes faciales como el entrenamiento del modelo requiere de recursos computacionales potentes. En consecuencia, se hace uso de un servidor para agilizar dichos procesos. No obstante, este servidor podría dejar de funcionar debido a cortes de luz, mantenimiento deficiente de la infraestructura universitaria o mala gestión del equipo físico. Esto supondría interrupciones en el desarrollo del trabajo, así como retrasos en la entrega de resultados.

  - Probabilidad de ocurrencia: 20 %
  - Impacto: 70 %

  La probabilidad de ocurrencia asignada a este primer riesgo es baja debido a que la infraestructura de equipos físicos del grupo de investigación es adecuada. De hecho, el servidor está ubicado en un CPD (Centro de Procesamiento de Datos), cumpliendo las medidas necesarias para garantizar su correcto funcionamiento. Sin embargo, puede ocurrir que se vaya la luz en la universidad, lo que supondría la inoperatividad del servidor. En cuanto al impacto, este sería notable, puesto que habría que esperar a que el servidor vuelva a estar disponible o buscar otra alternativa para continuar con el desarrollo del trabajo.

- **R2 - Modificaciones en la legislación**. En trabajos de reconocimiento facial, el uso de imágenes de rostros es un aspecto sensible y crítico. La manipulación de es-

te tipo de información a través de la Inteligencia Artificial implica legislación restrictiva que diferentes entidades, tanto nacionales como internacionales, podrían ver modificada debido a la actualización asociada a tecnologías novedosas, como es el caso de la Inteligencia Artificial, con el propósito de limitar la difusión de información facial para salvaguardar la privacidad de las personas. La razón principal radica en el uso indebido de esta información y la posible vulneración de los derechos individuales asociados a la privacidad. En el caso de que este riesgo se materialice, podría implicar tener que realizar cambios significativos en el desarrollo del trabajo para adaptarse a las nuevas regulaciones.

- Probabilidad de ocurrencia: 30 %
- Impacto: 90 %

En la actualidad, las entidades públicas están trabajando en la modificación de las leyes relacionadas con las limitaciones del uso de la Inteligencia Artificial, dada su rápida evolución en un periodo de tiempo relativamente corto. Sin embargo, el proceso de modificación de las leyes suele avanzar con lentitud. En consecuencia, se considera que la probabilidad de ocurrencia es baja en este periodo. Por otro lado, el impacto que supondría la modificación de la regulación de la Inteligencia Artificial sería significativo. Esto requeriría llevar a cabo un nuevo estudio y análisis de dicha ley, así como la posterior adaptación del trabajo a las disposiciones establecidas en la misma.

- **R3 - Desajustes del presupuesto**. Durante el desarrollo del trabajo pueden aparecer diversos desajustes que provoquen la necesidad de la modificación del presupuesto inicial. Estos pueden originarse por una incorrecta estimación de los gastos, lo que implica un aumento de los costes del trabajo. Por ello, estar preparado para posibles cambios económicos, como revisar las estimaciones iniciales, es transcendental.

  - Probabilidad de ocurrencia: 20 %
  - Impacto: 60 %

  El desajuste del presupuesto es un riesgo poco probable, dado que se ha realizado una exhaustiva planificación económica antes del inicio del trabajo. Además, los gastos que puedan originarse durante el transcurso del trabajo suelen tener poca relevancia en el presupuesto final. Por ende, se considera que el impacto es medio.

- **R4 - Incumplimiento de plazos**. Este último riesgo implica el retraso de las entregas de los resultados del trabajo. Estos pueden ser causados por diversos factores, como una inadecuada planificación, problemas originados durante el desarrollo del trabajo o falta de coordinación dentro del equipo.

  - Probabilidad de ocurrencia: 10 %
  - Impacto: 50 %

  Haber realizado una planificación exhaustiva y conservadora con plazos amplios que permitan amortiguar retrasos se traduce en una probabilidad mínima de ocurrencia. En cambio, en caso de que se produzcan retrasos, el impacto sería moderado.

7.2. Comparación de Riesgos

A continuación, en la Figura 17 se muestra la matriz de probabilidad-impacto, que es útil para concluir qué riesgos requieren más atención.

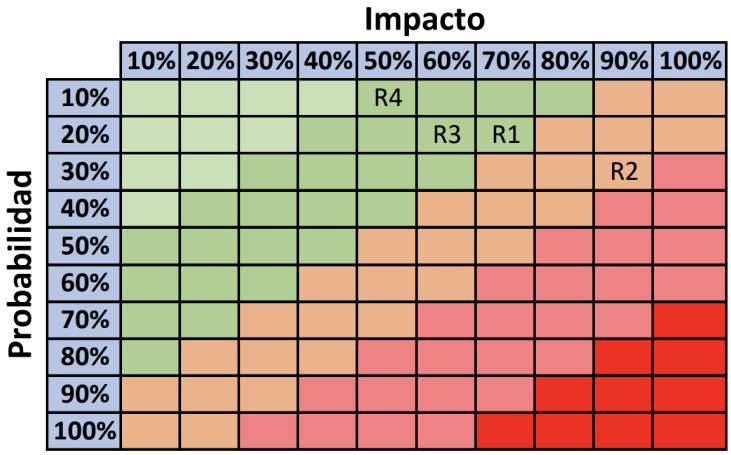


Figura 17: Matriz de probabilidad-impacto.

Con base en la matriz de probabilidad-impacto, se deduce que el riesgo que demanda atención continua es R2. Aunque su probabilidad de ocurrencia es baja, su impacto tendría consecuencias significativas. Por el contrario, los riesgos R1, R3 y R4 se encuentran en la zona aceptable de la matriz, ya que no representan una amenaza importante para el trabajo.

7.3. Plan de Contingencia

En el caso de que alguno de los riesgos se materialice, o para minimizar la probabilidad de ocurrencia, contar con un plan para reducir al máximo su impacto ayuda a mantener el proyecto en curso y mitigar sus efectos negativos. Seguidamente, se describe el plan de contingencia propuesto.

- **R1 - Inoperatividad del servidor**. Para evitar que el servidor sufra problemas que paralicen el desarrollo del trabajo, es importante llevara cabo mantenimientos regulares del servidor realizados por un equipo de soporte técnico. Además, la monitorización constante del servidor es esencial para detectar posibles problemas que puedan indicar la necesidad de una revisión o modificación. Respecto a los cortes de luz, implementar un sistema de alimentación ininterrumpida aseguraría la continuidad del funcionamiento del servidor.
- **R2 - Modificaciones en la legislación**. Este riesgo puede ser mitigado manteniéndose informado de las modificaciones realizadas en las leyes ya existentes. Para ello, es fundamental la colaboración con expertos en temas legales y regulaciones relacionadas con la Inteligencia Artificial y el reconocimiento facial.
- **R3 - Desajustes del presupuesto**. Para reducir el impacto de este riesgo, es necesario realizar un presupuesto inicial detallado, considerando todos los elementos

que conforman el mismo. También, dejar un margen para los posibles gastos imprevistos ayuda a que el impacto se vea reducido. En el caso de que la desviación del presupuesto sea significativa, es importante buscar una solución entre las diferentes partes que constituyen el trabajo.

- **R4 - Incumplimiento de plazos**. Para prevenir los retrasos en los plazos, se ha de realizar una planificación real donde las diversas fases del trabajo tomen el tiempo necesario. Asimismo, un seguimiento y control del avance del trabajo favorece a cumplir los plazos. En el caso de que no se cumplan los plazos definidos, una opción a tener en cuenta es solicitar una prórroga del trabajo para garantizar su finalización.

8. Descripción de la Solución

En este apartado se procede a explicar el diseño y la implementación de la solución propuesta.

Para una compresión e interpretación adecuada de los resultados, se considera imprescindible describir el diseño. Por ello, en primer lugar, se presenta la arquitectura general, para posteriormente describir en detalle los diversos módulos que la forman.

Al igual que el diseño se considera la base de todo trabajo, una explicación detallada de la implementación es esencial. Así, se garantiza que los resultados obtenidos son válidos, y se permite observar la viabilidad del diseño definido. El objetivo es incluir la descripción de la plataforma hardware y software utilizada para llevar a cabo la implementación del trabajo. Asimismo, se detallan las diferentes fases del proceso, incluyendo el papel que desempeñan las herramientas empleadas en cada una de ellas.

Antes de especificar la solución propuesta, se recuerda que la aplicación del sistema desarrollado en este TFM es adaptable a diversos escenarios. Esto abarca desde la identificación de pacientes con y sin una determinada patología hasta la distinción entre personas delincuentes y no delincuentes en entornos como aeropuertos y eventos deportivos. Sin embargo, en este apartado, para facilitar la lectura y la compresión, se emplean los términos *delincuentes* y *no delincuentes* como las dos categorías en las que se clasifican los rostros.

8.1. Diseño del Sistema

El diseño del sistema desarrollado se divide en 2 apartados. Primero, se presenta la visión general del sistema, que incluye el diseño de alto nivel. Posteriormente, se describen los módulos que componen el sistema, lo que equivale al diseño de bajo nivel.

8.1.1. Visión General del Sistema de Identificación

Mediante este sistema basado en el reconocimiento facial e Inteligencia Artificial, principalmente se busca identificar de manera precisa un tipo de rostros, clasificándolos como delincuentes. Asimismo, se desea evitar que los demás rostros, pertenecientes a otras personas, sean identificados o clasificados como delincuentes. En definitiva, el sistema tiene como propósito principal realizar una correcta identificación y diferenciación entre personas delincuentes y no delincuentes.

El sistema de reconocimiento facial en cuestión está dividido en 2 subsistemas principales: el sistema de reentrenamiento y el sistema de evaluación. Antes de presentar la

arquitectura general del sistema, se considera importante mencionar que el reentrenamiento de un modelo posibilita la adaptación a tareas específicas de un modelo que ha sido previamente entrenado de manera general.

En el ámbito del reconocimiento facial, existen diferentes motivos por el que el reentrenamiento de un modelo es importante; por ejemplo, cambios físicos en las personas, envejecimiento, etc. Sin embargo, en este caso se recurre el reentrenamiento debido a la falta de una base de datos masiva y a la capacidad computacional insuficiente para entrenar el modelo desde cero. Por tanto, en este TFM, debido a su dimensión y características, resulta fundamental hacer uso de un modelo en el que ya se ha invertido tiempo y dinero para su entrenamiento. En otras palabras, se emplea un modelo previamente entrenado con gran cantidad de imágenes, y se aprovechan las habilidades de las capas del modelo para adaptarlo a esta tarea específica.

En la Figura 18 se puede apreciar el diseño general de la arquitectura del sistema, los 2 subsistemas que lo componen y los elementos que forman cada uno de ellos.

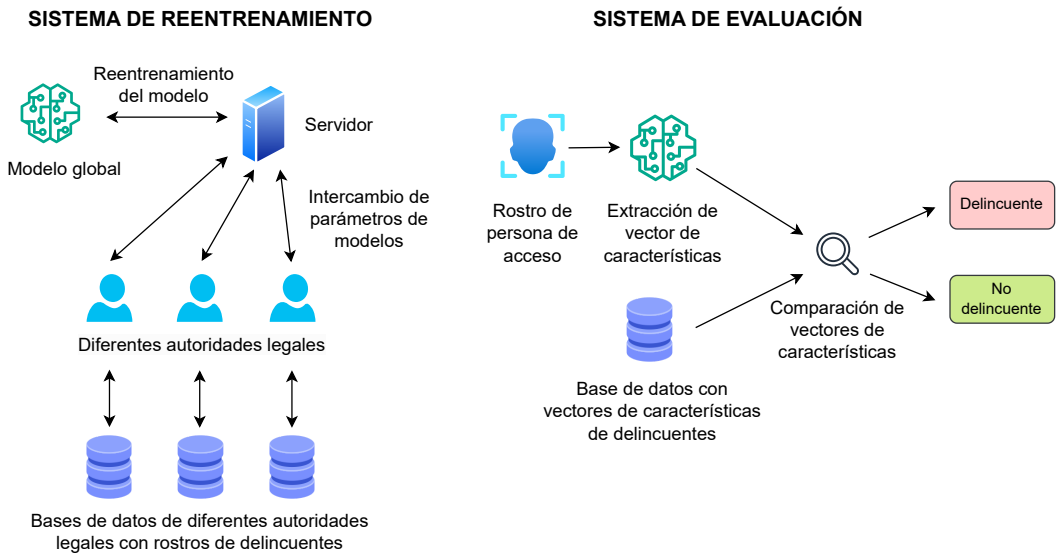


Figura 18: Diseño general del sistema de reconocimiento facial.

8.1.2. Diseño del Sistema de Reentrenamiento

Tal y como se ha indicado anteriormente, uno de los módulos principales del diseño es el sistema de reentrenamiento. Este sistema tiene como función fundamental reentrenar un sistema previamente entrenado con una cantidad masiva de imágenes de personas, con imágenes adicionales de las personas delincuentes a través del aprendizaje federado. A continuación, se describen los principales aspectos de este módulo.

8.1.2.1. Reentrenamiento del Modelo

En cuanto al reentrenamiento del modelo, se diferencian principalmente dos técnicas: reentrenamiento total y reentrenamiento parcial. El reentrenamiento total implica el ajuste de todos los parámetros entrenables del modelo al nuevo conjunto de datos de entrenamiento. En cambio, mediante el reentrenamiento parcial se busca congelar

ciertas capas y reentrenar las restantes, con el objetivo de mantener la capacidad de generalización.

En este trabajo se busca realizar un estudio para determinar qué técnica es la que mejor se adapta al problema en cuestión. En el caso del reentrenamiento total, todas las capas que contengan parámetros entrenables son modificadas. Por el contrario, en el caso del reentrenamiento parcial, se exploran diversos subescenarios, es decir, se realiza la congelación y ajuste de diferentes capas con el objetivo de encontrar la cantidad óptima de capas a congelar y ajustar. Mediante este estudio se busca diseñar el sistema de reentrenamiento más adecuado para el escenario de este trabajo.

8.1.2.2. Componentes

A continuación, se describen los componentes que forman parte del sistema de reentrenamiento.

8.1.2.2.1 Clientes

Los clientes representan las diferentes organizaciones que colaboran en el proceso de reentrenamiento del modelo. Este proceso se lleva a cabo sin la necesidad de que cada cliente exponga su información confidencial, que incluye la información de rostros que el sistema considera delincuentes. Esto se debe a problemas legales relacionados con cuestiones de privacidad.

Como el reentrenamiento se realiza en un entorno de aprendizaje federado, cada cliente entrena el modelo de manera local utilizando los parámetros iniciales del modelo global recibidos por el servidor y su propio conjunto de datos. Una vez terminado el reentrenamiento, cada uno de ellos envía al servidor los parámetros actualizados, con el propósito de que este realice la agregación de los diferentes parámetros actualizados obtenidos de los diversos clientes. Este proceso se explica en detalle en el apartado 2.3.3.

8.1.2.2.2 Bases de Datos de los Clientes

Las imágenes faciales de los delincuentes empleadas en el reentrenamiento del modelo se encuentran almacenadas en la base de datos de cada cliente. En otras palabras, cada cliente dispone de imágenes faciales de los delincuentes buscados, adquiridas mediante sus propios procesos de captura de rostros delictivos.

8.1.2.2.3 Modelo Global de Aprendizaje Automático

El modelo global de aprendizaje automático debe ser capaz de extraer características faciales y aprender diferentes patrones, haciendo diferenciaciones entre las distintas personas. Para ello, es importante tener un conjunto diverso de imágenes de rostros. Esto permite al modelo aprender y generalizar correctamente frente a imágenes faciales previamente no vistas.

Se considera ventajoso aprovechar el conocimiento de un modelo que ha sido previamente entrenado con una cantidad masiva de imágenes. Además, a este modelo

se le aplica un proceso de reentrenamiento y ajuste de parámetros con base en las imágenes faciales de los delincuentes buscados. Esta técnica es conocida como *fine-tuning* [15] (ver apartado 2.2.2).

8.1.2.2.4 Servidor

En el proceso de reentrenamiento mediante aprendizaje federado, el servidor se encarga de gestionar el intercambio de los parámetros, tanto globales como locales, entre los clientes y él mismo. Los parámetros globales están vinculados con el modelo global que se desea generar, mientras que los parámetros locales se relacionan con los modelos locales que los clientes reentrenan.

De igual manera, realiza la agregación de los parámetros de los modelos locales de los clientes. En otras palabras, es el responsable de salvaguardar la confidencialidad y seguridad del modelo.

8.1.2.2.5 Método de Agregación

Pese a que el método de agregación empleado en el servidor para realizar la agregación de los parámetros locales no aparece en la Figura 18, este desempeña un papel fundamental en el aprendizaje federado. La razón principal radica en que el modelo global debe ser capaz de consolidar todos los conocimientos obtenidos de forma local.

