



**Universidad
Europea**

UNIVERSIDAD EUROPEA DE MADRID

ESCUELA DE ARQUITECTURA, INGENIERÍA Y DISEÑO

GRADO EN INGENIERÍA EN MATEMÁTICAS APLICADA AL

ANÁLISIS DE DATOS

PROYECTO FIN DE GRADO

**MODELO DE PREDICCIÓN DE OCUPACIÓN DE
VIVIENDAS**

FRANCISCO JAVIER DÍAZ GARCÍA

Dirigido por

CHRISTIAN VLADIMIR SUCUZHANAY ARÉVALO

CURSO 2020-2021

TÍTULO: MODELO DE PREDICCIÓN DE OCUPACIÓN DE VIVIENDAS

AUTOR: FRANCISCO JAVIER DÍAZ GARCÍA

TITULACIÓN: GRADO EN INGENIERÍA MATEMÁTICA APLICADA AL ANÁLISIS DE DATOS

DIRECTOR/ES DEL PROYECTO: CHRISTIAN VLADIMIR SUCUZHANAY ARÉVALO

FECHA: JULIO de 2021

RESUMEN

Actualmente, debido a la crisis que afronta España tanto económica como social acentuada por la pandemia ocasionada por el virus COVID-19, se ha producido un gran aumento en los índices de ocupación ilegal de viviendas. Madrid, es una de las ciudades más afectadas y es por eso que el desarrollo del proyecto se ha centrado en esta capital.

Vivimos en una época de grandes avances tecnológicos, principalmente en el campo de la inteligencia artificial. Es por eso que, como solución a este problema, el objetivo de este proyecto ha sido la creación de un modelo de predicción automático basado en redes neuronales, más concretamente en *Deep Learning*, para poder predecir si una vivienda va a ser objetivo de ser ocupada y de esta manera, poder anticiparse para tomar medidas preventivas.

Para su desarrollo se ha estudiado el contexto y los factores influyentes en la ocupación, y de esta forma se han creado datos sintéticos con los que se ha entrenado el modelo.

Se ha conseguido desarrollar un modelo con muy buenos resultados, y una interfaz web completamente funcional para poder utilizarlo.

Palabras clave: *machine learning*, *web scraping*, *Deep Learning*, ocupación, datos sintéticos.

ABSTRACT

Currently, due to the economic and social crisis facing Spain, accentuated by the pandemic caused by the COVID-19 virus, there has been a large increase in the rates of squatting. Madrid is one of the most affected cities and that is why the development of the project has focused on this capital.

We live in a time of great technological advances, mainly in the field of artificial intelligence. That is why, as a solution to this problem, the aim of this project has been the creation of an automatic prediction model based on neural networks, more specifically on Deep Learning, to be able to predict whether a house is going to be the target of being occupied and thus be able to anticipate in order to take preventive measures.

For its development, the context and the factors influencing occupation have been studied, and in this way synthetic data have been created with which the model has been trained.

We have managed to develop a model with very good results, and a fully functional web interface to be able to use it.

Keywords: machine learning, web scraping, deep learning, occupation, synthetic data.

AGRADECIMIENTOS

Especial agradecimiento a todos los profesores que me han enseñado a navegar hasta este puerto.

"It is not in the stars to hold our destiny but in ourselves".

W.S.

TABLA RESUMEN

	DATOS
Nombre y apellidos:	Francisco Javier Díaz García
Título del proyecto:	MODELO DE PREDICCIÓN DE OCUPACIÓN DE VIVIENDAS
Directores del proyecto:	Francisco Javier Díaz García
El proyecto se ha realizado en colaboración de una empresa o a petición de una empresa:	NO
El proyecto ha implementado un producto	SI
El proyecto ha consistido en el desarrollo de una investigación o innovación:	SI
Objetivo general del proyecto:	Crear un modelo de predicción para comprobar si una vivienda de Madrid será ocupada ilegalmente

Tabla 1. Tabla resumen

Índice

RESUMEN	3
ABSTRACT	4
TABLA RESUMEN	7
Capítulo 1. RESUMEN DEL PROYECTO	12
1.1 Contexto y justificación	12
1.2 Planteamiento del problema	12
1.3 Objetivos del proyecto	12
1.4 Resultados obtenidos	12
1.5 Estructura de la memoria	12
Capítulo 2. ANTECEDENTES / ESTADO DEL ARTE	13
2.1 Estado del arte	13
2.2 Contexto y justificación	14
2.3 Planteamiento del problema	15
Capítulo 3. OBJETIVOS	16
3.1 Objetivos generales	16
3.2 Objetivos específicos	16
3.3 Beneficios del proyecto	16
Capítulo 4. DESARROLLO DEL PROYECTO	17
4.1 Planificación del proyecto	17
4.2 Descripción de la solución, metodologías y herramientas empleadas	19
4.3 Recursos requeridos	28
4.4 Presupuesto	29
4.5 Viabilidad	29
4.6 Resultados del proyecto	32
Capítulo 5. DISCUSIÓN	38
5.1 Resultados	38
5.2 Limitaciones de estudio	42
5.3 Cambios en el proyecto	42

Capítulo 6.	CONCLUSIONES	45
6.1	Conclusiones del trabajo	45
6.2	Conclusiones personales	45
Capítulo 7.	FUTURAS LÍNEAS DE TRABAJO	48
Capítulo 8.	REFERENCIAS.....	50
Capítulo 9.	ANEXOS	54
9.1	Uso de la aplicación web.....	54
9.2	Repositorio de GitHub.....	55

Índice de Figuras

Figura 1. Diagrama de Gantt (Convocatoria de junio)	17
Figura 2. Diagrama de Gantt (Convocatoria de julio).....	19
Figura 3. Fórmula de la función de activación ReLU [25].....	25
Figura 4. Fórmula para establecer el número de neuronas de la capa oculta [26]	25
Figura 5. Fórmula de la función de activación Sigmoid [27]	26
Figura 6. Fórmula de cálculo de precisión [34]	32
Figura 7. Gráfica de evaluación de precisión del modelo	33
Figura 8. Gráfica de evaluación de la pérdida del modelo.....	33
Figura 9. Interfaz de aplicación web	36
Figura 10. Interfaz de aplicación web - Pantalla de ocupado [35]	36
Figura 11. Interfaz de aplicación web - Pantalla de no ocupado [36].....	37
Figura 12. Gráfica de evaluación de precisión de modelo anterior	38
Figura 13. Interfaz de aplicación web - Ejemplo de campos completados	54

Índice de Tablas

Tabla 1. Tabla resumen	7
Tabla 2. Tabla de presupuesto	29
Tabla 3. Tabla de estimación de coste de Cloud Functions	30
Tabla 4. Tabla de precios de Cloud Storage	31
Tabla 5. Tabla de estimación de costes de Cloud Storage	31
Tabla 6. Tabla de resultados de modelo definitivo	35
Tabla 7. Tabla de resultados de modelo descartado I	39
Tabla 8. Tabla de resultados de modelo descartado II	40
Tabla 9. Tabla de resultados de modelo descartado de la Comunidad de Madrid	41

Capítulo 1. RESUMEN DEL PROYECTO

1.1 Contexto y justificación

La ocupación ilegal es un problema que ha adquirido una gran relevancia y que tiene una fuerte repercusión a nivel social y económico. Esta situación se ha visto acentuada por la pandemia ocasionada por el COVID-19 y las medidas y políticas de actuación por parte del Gobierno para combatirlo [1]. En los vecindarios con una tasa elevada de ocupación, se genera malestar, inseguridad y conflictividad, además de una devaluación de las propiedades ocupadas, de entre un 40 y 60% de su valor [2].

1.2 Planteamiento del problema

Actualmente, existen tres vías para solucionar un problema de ocupación ilegal: la vía legal/judicial, negociación con los presuntos ocupas y la contratación de empresas dedicadas al desalojo. El problema que se plantea es que las tres vías mencionadas son de actuación tras la ocupación del inmueble, no medidas preventivas. Se pueden instalar alarmas o puertas y ventanas de seguridad, pero estas instalaciones suponen un coste, por lo que lo apropiado es optimizarlo instalándolo únicamente si el inmueble es propenso a ser ocupado.

1.3 Objetivos del proyecto

Los principales objetivos del proyecto consisten en estudiar y comprender el estado de la ocupación en España, concretamente en Madrid. Tras esto, ver los factores que influyen directamente en la ocupación de una vivienda para poder obtener las variables más representativas y así generar los datos sintéticos con los que se desarrollará el modelo. Finalmente, se desarrollará un modelo de predicción automático con *machine learning*.

1.4 Resultados obtenidos

Finalmente, se ha conseguido desarrollar un modelo predictivo para ver si una vivienda con unas ciertas características va a ser ocupada. Tras una gran cantidad de modelos diferentes, se han obtenido unos resultados muy satisfactorios, ya que se ha conseguido llegar a uno que predice de manera bastante fiel a la realidad, basado en los diferentes estudios y documentación que se ha utilizado.

1.5 Estructura de la memoria

La memoria de este proyecto está organizada en nueve apartados. En este primero se resume brevemente en qué ha consistido el proyecto, mientras que, en el segundo, se contextualiza. Después se organizan los objetivos de trabajo para continuar explicando en detalle cómo se ha desarrollado. A continuación, se hace una retrospectiva del proyecto analizando los resultados obtenidos, las limitaciones de estudio y los cambios realizados, para terminar con las conclusiones y las futuras líneas de trabajo. Por último, en el capítulo ocho y nueve se recogen las fuentes utilizadas y los anexos.