8.1.2.3. Fases de la Operación

Una vez descritos los elementos que forman parte del sistema de reentrenamiento, se mencionan los pasos a seguir para realizar el reentrenamiento del modelo. Este proceso se estructura en 6 fases diferentes: recopilación de imágenes de delincuentes, división de imágenes de delincuentes en conjuntos de reentrenamiento y prueba, limpieza y procesamiento de imágenes de reentrenamiento de delincuentes, reentrenamiento del modelo y almacenamiento del modelo global.

Antes de profundizar en cada fase, es importante destacar que las imágenes, tanto de los delincuentes como de los no delincuentes, provienen de una base de datos. Esto implica que determinadas personas en la base de datos se consideran delincuentes y las restantes, no delincuentes.

8.1.2.3.1 Recopilación de Imágenes de Delincuentes

Las imágenes faciales de los delincuentes deben presentar diversas posiciones, expresiones faciales y condiciones de iluminación para que la diversidad de imágenes faciales permita al modelo aprender y reconocer patrones de manera más amplia en el proceso de reentrenamiento. Reentrenar el modelo solo con un cierto tipo de imágenes conlleva el riesgo de que el modelo resultante esté sesgado y no sea tan efectivo al enfrentarse con imágenes previamente no vistas.

Además, se considera necesario que las imágenes de los delincuentes tengan una calidad/resolución óptima para que posteriormente el modelo sea capaz de llevar a cabo

la identificación de forma adecuada. Por otro lado, conviene señalar que la base de datos debe estar etiquetada, puesto que es importante saber a qué delincuente pertenece cada imagen, y que el volumen de imágenes para reentrenar el modelo debe ser adecuado para identificar adecuadamente a los delincuentes.

8.1.2.3.2 División de Imágenes de Delincuentes en Conjuntos de Reentrenamiento y Prueba

Una vez que las imágenes de los delincuentes son recopiladas, la siguiente fase consiste en separarlas en 2 conjuntos de datos: reentrenamiento y prueba.

Por un lado, las imágenes de reentrenamiento se utilizan para reentrenar el modelo mediante aprendizaje federado. Con base en estas imágenes, los clientes ajustan los parámetros recibidos del modelo global de manera local. Por otro lado, las imágenes de prueba se emplean en el sistema de evaluación para medir la fiabilidad del modelo para identificar a dichos delincuentes.

Con base en las consideraciones comunes en el mundo de la Inteligencia Artificial, se opta por una división estándar de reentrenamiento y prueba de 80 % y 20 %, respectivamente. Esta proporción es muy habitual, puesto que permite al modelo aprender con una cantidad sustancial de información, y al mismo tiempo, se dispone de una cantidad adecuada de imágenes para evaluar después la fiabilidad del modelo.

8.1.2.3.3 Limpieza y Procesamiento de Imágenes de Reentrenamiento de Delincuentes

Las imágenes de reentrenamiento de los delincuentes pueden haber sido capturadas en diferentes situaciones, lo que implica que podrían variar en tamaño o contener elementos adicionales, como cuerpos o fondos. Sin embargo, al tratarse de un problema de reconocimiento facial, es crucial que las imágenes sean procesadas para asegurar que estas son del mismo tamaño y que solamente contienen rostros, sin ningún ruido extra.

8.1.2.3.4 Reentrenamiento del Modelo

Llegados a este punto, se dispone de una base de datos de delincuentes procesada y separada en los conjuntos de reentrenamiento y prueba. El siguiente paso implica el reentrenamiento del modelo mediante aprendizaje federado. Por consiguiente, es importante seleccionar adecuadamente los parámetros relacionados con el aprendizaje federado.

- Número de rondas de reentrenamiento. Este parámetro se refiere al número de iteraciones en las que se reentrena el modelo global mediante el aprendizaje federado.
- Número de clientes que participan en el reentrenamiento.
- Número de clientes que participan en cada ronda del reentrenamiento. Se pueden dar situaciones en las que no todos los clientes participen en todas las rondas del reentrenamiento, debido a problemas de disponibilidad de datos, capacidad computacional, conectividad, etc.

- Método de agregación de los parámetros de los modelos locales (ver apartado 2.3.5).
- Tipo de distribución de las imágenes de los clientes. Este parámetro se refiere a la similitud en la cantidad de imágenes de cada delincuente que utiliza cada cliente para reentrenar el modelo. Para este caso en concreto, se considera el uso de la distribución IID (ver apartado 2.3.4).

Asimismo, también se deben elegir correctamente los principales parámetros vinculados con el aprendizaje realizado por cada cliente, es decir, los parámetros de *Deep Learning*.

- Arquitectura del modelo preentrenado con imágenes faciales. Se refiere a la red convolucional empleada para realizar la correcta identificación de delincuentes y no delincuentes.
- Número de épocas en cada reentrenamiento local. Se refiere al número de iteraciones con las que se reentrena el modelo de manera local.
- Tamaño de lote de reentrenamiento y prueba. Este parámetro se establece con el objetivo de que el modelo no procese todas las imágenes en una sola interacción, es decir, las imágenes se organizan en lotes para que el modelo sea capaz de procesarlas de manera más eficiente y aprenda de forma continua.
- Tasa de aprendizaje. Indica la magnitud con la que los parámetros del modelo son ajustados en el proceso de reentrenamiento. Una tasa de aprendizaje alta permite que el modelo converja rápido, pero puede llevar a oscilaciones alrededor del mínimo global. En cambio, una tasa de aprendizaje baja implica que el modelo converja lentamente con el riesgo de quedarse atrapado en mínimos locales, no en el mínimo global.
- Uso de procesador (CPU, Central Processing Unit) o tarjeta gráfica (GPU, Graphics Processing Unit). En caso de contar con una GPU, se recomienda su uso para cumplir los plazos definidos para este TFM. Además, su diseño permite procesar imágenes de forma más eficiente y veloz.

8.1.2.3.5 Almacenamiento del Modelo Global

Una vez el modelo global es reentrenado de manera exitosa, debe ser almacenado de forma segura para posteriormente ser implementado en los escenarios en los que se vaya a aplicar.

8.1.3. Diseño del Sistema de Evaluación

Todo modelo entrenado debe ser evaluado frente a imágenes o datos previamente no vistos en el reentrenamiento para concluir si el modelo es adecuado cuando se enfrenta a la identificación de datos nuevos. En consecuencia, a través de este sistema de evaluación se busca evaluar el modelo reentrenado en el sistema previo. En otras palabras, el sistema de evaluación se centra en evaluar y valorar el funcionamiento y la fiabilidad del modelo previamente reentrenado con imágenes de los delincuentes.

El funcionamiento del sistema de evaluación es el siguiente: la imagen capturada de la persona que se pretende identificar en una de las categorías definidas (delincuente o no delincuente) es comparada con las imágenes almacenadas de los delincuentes buscados. Si la persona es delincuente, el sistema así lo indica; sin embargo, si es una persona no delincuente, no se le identifica incorrectamente como delincuente. En la Figura 18 se puede ver el proceso de evaluación de forma gráfica.

8.1.3.1. Componentes

Al igual que el sistema de reentrenamiento, el sistema de evaluación está formado por una variedad de componentes.

8.1.3.1.1 Base de Datos de Delincuentes y No Delincuentes

Esta base de datos está constituida por las imágenes de prueba de los delincuentes, así como por una cantidad adicional de imágenes de personas no delincuentes. Es importante destacar que las imágenes de los delincuentes utilizadas en el reentrenamiento del modelo no son empleadas en este sistema de evaluación.

8.1.3.1.2 Extractor de Características Faciales

El extractor de características faciales corresponde a la parte convolucional del modelo previamente reentrenado. En el sistema de evaluación solo interesa trabajar con la parte convolucional, ya que es la encargada de devolver un vector de características asociado a la imagen ingresada. Cuanto más diferentes sean estos vectores, mejor se podrá diferenciar a las personas en general.

8.1.3.1.3 Comparador de Similitud de Rostros

El último componente que forma el sistema de evaluación es el comparador. Este se encarga de comparar e indicar cuánto se parecen o no parecen la imagen que se pretende identificar con las imágenes almacenadas de los delincuentes, con el propósito de determinar si dicha persona es delincuente o no.

8.1.3.2. Fases de la Operación

A continuación, se describen las diferentes fases que se llevan a cabo en el sistema de evaluación. Este proceso se estructura en 7 fases diferentes: recopilación de imagen a identificar, recopilación de imágenes de delincuentes, limpieza y procesamiento de imágenes, extracción de características faciales, comparación de características faciales y evaluación de fiabilidad del modelo.

Es importante mencionar que este proceso es ejecutado múltiples veces con diferentes imágenes a identificar, tanto de delincuentes como de no delincuentes, con el objetivo de observar el comportamiento del sistema.

8.1.3.2.1 Recopilación de Imagen a Identificar

Esta primera fase comienza con la captura de la imagen que se desea identificar, designada como imagen a identificar. Esta imagen es la que se emplea posteriormente en la comparación para determinar su estatus delictivo.

8.1.3.2.2 Recopilación de Imágenes de Delincuentes

Seguidamente, se recopilan imágenes de los delincuentes que se desean identificar. Estas son conocidas como imágenes almacenadas de los delincuentes, y se utilizan para comparar con la imagen del rostro que se pretende reconocer. Todas ellas deben ser imágenes en las que se identifique adecuadamente el rostro, lo que garantiza una correcta comparación.

8.1.3.2.3 Limpieza y Procesamiento de las Imágenes

Posteriormente, se realiza el mismo procesado que en el sistema de reentrenamiento a la imagen a identificar y a las imágenes almacenadas de los delincuentes.

8.1.3.2.4 Extracción de Características Faciales

Una vez procesadas la imagen a identificar y las imágenes almacenadas de los delincuentes, se lleva a cabo la extracción de características faciales de todas ellas, para su posterior comparación. Para ello, se hace uso de la parte convolucional del modelo reentrenado, la cual genera un vector de características.

8.1.3.2.5 Comparación de Características Faciales

Tras obtener todos los vectores de características, se procede a comparar el vector de características de la imagen a identificar con los vectores de características de las imágenes almacenadas de los delincuentes. El objetivo es determinar a qué delincuente se asemeja más la persona que se está procesando. Dicho en otras palabras, la salida de esta fase consta de tres puntos: el nombre de la persona procesada, el nombre del delincuente al que más se parece y la similitud entre ambos.

8.1.3.2.6 Evaluación de Fiabilidad del Modelo

Para concluir, se realiza la evaluación de la fiabilidad del modelo, es decir, se decide si el modelo efectúa una correcta identificación. Con tal propósito, se emplea la matriz de confusión, analizando los verdaderos positivos, falsos negativos, falsos positivos y verdaderos negativos obtenidos. Además, se calculan las métricas de sensibilidad y especificidad para evaluar la fiabilidad del modelo (ver apartado 2.1.1.2.2). Estas métricas indican el porcentaje de delincuentes y no delincuentes correctamente identificados, respectivamente (más adelante se describen detalladamente).

8.2. Implementación del Sistema

Después de haber definido el diseño del sistema, a continuación, se especifica en detalle la implementación llevada a cabo. Esto implica la descripción de la plataforma desarrollada y las diversas fases de la implementación.

8.2.1. Plataforma de Desarrollo

Para replicar la implementación descrita más adelante, se considera necesario describir la plataforma tanto hardware como software.

8.2.1.1. Plataforma Hardware

Para llevar a cabo el desarrollo de este sistema, se hace uso de un servidor Dell PowerEdge R730. Este cuenta con un procesador Intel(R) Xeon(R) CPU E5-2690 v3 @ 2.60GHz y una memoria RAM de 128 GB. Este servidor se encuentra ubicado en el CPD del grupo de investigación, donde se cumplen los requisitos de seguridad y conectividad necesarios.

8.2.1.2. Plataforma Sofware

En cuanto al software, el servidor empleado dispone del sistema operativo Ubuntu 20.04.6 LTS junto a la versión 3.9.5 de Python.

Con el propósito de simular tanto a las personas delincuentes que están en búsqueda como a las personas no delincuentes, se hace uso de la base de datos Pins Face Recognition [49] de Kaggle. Esta contiene imágenes de 105 personas, y se trata de una base de datos etiquetada. Además, se pueden encontrar imágenes tanto de hombres como de mujeres con diferentes resoluciones, posiciones faciales e iluminación.

Asimismo, en el ámbito del *Deep Learning*, se opta por utilizar la librería PyTorch [45] de Python debido a que ofrece flexibilidad en la construcción y entrenamiento de modelos. Esta herramienta ofrece una documentación completa y de simple interpretación, lo que facilita su uso desde el principio. Además, PyTorch implementa otras sublibrerías de las que se puede hacer uso en función del problema que se desee resolver. Por ejemplo, para trabajos relacionados con el procesamiento de imágenes o vídeos, proporciona la sublibrería torchvision [50], la cual es utilizada en este trabajo.

Aparte de PyTorch, se emplea la librería facenet-pytorch [51] de Python, la cual ha sido desarrollada y adaptada para su uso con PyTorch, consecuencia del artículo [32]. Para este trabajo en particular, se hace uso de ella para aprovechar 2 de sus componentes.

En primer lugar, el modelo base empleado en los sistemas de reentrenamiento y evaluación para el desarrollo del sistema de identificación de delincuentes se trata de InceptionResnetV1. La mencionada librería facenet-pytorch permite cargar, junto al modelo, los parámetros preentrenados, los cuales se basan en el entrenamiento realizado con la base de datos VGGFace2 [52] o Casia-WebFace [53]. Dicha acción ofrece la facilidad de cargar un modelo previamente entrenado con una cantidad masiva de rostros. Como resultado, se logra un gran ahorro de tiempo al no tener que entrenar un modelo desde cero con un gran número de imágenes faciales.

Este modelo está compuesto por diferentes módulos, donde cada uno de ellos contiene diversas capas (reentrenables o no). Los módulos principales del modelo son *conv2d_1a, conv2d_2a, conv2d_2b, maxpool_3a, conv2d_3b, conv2d_4a, conv2d_4b, repeat_1, mixed_6a, repeat_2, mixed_7a, repeat_3, block8, avgpool_1a, dropout, last_linear, last_bn* y *logits*.

En la Figura 19 se muestra la arquitectura del modelo, donde los módulos en rojo representan aquellos que contienen capas con parámetros no entrenables, mientras que el módulo en gris es el encargado principalmente de realizar la clasificación. Los demás módulos con colores diferentes contienen capas usuales con parámetros entrenables.

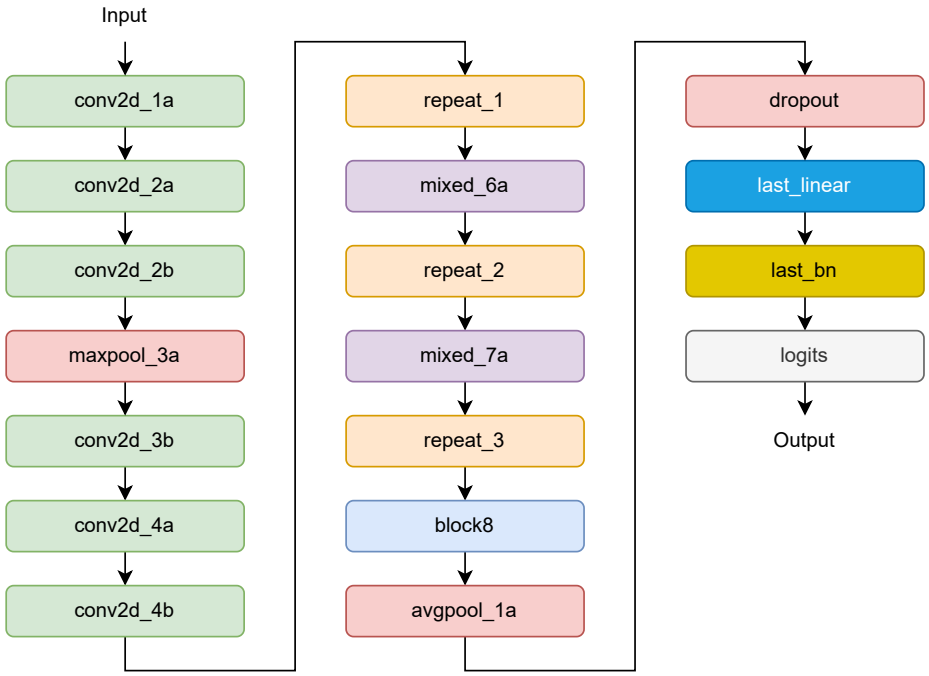


Figura 19: Arquitectura general del modelo InceptionResnetV1.