Capítulo 2. ANTECEDENTES / ESTADO DEL ARTE

2.1 Estado del arte

La ocupación ilegal es un problema que ha conseguido una gran relevancia y que tiene una fuerte repercusión a nivel social y económico. Esta situación se ha visto acentuada por la pandemia ocasionada por el COVID-19 y las medidas por parte del Gobierno para combatirlo [1]. Este problema de ocupación al que nos enfrentamos genera, en los vecindarios con una tasa elevada de ocupación, malestar, inseguridad y conflictividad, además de una pérdida importante de valor de las propiedades que han sido ocupadas, de entre un 40 y 60% de su valor [2].

Cada año, el número de denuncias por ocupación de inmuebles tiende a aumentar, ya que, según los datos publicados por el Ministerio del Interior, en tan solo cuatro años este número de denuncias ha crecido un 41%, de 10.376 denuncias en 2015 a las 14.621 con las que se cerró en el 2019. En el primer semestre de 2020 (últimos datos publicados por el Ministerio de Interior hasta la fecha) el número de denuncias creció un 5% respecto a las de 2019, con 7.450 denuncias en 2020 frente a las 7.093 de 2019 [3, 4].

Con estos datos, se puede ver que la ocupación ilegal es un problema creciente en España y para el que no hay soluciones realmente buenas para este problema.

Hoy en día existen tres vías por las que se puede solucionar este problema:

1. La vía legal/judicial

El primer problema al que se enfrenta esta solución es que un delito de ocupación tiene que ser denunciado en menos de 48 horas. Dentro de este margen de tiempo, la policía puede acudir a la propiedad ocupada y desalojar a los ocupas, pero no por ocupación si no por un supuesto delito de allanamiento de morada [5].

Una vez transcurrido este período, el problema se hace mayor debido a la lentitud del sistema judicial. Esto consistiría en interponer una demanda por precario contra el ocupa del inmueble. La base de esta acción es que el presunto ocupa no tiene ningún título que justifique su ocupación, y entonces, lo que realmente reclama el dueño es una restitución de la posesión [5, 6]. En muchas ocasiones la resolución del caso, que finalizaría con el desalojo de los ocupas, puede alargarse hasta periodos superiores al año, o más si la familia ocupa cuenta con menores a su cargo viviendo en el inmueble.

2. Negociando con los ocupas ilegales del inmueble

Otra posible solución que nos podemos encontrar es la negociación con los ocupas. Estas negociaciones son de carácter monetario y suelen ser ofrecer una cantidad de dinero a cambio de que los inquilinos se marchen [6].

Los problemas que presenta esta solución son varios. El primero es que esta negociación no se hace una forma legal, ya que no tiene ningún tipo de documento jurídico que alegue las condiciones del pacto de la negociación. Otro problema que presenta, derivado del anterior, es que al final se realiza un pago en dinero negro y sin ninguna garantía, puesto que los ocupas podrían volver y supondría un desperdicio de dinero. El tercer y último inconveniente que presenta es que en la mayoría de los casos las ocupaciones están gestionadas por mafias, y se aprovechan de estas negociaciones para obtener dinero de sus víctimas.

3. Contratando empresas de desocupación

Esta opción es la más rápida de todas. Consiste en la contratación del servicio de empresas que se dedican a desalojar a los ocupas ilegales. Estas empresas han emergido debido a la necesidad de recuperación de las viviendas ocupadas y debido a la lentitud del sistema judicial.

El principal problema que presenta la contratación de estas empresas es que actúan de una manera polémica, ya que “bailan en la frontera de la legalidad”. Esto se debe a que la forma de desalojar es mediante extorsión y amenazas a los inquilinos, y los propietarios originales, que sufren la ocupación de su propiedad, podrían enfrentarse a demandas por allanamiento de morada, injurias, amenazas o agresión [7].

2.2 Contexto y justificación

Ante este problema económico y social, las soluciones existentes no son muy eficaces. Para poder solucionarlo se debería cambiar la Ley para que proteja al propietario. Pero mientras esto no ocurra la mejor solución es la prevención.

La solución que se plantea es la de poner vigilancia en las viviendas susceptibles de ser ocupadas, estas sean porque se encuentran en venta, porque son segundas residencias o simplemente están vacías temporalmente. Aunque eficaz, esta solución no sería eficiente, ya que la vigilancia es un servicio caro y no merece la pena contratarlo para tener controladas todas las propiedades por prevención. La única forma de que esta solución fuese plausible sería conociendo la probabilidad de que una vivienda pueda ser ocupada, optimizándose el proceso de contratación de vigilancia o servicio de alarma.

Para ello, el mayor aliado son las matemáticas, los datos y el *machine learning*. Con estas herramientas tan potentes, se puede crear un modelo con el que ver la probabilidad de que un inmueble pueda ser ocupado y actuar en función a esa información.

El modelo desarrollado sería de gran interés para empresas con múltiples propiedades, como puede ser un banco, ya que como se ha comentado previamente, al tener muchas propiedades no es viable la contratación de vigilancia en todos los inmuebles. De esta forma, se ahorraría en costes y además se evitaría la ocupación de las viviendas, evitando así, pérdidas económicas. Además de bancos, es un producto de gran interés para las propias empresas de

alarmas, ya que utilizándolo ellos pueden realizar estimaciones de los lugares más propensos a ser ocupados y poder realizar campañas de marketing en esos lugares, y pudiendo establecer tarifas de sus servicios. Un caso muy similar sería el uso del modelo por parte de constructoras, puesto que también podrían utilizar el modelo para establecer en qué lugares es más seguro construir y realizar ahí sus promociones.

Actualmente no existe ningún modelo de predicción similar ni herramienta que realice esta función en el mundo empresarial, pero sí que existe la tecnología para poder crearse.

2.3 Planteamiento del problema

Los efectos de la pandemia provocada por el virus COVID-19, están afectando al mercado inmobiliario, y cada vez la decadencia en este sector va aumentando debido al impacto económico, sanitario y social del coronavirus. Se identifican dos consecuencias fundamentales que acentúan este problema: los desequilibrios y el aumento de la amenaza que supone la ocupación ilegal [8]. Este problema afecta principalmente a grandes ciudades capitales de provincia, las más afectadas son Madrid y Barcelona.

La ocupación genera pérdidas económicas a los grandes propietarios de inmuebles, ya que cuando una vivienda ha sido ocupada es el propietario quien tiene que hacerse cargo del proceso de desahucio, el cual conlleva un coste, y a parte de este coste de gestión la vivienda se devalúa hasta un 42.4% [9]. Por otro lado, aparte de producir pérdidas económicas también genera una situación de malestar y conflictos a los vecinos de las zonas con alto índice de ocupación.

Por lo tanto, el problema identificado es la ocupación ilegal de inmuebles y todos los daños colaterales que esta situación acarrea, y la necesidad de los propietarios de solucionarlo en el menor tiempo posible, ya que como se ha visto en el apartado 2.1 Estado del arte, las posibles vías para solucionar este problema son muy lentas o implican riesgos legales para el propietario.

La solución que se plantea a este problema es el de desarrollar un modelo de predicción automática con *machine learning*, con el que una persona pueda anticiparse a que su vivienda pueda llegar a ser ocupada y de esta forma evitar todos los problemas, ya comentados, que esto acarrea tanto para la vivienda y barrio donde se sitúa, como para el propietario.

Es ese por tanto el objetivo de este proyecto, conseguir crear e implementar un modelo con el que poder anticiparse a la ocupación y actuar en consecuencia, y de esta manera se puede poner solución al problema identificado.

Capítulo 3. OBJETIVOS

3.1 Objetivos generales

El objetivo general del presente trabajo consiste en desarrollar un modelo predictivo, con *machine learning*, para realizar la predicción de si una vivienda va a ser ocupada de forma ilegal.

3.2 Objetivos específicos

Objetivos específicos:

- Documentación e investigación para el desarrollo del proyecto.
- Recopilar datos e información mediante *web scraping*.
- Crear un *dataset* con los datos necesarios para la creación de datos sintéticos.
- Creación de datos sintéticos.
- Desarrollo del modelo de predicción.
- Pruebas para el modelo.
- Reajustes.
- Crear un nuevo modelo realizando un preprocesado más exhaustivo de los datos sintéticos y desarrollarlo en la nube.
- Desarrollo de una aplicación web utilizando Flask.

3.3 Beneficios del proyecto

El modelo de predicción pretende ser una herramienta para ayudar a los propietarios, principalmente optimizando costes, a la hora de prevenir que una vivienda sea ocupada. Con optimización de costes, se refiere a que se podría poner vigilancia y alarma en todos los inmuebles, pero eso no es eficiente porque contratar un servicio de vigilancia para una vivienda cuya probabilidad de ser ocupada es mínima, supone un derroche.

Además de ayudar a prevenir la ocupación, como ya se ha mencionado en el apartado 2.2 Contexto y justificación, actualmente las medidas de actuación para recuperar una vivienda ocupada son muy lentas o no muy eficaces, y en ese período de tiempo en el que la vivienda está ocupada, los ocupas la deterioran y destrozan, lo que conlleva pérdidas económicas.

La mejor solución que se puede plantear, al menos con la ley actual, es prevenir que sea ocupada, ya que así se evitan problemas. Para prevenirlo hay que anticiparse, y es ahí donde entra en juego el modelo de predicción, aportando así todo su valor. Por lo tanto, ese es el beneficio del proyecto, conseguir con él poner solución a un problema de gran envergadura como es la ocupación ilegal.

Capítulo 4. DESARROLLO DEL PROYECTO

4.1 Planificación del proyecto

4.1.1 Planificación del proyecto de convocatoria de junio



Figura 1. Diagrama de Gantt (Convocatoria de junio)

- **Documentar la situación actual y evolución del tema de la ocupación** (15-ene, 29-ene)
- **Documentar el estado del arte** (30-ene, 06-feb)
- **Estudiar localización geográfica dónde se aplicará el modelo** (07-feb, 14-feb)

Tras haber realizado la documentación, estudiar dónde es mayor el índice de ocupación y, por tanto, dónde se aplicará el modelo.

- **Estudiar las variables y factores que influyen en la ocupación** (15-feb, 01-mar)
- **Recopilar datos** (01-mar, 09-abr)

La recopilación de datos se refiere a la tarea de obtener datos para la creación de los datos sintéticos. Estos datos se obtienen mediante *web scraping* y mediante recopilación de información de *datasets* públicos.

- **Desarrollar script para realizar *web scraping* en diferentes dominios web para la obtención de datos e información** (01-mar, 22-mar)

- **Desarrollar script de tratamiento de texto para procesar los datos obtenidos con el programa de *web scraping*** (23-mar, 30-mar)
- **Almacenar los datos** (30-mar, 31-mar)
- **Obtener datos recopilados en *datasets* de fuentes fiables** (INE, Ministerio de interior, Ayuntamiento de Madrid, etc.) (01-abr, 09-abr)
- **Crear un *dataset* con los datos necesarios para la creación de datos sintéticos** (10-abr, 17-abr)

Este *dataset*, es un conjunto de datos de Madrid capital considerados importantes para el modelo, y con ellos se crearán los datos sintéticos.

- **Crear datos sintéticos para el modelo** (18-abr, 20-abr)

Establecer las variables que se van a utilizar y realizar los cálculos y operaciones pertinentes.
- **Idear una fórmula de probabilidad y de requisitos, para establecer que una vivienda ha sido ocupada** (21-abr, 28-abr)

Como los datos son sintéticos, para asignar a una vivienda que ha sido ocupada, hay que crear una fórmula con la que se calculará una probabilidad y, a partir de ella, asignar si ha sido o no ocupada.
- **Crear los datos** (29-abr, 30-abr)
- **Preprocesar los datos sintéticos para el entrenamiento del modelo** (01-may, 02-may)

Preparar los datos para entrenar el modelo realizando preprocesado de los mismos.
- **Diseñar y programar la red neuronal** (03-may, 04-may)
- **Entrenar el modelo con los datos sintéticos procesados** (03-may, 05-may)
- **Crear un test para probar el modelo con diferentes domicilios de viviendas** (04-may, 05-may)
- **Realizar pruebas para comprobar los resultados** (05-may, 06-may)
- **Reajustes** (06-may, 18-may)

En función de los resultados obtenidos, realizar ciertos reajustes en los datos sintéticos, en el preprocesado, en el test, etc.

4.1.2 Planificación del proyecto de convocatoria de junio

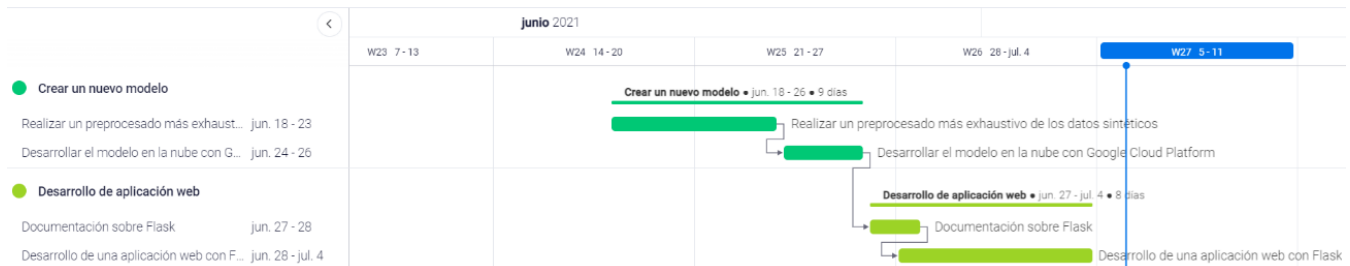


Figura 2. Diagrama de Gantt (Convocatoria de julio)

- **Mejorar el modelo realizando un nuevo preprocesado de los datos para el entrenamiento** (18-jun, 23-jun)
- **Desarrollar el modelo en la nube con Google Cloud Platform** (24-jun, 26-jun)
Para ello se ha creado una máquina virtual en la plataforma de Google GCP, para de esta manera tener una mayor potencia de cómputo.
- **Desarrollo de una aplicación web utilizando Flask** (27-jun, 3-jul)
Se decidió trabajar la parte visual del modelo, y para ello se ha creado una aplicación web para poder utilizarse, y para ello se ha utilizado el *framework* Flask, el cual permite desarrollar aplicaciones web con Python.
- **Mejora de la memoria.** (18-jun, 6-jul)
Para la convocatoria de julio, ha habido que realizar cambios en la memoria y corrección de algunos apartados.

4.2 Descripción de la solución, metodologías y herramientas empleadas

El objetivo del proyecto era el desarrollo de un modelo de predicción y clasificación automática para predecir si una vivienda va a ser ocupada ilegalmente. Para lograrlo, y llegar a la solución esperada, es imprescindible obtener datos sobre la ocupación, ya que es nuestra variable objetivo ver si va a ser ocupada.

4.2.1 Documentación

El primer paso consistía en documentarse sobre el tema. Se comenzó estudiando el estado del arte para investigar las alternativas existentes en el mercado al modelo desarrollado y comprobar si existe algún producto o servicio similar. Con el estudio del estado del arte, se procedió a documentar el contexto social y económico en el que se encuentra el problema

identificado y al que se quiere poner solución. Toda esta información está recogida y detallada en el punto 2. Antecedentes/Estado del arte.

Una vez estudiado el contexto y estado del arte, la prioridad era identificar los diferentes factores y variables que influyen en el proceso de ocupación ilegal. Para esta investigación, la mayor parte de la información proviene de noticias publicadas sobre algunos casos. Esto es debido a que no existen estudios rigurosos sobre este tema, solamente hay algunos datos muy generales en los que se agrupan zonas o regiones y lo más específico que se puede encontrar es un dato del número de viviendas ocupadas en un distrito de Madrid, e incluso en ocasiones, este dato está agrupado por pares de distritos [10].

Lo ideal para obtener los datos más rigurosos y de mayor calidad, sería poder tener acceso a todas las denuncias y procesos judiciales sobre la ocupación en Madrid, ya que estos recogen la dirección concreta de los domicilios ocupados. Estos datos no son accesibles por motivos de privacidad y leyes de protección de datos.

4.2.2 Extracción de datos

Una vez se ha documentado y se conoce el estado del arte, contexto que engloban el problema que se quiere abordar y se tiene una idea de los factores que influyen en la ocupación de viviendas de Madrid, se procedió a la extracción de datos y de información relevante e interés con la que se han desarrollado los datos sintéticos que han entrenado el modelo.

La recolección de toda esta cantidad de datos e información se ha utilizado una técnica conocida como *web scraping*. El *web scraping* es una técnica de extracción de información de sitios web utilizando programas de software [11]. Con esta técnica se simula navegar por el dominio web como lo haría una persona, pero al hacerlo con un programa de *software*, la cantidad de información que se puede procesar es mucho mayor que navegando a mano, ya que se puede recoger toda la información publicada en el dominio web de una sola vez y almacenarse en cuestión de segundos. En este caso, se ha realizado en Python con las librerías BeautifulSoup y Requests.

Se ha desarrollado un *crawler*, que es como se denomina al programa que realiza el *web scraping*, en el que dada cualquier noticia relacionada con la ocupación se extraen todos los datos e información relevante que pueda ser útil para el desarrollo de los datos sintéticos. Este *crawler*, accede a un documento con formato .txt, el cual contiene todos los *links* con noticias, artículos y cualquier sitio web que trate el tema de la ocupación. El *crawler* accede a cada *link* y guarda toda la información pública que contiene ese dominio. La información guardada, ya puede utilizarse, pero se consideró que lo más adecuado era procesarla para obtener únicamente datos concretos sobre ocupación, además de reducir el tamaño de los archivos guardados.

Para el procesado del texto, se han seguido una serie de criterios.

Lo primero fue identificar qué información era de interés. Como a partir de esta información se crean los datos sintéticos, lo que interesa son los datos numéricos, promedios y

porcentajes. Tras haber identificado qué tipo de información se quería obtener, se filtra el texto quedándose solamente las oraciones que contienen números, promedios y portajes.

En segundo lugar, hay que tener en cuenta que la notación no es la misma en todas las fuentes de información. En algunos sitios web separaban los decimales por un punto y las unidades de millar con una coma, en cambio, en otros dominios es al revés o incluso no hay signos de puntuación que separen las unidades de millar. Estas diferencias de notación hacen que, por ejemplo, los datos numéricos se corten, ya que el punto lo identifica como final de una oración. Por lo tanto, el primer paso era cambiar todos los sistemas de separación y dejarlos en uno solo. Se consideró que lo más adecuado era no tener separación en las unidades de millar y poner una coma para separar los decimales.

Tras haber pasado todos los números al mismo formato y haber cogido las oraciones con números, se prosigue eliminando las fechas, ya que realmente no nos interesan puesto que la información relevante son datos de promedios, distribuciones, índices y porcentajes, entonces se eliminan las oraciones numéricas que solamente contienen fechas.