Además del mencionado modelo, es necesario realizar la identificación y recorte de rostros en imágenes y videos para simplemente procesar imágenes faciales. Para ello, se emplea una red neuronal convolucional en cascada multitarea (MTCNN, Multi-Task Cascaded Convolutional Neural Network). Su diseño permite identificar rostros con precisión en entornos que contienen elementos adicionales. De esta manera, se elimina el ruido que pueda encontrarse en las imágenes originales.

Para finalizar, la librería de Python utilizada para dividir las imágenes de los delincuentes en los conjuntos de reentrenamiento y prueba de manera rápida y eficaz es split-folders [54], asignando el 80 % de las imágenes para el reentrenamiento y el 20 % restante para la prueba, tal como se especificó en el diseño inicial.

A continuación, en la Tabla 4 se muestran las versiones de las principales herramientas software utilizadas.

Herramienta software	Versión
Python	3.9.5
PyTorch	2.2.2
Torchvision	0.17.2
Python	3.9.5
Facenet-pytorch	2.5.3
Split-folders	0.5.1

Tabla 4: Versiones de las herramientas software utilizadas.

8.2.2. Descripción de las Fases de la Implementación

A continuación, se procede a explicar la función desempeñada por cada una de las herramientas software previamente mencionadas en los sistemas de reentrenamiento y evaluación.

8.2.2.1. Fases del Sistema de Reentrenamiento

En las fases del reentrenamiento del sistema se lleva a cabo el reentrenamiento y desarrollo del modelo para su posterior implementación. Seguidamente, se detallan las distintas fases.

8.2.2.1.1 Recopilación de Imágenes de Delincuentes

Una vez descargada la base de datos Pins Face Recognition, se procede a su separación manual entre delincuentes y no delincuentes. Dado que la base de datos contiene imágenes de 105 individuos, se opta por seleccionar a 15 de ellos como delincuentes, dejando a los 90 restantes como no delincuentes.

8.2.2.1.2 División de Imágenes de Delincuentes en Conjuntos de Reentrenamiento y Prueba

Tal como se ha mencionado en el diseño, las imágenes de los delincuentes son divididas en los conjuntos de reentrenamiento y prueba, a través de la librería split-folders. En consecuencia, se asigna un 80 % de las imágenes de los delincuentes para el reentrenamiento y un 20 % para las pruebas. El motivo de esta división es que en el proceso de evaluación se desea tener imágenes de delincuentes que el modelo previamente no haya procesado o visto en el reentrenamiento. Se cuenta con entre 100 y 190 imágenes de reentrenamiento y entre 20 y 50 imágenes de test por delincuente, depende de cada uno.

En la Figura 20 se puede observar cómo se tienen almacenadas diferentes cantidades de imágenes de cada delincuente. Esto puede introducir sesgos en el proceso de reentrenamiento, ya que los delincuentes con mayor cantidad de imágenes pueden ser identificados con mayor fiabilidad y facilidad. Sin embargo, esto también depende de la calidad y diversidad de las imágenes de reentrenamiento empleadas para cada clase.

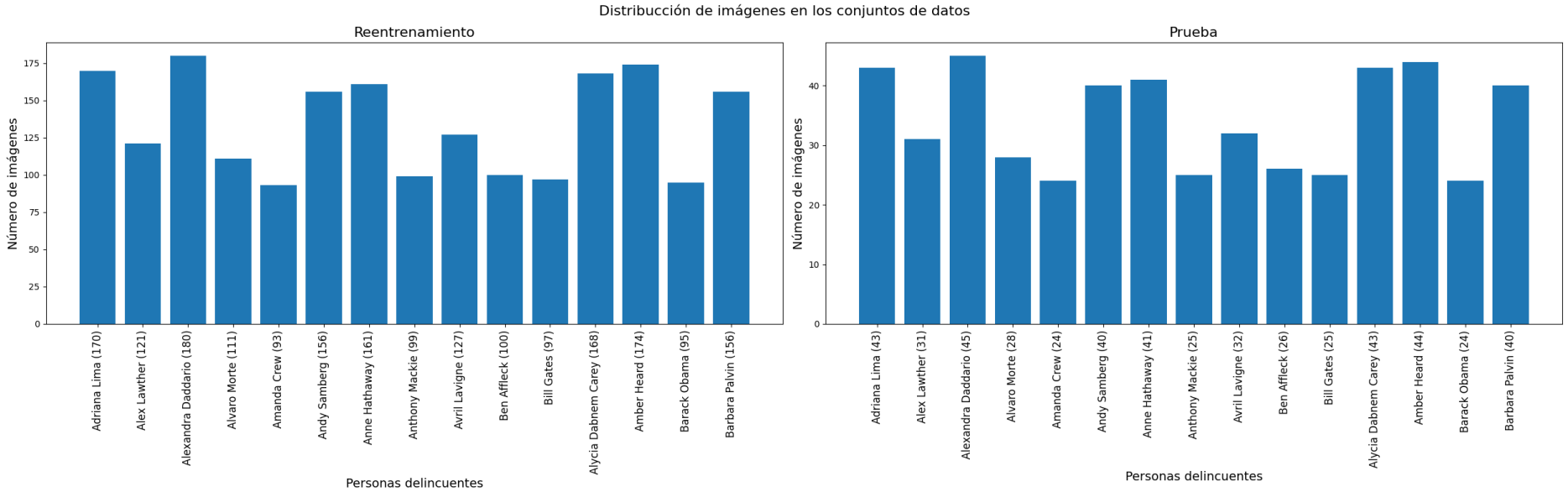


Figura 20: Cantidad de imágenes de reentrenamiento y prueba de cada delincuente.

8.2.2.1.3 Limpieza y Procesamiento de Imágenes de Reentrenamiento de Delincuentes

Posteriormente, a través de la red neuronal convolucional MTCNN, se realiza el procesado de las imágenes de reentrenamiento de los delincuentes. Para ello, dichas imágenes son introducidas en esta red neuronal convolucional con el propósito de almacenar únicamente las imágenes de los rostros de los delincuentes.

Igualmente, se ajusta el tamaño de las imágenes debido a que las originales son de diferentes tamaños. Dado que el modelo fue entrenado con imágenes de tamaño 160x160, se redimensionan las imágenes faciales para que coincidan con esta forma.

8.2.2.1.4 Reentrenamiento del Modelo

Para continuar, se lleva a cabo el reentrenamiento del modelo a través de las imágenes de reentrenamiento de los delincuentes mediante aprendizaje federado. En este proceso, el modelo aprende características faciales de los delincuentes, lo que permite detectar a estos rostros con mayor eficacia.

En la Figura 21 se observa cómo cada cliente que participa en el reentrenamiento emplea la misma cantidad de imágenes de cada delincuente, ya que se trata de un entorno IID.

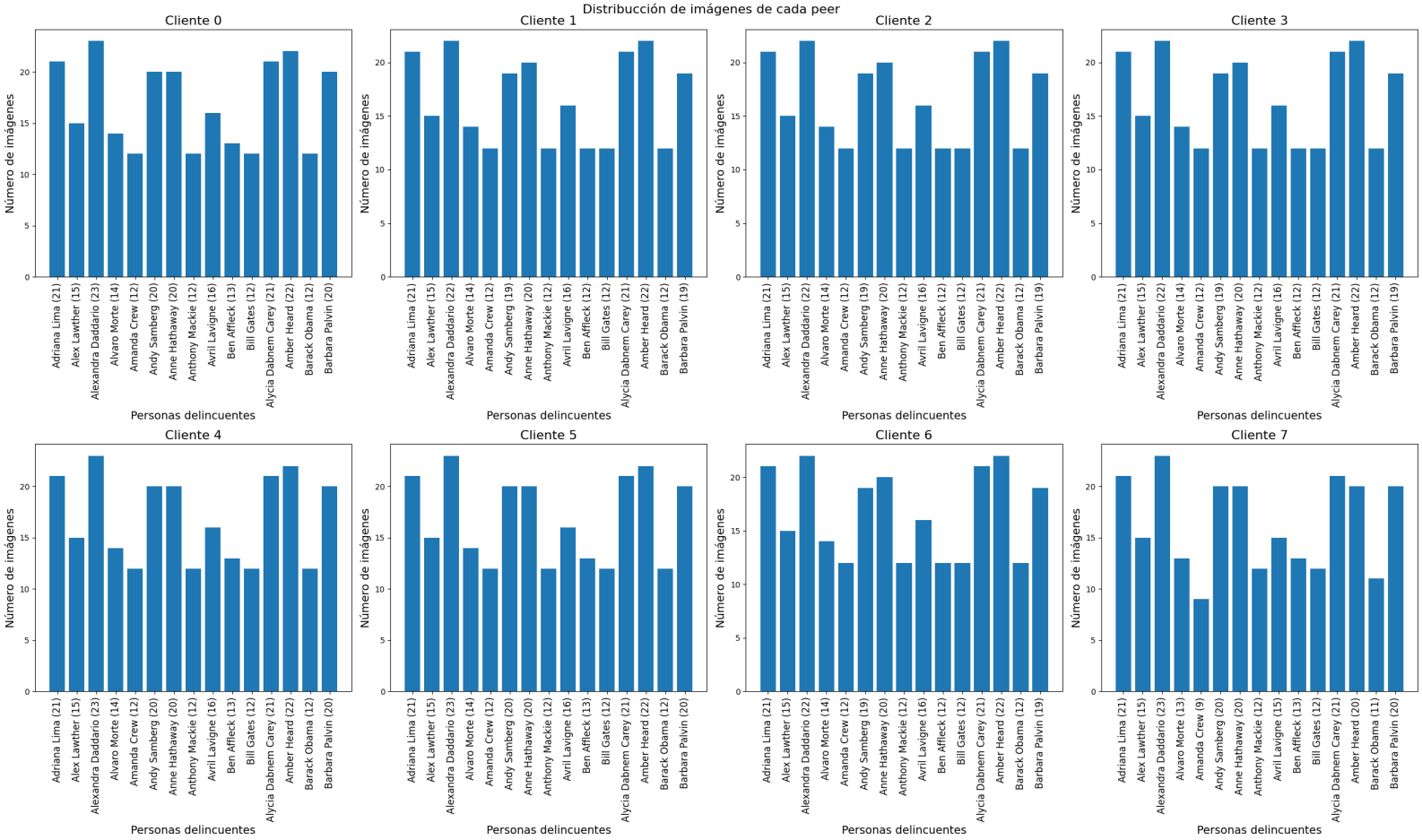


Figura 21: Distribución de imágenes de delincuentes de cada cliente.

8.2.2.1.5 Almacenamiento del Modelo Global

Para terminar, el modelo ya reentrenado se almacena en el servidor para su posterior implementación en el sistema de evaluación.

8.2.2.2. Fases del Sistema de Evaluación

El modelo reentrenado, al ser almacenado en el servidor, permite poder implementarlo en diversos escenarios que tengan como objetivo identificar a los delincuentes.

Es importante recordar que el proceso descrito seguidamente para la imagen a identificar es repetido en múltiples ocasiones para evaluar el sistema en su totalidad con imágenes distintas. Además, se ha de tener en cuenta que la imagen a identificar puede pertenecer tanto a un delincuente como a un no delincuente.

A continuación, se describe el proceso de evaluación.

8.2.2.2.1 Recopilación de Imagen a Identificar

Con el propósito de simular un escenario real de reconocimiento facial, se selecciona una imagen que representa a la persona que se desea identificar.

8.2.2.2.2 Recopilación de Imágenes de Delincuentes

Seguidamente, se almacenan 3 imágenes de cada individuo delincuente, las cuales se utilizan como referencia para comparar con la imagen a identificar con el objetivo de determinar el estatus delictivo de esa persona.

8.2.2.2.3 Limpieza y Procesamiento de las Imágenes

La imagen a identificar y las imágenes almacenadas de los delincuentes son recortadas para obtener el rostro solamente, el cual luego se ajusta al tamaño de 160x160. Llegados a este punto, se poseen 2 conjuntos de datos procesados.

- Imagen a identificar. Contiene una imagen facial de una persona.
- Imágenes almacenadas de los delincuentes. Contiene 3 imágenes faciales de cada delincuente, sumando un total de 45 imágenes.

Es importante recalcar que ambos conjuntos de datos contienen imágenes faciales de resolución óptima, donde un ser humano es capaz de identificar correctamente a cada persona. Dicho en otras palabras, los rostros son claramente visibles y reconocibles.

8.2.2.2.4 Extracción de Características Faciales

Para continuar, se hace uso de la parte convolucional del modelo reentrenado InceptionResnetV1. Este proceso se lleva a cabo con el fin de obtener el vector de

características tanto de la imagen a identificar como de las imágenes almacenadas de los delincuentes. En este caso, los vectores de características tienen una longitud de 512 valores numéricos.

8.2.2.2.5 Comparación de Características Faciales

Para realizar un apropiado reconocimiento facial, es crucial seleccionar una métrica de comparación de vectores de características que sea fácil de interpretar y que permita establecer umbrales de similitud. Por lo tanto, se hace uso de la similitud del coseno para comparar el vector de características del usuario a identificar con los de los delincuentes almacenados. De esta manera, se determina a qué delincuente se asemeja más el individuo que se desea reconocer, dejando la toma de decisión para la siguiente fase.

Los valores de la similitud del coseno oscilan entre -1 y 1, donde 1 indica que ambos vectores de características son idénticos y -1 indica que son totalmente diferentes.

Existe la posibilidad de que el sistema identifique al individuo que se analiza como más de un delincuente. Sin embargo, este caso no se ha tenido en cuenta debido a que el valor de la similitud está redondeado a 3 decimales y es un caso poco probable. Aun así, es un caso que se deberá considerar en el futuro.

8.2.2.2.6 Evaluación de Fiabilidad del Modelo

La similitud del coseno, al ser una métrica que permite establecer umbrales, requiere de una optimización de dicho umbral con el objetivo de identificar a las personas de manera adecuada.

Con ese fin, se hace uso de la matriz de confusión. Esta herramienta permite evaluar cómo el modelo identifica a las distintas personas, es decir, valora cómo de bueno es el modelo clasificando. El objetivo principal es maximizar los verdaderos positivos y negativos, y minimizar los falsos positivos y negativos. Así, los individuos delincuentes y no delincuentes son correctamente identificados.

En este trabajo, la interpretación de la matriz de la confusión se realiza de la siguiente manera.

- Verdadero positivo. Individuo delincuente identificado como delincuente.
- Falso positivo. Individuo no delincuente identificado como delincuente.
- Falso negativo. Individuo delincuente identificado como no delincuente.
- Verdadero negativo. Individuo no delincuente identificado como no delincuente.

9. Evaluación de la Solución Propuesta

Con base en la solución propuesta en el apartado 8, a continuación, se presentan los resultados y aspectos más relevantes de la evaluación de la misma.

Este apartado está organizado de modo que primero se define un plan de pruebas para realizar la parametrización óptima del sistema de reconocimiento facial. Seguidamente, se describen las métricas de evaluación utilizadas para este trabajo. Para terminar, se lleva a cabo un análisis de los resultados obtenidos para comprender el impacto y la eficacia de las implementaciones realizadas, sacar conclusiones e identificar posibles mejoras.

9.1. Diseño del Plan de Pruebas

Un plan de pruebas no solo posibilita evaluar el desempeño del sistema desarrollado, sino que también permite realizar análisis de fiabilidad y robustez. Además, detalla los procedimientos a seguir para probar el sistema en una variedad de escenarios.

En este trabajo en concreto, se lleva a cabo una comparación de diversos escenarios dentro de un entorno de aprendizaje federado, donde se realiza un análisis de fiabilidad tanto del reentrenamiento total del modelo como del reentrenamiento parcial del modelo. Esto se efectúa con el objetivo de determinar la configuración óptima del sistema de reconocimiento facial.

En el caso del reentrenamiento total del modelo, se reentrenan todos los módulos del modelo que contienen capas con parámetros entrenables. Este escenario se denomina escenario E0. No obstante, en el caso del reentrenamiento parcial, en la Tabla 5 se indica hasta qué capa se realiza la congelación de los módulos en cada escenario, mientras que los restantes son reentrenados (ver apartado 8.2.1.2).

Escenario	Última capa congelada
E1	*conv2d_4b*
E2	*repeat_1*
E3	*mixed_6a*
E4	*repeat_2*
E5	*mixed_7a*
E6	*repeat_3*

Tabla 5: Congelación de módulos durante el reentrenamiento parcial.

Dentro de cada escenario de reentrenamiento, se examinan 2 subescenarios diferen-

tes. Estos son comparados con el subescenario SE1 donde el modelo no es reentrenado.