Una vez filtrado el texto, se reduce la información inicial, en ocasiones de medio megabyte a 1 ó 2 kilobytes, con únicamente la información relevante.

Por último, en el procesado de texto, se crea un documento a parte que filtra toda la información y recoge los datos de ocupación, pero en este caso relacionado con los bancos más importantes de España. De esta manera se puede observar que los bancos son los más afectados por la ocupación ilegal [12].

Cabe destacar que, para la extracción de texto, no solo se ha empleado la técnica del *web scraping*. Algunos dominios web tienen protección contra esta técnica, y por tanto la información se tiene que obtener de forma manual, y otros datos ya estaban regidos en *datasets* publicados por el Ministerio Del Interior [4], por el Ayuntamiento de Madrid [13] y por el Instituto Nacional de Estadística (INE) [14, 15], por lo que no ha sido necesaria ninguna técnica de extracción de información.

4.2.3 Creación de datos sintéticos

Ante la falta de datos reales, se tomó la decisión de crear datos sintéticos. Los datos sintéticos son datos que, como su propio nombre indica, no son reales pero que son creados a partir de otros datos reales y que tienen las mismas propiedades estadísticas que los datos reales [16].

Para la creación de estos datos sintéticos se han utilizado los datos recogidos de diferentes *datasets* de Madrid y la información obtenida mediante el *web scraping*.

Para la generación de los datos sintéticos, se siguen una serie de pasos:

En primer lugar, se genera un distrito de forma aleatoria, pero la probabilidad de un distrito que tiene más viviendas vacías es mayor, ya que, para la selección del distrito, el número aleatorio que se genera entre 0 y 1 se compara con la frecuencia absoluta de viviendas vacías

del distrito. De esta forma va a ser mayor la probabilidad de que se salga un distrito con más viviendas vacías.

Una vez generado el distrito, se accede a un data set, de creación propia, en el que están almacenados los distritos con los códigos postales que hay en cada uno. Cada distrito tiene uno o varios códigos postales, entonces se accede aleatoriamente a uno de los códigos postales del distrito generado en el paso anterior, y con el código postal seleccionado se accede a otro data set, también de creación propia, en el que están todas las calles de Madrid capital con su correspondiente código postal. De esta manera, entre todas las calles con el mismo código postal del distrito generado de forma aleatoria, se selecciona, nuevamente de manera aleatoria una dirección para el dato.

A continuación, se genera otro número aleatorio entre 0 y 1, y se compara con el promedio de casas ocupadas entre viviendas vacías del distrito. Si el número generado es menor o igual, se le asigna que está ocupada.

Un hecho muy importante, que se ha tenido en cuenta, es que el entorno es un factor muy influyente en lo referido a la ocupación. Con el entorno se quiere hacer referencia a que cuando un bloque tiene un alto porcentaje de viviendas ocupadas, es muy probable que las viviendas restantes que no están ocupadas acaben siéndolo. Este hecho se ha reflejado en los datos sintéticos de la siguiente forma:

- Al asignarse que una dirección ha sido ocupada, la calle en la que se encuentra esa vivienda se almacena en una lista.
- Esta lista almacena, por tanto, todas las calles de las viviendas que han sido ocupadas para poder compararla con las direcciones postales generadas posteriores.
- En caso de que la calle de una dirección generada se encuentre en esta lista, la probabilidad de que la vivienda con ese domicilio acabe siendo ocupada se habrá incrementado.
- De esta forma se está teniendo en cuenta el entorno.

En cambio, si el número generado entre 0 y 1 es mayor que el promedio de casas ocupadas entre viviendas vacías del distrito, y que en caso de haber salido una calle que está en la lista de calles con viviendas ocupadas, sigue saliendo mayor, a la vivienda se le asignará que no está ocupada.

Una vez a una dirección se le ha asignado si ha sido ocupada o no, se generan dos datos nuevos: si la vivienda tenía o tiene alarma y el propietario de la vivienda.

Por un lado, las viviendas con alarma es más difícil que sean ocupadas, ya que, aparte de ser un elemento disuasorio, en caso de que alguien entre en una casa, por ley se tiene que actuar en un plazo de 48h para desalojar al individuo, si alguien entra y salta la alarma, la policía acudirá al lugar del delito y detendrá a los presuntos usurpadores [17]. Por lo tanto, en la creación de los datos sintéticos, este factor se ha tenido en cuenta de la siguiente forma:

- Si la vivienda está marcada como ocupada, la probabilidad de que se le asigne que tenía o tiene alarma es muy baja.

- En cambio, si la vivienda está indicada como que no está ocupada, la probabilidad de que le asigne que tiene alarma es mayor, pero no muy alta, ya que la mayoría de viviendas vacías no tienen alarma.

Por otro lado, el otro dato que se genera es el propietario. La mayoría de viviendas ocupadas son propiedades del banco o públicas [18, 19]. El problema con estos datos es que las viviendas públicas que están ocupadas no suelen contabilizarse como ocupadas. Esto es debido a que el objetivo de la construcción de viviendas públicas es dar un hogar a familias con muy pocos o sin ningún recurso, y este perfil de personas son las que tienden a ocupar viviendas. Por lo tanto, ellos ocupan la vivienda de forma ilegal pero técnicamente era para ellos, por lo que no se contabiliza como ocupada. Para tener en cuenta estos factores, se ha seguido el siguiente criterio:

- Si la vivienda ha sido marcada como ocupada, tendrá una probabilidad muy alta de ser del banco, una probabilidad un poco menor de ser del ayuntamiento y una probabilidad muy baja de ser de un particular.
- En cambio, si ha sido marcada como no ocupada, tendrá una probabilidad muy alta de ser de un particular, una probabilidad bastante baja de ser del ayuntamiento y una probabilidad baja de ser de un banco.

Una vez establecidos todos los criterios para generar los datos sintéticos, se crean los datos con las siguientes variables:

- Calle del domicilio.
- Localización (distrito de Madrid en el que está situada la vivienda).
- Número de habitantes del distrito.
- El paro registrado en el distrito.
- El número de actuaciones policiales en el último año.
- Las detenciones por habitante.
- El número total de viviendas que hay en el distrito.
- El número de viviendas vacías.
- La cantidad de viviendas ocupadas del distrito.
- La renta media bruta anual por persona.
- Promedio de extranjeros.
- Número de extranjeros.

Se han generado 100.000 datos sintéticos para entrenar el modelo. Se decidió generar ese número porque se consideró suficiente y, además es un número cercano a la realidad, ya que en Madrid hay aproximadamente 130.000 viviendas vacías. Cabe destacar que como la creación de cada uno de los datos tiene muchos procesos (generación de números aleatorios, búsquedas de datos en *datasets* de gran dimensión, recorrido de listas, comparación de elementos y diversos cálculos), el tiempo que se tardaba en crear los 100.000 datos era de más de cinco horas. Aunque puede parecer que es un proceso que se ejecuta una sola vez y que por tanto tampoco es mucho tiempo, la realidad es que se tuvo que ejecutar en numerosas ocasiones para ir ajustando las probabilidades, proporciones y los diferentes cálculos hasta dar con los valores que creaban datos con sentido y lo más reales posibles.

4.2.4 Creación del modelo de predicción

Una vez creados los datos sintéticos, se procedió a desarrollar el modelo de predicción.

Para ello, se ha utilizado TensorFlow. TensorFlow es una plataforma de código abierto que sirve para el aprendizaje automático. Cuenta con una gran variedad de herramientas, librerías y recursos, todos ellos para ayudar a los desarrolladores a compilar e implementar aplicaciones con tecnología de aprendizaje automático [20].

Se ha decidido decantarse por un modelo de clasificación binaria, ya que la salida es 0 si no va a ser ocupada y 1 si va a ser ocupada, basado en *Deep Learning*.

La regresión logística, es un algoritmo utilizado para clasificar cuando la variable de salida, en nuestro caso si la vivienda va a ser ocupada, es binaria. El problema que presenta la regresión logística es que no tiene margen de maniobrabilidad, y es por eso que la elección final ha sido el *Deep Learning*. El *Deep Learning*, traducido como aprendizaje profundo, es un algoritmo automático, compuesto por capas de redes neuronales entrelazadas, que simula el funcionamiento de las redes neuronales humanas y es capaz de aprender y encontrar patrones después de un periodo previo de entrenamiento [21, 22].

Se ha optado por el *Deep Learning* porque después de haber realizado un número exhaustivo de *tests*, este algoritmo tiene la ventaja de que se puede aumentar su precisión cambiando sus distintos parámetros, como el número de capas de neuronas, el número de neuronas, número de *epochs* (las *epochs* son cada uno de los ciclos de la red neuronal en el proceso de entrenamiento), etc.

Antes de entrenar el modelo con los datos sintéticos, se realizó un preprocesado de los mismos para que tuviesen la estructura y formato adecuado para el entrenamiento. Se han realizado diferentes pasos para preprocesar los datos:

- Lo primero, se ha realizado una técnica conocida como *upsampling*. El *upsampling* es un método para manejar el desequilibrio de los datos que se utilizan [23]. La cuestión es que la mayoría de los datos representan un comportamiento, en este caso es que no van a ser ocupadas, y la minoría indica otro, que van a ocuparse. Entonces es necesario balancear los datos para que haya más o menos la misma cantidad de cada uno en el *dataset*. En este caso, lo que se ha hecho ha sido aumentar la cantidad de datos de viviendas que van a ser ocupadas (aproximadamente el 3%) para obtener 200.000 datos en los que la cantidad de ocupadas y no ocupadas esté balanceada, una vez ampliados se mezclan los 200.000 datos.
- Después de realizar el *upsampling*, se separan las variables en numéricas y categóricas, ya que se tratan de forma diferente.
- Las variables numéricas se normalizan por columnas para que comprendan valores entre 0 y 1.
- Para el procesado de las variables categóricas, se ha empleado la función `get_dummies()`. Esta función crea una columna nueva por cada valor de una variable concreta y le asigna 0 ó 1 si la cumple, excepto si esa variable solo tiene dos valores, en ese caso asigna 0 ó 1 dentro de la misma variable [24]. Por ejemplo: la variable alarma

tiene dos posibles valores, sí o no, entonces esta función asigna 1 a sí y 0 a no; en cambio, la variable distrito tiene 21 posibles valores, en este caso crea una columna por cada distrito y le asigna un 1 al distrito al que pertenece y cero al resto.

Con este proceso fue con el que se vio que Google Colab no tiene la suficiente potencia, ya que para la variable 'Calle', como tiene muchos valores diferentes, crea muchas columnas (más de 8200) y no puede procesar toda esa información. Por ello, se creó una máquina virtual con Google Cloud Platform (GCP) y desplegar así el modelo en la nube.

La red neuronal creada, tiene una serie de características concretas. Estas características han sido resultado de prueba y error y comprobar con qué configuración se han obtenido los mejores resultados.

- En primer lugar, la red neuronal está formada por tres capas de neuronas:
 - La primera capa está compuesta por 10 neuronas, y se activa con la función *ReLU (Rectificador Lineal Unitario)*. Se ha escogido la función *ReLU* como función de activación porque por norma general es la que más se utiliza en *Deep Learning*, ya que debido a su simpleza es con la que mejores resultados se obtienen [25]. Su función viene dada por:

$$f(x) = \max(0, x)$$

Figura 3. Fórmula de la función de activación ReLU [25]

- La segunda capa, conocida como capa oculta, se ha establecido siguiendo la siguiente fórmula:

$$h = \sqrt{i \cdot o}$$

Figura 4. Fórmula para establecer el número de neuronas de la capa oculta [26]

Donde:

h = número de neuronas de la capa oculta

i = número de neuronas de la capa anterior (*input*)

o = número de neuronas de la siguiente capa (*output*)

Esta capa oculta se ha activado también con la función de activación ReLU, siguiendo el mismo criterio de la capa anterior.

- La tercera, y última capa, como el modelo es de clasificación binaria, lo que se busca es que la salida sea 1 si va a ser ocupada, ó 0 si no lo va a ser. Entonces, esta última capa cuenta con tan solo una neurona. En este caso, esta capa se activa con la función de activación *Sigmoid*. Esta función de activación transforma los valores a una escala de 0 a 1 [25, 27]. Su función es la siguiente:

$$f(x) = \frac{1}{1 + e^{-x}}$$

Figura 5. Fórmula de la función de activación Sigmoid [27]

4.2.5 Creación de Máquina Virtual en Google Cloud Platform

Como se ha comentado en el apartado anterior, 4.2.5 Creación de Máquina Virtual en Google Cloud Platform, en lo relativo al preprocesado de los datos, Google Colab no cuenta con la potencia necesaria para procesar los datos de entrenamiento del modelo.

Para solucionarlo, se creó una máquina virtual en la plataforma Google Cloud Platform de Google. La máquina virtual reunía las siguientes características y especificaciones técnicas [28]:

- 8 CPU's virtuales
- 32 Gb de memoria RAM
- Zona: europe-west6-a
- Imagen: Google, Deep Learning Image: TensorFlow Enterprise 2.5, m73 CUDA 110, A debian-10 Linux based image with TensorFlow Enterprise 2.5 (With CUDA 110 and Intel (TM) MKL-DNN, Intel (TM) MKL) plus Intel (TM) optimized NumPy, SciPy, and scikit-learn.

Una vez creada la máquina virtual con la potencia suficiente para el entrenamiento del modelo, se realizó una conexión a dicha máquina mediante un protocolo SSH desde la consola de comandos del ordenador en el que se desarrolló el proyecto. Los comandos utilizados para realizar esta conexión segura fueron los siguientes [29]:

```
$ gcloud init
```

Una vez realizada la configuración de uso, siguiendo los pasos que se indican, se establece la conexión con el comando:

```
$ gcloud compute ssh instance-1 --zone europe-west6-a -- -L 8090:localhost:8090
```

Después de realizar la conexión con la máquina virtual, hay que dar permisos a Google Colab para poder ejecutar el notebook en el puerto indicado, en este caso el 8090. Esto se realiza mediante el siguiente comando:

```
$ jupyter notebook --NotebookApp.allow_origin='https://colab.research.google.com' --  
port=8090 --no-browser
```

El último comando devuelve un link que se copia y pega en el entorno de ejecución del notebook donde se desarrolla el modelo, y finalmente se establece la conexión con la máquina virtual.

4.2.6 Entrenamiento del modelo

Por último, una vez establecida la conexión con la máquina virtual y tener la potencia suficiente para entrenar el modelo, se procedió a su entrenamiento.

Para ello, se los datos de entrenamiento se dividieron en una proporción 30 - 70. El 30% en test y el 70% para el training.

En lo relativo al número de *epochs*, que como se ha explicado antes son los ciclos de la red neuronal en el entrenamiento, se han establecido 15 *epochs*.

Para definir estos parámetros, se ha seguido el mismo criterio que con la red neuronal, ha sido un proceso de prueba y error para ver cuáles eran los parámetros con los que mejores resultados se obtenían.

4.2.7 Evaluación del modelo

Finalmente, una vez creado y entrenado el modelo, se procedió a realizar una evaluación de cómo había ido evolucionando por cada *epoch* el modelo. Para ello se utilizaron las medidas de evaluación *accuracy* (precisión) y *loss* (pérdida).

Esta evolución está representada en dos gráficas indicadas y explicadas en el apartado 4.6.1 Medidas de evaluación.

Es importante añadir que, para la evaluación del modelo se han creado una serie de datos sintéticos, pero sin la variable 'Ocupada'. Estos datos se han utilizado para pasarles el modelo y comprobar a cuáles de los domicilios generados les indica que van a ser ocupadas y cuáles no.

Los resultados obtenidos han sido muy satisfactorios y fieles a la realidad, pero de nuevo, toda la evaluación se detalla en el apartado 4.6 Resultados del proyecto.

4.2.8 Desarrollo de aplicación web

Para poder utilizar el modelo de forma cómoda sin tener que acceder al código del modelo para poder introducir una dirección de una vivienda y sus características, se tomó la decisión de desarrollar una aplicación web muy simple. Esta interfaz permite a cualquier usuario introducir cuatro variables:

- La calle del domicilio
- El distrito de Madrid en el que se encuentra
- El propietario (Particular, Banco o Ayuntamiento)
- Si tiene alarma

Únicamente son necesarias estas cuatro variables y la aplicación de forma interna accede a los diferentes conjuntos de datos que se han creado para el desarrollo de los datos sintéticos y completa la información faltante.

En el apartado 4.6 Resultados del proyecto, se muestra en detalle la aplicación web.

4.3 Recursos requeridos

- Ordenador Lenovo ideapad 720
- Visual Studio Code
- Google Colab
- Tensorflow
- Keras
- Google Drive
- Microsoft Excel
- Microsoft Word
- GitHub
- Documentación (Google, periódicos, noticias, Ministerio de Interior, INE (Instituto Nacional de Estadística, Datos del Ayuntamiento de Madrid, etc.)
- Python 3.7 (Librerías)
- Google Cloud Platform (GCP)
- Monday (Herramienta de gestión de proyectos)
- Flask
 - Flask es un *framework* de programación escrito en Python que permite crear aplicaciones web [30].
- Librerías de Python utilizadas:
 - BeautifulSoup: librería utilizada para poder realizar *web scraping*.
 - Django.utils.encoding: librería utilizada para decodificar caracteres extraños.
 - Io: librería que provee diferentes facilidades de Python para manejar texto.
 - Matplotlib.pyplot: librería utilizada para hacer gráficos.
 - Numpy: librería utilizada para cálculo numérico.

- Os: librería utilizada para acceder a funcionalidades dependientes del sistema operativo.
- Pandas: librería utilizada para poder trabajar con *dataframes*.
- Re: librería utilizada para realizar operaciones de expresiones regulares.
- Requests: librería utilizada para poder realizar *web scraping*.
- Sklearn: librería utilizada para el aprendizaje automático del modelo.
- Uuid: librería utilizada para poder generar números hash.

4.4 Presupuesto

Tipo de coste	Valor	Comentarios
Horas de trabajo en el proyecto	650 horas	El trabajo ha sido realizado íntegramente por Francisco Javier Díaz García
Equipo técnico utilizado	800 €	Ordenador Lenovo ideapad 720
Software utilizado	40 €	Los 40€ corresponden a gasto en uso de herramientas de Google Cloud Platform Las demás aplicaciones, IDEs y herramientas que se han utilizado son gratis o sus versiones gratuitas
Estudios e informes	0 €	Toda la documentación utilizada ha sido gratuita.
Materiales empleados	0 €	No se ha utilizado ningún material a parte del ordenador.

Tabla 2. Tabla de presupuesto

4.5 Viabilidad

4.5.1 Coste del proyecto

Como se puede apreciar en la tabla de presupuesto, este proyecto no ha tenido costes económicos.

El precio del equipo técnico utilizado no ha sido realmente un coste, ya que lo que se ha utilizado ha sido el ordenador personal, y no se ha específicamente comprado para la realización del proyecto. En cambio, el coste que ha tenido el uso de la plataforma GCP de Google, sí que

ha sido coste real pero técnicamente no ha costado dinero, ya que los 40€ han formado parte del conjunto de créditos gratuitos que otorga Google cuando te das de alta en la plataforma por primera vez.