- SE1 - Uso del modelo InceptionResnetV1 preentrenado con la base de datos VGGFace2.
- SE2 - Uso del modelo InceptionResnetV1 preentrenado con la base de datos VGGFace2 y reentrenado con las imágenes de reentrenamiento de los delincuentes durante 15 rondas de aprendizaje federado.
- SE3 - Uso del modelo InceptionResnetV1 preentrenado con la base de datos VGGFace2 y reentrenado con las imágenes de reentrenamiento de los delincuentes durante 5 rondas de aprendizaje federado.

El propósito de crear 2 subescenarios con diferentes números de rondas es analizar si un entrenamiento más dilatado resulta en una identificación más fiable de delincuentes y no delincuentes.

Cabe destacar que el número de clientes que participan en el reentrenamiento del modelo es 8, en el cual todos participan en todas las rondas del aprendizaje federado. Del mismo modo, se emplea la distribución de las imágenes IID, donde cada cliente reentrena el modelo recibido por el servidor durante 5 épocas, utilizando un tamaño de lote de 16 y una tasa de aprendizaje de 0,001. Por último, cabe destacar que se hace uso del procesador del servidor, ya que no se cuenta con una GPU.

Para concluir, en los subescenarios SE2 y SE3 se aplican los 3 métodos de agregación descritos en el contexto: media, mediana y media recortada. Es relevante señalar que, con respecto a la agregación de media recortada, se establece un umbral de recorte del 25 % (ver apartado 2.3.5).

En resumen, el plan de pruebas propuesto pretende realizar la comparación de diversos escenarios a diferentes niveles de profundidad bajo el aprendizaje federado. En la Figura 22 se presenta un resumen del plan de pruebas propuesto.

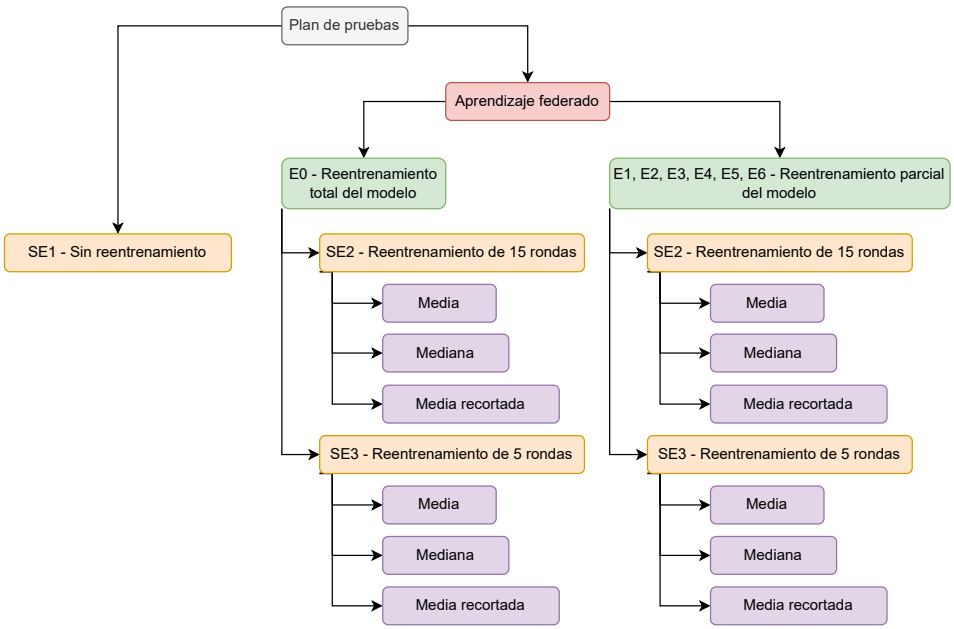


Figura 22: Esquema del diseño del plan de pruebas.

9.2. Métricas de Evaluación

Tal como se ha mencionado a lo largo de la documentación, el objetivo principal del sistema propuesto es identificar correctamente a los delincuentes, teniendo como objetivo adicional y secundario que los individuos no delincuentes no sean clasificados como delincuentes. Esto se traduce en maximizar los valores de verdaderos positivos y negativos, y minimizar los de los falsos positivos y negativos. Para evaluar dichos valores se hace uso de métricas como la sensibilidad (TPR) y la especificidad (TNR), que miden la fiabilidad para clasificar correctamente a los delincuentes y no delincuentes, respectivamente. Estas se calculan a través de las ecuaciones 9.1 y 9.2.

$$TPR = \frac{verdaderos\ positivos}{verdaderos\ positivos + falsos\ negativos} \tag{9.1}$$

$$TNR = \frac{verdaderos\ negativos}{verdaderos\ negativos + falsos\ positivos} \tag{9.2}$$

Siguiendo el diseño establecido, en cada subescenario se optimiza el umbral de similitud del coseno para valores entre 0,5 y 0,95. El motivo principal de esta decisión es que cada subescenario requiere un umbral diferente para alcanzar los objetivos mencionados al principio de este subapartado.

También, se mide la similitud del coseno media de los delincuentes para evaluar qué tan fiable es esa predicción sobre ellos. Para ello, se calcula la media de las similitudes del coseno de los delincuentes correctamente clasificados como delincuentes.

9.3. Análisis de los Resultados

Tras haber definido las métricas que se utilizan en el presente trabajo, se procede a analizar los resultados obtenidos y a extraer conclusiones.

En las tablas siguientes, se utiliza el color azul para destacar los resultados del subescenario SE1, mientras que la negrita se emplea para destacar los resultados que superan al del subescenario SE1. La comparación con el subescenario SE1 se realiza con el propósito de mejorar los resultados de dicho modelo a través del reentrenamiento del modelo.

Además, cabe destacar que los resultados del subescenario SE1 son iguales para todos los escenarios, como se espera, dado que no se realiza ningún proceso de reentrenamiento del modelo. Estos se incluyen en las tablas para facilitar la comparación entre los diferentes subescenarios.

9.3.1. Comparación de Diversos Escenarios Dentro de un Entorno de Aprendizaje Federado

A continuación, se presentan los resultados obtenidos en cada escenario, y se extraen las conclusiones más significativas.

9.3.1.1. Reentrenamiento Total del Modelo

Para comenzar, en este subapartado se presentan los resultados obtenidos para el escenario en la que todos los módulos con capas con parámetros entrenables son reentrenados.

A continuación, en la Tabla 6 se exponen los resultados obtenidos en los subescenarios SE1, SE2 y SE3.

		TPR	TNR	Umbral	Similitud media de delincuentes
SE1 (Sin reentrenamiento)		100 %	97,778 %	0,6	0,807
SE2 (15 rondas)	Media	100 %	90 %	0,85	0,935
	Mediana	100 %	83,333 %	0,8	0,938
	Media recortada	100 %	93,333 %	0,85	0,937
SE3 (5 rondas)	Media	100 %	90 %	0,85	0,938
	Mediana	100 %	85,556 %	0,85	0,935
	Media recortada	100 %	80 %	0,85	0,939

Tabla 6: Comparación de los subescenarios del escenario E0.

Con base en los resultados presentados, se puede concluir que, a pesar de reentrenar el modelo en su totalidad con imágenes de delincuentes, la fiabilidad o la eficacia para detectar los mencionados delincuentes es la misma que la del modelo preentrenado. Por otro lado, es interesante señalar que a la hora de detectar individuos no delincuentes, se obtienen resultados peores. Esto se traduce en una no necesidad de reentrenar todos los módulos del modelo. Los motivos se deben a que el modelo del subescenario SE1 está lo suficientemente capacitado para hacer diferenciaciones entre diversas personas, gracias al preentrenamiento realizado con una cantidad masiva de rostros faciales de diferentes individuos. Como resultado, este modelo generaliza de manera correcta frente a imágenes no vistas previamente.

En resumen, todos los modelos, independientemente del método de agregación empleado, logran identificar de manera correcta a los delincuentes, adaptando los umbrales de similitud del coseno. Sin embargo, la diferencia radica en cómo se clasifican las personas no delincuentes. En los subescenarios SE2 y SE3, al reentrenar los modelos con imágenes de delincuentes, las capas encargadas de extraer características pierden cierta capacidad de diferenciación aprendida en el entrenamiento previo. Esto lleva a que los individuos no delincuentes no involucrados en el reentrenamiento sean clasificados erróneamente como delincuentes. En otras palabras, el valor de TNR se ve afectado, y ninguno iguala o supera el valor del subescenario SE1, que es el más alto de todos. Esto se debe a que los parámetros del modelo son ajustados a las imágenes de los delincuentes, produciendo más falsos positivos y reduciendo los verdaderos negativos.

Al observar las dos últimas columnas de la Tabla 6, se puede ver cómo difieren los valores para el subescenario SE1 y para los subescenarios SE2 y SE3. Para el caso del subescenario SE1, el umbral óptimo que maximiza los valores de TPR y TNR es 0,6, mientras que para los subescenarios SE2 y SE3 tienden a ser 0,8 y 0,85. De manera similar, la similitud media de los delincuentes para el subescenario SE1 es de 0,807, mientras que para los subescenarios SE2 y SE3 se encuentra entre 0,935 y 0,939. La principal causa de estas diferencias se debe al reentrenamiento realizado con las imágenes de los delincuentes. El modelo del subescenario E1, al no haber sido reentrenado con imágenes de delincuentes, obtiene una similitud media de delincuentes menor que la de los subescenarios SE2 y SE3, como es de esperar. Esto se traduce en la necesidad de tener que modificar el umbral de los subescenarios SE2 y SE3 para minimizar el impacto de los

falsos positivos. En resumen, una mayor similitud media de delincuentes implica tener que ajustar el umbral para mantener la precisión en la detección de delincuentes y no delincuentes.

9.3.1.2. Reentrenamiento Parcial del Modelo

Después de analizar los resultados del escenario E0 que implica el reentrenamiento total del modelo, se considera necesario realizar un estudio de la técnica de *fine-tuning* a través de un enfoque parcial. Con ese fin, se ponen en marcha los escenarios presentados en la Tabla 5, que se resumen en el reentrenamiento de solo determinadas capas. En las Tablas 7, 8, 9, 10, 11 y 12 se presentan los resultados obtenidos.

		TPR	TNR	Umbral	Similitud media de delincuentes
SE1 (Sin reentrenamiento)		100 %	97,778 %	0,6	0,807
SE2 (15 rondas)	Media	100 %	85,556 %	0,85	0,935
	Mediana	100 %	94,444 %	0,85	0,930
	Media recortada	100 %	93,333 %	0,85	0,932
SE3 (5 rondas)	Media	100 %	68,889 %	0,8	0,929
	Mediana	100 %	88,889 %	0,85	0,938
	Media recortada	100 %	74,444 %	0,8	0,93

Tabla 7: Comparación de los subescenarios del escenario E1.

		TPR	TNR	Umbral	Similitud media de delincuentes
SE1 (Sin reentrenamiento)		100 %	97,778 %	0,6	0,807
SE2 (15 rondas)	Media	100 %	91,111 %	0,85	0,942
	Mediana	100 %	82,222 %	0,8	0,94
	Media recortada	100 %	86,667 %	0,85	0,945
SE3 (5 rondas)	Media	100 %	82,222 %	0,85	0,959
	Mediana	100 %	35,556 %	0,75	0,946
	Media recortada	100 %	80 %	0,85	0,951

Tabla 8: Comparación de los subescenarios del escenario E2.

		TPR	TNR	Umbral	Similitud media de delincuentes
SE1 (Sin reentrenamiento)		100 %	97,778 %	0,6	0,807
SE2 (15 rondas)	**Media**	**100 %**	**98,889 %**	**0,9**	**0,953**
	Mediana	100 %	87,778 %	0,85	0,95
	Media recortada	100 %	92,222 %	0,85	0,95
SE3 (5 rondas)	Media	100 %	72,222 %	0,9	0,96
	Mediana	100 %	85,556 %	0,9	0,958
	Media recortada	100 %	61,111 %	0,85	0,96

Tabla 9: Comparación de los subescenarios del escenario E3.

		TPR	TNR	Umbral	Similitud media de delincuentes
SE1 (Sin reentrenamiento)		100 %	97,778 %	0,6	0,807
SE2 (15 rondas)	**Media**	**100 %**	**100 %**	**0,9**	**0,972**
	Mediana	100 %	92,222 %	0,85	0,968
	Media recortada	**100 %**	**98,889 %**	**0,9**	**0,965**
SE3 (5 rondas)	Media	100 %	92,222 %	0,9	0,981
	Mediana	100 %	83,333 %	0,85	0,974
	Media recortada	100 %	94,444 %	0,95	0,978

Tabla 10: Comparación de los subescenarios del escenario E4.

		TPR	TNR	Umbral	Similitud media de delincuentes
SE1 (Sin reentrenamiento)		100 %	97,778 %	0,6	0,807
SE2 (15 rondas)	Media	100 %	94,444 %	0,85	0,955
	Mediana	100 %	86,667 %	0,8	0,955
	Media recortada	100 %	95,556 %	0,85	0,96
SE3 (5 rondas)	Media	100 %	77,778 %	0,8	0,96
	Mediana	100 %	85,556 %	0,85	0,964
	Media recortada	100 %	90 %	0,85	0,964

Tabla 11: Comparación de los subescenarios del escenario E5.

		TPR	TNR	Umbral	Similitud media de delincuentes
SE1 (Sin reentrenamiento)		100 %	97,778 %	0,6	0,807
SE2 (15 rondas)	Media	100 %	83,333 %	0,75	0,941
	Mediana	100 %	92,222 %	0,8	0,946
	Media recortada	100 %	93,333 %	0,8	0,951
SE3 (5 rondas)	Media	100 %	92,222 %	0,8	0,951
	Mediana	100 %	95,556 %	0,8	0,952
	Media recortada	100 %	83,333 %	0,75	0,951

Tabla 12: Comparación de los subescenarios del escenario E6.

Según los resultados, se llega a la conclusión de que dentro del escenario E4, el subescenario SE2, cuando se emplea el método de agregación media, exhibe la mayor fiabilidad. Este subescenario implica el reentrenamiento de los módulos *mixed_7a, repeat_3, block8, last_linear, last_bn* y *logits*, mientras que los módulos restantes se mantienen congelados. No obstante, existen otros dos subescenarios en los escenarios E3 y E4 en los que el reentrenamiento del modelo presenta una mejora menor en los resultados.

Centrando la atención en el resultado más destacado, en la Tabla 10 se puede observar cómo el valor de la similitud media de los delincuentes aumenta hasta 0,972, que es el mayor valor conseguido entre todos los subescenarios SE2. Además, el valor de TNR crece hasta el 100 %, lo que se traduce en una detección impecable de los individuos no delincuentes. Esto supone mejoras en la capacidad del modelo para distinguir entre delincuentes y no delincuentes, pese a que el modelo es solamente reentrenado con imágenes de delincuentes. La razón principal de estas mejoras en cuanto a TNR y similitud media de los delincuentes radica en el ajuste de los parámetros del modelo mediante el reentrenamiento. Estos parámetros son modificados para detectar con mayor fiabilidad a los delincuentes. Asimismo, el congelamiento de determinadas capas evita un sobreajuste respecto a las imágenes de reentrenamiento de los delincuentes, como ocurre en el caso del reentrenamiento total descrito en el apartado 9.3.1.1.

Es importante resaltar que los 3 subescenarios destacados en negrita que representan una mejora de resultados respecto al subescenario SE1 comparten el mismo valor de umbral, que es 0,9. Esto sugiere que 0,9 es la similitud del coseno óptima para diferenciar entre las personas delincuentes y no delincuentes, después de realizar un reentrenamiento parcial del modelo.

En las Figuras 23 y 24 se resumen los valores de TNR y similitud media de los delincuentes de los escenarios de las Tablas 7, 8, 9, 10, 11 y 12. En ambas figuras se puede observar cómo a través del escenario E4 se obtienen los valores más altos de TNR y similitud media de delincuentes. Esto supone que el módulo *mixed_7a* es necesario que sea reentrenado para obtener una fiabilidad mayor tanto con individuos delincuentes como no delincuentes.