Todas las demás herramientas que se han utilizado han sido *open source*, que significa código abierto, y por tanto gratuitas. Otras plataformas o aplicaciones sí que tienen versión de pago, pero se ha utilizado la parte gratuita, como es el caso de Google Colab. Esta plataforma tiene opción de usar la versión Pro, la cual es mucho más potente, da prioridad a la hora de utilizar las GPU's y TPU's, permite ejecutar los notebooks durante más tiempo y ofrece más cantidad de memoria RAM.

4.5.2 Coste de cara al futuro

Para estudiar la viabilidad del modelo de cara al futuro, se debe tener en cuenta que el proyecto tendría un objetivo de generar valor económico y por tanto habría que realizar una evaluación del mismo para determinar si tiene sentido operativa y económicamente.

En primer lugar, hay que calcular el coste que supondría tener un modelo operativo 24 horas al día en la nube.

Por un lado, el coste de desarrollo sería igual al que ha costado en la realización de este proyecto. Sin embargo, sí que es cierto que si se hubiese contando con algún tipo de presupuesto, seguramente se podrían haber utilizado herramientas de pago que fuesen más sencillas de utilizar, o se podía haber contratado a un equipo de soporte y desarrollo para que contribuyese en el proceso.

Por otro lado, si lo que se pretende es obtener un beneficio buscando una aplicación en el mundo empresarial, hay costes en infraestructura. Sería necesario tener el modelo desplegado en Google Cloud Platform o en Amazon Web Services, lo que conlleva un coste, porque estas dos plataformas tienen créditos de prueba gratuita, pero son insuficientes porque si se realizan muchas peticiones, el crédito se gasta rápidamente y además las pruebas gratuitas tienen un tiempo límite. En el caso de GCP son 300\$ \approx 2553,92€ durante 90 días, y en AWS son 12 meses con los recursos limitados.

A continuación, se expone el cálculo estimado del coste en Google Cloud Platform.

Métrica	Valor bruto	Nivel gratuito	Valor neto	Precio por unidad	Precio total
Invocaciones	10.000.000	2.000.000	8.000.000	0,0000004 USD	3,20 USD
GB por segundo	375.000	400.000	<0	0,0000025 USD	0,00 USD
GHz por segundo	600.000	200.000	400.000	0,0000100 USD	4,00 USD
Redes	0	5	0	0,12 USD	0,00 USD
Total/mes					7,20 USD

Tabla 3. Tabla de estimación de coste de Cloud Functions

Como se puede ver en la *Tabla 3* el coste sería de 7,2\$ mensuales, aproximadamente 6,9€. Esta estimación, es utilizando Cloud Functions de GCP. El problema de estimaciones de las herramientas de Cloud es que varían mucho en función al uso [31].

También sería necesario utilizar Cloud Storage [32]. Su tarifa de precios es:

Standard Storage	Nearline Storage	Coldline Storage	Archive Storage
(por GB al mes)	(por GB al mes)	(por GB al mes)	(por GB al mes)
\$0.023	\$0.013	\$0.007	\$0.0025

Tabla 4. Tabla de precios de Cloud Storage

Realizar un cálculo estimado de precio en esta plataforma, sería:

Categoría de precio	Cálculo	Coste
Almacenamiento de datos	50 GB de Standard Storage * 0,020 USD por GB	1,00 USD
Red	1 GB de salida * 0,12 USD por GB	0,12 USD
Operaciones	10.000 operaciones de clase A * 0,05 USD por cada 10.000 operaciones	0,05 USD
Operaciones	50.000 operaciones de clase B * 0,004 USD por cada 10.000 operaciones	0,02 USD
Total		1,19 USD

Tabla 5. Tabla de estimación de costes de Cloud Storage

El precio de almacenamiento con esas características y esos accesos sería de 1,19\$ mensuales, lo que son aproximadamente 1,01€.

Otro servicio que se utilizaría es Cloud Engine. Para esta plataforma, existen gran cantidad de tipos de máquinas diferentes. El precio varía en función qué máquina se utilice, dónde está situada, la demanda, el tiempo y la potencia [33].

Google Cloud Platform sería la plataforma escogida, ya que es la que se ha utilizado para el desarrollo del proyecto y, por tanto, en la que se tienen experiencia y conocimiento en su uso.

Entonces, si nos basamos en el modelo desarrollado y tenemos en cuenta los gastos que supondría el mantenimiento del proyecto, a priori no parece suponer un gran gasto en infraestructura.

Este modelo, es una herramienta muy útil para empresas con un gran número de bienes inmuebles, para empresas de alarmas, constructoras o incluso de uso público para el Ayuntamiento de Madrid. Todos ellos pueden obtener un gran beneficio del modelo, ya que al banco le ahorrará costes en optimización a la hora de establecer medidas de seguridad en sus propiedades, a las empresas de alarmas les servirá para establecer tarifas de sus servicios, a las constructoras para evaluar las zonas donde se tiene intención de construir y al Ayuntamiento de Madrid para tomar medidas y establecer políticas de actuación contra la ocupación.

El beneficio es muy elevado en relación al coste que se calcula que supone.

4.6 Resultados del proyecto

4.6.1 Medidas de evaluación

Durante el proceso de entrenamiento del modelo de predicción automática, se han tomado dos medidas para estudiar su calidad:

- *Accuracy* (precisión): La precisión es una métrica utilizada para evaluar la proporción de acierto del modelo utilizando para ello el *test* [34]. En los modelos de clasificación binaria la fórmula por la que se rige esta métrica es:

$$Accuracy = \frac{VP + VN}{VP + VN + FP + FN}$$

Figura 6. Fórmula de cálculo de precisión [34]

Donde:

VP: Verdadero positivo

VN: Verdadero negativo

FP: Falso positivo

FN: Falso negativo

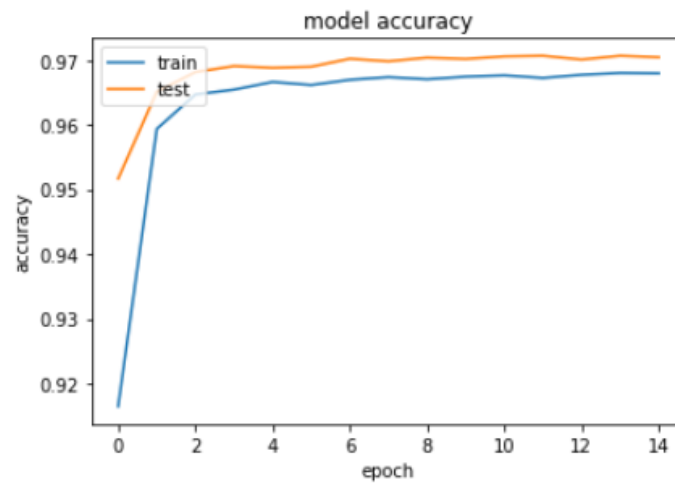


Figura 7. Gráfica de evaluación de precisión del modelo

Como se puede observar en la *Figura 7*, la precisión del modelo es muy buena. Este valor, con la última *epoch*, se queda cercano al 0.98 lo que quiere decir que el modelo tiene un 98% de precisión a la hora de predecir un resultado.

- *Loss* (pérdida): Este valor indica la diferencia entre el valor predicho por el modelo y el valor de la salida deseada.

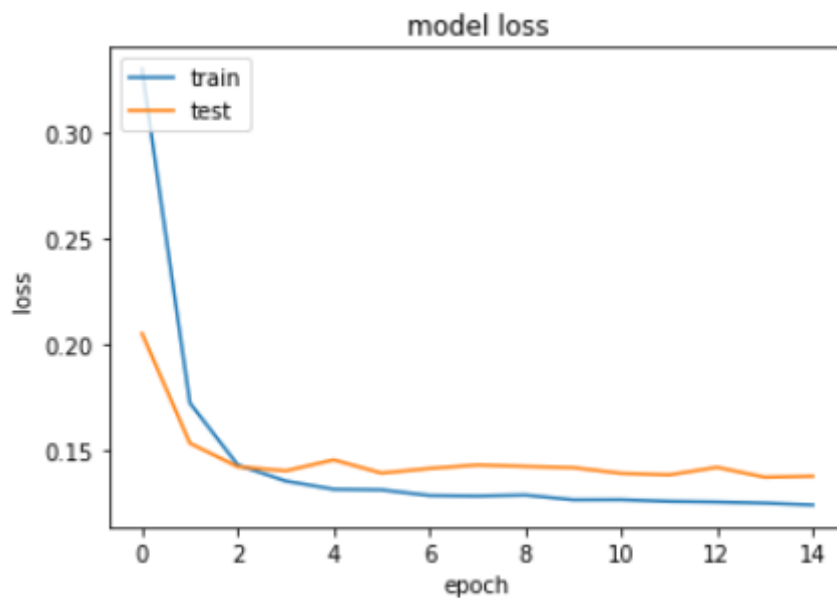


Figura 8. Gráfica de evaluación de la pérdida del modelo

Como se puede observar en la *Figura 8*, la pérdida del modelo es baja. Comienza por

encima de 0,2 y termina con una pérdida por debajo de 0,15.