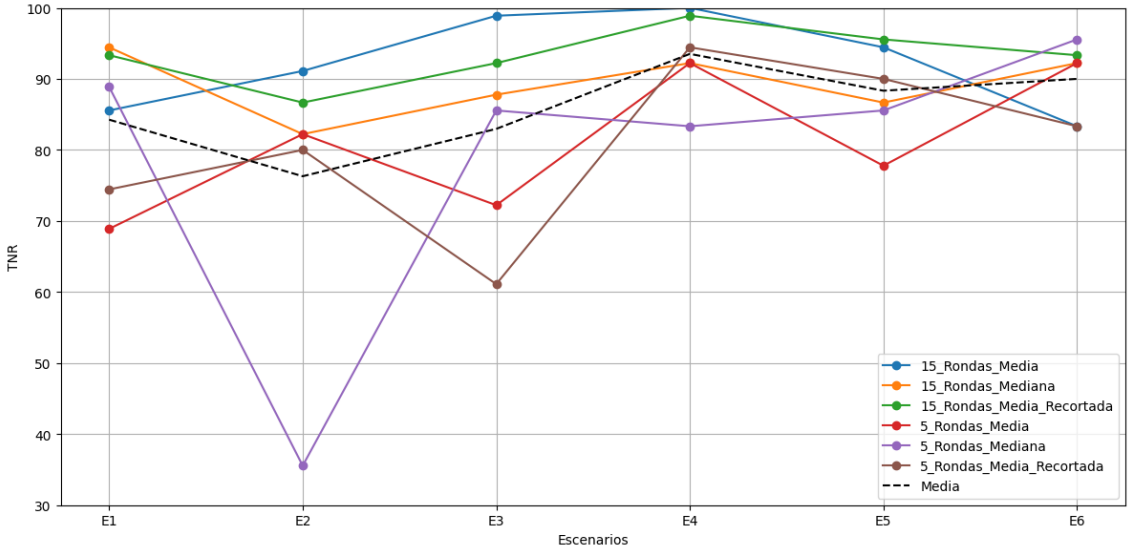


Figura 23: Valores de TNR de los diversos escenarios.

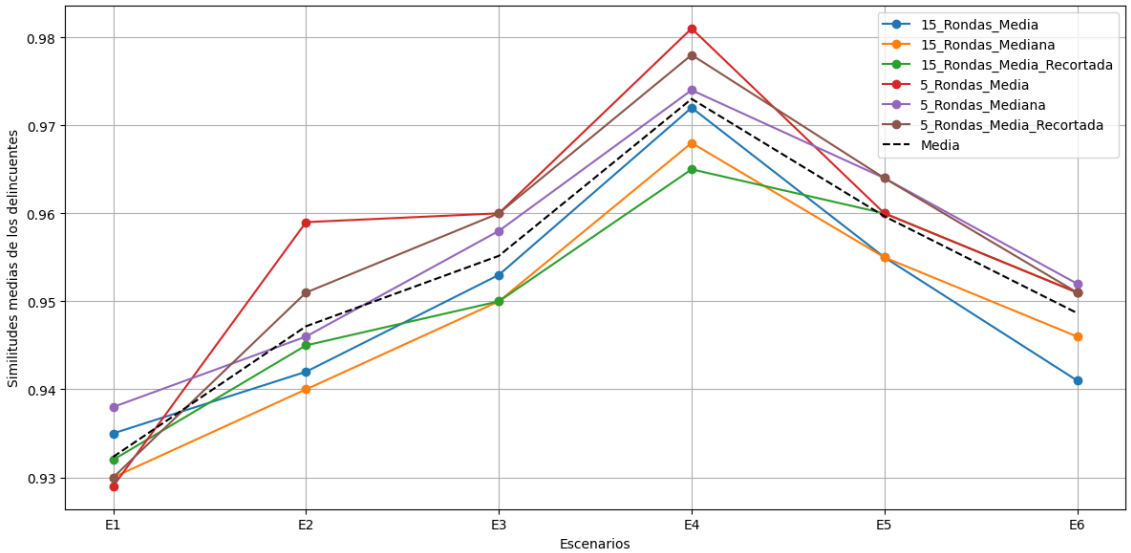


Figura 24: Valores de similitudes media de los delincuentes de los diversos escenarios.

Asimismo, en las Figuras 25 y 26 se presentan los mapas de calor donde se especifican las similitudes de los individuos delincuentes y no delincuentes a identificar, en comparación con las imágenes de los delincuentes. Dichas figuras pertenecen al subescenario SE2 del escenario E4, donde se utiliza el método de agregación media.

En la Figura 26 se puede apreciar cómo ningún individuo no delincuente está asociado con alguno de los delincuentes con una similitud mayor a 0,9, lo que resulta en un valor de 100 % para TNR.

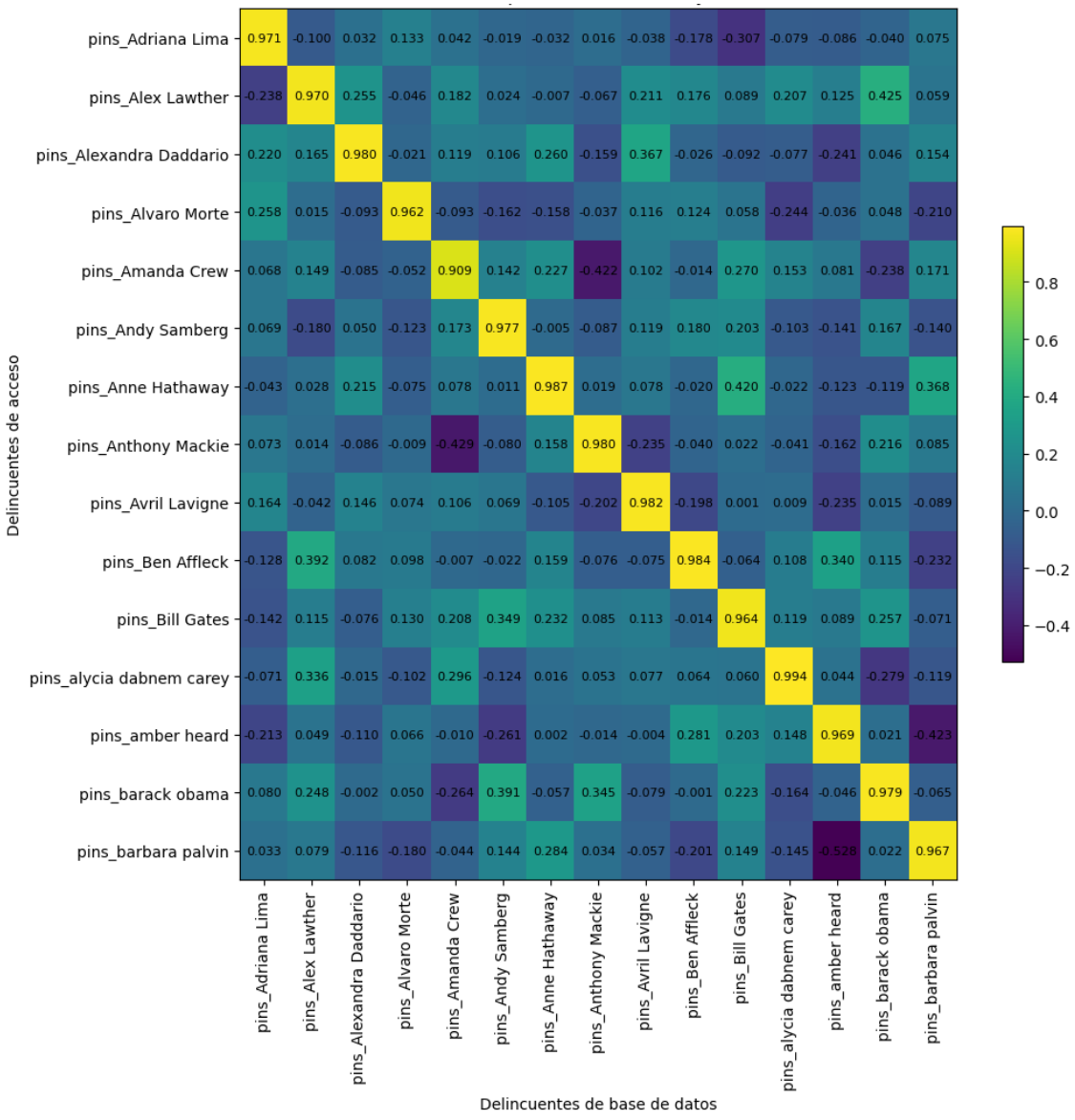
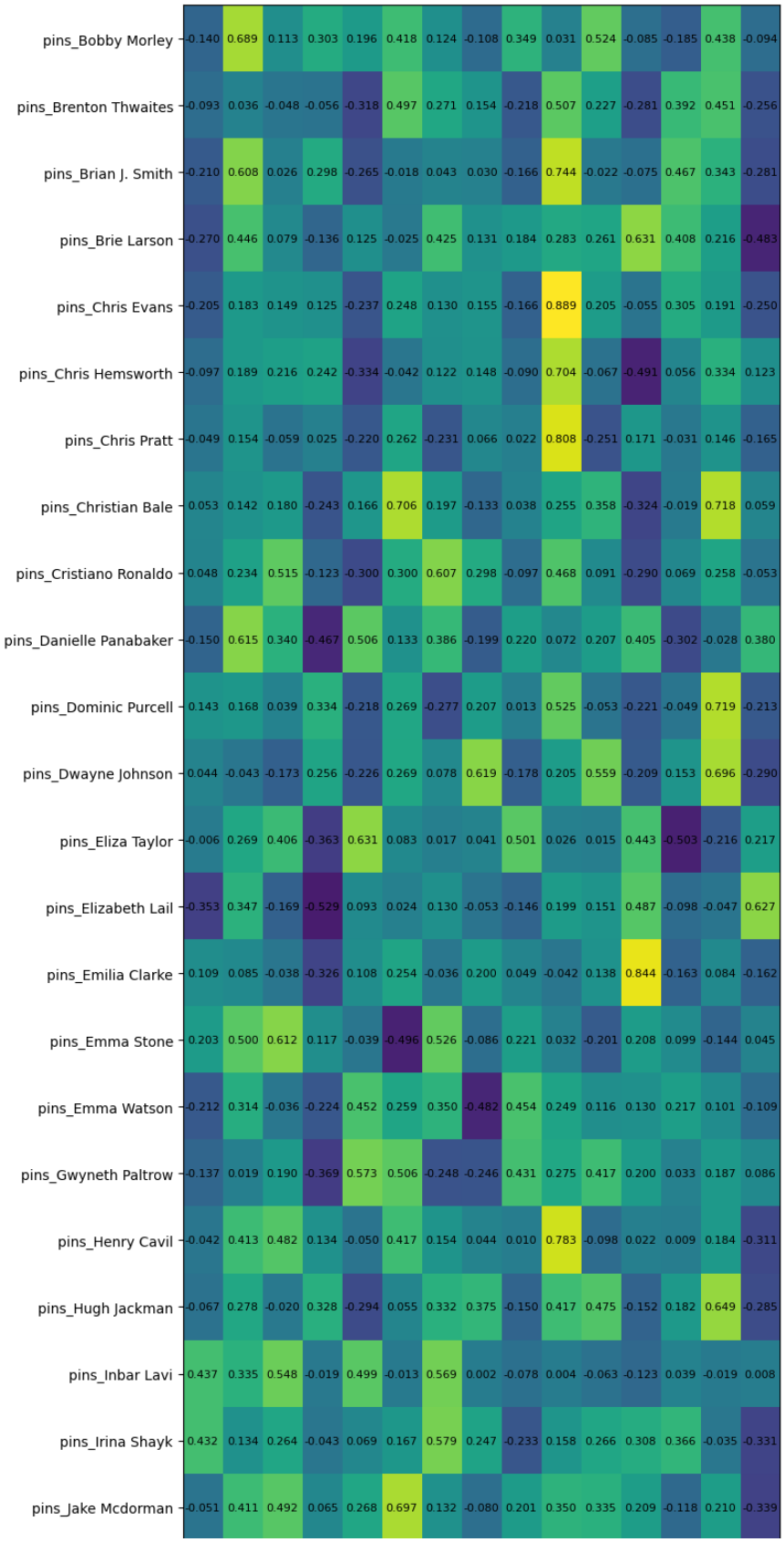
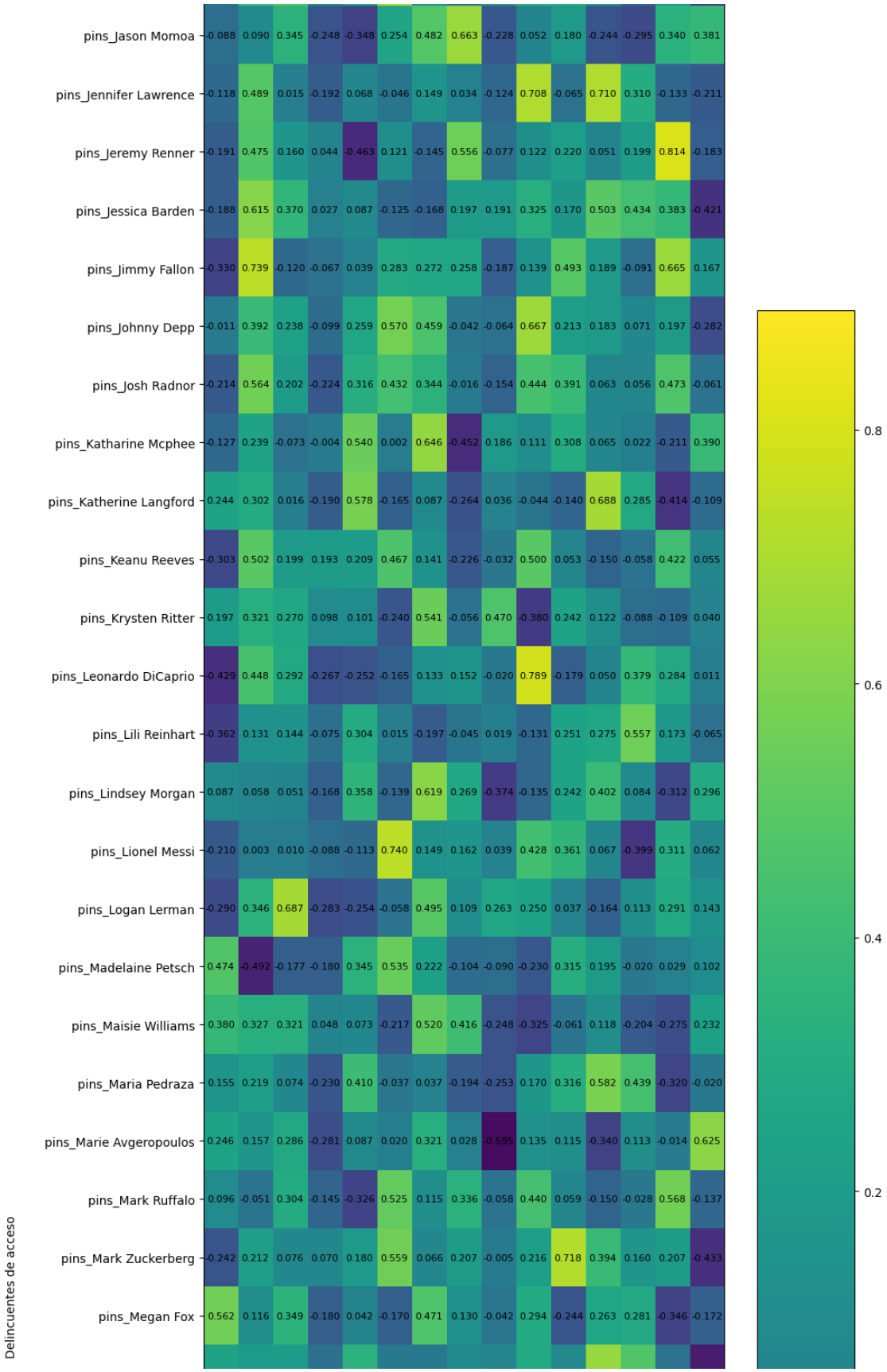
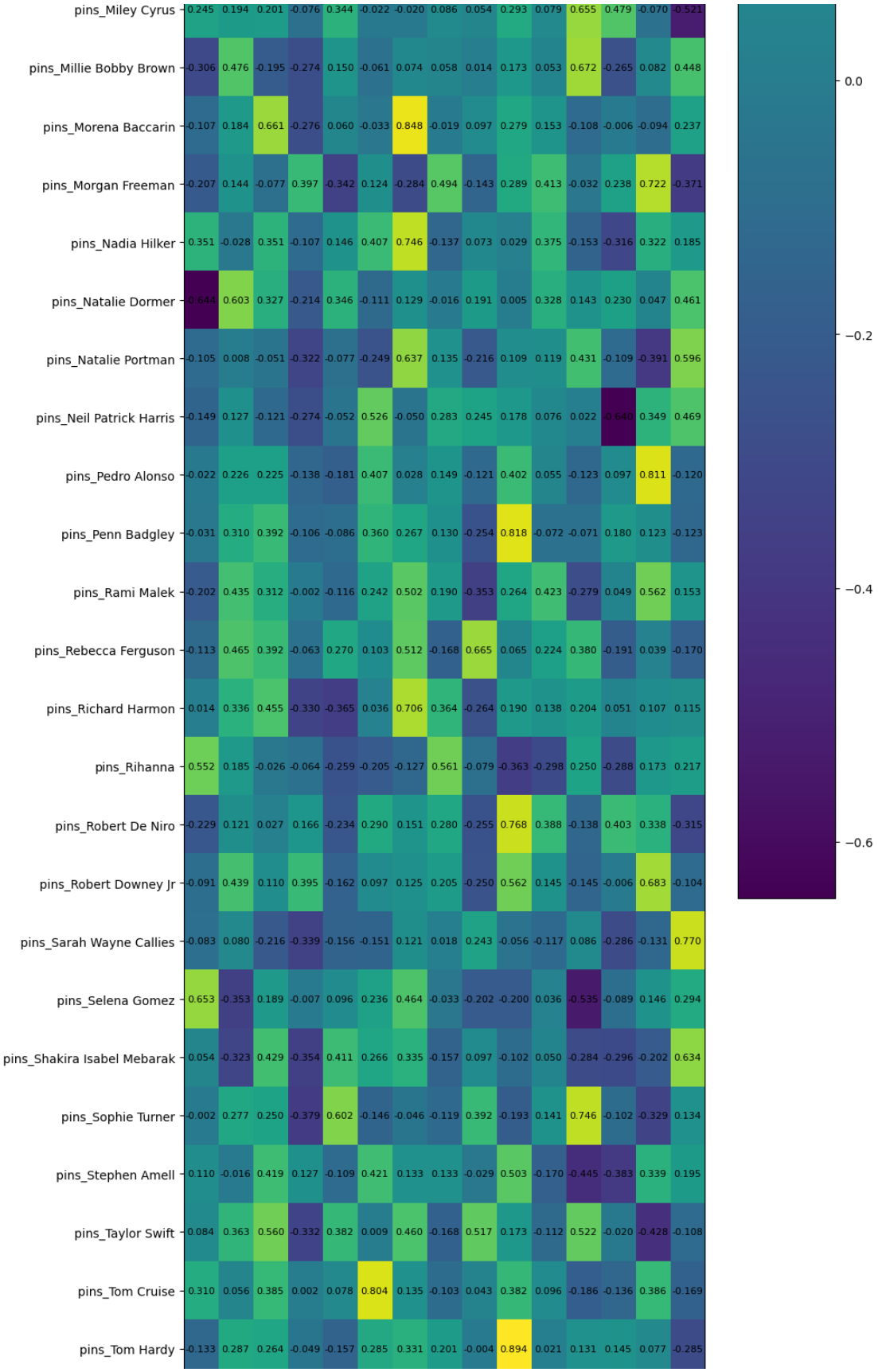


Figura 25: Mapa de calor de similitudes de individuos delincuentes.


pins_Bobby Morley -0.140 0.689 0.113 0.303 0.196 0.418 0.124 -0.108 0.349 0.031 0.524 -0.085 -0.185 0.438 -0.094
pins_Brenton Thwaites -0.093 0.036 -0.048 -0.056 -0.318 0.497 0.271 0.154 -0.218 0.507 0.227 -0.281 0.392 0.451 -0.256
pins_Brian J. Smith -0.210 0.608 0.026 0.298 -0.265 -0.018 0.043 0.030 -0.166 0.744 -0.022 -0.075 0.467 0.343 -0.281
pins_Brie Larson -0.270 0.446 0.079 -0.136 0.125 -0.025 0.425 0.131 0.184 0.283 0.261 0.631 0.408 0.216 -0.483
pins_Chris Evans -0.205 0.183 0.149 0.125 -0.237 0.248 0.130 0.155 -0.166 0.889 0.205 -0.055 0.305 0.191 -0.250
pins_Chris Hemsworth -0.097 0.189 0.216 0.242 -0.334 -0.042 0.122 0.148 -0.090 0.704 -0.067 -0.491 0.056 0.334 0.123
pins_Chris Pratt -0.049 0.154 -0.059 0.025 -0.220 0.262 -0.231 0.066 0.022 0.808 -0.251 0.171 -0.031 0.146 -0.165
pins_Christian Bale 0.053 0.142 0.180 -0.243 0.166 0.706 0.197 -0.133 0.038 0.255 0.358 -0.324 -0.019 0.718 0.059
pins_Cristiano Ronaldo 0.048 0.234 0.515 -0.123 -0.300 0.300 0.607 0.298 -0.097 0.468 0.091 -0.290 0.069 0.258 -0.053
pins_Danielle Panabaker -0.150 0.615 0.340 -0.467 0.506 0.133 0.386 -0.199 0.220 0.072 0.207 0.405 -0.302 -0.028 0.380
pins_Dominic Purcell 0.143 0.168 0.039 0.334 -0.218 0.269 -0.277 0.207 0.013 0.525 -0.053 -0.221 -0.049 0.719 -0.213
pins_Dwayne Johnson 0.044 -0.043 -0.173 0.256 -0.226 0.269 0.078 0.619 -0.178 0.205 0.559 -0.209 0.153 0.696 -0.290
pins_Eliza Taylor -0.006 0.269 0.406 -0.363 0.631 0.083 0.017 0.041 0.501 0.026 0.015 0.443 -0.503 -0.216 0.217
pins_Elizabeth Lail -0.353 0.347 -0.169 -0.529 0.093 0.024 0.130 -0.053 -0.146 0.199 0.151 0.487 -0.098 -0.047 0.627
pins_Emilia Clarke 0.109 0.085 -0.038 -0.326 0.108 0.254 -0.036 0.200 0.049 -0.042 0.138 0.844 -0.163 0.084 -0.162
pins_Emma Stone 0.203 0.500 0.612 0.117 -0.039 -0.496 0.526 -0.086 0.221 0.032 -0.201 0.208 0.099 -0.144 0.045
pins_Emma Watson -0.212 0.314 -0.036 -0.224 0.452 0.259 0.350 -0.482 0.454 0.249 0.116 0.130 0.217 0.101 -0.109
pins_Gwyneth Paltrow -0.137 0.019 0.190 -0.369 0.573 0.506 -0.248 -0.246 0.431 0.275 0.417 0.200 0.033 0.187 0.086
pins_Henry Cavil -0.042 0.413 0.482 0.134 -0.050 0.417 0.154 0.044 0.010 0.783 -0.098 0.022 0.009 0.184 -0.311
pins_Hugh Jackman -0.067 0.278 -0.020 0.328 -0.294 0.055 0.332 0.375 -0.150 0.417 0.475 -0.152 0.182 0.649 -0.285
pins_Inbar Lavi 0.437 0.335 0.548 -0.019 0.499 -0.013 0.569 0.002 -0.078 0.004 -0.063 -0.123 0.039 -0.019 0.008
pins_Irina Shayk 0.432 0.134 0.264 -0.043 0.069 0.167 0.579 0.247 -0.233 0.158 0.266 0.308 0.366 -0.035 -0.331
pins_Jake Mcdorman -0.051 0.411 0.492 0.065 0.268 0.697 0.132 -0.080 0.201 0.350 0.335 0.209 -0.118 0.210 -0.339


pins_Jason Momoa -0.088 0.090 0.345 -0.248 -0.348 0.254 0.482 0.663 -0.228 0.052 0.180 -0.244 -0.295 0.340 0.381
pins_Jennifer Lawrence -0.118 0.489 0.015 -0.192 0.068 -0.046 0.149 0.034 -0.124 0.708 -0.065 0.710 0.310 -0.133 -0.211
pins_Jeremy Renner -0.191 0.475 0.160 0.044 -0.463 0.121 -0.145 0.556 -0.077 0.122 0.220 0.051 0.199 0.814 -0.183
pins_Jessica Barden -0.188 0.615 0.370 0.027 0.087 -0.125 -0.168 0.197 0.191 0.325 0.170 0.503 0.434 0.383 -0.421
pins_Jimmy Fallon -0.330 0.739 -0.120 -0.067 0.039 0.283 0.272 0.258 -0.187 0.139 0.493 0.189 -0.091 0.665 0.167
pins_Johnny Depp -0.011 0.392 0.238 -0.099 0.259 0.570 0.459 -0.042 -0.064 0.667 0.213 0.183 0.071 0.197 -0.282
pins_Josh Radnor -0.214 0.564 0.202 -0.224 0.316 0.432 0.344 -0.016 -0.154 0.444 0.391 0.063 0.056 0.473 -0.061
pins_Katharine Mcphee -0.127 0.239 -0.073 -0.004 0.540 0.002 0.646 -0.452 0.186 0.111 0.308 0.065 0.022 -0.211 0.390
pins_Katherine Langford 0.244 0.302 0.016 -0.190 0.578 -0.165 0.087 -0.264 0.036 -0.044 -0.140 0.688 0.285 -0.414 -0.109
pins_Keanu Reeves -0.303 0.502 0.199 0.193 0.209 0.467 0.141 -0.226 -0.032 0.500 0.053 -0.150 -0.058 0.422 0.055
pins_Krysten Ritter 0.197 0.321 0.270 0.098 0.101 -0.240 0.541 -0.056 0.470 -0.380 0.242 0.122 -0.088 -0.109 0.040
pins_Leonardo DiCaprio -0.429 0.448 0.292 -0.267 -0.252 -0.165 0.133 0.152 -0.020 0.789 -0.179 0.050 0.379 0.284 0.011
pins_Lili Reinhart -0.362 0.131 0.144 -0.075 0.304 0.015 -0.197 -0.045 0.019 -0.131 0.251 0.275 0.557 0.173 -0.065
pins_Lindsey Morgan 0.087 0.058 0.051 -0.168 0.358 -0.139 0.619 0.269 -0.374 -0.135 0.242 0.402 0.084 -0.312 0.296
pins_Lionel Messi -0.210 0.003 0.010 -0.088 -0.113 0.740 0.149 0.162 0.039 0.428 0.361 0.067 -0.399 0.311 0.062
pins_Logan Lerman -0.290 0.346 0.687 -0.283 -0.254 -0.058 0.495 0.109 0.263 0.250 0.037 -0.164 0.113 0.291 0.143
pins_Madelaine Petsch 0.474 -0.492 -0.177 -0.180 0.345 0.535 0.222 -0.104 -0.090 -0.230 0.315 0.195 -0.020 0.029 0.102
pins_Maisie Williams 0.380 0.327 0.321 0.048 0.073 -0.217 0.520 0.416 -0.248 -0.325 -0.061 0.118 -0.204 -0.275 0.232
pins_Maria Pedraza 0.155 0.219 0.074 -0.230 0.410 -0.037 0.037 -0.194 -0.253 0.170 0.316 0.582 0.439 -0.320 -0.020
pins_Marie Avgeropoulos 0.246 0.157 0.286 -0.281 0.087 0.020 0.321 0.028 -0.595 0.135 0.115 -0.340 0.113 -0.014 0.625
pins_Mark Ruffalo 0.096 -0.051 0.304 -0.145 -0.326 0.525 0.115 0.336 -0.058 0.440 0.059 -0.150 -0.028 0.568 -0.137
pins_Mark Zuckerberg -0.242 0.212 0.076 0.070 0.180 0.559 0.066 0.207 -0.005 0.216 0.718 0.394 0.160 0.207 -0.433
pins_Megan Fox 0.562 0.116 0.349 -0.180 0.042 -0.170 0.471 0.130 -0.042 0.294 -0.244 0.263 0.281 -0.346 -0.172
Delincuentes de acceso
0.8
0.6
0.4
0.2


pins_Miley Cyrus 0.245 0.194 0.201 -0.076 0.344 -0.022 -0.020 0.086 0.054 0.293 0.079 0.655 0.479 -0.070 -0.521
pins_Millie Bobby Brown -0.306 0.476 -0.195 -0.274 0.150 -0.061 0.074 0.058 0.014 0.173 0.053 0.672 -0.265 0.082 0.448
pins_Morena Baccarin -0.107 0.184 0.661 -0.276 0.060 -0.033 0.848 -0.019 0.097 0.279 0.153 -0.108 -0.006 -0.094 0.237
pins_Morgan Freeman -0.207 0.144 -0.077 0.397 -0.342 0.124 -0.284 0.494 -0.143 0.289 0.413 -0.032 0.238 0.722 -0.371
pins_Nadia Hilker 0.351 -0.028 0.351 -0.107 0.146 0.407 0.746 -0.137 0.073 0.029 0.375 -0.153 -0.316 0.322 0.185
pins_Natalie Dormer -0.644 0.603 0.327 -0.214 0.346 -0.111 0.129 -0.016 0.191 0.005 0.328 0.143 0.230 0.047 0.461
pins_Natalie Portman -0.105 0.008 -0.051 -0.322 -0.077 -0.249 0.637 0.135 -0.216 0.109 0.119 0.431 -0.109 -0.391 0.596
pins_Neil Patrick Harris -0.149 0.127 -0.121 -0.274 -0.052 0.526 -0.050 0.283 0.245 0.178 0.076 0.022 -0.640 0.349 0.469
pins_Pedro Alonso -0.022 0.226 0.225 -0.138 -0.181 0.407 0.028 0.149 -0.121 0.402 0.055 -0.123 0.097 0.811 -0.120
pins_Penn Badgley -0.031 0.310 0.392 -0.106 -0.086 0.360 0.267 0.130 -0.254 0.818 -0.072 -0.071 0.180 0.123 -0.123
pins_Rami Malek -0.202 0.435 0.312 -0.002 -0.116 0.242 0.502 0.190 -0.353 0.264 0.423 -0.279 0.049 0.562 0.153
pins_Rebecca Ferguson -0.113 0.465 0.392 -0.063 0.270 0.103 0.512 -0.168 0.665 0.065 0.224 0.380 -0.191 0.039 -0.170
pins_Richard Harmon 0.014 0.336 0.455 -0.330 -0.365 0.036 0.706 0.364 -0.264 0.190 0.138 0.204 0.051 0.107 0.115
pins_Rihanna 0.552 0.185 -0.026 -0.064 -0.259 -0.205 -0.127 0.561 -0.079 -0.363 -0.298 0.250 -0.288 0.173 0.217
pins_Robert De Niro -0.229 0.121 0.027 0.166 -0.234 0.290 0.151 0.280 -0.255 0.768 0.388 -0.138 0.403 0.338 -0.315
pins_Robert Downey Jr -0.091 0.439 0.110 0.395 -0.162 0.097 0.125 0.205 -0.250 0.562 0.145 -0.145 -0.006 0.683 -0.104
pins_Sarah Wayne Callies -0.083 0.080 -0.216 -0.339 -0.156 -0.151 0.121 0.018 0.243 -0.056 -0.117 0.086 -0.286 -0.131 0.770
pins_Selena Gomez 0.653 -0.353 0.189 -0.007 0.096 0.236 0.464 -0.033 -0.202 -0.200 0.036 -0.535 -0.089 0.146 0.294
pins_Shakira Isabel Mebarak 0.054 -0.323 0.429 -0.354 0.411 0.266 0.335 -0.157 0.097 -0.102 0.050 -0.284 -0.296 -0.202 0.634
pins_Sophie Turner -0.002 0.277 0.250 -0.379 0.602 -0.146 -0.046 -0.119 0.392 -0.193 0.141 0.746 -0.102 -0.329 0.134
pins_Stephen Amell 0.110 -0.016 0.419 0.127 -0.109 0.421 0.133 0.133 -0.029 0.503 -0.170 -0.445 -0.383 0.339 0.195
pins_Taylor Swift 0.084 0.363 0.560 -0.332 0.382 0.009 0.460 -0.168 0.517 0.173 -0.112 0.522 -0.020 -0.428 -0.108
pins_Tom Cruise 0.310 0.056 0.385 0.002 0.078 0.804 0.135 -0.103 0.043 0.382 0.096 -0.186 -0.136 0.386 -0.169
pins_Tom Hardy -0.133 0.287 0.264 -0.049 -0.157 0.285 0.331 0.201 -0.004 0.894 0.021 0.131 0.145 0.077 -0.285
0.0
−0.2
−0.4
−0.6