Viendo los resultados obtenidos con las métricas *accuracy* y *loss*, se puede deducir que no hay *overfitting* (sobreentrenamiento). El *overfitting* se suele producir cuando el entrenamiento tiene muchas *epochs* y el modelo memoriza los datos de entrenamiento. De esta forma, el modelo solo sabrá predecir correctamente los datos con los que entrena y no datos nuevos. Un indicativo de sobreentrenamiento sería que la precisión se quede en 1 y que la pérdida presentase una tendencia ascendente. Otro indicativo sería cuando en la precisión, la gráfica del *train* queda por encima del *test*.

4.6.2 Pruebas

A continuación, se expone una pequeña muestra de la tabla de predicciones resultantes del modelo:

Calle	Localización	Ocupada
C/ La Mezquita	Villaverde	0
C/ Albatros	Carabanchel	1
C/ Camino Antequina	Moncloa-Aravaca	0
C/ Cabo Machichaco	Puente de Vallecas	1
C/ Los Barros	Puente de Vallecas	0
C/ Paz	Centro	0
C/ Plaza San Ildefonso	Hortaleza	1
C/ San Antonio	Tetuán	0
C/ Miguel Arredondo	Arganzuela	0
C/ Los Nogales	Arganzuela	1
C/ Covalada	Hortaleza	1
C/ Camino Viejo De Villaverde	Villaverde	1
C/ Silvina Ocampo	Villa de Vallecas	1
C/ Condor	Carabanchel	0
C/ Almodovar	Latina	0
C/ Plaza Carros	Arganzuela	0
C/ Carcastillo	Carabanchel	1
C/ Jose Calvo	Tetuán	1
C/ E (El Salobral)	Villaverde	1
C/ Maudes	Chamberí	0

Tabla 6. Tabla de resultados de modelo definitivo

Como se puede observar en la *Tabla 6*, esta muestra de la tabla de resultados de una serie de viviendas a las que se les ha pasado, el modelo predice que viviendas situadas en los distritos de Carabanchel, Puente de Vallecas, Villaverde, Tetuán y Hortaleza, tienen una alta probabilidad de ser ocupados. Estos resultados son muy satisfactorios, ya que principalmente los distritos de Carabanchel, Puente de Vallecas, Villaverde y Tetuán, son distritos con alta tasa de ocupación y que reúnen las características principales de ser lugares propensos a ocupación. Además, se podría pensar que el modelo ha identificado estos distritos como alto riesgo de ocupación, pero como se puede observar hay domicilios en esos distritos que no cataloga de propensos a ocupación. Esto se debe a que no tiene en cuenta únicamente el distrito, sino que hay otra serie de factores influyentes, que probablemente en estos distritos tengan mayor probabilidad de darse.

En el apartado 5.1 Resultados, se mostrarán resultados obtenidos con modelos anteriores.

4.6.3 Aplicación web

Para el uso del modelo, como se explica en el apartado 4.2 Descripción de la solución, metodologías y herramientas empleadas, metodologías y herramientas empleadas, se ha creado una aplicación web funcional.

A la hora de su desarrollo, se ha decidido priorizar la funcionalidad al diseño. El motivo de ello ha sido que se quiere una aplicación útil y muy simple en la que se introducen los valores de las variables y se predice en función a ellos.

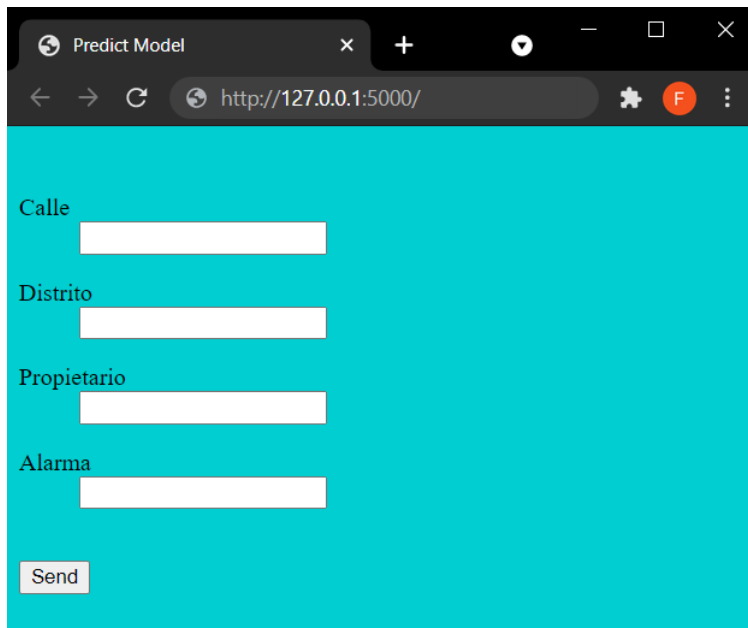
A screenshot of a web browser window titled 'Predict Model'. The address bar shows 'http://127.0.0.1:5000/'. The main content area has a light blue background and contains four white input fields stacked vertically, labeled 'Calle', 'Distrito', 'Propietario', and 'Alarma'. Below these fields is a white button with the text 'Send'.

Figura 9. Interfaz de aplicación web

Si el modelo predice que esa vivienda va a ser ocupada, muestra lo siguiente:

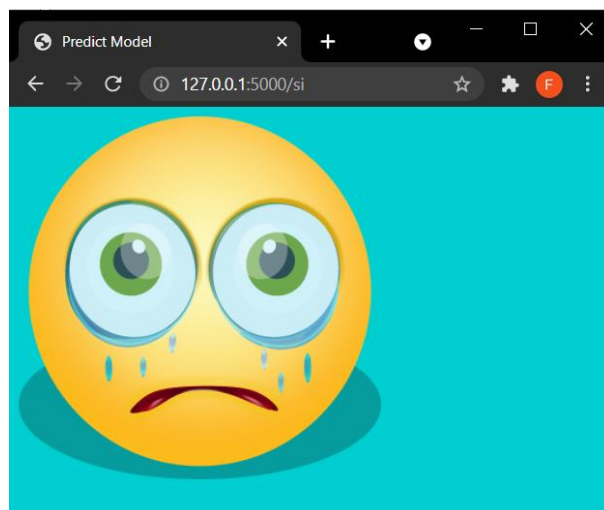


Figura 10. Interfaz de aplicación web - Pantalla de ocupado [35]

En caso contrario, si predice que no va a ser ocupada devuelve la siguiente imagen:

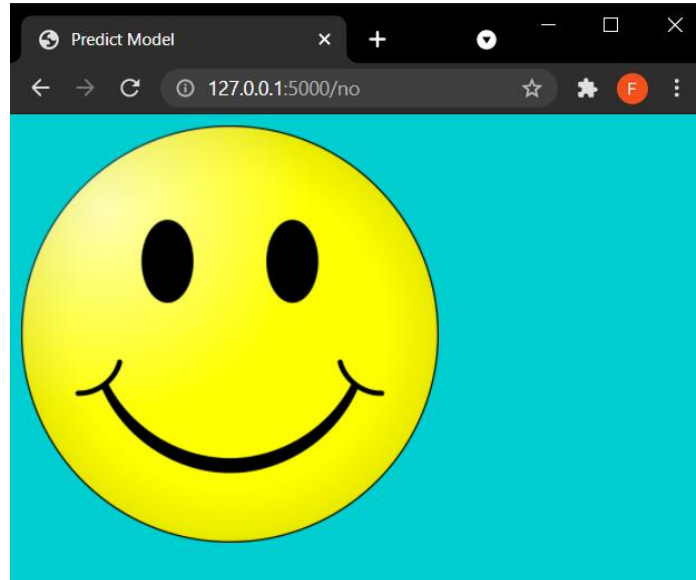


Figura 11. Interfaz de aplicación web - Pantalla de no ocupado [36]

Cabe destacar, que las imágenes utilizadas no tienen derechos de autor.

Capítulo 5. DISCUSIÓN

5.1 Resultados

En este apartado, se van a comparar los resultados obtenidos del modelo definitivo con los obtenidos en otros modelos.

A continuación, se muestra el gráfico que muestra la precisión del modelo anterior, el modelo que no se había desarrollado en la nube, ni tenía los datos balanceados.

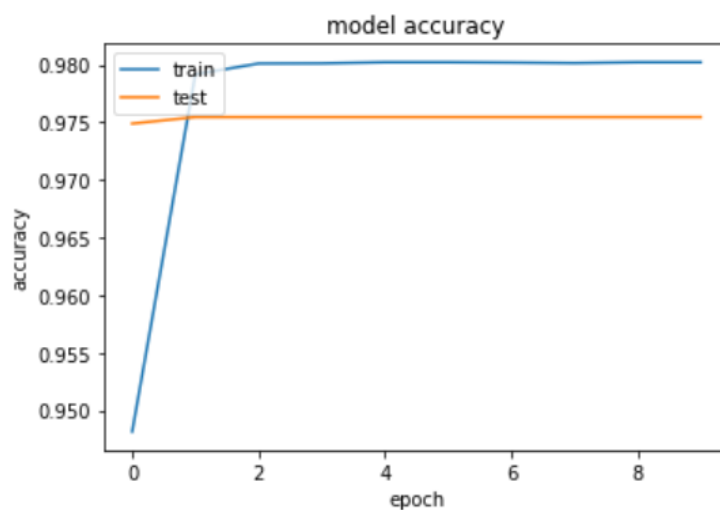


Figura 12. Gráfica de evaluación de precisión de modelo anterior

Si este resultado de la *Figura 7*, se compara con el de *Figura 12* en el que no se realizó la técnica del *upsampling*, ni se entrenó con la variable 'Calle' de las viviendas, se puede observar que, aunque la precisión es también muy alta, esta no evoluciona. Esto se debe a que la mayoría de los datos tienen como valor en la variable 'Ocupación' un 0. De esta forma el modelo, estadísticamente va a acertar un 97% de las veces, que es la proporción que había en los datos de viviendas sin ocupar.