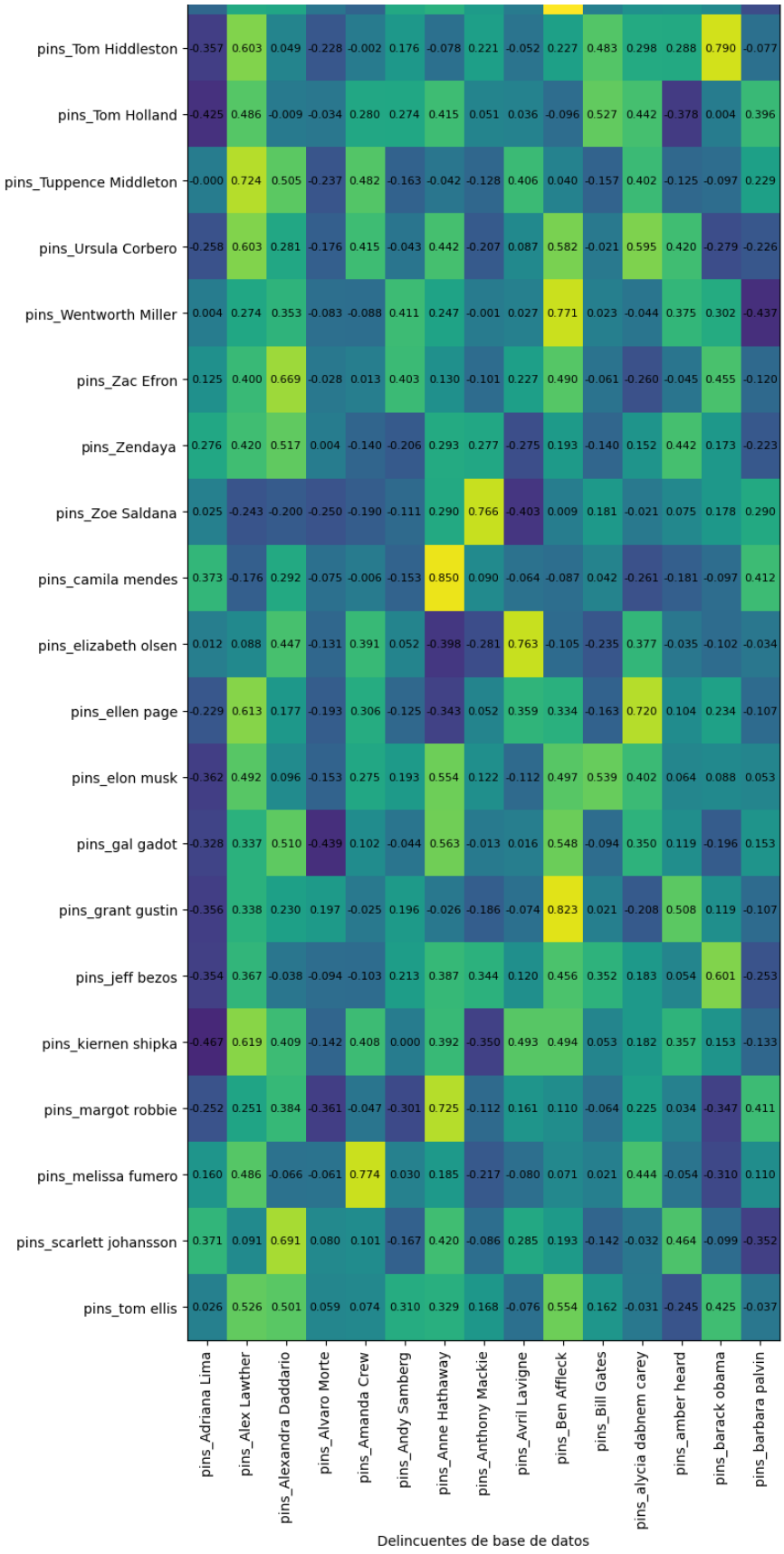


Figura 26: Mapa de calor de similitudes de individuos no delincuentes.

Para finalizar con la presente comparación, se deduce que el subescenario SE3 no es efectivo, puesto que en ninguna implementación alcanza los resultados del subescenario SE1 en términos de TPR y TNR.

10. Descripción de Tareas

En este apartado, se procede a explicar la planificación y la metodología del trabajo. Primero de todo, se nombran las personas que constituyen el equipo de trabajo, así como el rol que desempeña cada una de ellas. Además, se mencionan los recursos físicos empleados. Seguidamente, se describen los diversos paquetes de trabajos (PT) y las tareas (T) asociadas a cada uno de ellos. Para finalizar, se presentan los hitos y el diagrama de Gantt, que resume visualmente la planificación del trabajo.

10.1. Recursos Humanos y Materiales

En la Tabla 13 se presenta el equipo de trabajo que ha llevado a cabo el diseño, la implementación y la redacción del presente trabajo.

Identificador	Nombre	Cargo
MVH	Maria Victoria Higuero	Directora del trabajo
JA	Jasone Astorga	Codirectora del trabajo
AG	Ander Galván	Proyectista

Tabla 13: Recursos humanos.

Tanto la directora como la codirectora del trabajo son ingenieras senior de telecomunicaciones con un alto nivel de conocimientos en *Deep Learning* y reconocimiento facial. Su misión principal consiste en guiar al proyectista en la realización de las diversas tareas con el propósito de cumplimentar los objetivos del trabajo. De igual manera, son las personas encargadas de determinar el plan de trabajo y supervisar los procedimientos establecidos para alcanzar los resultados esperados.

En cambio, el proyectista es un ingeniero junior de telecomunicaciones con conocimientos adecuados para llevar a cabo el diseño del sistema y su implementación. Además, su labor incluye la realización de las diferentes tareas que conforman el proyecto, siguiendo las indicaciones de la directora y la codirectora. Asimismo, es responsable de la redacción de la documentación del presente trabajo.

En cuanto a los recursos físicos, en la Tabla 14 se presentan los equipos empleados en este trabajo.

Identificador	Material	Cantidad
PC	Asus Zenbook	1
SD	Servidor Dell PowerEdge R730	1

Tabla 14: Recursos físicos.

10.2. Definición de los Paquetes de Trabajo y Tareas

A continuación, se describen los paquetes de trabajo y las tareas asociadas a cada uno de ellos. Por cada tarea se especifica la fecha de inicio, fecha de finalización, el número de horas invertidas por cada persona y la cantidad de horas que se han utilizado los recursos físicos.

10.2.1. PT1 - Definición del Trabajo

Para empezar, este primer paquete de trabajo tiene como objetivo principal establecer los objetivos y la planificación, así como el presupuesto y el análisis de riesgos correspondiente al desarrollo del trabajo. Por ende, se trata de uno de los paquetes de trabajo más importantes de todos.

- **T1.1 - Definición de los objetivos**. Se definen los objetivos generales y específicos a cumplimentar a la finalización del trabajo. Estos deben ser claros y concisos, de manera que permitan una evaluación precisa de su cumplimiento.
    - MVH, JA: 10 horas cada persona.
    - AG: 20 horas.
    - PC: 20 horas.
- **T1.2 - Definición de la planificación y presupuesto**. Esta tarea implica la planificación del trabajo describiendo los diferentes paquetes de trabajo y tareas. Además, se realiza una estimación detallada de los costes del trabajo, dejando margen para otros posibles costes originados durante el desarrollo del mismo. Cabe destacar que esta tarea desempeña un papel fundamental, dado que determina tanto los plazos como el aspecto económico del trabajo.
    - MVH, JA: 10 horas cada persona.
    - AG: 20 horas.
    - PC: 20 horas.
- **T1.3 - Análisis de riesgos**. Con el fin de minimizar el impacto de los riesgos que puedan surgir, se lleva a cabo un estudio y análisis de los posibles riesgos que pueden interrumpir el desarrollo del trabajo. Al mismo tiempo, se diseña un plan de contingencia.
    - MVH, JA: 5 horas cada persona.
    - AG: 20 horas.
    - PC: 20 horas.

En la Tabla 15 se resumen las fechas de inicio y finalización de cada tarea.

Tarea	Fecha de inicio	Fecha de finalización
T1.1	01/02/2024	12/02/2024
T1.2	01/02/2024	12/02/2024
T1.3	12/02/2024	15/02/2024

Tabla 15: Tareas del paquete de trabajo de definición del trabajo.

10.2.2. PT2 - Diseño del Sistema de Reconocimiento Facial

Esta fase conlleva la realización del diseño del sistema de reconocimiento facial con base en los objetivos definidos previamente. Es fundamental realizar un diseño adecuado que facilite su posterior desarrollo sin impedimentos.

- **T2.1 - Análisis de alternativas**. Se realiza un estudio exhaustivo de todas las alternativas posibles para el diseño y la implementación del sistema. El objetivo de este estudio es evaluar distintas opciones y seleccionar las más adecuadas para cumplir los objetivos del trabajo.
    - MVH, JA: 5 horas cada persona.
    - AG: 30 horas.
    - PC: 30 horas.
- **T2.2 - Diseño de alto nivel**. Se diseña la arquitectura del sistema de reconocimiento facial, definiendo los módulos principales, sus funcionalidades y sus respectivas relaciones.
    - MVH, JA: 5 horas cada persona.
    - AG: 40 horas.
    - PC: 40 horas.
- **T2.3 - Diseño de los módulos**. Se diseñan los 2 módulos que conforman el sistema de reconocimiento facial basado en *Deep Learning* y aprendizaje federado. Esta tarea se denomina diseño de bajo nivel, donde se especifica detalladamente la arquitectura de cada módulo y su integración. Del mismo modo, se definen otros tantos aspectos relacionados con los mencionados módulos, tales como las imágenes con las que se va a trabajar, el volumen de las mismas, arquitectura del modelo de *Deep Learning*, entre otros.
    - MVH, JA: 5 horas cada persona.
    - AG: 40 horas.
    - PC: 40 horas.

En la Tabla 16 se resumen las fechas de inicio y finalización de cada tarea.

Tarea	Fecha de inicio	Fecha de finalización
T2.1	15/02/2024	22/02/2024
T2.2	22/02/2024	29/02/2024
T2.3	29/02/2024	07/03/2024

Tabla 16: Tareas del paquete de trabajo de diseño del sistema de reconocimiento facial.

10.2.3. PT3 - Implementación del Sistema de Reconocimiento Facial

Con base en el diseño tanto de alto nivel como de bajo nivel, se procede a realizar la implementación del sistema con el fin de valorar la viabilidad del mismo.

- **T3.1 - Implementación de los módulos**. En esta tarea se programan y desarrollan los módulos diseñados en la tarea T2.3, es decir, se traducen los diseños a código. Asimismo, se establecen las relaciones entre ellos para el correcto funcionamiento del sistema general.
    - MVH, JA: 10 horas cada persona.
    - AG: 160 horas.
    - PC: 160 horas.
    - SD: 160 horas.

En la Tabla 17 se resumen las fechas de inicio y finalización de cada tarea.

Tarea	Fecha de inicio	Fecha de finalización
T3.1	07/03/2024	15/04/2024

Tabla 17: Tareas del paquete de trabajo de implementación del sistema de reconocimiento facial.

10.2.4. PT4 - Diseño, Realización y Evaluación del Plan de Pruebas

Para finalizar el desarrollo del sistema, es necesario realizar su validación a través de diversas pruebas. Esto implica el diseño y la realización de un plan de pruebas de fiabilidad, para posteriormente extraer conclusiones significativas.

- **T4.1 - Diseño del plan de pruebas de fiabilidad**. Esta tarea conlleva la elaboración del diseño del plan de pruebas con el propósito de realizar un estudio de la configuración óptima del sistema de reconocimiento facial.
    - MVH, JA: 5 horas cada persona.
    - AG: 40 horas.
    - PC: 40 horas.
- **T4.2 - Realización de las pruebas de fiabilidad**. Se efectúan las pruebas diseñadas en la tarea T4.1.
    - MVH, JA: 5 horas cada persona.
    - AG: 100 horas.
    - PC: 100 horas.
    - SD: 100 horas.
- **T4.3 - Evaluación de los resultados obtenidos en las pruebas de fiabilidad**. Se procede a evaluar los resultados obtenidos y a extraer conclusiones pertinentes. Además, se deduce la configuración óptima del sistema para su posterior implementación en un caso de uso real futuro.
    - MVH, JA: 5 horas cada persona.
    - AG: 50 horas.
    - PC: 50 horas.

En la Tabla 18 se resumen las fechas de inicio y finalización de cada tarea.

Tarea	Fecha de inicio	Fecha de finalización
T4.1	15/04/2024	22/04/2024
T4.2	22/04/2024	20/05/2024
T4.3	20/05/2024	31/05/2024

Tabla 18: Tareas del paquete de trabajo de diseño, realización y evaluación del plan de pruebas.

10.2.5. PT5 - Gestión del Trabajo

La gestión del trabajo es el paquete de trabajo que abarca todo el transcurso del mismo. Al mismo tiempo que se desarrolla la parte técnica del trabajo, es indispensable realizar un seguimiento, así como la documentación del mismo.

- **T5.1 - Seguimiento del trabajo**. Con el propósito de verificar que el desarrollo del sistema marcha como es debido, se supervisa mediante reuniones periódicas. Se asegura que el avance del trabajo está alineado con la planificación definida en la tarea T1.2.
    - MVH, JA: 15 horas cada persona.
    - AG: 20 horas.
    - PC: 20 horas.
- **T5.2 - Redacción de la memoria**. Al mismo tiempo que se desarrollan los anteriores paquetes de trabajos, se redacta la documentación ligada al trabajo.
    - MVH, JA: 10 horas cada persona.
    - AG: 80 horas.
    - PC: 80 horas.

En la Tabla 19 se resumen las fechas de inicio y finalización de cada tarea.

Tarea	Fecha de inicio	Fecha de finalización
T5.1	01/02/2024	06/06/2024
T5.2	01/02/2024	06/06/2024

Tabla 19: Tareas del paquete de trabajo de gestión del trabajo.

10.3. Hitos del Proyecto

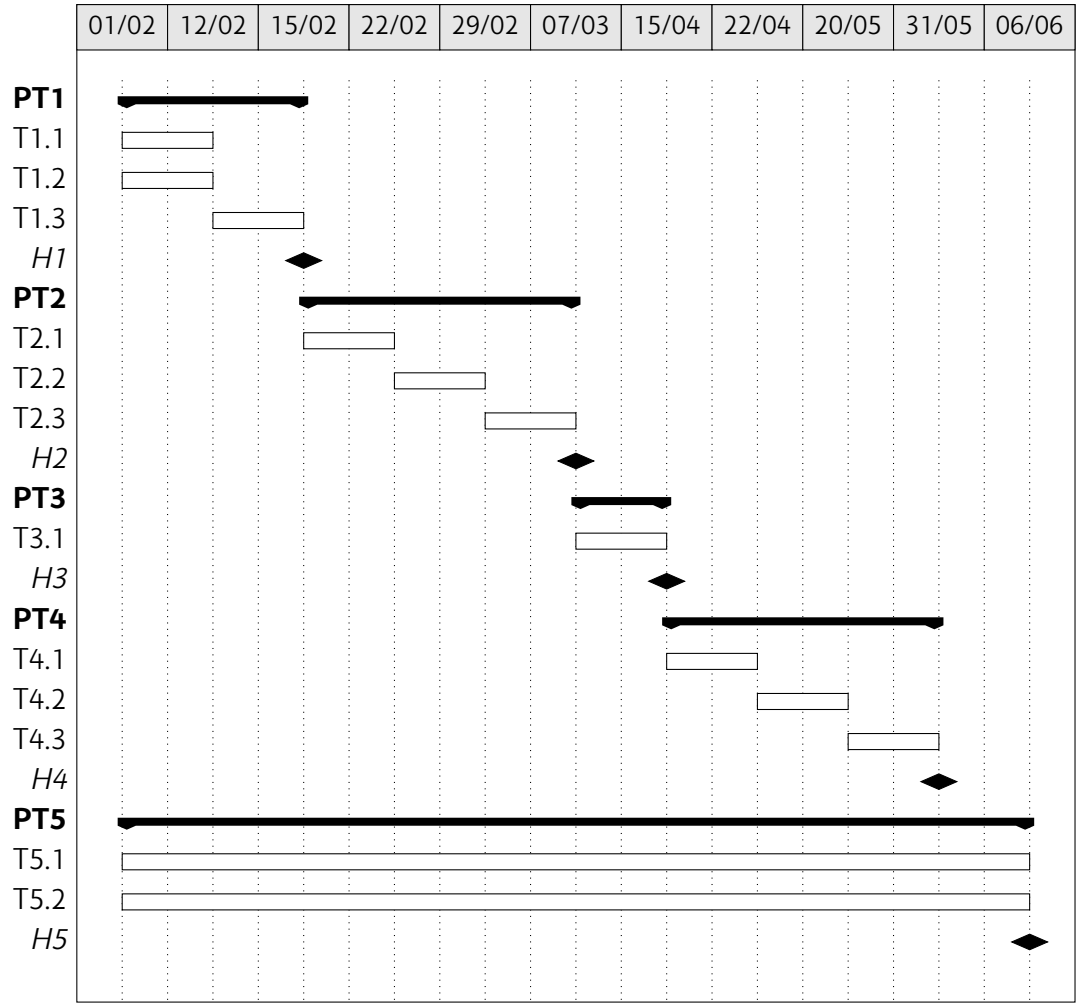
Los hitos son puntos de referencia de la planificación que marcan el cumplimiento de objetivos o eventos importantes. En la Tabla 20 se muestran los diversos hitos propuestos para este trabajo, junto con sus fechas de entrega y el paquete de trabajo al que pertenecen.

Identificador	Hito	Fecha de entrega	Paquete de trabajo
H1	Objetivos, planificación, presupuesto y análisis de riesgos definidos	15/02/2024	PT1
H2	Diseño del sistema realizado	07/03/2024	PT2
H3	Modulos desarrollados e implementados	15/04/2024	PT3
H4	Pruebas de fiabilidad efectuadas	31/05/2024	PT4
H5	Trabajo terminado y documentación redactada	06/06/2024	PT5

Tabla 20: Hitos del proyecto.

10.4. Diagrama de Gantt

Para finalizar la planificación, se resume que el trabajo comenzó el 1 de febrero de 2024 y finaliza el día 6 de junio de 2024, con la entrega de la memoria final en la plataforma ADDI. A continuación, se presenta el diagrama de Gantt.



11. Resumen de Costes

Un aspecto imprescindible en la realización de un trabajo es la estimación de los costes. Esta ayuda a verificar que los recursos disponibles se emplean de manera eficiente, así como a mantener el control financiero del trabajo. Por este motivo, a continuación se presenta el resumen de costes de este trabajo.

11.1. Horas Internas

Primeramente, cabe destacar que el desarrollo del presente trabajo es realizado por 2 ingenieras senior y 1 ingeniero junior, tal y como se ha mencionado en la planificación. Los costes asociados a los recursos humanos se presentan en la Tabla 21.

Identificador	Nombre	Dedicación (h)	Coste horario (€/h)	Coste (€)
MVH	Maria Victoria Higuero	90	55	4.950
JA	Jasone Astorga	90	55	4.950
AG	Ander Galván	620	20	12.400
			TOTAL	**22.300**

Tabla 21: Horas internas del trabajo.

11.2. Amortizaciones

En cuanto a los recursos físicos empleados, cabe mencionar que ninguno de ellos ha sido adquirido específicamente para el desarrollo de este trabajo. Por ello, en la Tabla 22 se muestran las amortizaciones asociadas a dichos equipos. Asimismo, las herramientas software utilizadas son todas de código abierto, lo que contribuye positivamente en el aspecto económico.

Identificador	Nombre	Coste adquisición (€)	Vida útil (h)	Uso (h)	Coste (€)
PC	Asus Zenbook	1.500	26.000	620	34,62
SD	Servidor Dell PowerEdge R730	3.000	44.000	260	17,73
				TOTAL	**52,35**

Tabla 22: Amortizaciones del trabajo.

11.3. Gastos

Este concepto corresponde al coste de los gastos no incluidos en otras categorías, incurridos durante el desarrollo del proyecto. El desglose de estos costes se muestra a

continuación en la Tabla 23.

Concepto	Número de horas (h)	Coste unitario (€/h)	Coste (€)
Facturas de Internet	-	-	430
Facturas de electricidad	-	-	480
		TOTAL	**910**

Tabla 23: Gastos del trabajo.

11.4. Subcontrataciones

Al realizar el trabajo en un grupo de investigación, no es necesaria la subcontratación de personal externo. Como se puede observar en la Tabla 24, en este proyecto no se ha subcontratado ningún trabajo, lo que hace que el coste de las subcontrataciones sea nulo.

Concepto	Número de horas (h)	Coste unitario (€/h)	Coste (€)
-	-	-	-
		TOTAL	**0**

Tabla 24: Subcontrataciones del trabajo.

11.5. Coste Total del Trabajo

Para finalizar, en la Tabla 25 se presenta el resumen de costes completo del trabajo, que asciende a **25.588,58 €** (IVA incluido). Cabe destacar que se reserva un 10 % del coste directo a futuros gastos indirectos que puedan originarse durante el desarrollo del trabajo.

Concepto	Coste (€)
Horas internas	22.300
Amortizaciones	52,35
Gastos	910
Subcontrataciones	0
SUBTOTAL	**23.262,35**
Costes indirectos (10 %)	2.326,23
TOTAL	**25.588,58**

Tabla 25: Resumen de costes del trabajo.

12. Conclusiones

La conclusión primordial y más destacada de este TFM es que se han alcanzado con éxito los objetivos planteados para el presente trabajo. De igual manera, la planificación y el presupuesto se han gestionado de manera eficaz, respetando los plazos y manteniéndose dentro de los costes estimados.

De forma más concreta, este trabajo ha demostrado cómo la Inteligencia Artificial es una herramienta muy poderosa en su aplicación junto al reconocimiento facial para el desarrollo de un sistema de identificación biométrica.

El estado del arte presentado manifiesta cómo la investigación del funcionamiento de las redes neuronales es un ámbito de actualidad. En consecuencia, este trabajo se ha centrado en el estudio del entrenamiento basado en dichas redes. Además, se ha elaborado un análisis de diversos escenarios para optimizar la fiabilidad del sistema. Para ello, se ha hecho uso de las técnicas *transfer learning* y *fine-tuning*. Con base en los resultados obtenidos, se ha concluido que el *fine-tuning* mejora significativamente la precisión del sistema de reconocimiento facial.

Es importante destacar que en el sistema desarrollado se garantiza la privacidad de los conjuntos de datos empleados mediante la utilización de la técnica denominada aprendizaje federado. Esta permite lograr una fiabilidad adecuada, a la vez que se proporciona la privacidad de los datos.

Asimismo, la solución propuesta es adaptable e implementable en una variedad de casos de uso reales, como la identificación de personas en distintos contextos, incluyendo seguridad, salud, entre otros.

En resumen, el presente trabajo contribuye al continuo avance de la investigación en los ámbitos de la Inteligencia Artificial y el reconocimiento facial. De cara al futuro, se tiene previsto realizar numerosos trabajos relacionados con el desarrollo de este proyecto.

- Validación de la fiabilidad del sistema frente a una mayor cantidad de imágenes con el objetivo de confirmar su escalabilidad y robustez, garantizando que se mantenga el nivel de precisión.
- Estudio de fiabilidad del sistema frente a imágenes faciales con distintas condiciones de iluminación y ángulos.
- Comparación con un escenario de aprendizaje centralizado para analizar el *trade-off* entre la precisión del sistema y la privacidad de los datos.

Referencias

[1] R. Stuart and P. Norvig, *Inteligencia Artificial: Un Enfoque Moderno*, 2004.

[2] "El año de la inteligencia artificial | Política Exterior." [Online]. Available: https://www.politicaexterior.com/articulo/el-ano-de-la-inteligencia-artificial/

[3] "AI Index Report 2024 – Artificial Intelligence Index." [Online]. Available: https://aiindex.stanford.edu/report/

[4] R. Mahato, *Artificial Intelligence, What is it?*, January 2022.

[5] P. Saikrishna and K. A. Basith, "An Efficient Application of Hybrid Optimization with Deep Learning Approach in the Prediction of Cardiovascular Disease," in *2023 3rd International Conference on Pervasive Computing and Social Networking (ICPCSN)*, 2023, pp. 692–697.

[6] D. Gostimirovic, D.-X. Xu, O. Liboiron-Ladouceur, and Y. Grinberg, "Deep Learning-Based Prediction of Fabrication-Process-Induced Structural Variations in Nanophotonic Devices," *ACS Photonics*, vol. 9, no. 8, pp. 2623–2633, 2022. [Online]. Available: https://doi.org/10,1021/acsphotonics,1c01973

[7] B. Lin, B. Ghaddar, and J. Nathwani, "Deep Reinforcement Learning for Electric Vehicle Routing Problem with Time Windows," *CoRR*, vol. abs/2010.02068, 2020. [Online]. Available: https://arxiv.org/abs/2010,02068

[8] J. Jha, A. K. Vishwakarma, C. N, A. Nithin, A. Sayal, A. Gupta, and R. Kumar, "Artificial Intelligence and Applications," in *2023 1st International Conference on Intelligent Computing and Research Trends (ICRT)*, 2023, pp. 1–4.

[9] M. M. Taye, "Understanding of Machine Learning with Deep Learning: Architectures, Workflow, Applications and Future Directions," *Computers*, vol. 12, no. 5, 2023. [Online]. Available: https://www.mdpi.com/2073-431X/12/5/91

[10] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *Journal of Big Data 2021 8:1*, vol. 8, pp. 1–74, 3 2021. [Online]. Available: https://journalofbigdata.springeropen.com/articles/10,1186/s40537-021-00444-8

[11] C. Janiesch, P. Zschech, and K. Heinrich, "Machine learning and deep learning," *CoRR*, vol. abs/2104.05314, 2021. [Online]. Available: https://arxiv.org/abs/2104,05314

[12] S. Madakam, T. Uchiya, S. Mark, and Y. Lurie, "Artificial Intelligence, Machine Learning and Deep Learning (Literature: Review and Metrics)," *Asia-Pacific Journal of Management Research and Innovation*, vol. 18, no. 1-2, pp. 7–23, 2022. [Online]. Available: https://doi.org/10,1177/2319510X221136682

[13] N. E. S. Rojas, B. S. M. Serna, E. M. P. Velásquez, and O. S. R. Galeano, "Reconocimiento del abecedario de la lengua de señas colombiana con Redes Neuronales Convolucionales," *Orinoquia*, vol. 25, pp. 25–30, 6 2021. [Online]. Available: https://orinoquia.unillanos.edu.co/index.php/orinoquia/article/view/680

[14] R. Ribani and M. Marengoni, "A Survey of Transfer Learning for Convolutional Neural Networks," in *2019 32nd SIBGRAPI Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T)*, 2019, pp. 47–57.

[15] T.-W. Li and G.-C. Lee, "Performance Analysis of Fine-tune Transferred Deep Learning," in *2021 IEEE 3rd Eurasia Conference on IOT, Communication and Engineering (ECICE)*, 2021, pp. 315–319.

[16] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated Machine Learning: Concept and Applications," *CoRR*, vol. abs/1902.04885, 2019. [Online]. Available: http://arxiv.org/abs/1902,04885

[17] Q. Li, Z. Wen, Z. Wu, S. Hu, N. Wang, X. Liu, and B. He, "A Survey on Federated Learning Systems: Vision, Hype and Reality for Data Privacy and Protection," *CoRR*, vol. abs/1907.09693, 2019. [Online]. Available: http://arxiv.org/abs/1907,09693

[18] "Data protection in the EU - European Commission." [Online]. Available: https://commission.europa.eu/law/law-topic/data-protection/data-protection-eu_en

[19] M. Arafeh, A. Hammoud, H. Otrok, A. Mourad, C. Talhi, and Z. Dziong, "Independent and Identically Distributed (IID) Data Assessment in Federated Learning," in *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*, 2022, pp. 293–298.

[20] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated Learning with Non-IID Data," *CoRR*, vol. abs/1806.00582, 2018. [Online]. Available: http://arxiv.org/abs/1806,00582

[21] M. Moshawrab, M. Adda, A. Bouzouane, H. Ibrahim, and A. Raad, "Reviewing Federated Learning Aggregation Algorithms; Strategies, Contributions, Limitations and Future Perspectives," *Electronics*, vol. 12, no. 10, 2023. [Online]. Available: https://www.mdpi.com/2079-9292/12/10/2287

[22] A. K. Jain, A. Ross, and K. Nandakumar, "An introduction to biometrics," in *2008 19th International Conference on Pattern Recognition*, 2008, pp. 1–1.

[23] Y. Rawat, Y. Gupta, G. Khothari, A. Mittal, and D. Rautela, "The Role of Artificial Intelligence in Biometrics," in *2023 2nd International Conference on Edge Computing and Applications (ICECAA)*, 2023, pp. 622–626.

[24] S. Minaee, A. Abdolrashidi, H. Su, M. Bennamoun, and D. Zhang, "Biometrics recognition using deep learning: a survey," *Artif. Intell. Rev.*, vol. 56, no. 8, pp. 8647–8695, aug 2023.

[25] X. Wang, J. Peng, S. Zhang, B. Chen, Y. Wang, and Y. Guo, "A Survey of Face Recognition," 2022.

[26] D. Aggarwal, J. Zhou, and A. K. Jain, "FedFace: Collaborative Learning of Face Recognition Model," 2021.

[27] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu, "CosFace: Large Margin Cosine Loss for Deep Face Recognition," 2018.

[28] Y. Ding, X. Wu, Z. Li, Z. Wu, S. Tan, Q. Xu, W. Pan, and Q. Yang, "An Efficient Industrial Federated Learning Framework for AIoT: A Face Recognition Application," 2022.

[29] L. Liu, Y. Zhang, H. Gao, X. Yu, and J. Cheng, "FedFV: federated face verification via equivalent class embeddings," *Multimedia Systems*, vol. 28, pp. 1833–1843, 10 2022. [Online]. Available: https://link.springer.com/article/10,1007/s00530-022-00927-5

[30] Y. Niu and W. Deng, "Federated Learning for Face Recognition with Gradient Correction," 2021.

[31] I. Br nescu, R.-I. Ciobanu, C. Dobre, and C. Mavromoustakis, "Decentralized Machine Learning for Face Recognition," in *2023 22nd International Symposium on Parallel and Distributed Computing (ISPDC)*, 2023, pp. 1–8.

[32] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 07-12-June-2015, pp. 815–823, 3 2015. [Online]. Available: https://arxiv.org/abs/1503,03832v3

[33] M. Heidari and K. Fouladi-Ghaleh, "Using Siamese Networks with Transfer Learning for Face Recognition on Small-Samples Datasets," in *2020 International Conference on Machine Vision and Image Processing (MVIP)*, 2020, pp. 1–4.

[34] "Very Deep Convolutional Networks for Large-Scale Image Recognition, author=Karen Simonyan and Andrew Zisserman," 2015.

[35] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[36] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments." [Online]. Available: http://vis-www.cs.umass.edu/lfw/.

[37] J. Zhang, X. Jin, Y. Liu, A. K. Sangaiah, and J. Wang, "Small Sample Face Recognition Algorithm based on Novel Siamese Network," *Journal of Information Processing Systems*, vol. 14, pp. 1464–1479, 2018. [Online]. Available: https://doi.org/10,3745/JIPS,02,0101

[38] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, 2005, pp. 539–546 vol. 1.

[39] H. Zheng, B. Li, G. Liu, Y. Li, Y. Zhang, W. Gao, and X. Zhao, "Blockchain-based Federated Learning Framework Applied in Face Recognition," in *2022 7th International Conference on Signal and Image Processing (ICSIP)*, 2022, pp. 265–269.

[40] S. Khan, E. Ahmed, M. H. Javed, S. A. A Shah, and S. U. Ali, "Transfer Learning of a Neural Network Using Deep Learning to Perform Face Recognition," in *2019 International Conference on Electrical, Communication, and Computer Engineering (ICECCE)*, 2019, pp. 1–5.

[41] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks." [Online]. Available: http://code.google.com/p/cuda-convnet/

[42] M. Zulfiqar, F. Syed, M. J. Khan, and K. Khurshid, "Deep Face Recognition for Biometric Authentication," in *2019 International Conference on Electrical, Communication, and Computer Engineering (ICECCE)*, 2019, pp. 1–6.

[43] "PyTorch vs TensorFlow vs Keras for Deep Learning: A Comparative Guide | DataCamp." [Online]. Available: https://www.datacamp.com/tutorial/pytorch-vs-tensorflow-vs-keras

[44] "TensorFlow." [Online]. Available: https://www.tensorflow.org/?hl=es

[45] "PyTorch." [Online]. Available: https://pytorch.org/

[46] "Keras: Deep Learning for humans." [Online]. Available: https://keras.io/

[47] N. M. Jebreel, J. Domingo-Ferrer, A. Blanco-Justicia, and D. Sánchez, "Enhanced Security and Privacy via Fragmented Federated Learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 5, p. 6703–6717, May 2024. [Online]. Available: http://dx.doi.org/10,1109/TNNLS,2022,3212627

[48] "Understanding Distance Metrics in Vector Embeddings: Cosine Similarity, Euclidean Distance, and Dot Product | LinkedIn." [Online]. Available: https://www.linkedin.com/pulse/understanding-distance-metrics-vector-embeddings-cosine-bilal-shaikh-qunwf/

[49] "Pins Face Recognition." [Online]. Available: https://www.kaggle.com/datasets/hereisburak/pins-face-recognition

[50] "torchvision — Torchvision 0.18 documentation." [Online]. Available: https://pytorch.org/vision/stable/index.html

[51] "timesler/facenet-pytorch: Pretrained Pytorch face detection (MTCNN) and facial recognition (InceptionResnet) models." [Online]. Available: https://github.com/timesler/facenet-pytorch

[52] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, "VGGFace2: A dataset for recognising faces across pose and age," *Proceedings - 13th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2018*, pp. 67–74, 10 2017. [Online]. Available: https://arxiv.org/abs/1710,08092v2

[53] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Learning Face Representation from Scratch," 11 2014. [Online]. Available: https://arxiv.org/abs/1411,7923v1

[54] "jfilter/split-folders: Split folders with files (i.e. images) into training, validation and test (dataset) folders." [Online]. Available: https://github.com/jfilter/split-folders/tree/main