Seguidamente, se exponen muestras de resultados de modelos previos. Son resultados muy imprecisos y poco reales, por lo que estos modelos fueron descartados.

Calle	Localización	Ocupada
C/ San Casimiro	Usera	0
C/ Juan De Austria	Chamberí	0
C/ Fantasia	Puente de Vallecas	0
C/ Nuestra Señora De Begoña (Villaverde)	Villaverde	0
C/ Guindos	Tetuán	0
C/ Recoletos	Salamanca	0
C/ Jaen	Tetuán	0
C/ Conde De Eleta	Carabanchel	0
C/ Tamarindo	Arganzuela	0
C/ Riocabado	Latina	0
C/ Historias De La Radio	Puente de Vallecas	0
C/ Jose Castan Tobeñas	Barajas	0
C/ Francisco De Diego	Moncloa-Aravaca	0
C/ Francisco Medrano	Tetuán	0
C/ Luis Ruiz	Ciudad Lineal	0
C/ Fuente De Piedra	Puente de Vallecas	0
C/ Plaza Carmona	Latina	0
C/ Plaza La Orotava	Ciudad Lineal	0
C/ Cartago	San Blas-Canillejas	0
C/ Mateo Garcia	Ciudad Lineal	0

Tabla 7. Tabla de resultados de modelo descartado I

Como se muestra en la Tabla 7, en este caso, el modelo predijo que ninguna vivienda iba a ser ocupada.

Calle	Localización	Ocupada
C/ Marques Del Vasto	Chamberí	1
C/ Paseo Castellana	Chamartín	1
C/ Alberto Marcos	Villa de Vallecas	1
C/ Tiempos Modernos	Puente de Vallecas	1
C/ Del Cuarto	Carabanchel	1
C/ Jose De Blas	San Blas-Canillejas	1
C/ Prado Alegre	Latina	1
C/ Juana Urosa	Carabanchel	1
C/ Matadero De El Pardo	Fuencarral-El Pardo	1
C/ Villanueva	Salamanca	1
C/ Pasaje Lucero	Latina	1
C/ Hermanos Garcia	Puente de Vallecas	1
C/ Monte Torozo	Moratalaz	1
C/ Najarra	Puente de Vallecas	1
C/ Espartinas	Salamanca	1
C/ Plaza Las Sufragistas	Villaverde	1
C/ Violonchelos	Carabanchel	1
C/ Hileras	Centro	1
C/ San Clodoaldo	Ciudad Lineal	1
C/ Juan De La Hoz	Vicálvaro	1

Tabla 8. Tabla de resultados de modelo descartado II

Los resultados mostrados en la *Tabla 8*, corresponden a un modelo en el que se predecía que la gran mayoría de viviendas iba a ser ocupada.

Por último, para finalizar el apartado de resultados, se quiere hacer una mención al modelo descartado que más cerca estuvo de ser el seleccionado como el mejor. Nos referimos al modelo que estaba restringido a la Comunidad de Madrid, con todos sus municipios y distritos.

A continuación, se muestra un ejemplo de uno de los resultados obtenidos con ese modelo:

ID	Localización	Ocupada
91e334d9-48df-4aa2-9646-b6c5b118998c	Alcalá de Henares	0
6e496767-2dbf-4563-98c4-bbe347f5079e	Tres Cantos	0
669e8fbb-2b83-4de8-bc2e-64dca70b16b1	Cobeña	0
cf725ecb-714c-411a-b18b-d58611eeeb2d	Arganzuela	0
37fd617a-2c07-463a-8483-4dc9f740a33d	Arganzuela	0
318261ef-13f5-404f-aae4-97fdd830f649	San Martín de la Vega	0
db7d6b83-e706-4fad-80bb-d79b8aaa40f1	Perales de Tajuña	0
0c16a9eb-642e-450b-abef-42bc1055ba32	Arganda del Rey	0
f65f46b0-95e9-4e0a-8253-ad8a6033cd75	Centro	0
0115984e-a3f6-4b0f-b72e-ea6eb74bdd59	Ciudad Lineal	0
3b47f621-1b86-44b2-bf11-6a77593ab500	Salamanca	0
6c727387-6a0f-48bc-88b2-694855b3d72e	Fuenlabrada	1
f0960072-a1ce-45f6-8501-cc7c5e4c56db	Chamartín	0
3e7ae902-8738-4540-918e-048f3974b508	Puente de Vallecas	0
42d1728b-c17c-451c-a40d-000ba700a1b2	Hortaleza - Barajas	0
9ac8e636-2d28-477c-8009-8cb3e85c9a11	San Blas - Vicálvaro	0
c602221f-6aa8-4c0a-899d-574204e4f37a	Hortaleza - Barajas	0
901aef2-03b7-4b40-a47d-7236cddda5e7	Moratalaz	1
c03af77e-e2dc-4630-ba56-f12d558b7447	Centro	0
c33879df-475f-4f32-a480-4601269e9e75	Ciudad Lineal	0

Tabla 9. Tabla de resultados de modelo descartado de la Comunidad de Madrid

Con los resultados mostrados en la *Tabla 9*, se puede observar que solo aparecen dos viviendas ocupadas: una en el municipio de Fuenlabrada y otra en Moratalaz. Como ya se ha explicado, este modelo tenía mucho ruido, datos faltantes y aún no se tenían en cuenta variables muy importantes. Como consecuencia de ello, sus predicciones no eran acertadas. Por ejemplo, identificaba un municipio y todas las viviendas de ese municipio se clasificaban como ocupadas.

Este modelo fue el elegido durante un tiempo hasta que hubo un punto de inflexión y se decidió restringir a Madrid capital, y esta decisión consideramos que fue muy acertada ya que, se ha conseguido un modelo de mayor calidad.

5.2 Limitaciones de estudio

A la hora de desarrollar el proyecto, se han encontrado dos limitaciones importantes:

La primera limitación, es la que se comentó varias veces a lo largo de la memoria, y es limitación en la obtención de ciertos datos. Hay ciertas variables influyentes que se consideraban cruciales en el proceso de ocupación, que no existen registros o no estaban públicos. Por ejemplo, las viviendas de promoción pública, que son viviendas de construye el ayuntamiento para familias desfavorecidas y sin recursos económicos, por lo que hemos comprobado son viviendas que tienden a ocuparse. Aunque luego este factor se ha tenido en cuenta a la hora de desarrollar los datos sintéticos (explicado en el apartado 4.2 Descripción de la solución, metodologías y herramientas empleadas), no ha sido posible encontrar ningún registro de este tipo de viviendas.

La segunda limitación que entorpeció el desarrollo del proyecto fue de recursos a la hora de programar el modelo. El modelo se ha programado en Google Colab, el cual funciona muy bien y para crear modelos es lo más recomendado, ya que es gratuito. El problema es que al ser gratuito tiene ciertas limitaciones. Tiene una cantidad limitada de GPU que se puede utilizar diariamente, entonces mientras se puede utilizar todo se procesa muy deprisa, pero en cuanto se termina ya casi ni merece la pena seguir. Otra limitación de Google Colab es la cantidad de memoria RAM gratuita que se puede utilizar. Como se explica en el apartado 4.2 Descripción de la solución, metodologías y herramientas empleadas, se intentó añadir como variable, la calle en la que se encontraba la vivienda. Al utilizar esta variable para el modelo, el conjunto de datos sintéticos para el entrenamiento crecía de tal forma que no lo podía procesar. La alternativa era pagar una suscripción para poder tener más capacidad de memoria RAM. Aunque esta segunda limitación se solventó creando una máquina virtual en Google Cloud Platform, no deja de ser una limitación que entorpeció y quitó tiempo durante el proyecto.

5.3 Cambios en el proyecto

A lo largo del proyecto se han realizado algunos cambios a nivel de organización de trabajo, se han descartado algunas ideas y se han ido adaptando otras nuevas. Todos estos cambios no han sido muy grandes ni han tenido un gran impacto en el proyecto, ya que son cambios que se han producido de manera natural, es decir, eran resultados y conclusiones e ideas que surgían mientras se trabajaba.

Sin embargo, sí que ha habido dos cambios muy significativos. Uno de gran magnitud que supuso un giro en el proyecto y otro que sirvió para mejorar la calidad del modelo resultante del último cambio.

El primer cambio consistió en la creación de un nuevo modelo, lo que prácticamente implica, hacer un nuevo proyecto, aunque sin partir desde el punto cero.

El primer modelo que se creó era un modelo predictivo, pero en vez de centrarse en Madrid capital, se restringió a toda la Comunidad de Madrid.

Este primer modelo, al que se le podría considerar como la fase 0, tenía ciertos inconvenientes:

- En primer lugar, los datos de este modelo eran del año 2017, ya que los datos públicos más actualizados que reuniesen el dato de ocupación para toda la comunidad. El problema de que fuesen de hace tantos años era que las condiciones de la ocupación han cambiado de esos años hasta la actualidad, ya que es una fecha anterior a la pandemia y anterior a la inestabilidad política y social en Madrid producida por el Gobierno de coalición. Este Gobierno, supone un cambio importante, ya que aprobó leyes y se adoptaron políticas sociales a favor de deudores y arrendatarios [8].
- En segundo lugar, para un modelo que recoge datos de tantos municipios y distritos diferentes, exactamente 127, había muchos datos faltantes y algunos datos mal recogidos. Muchos de los datos faltantes o que estaban mal se intentó solucionar con cálculo de medias, medianas, modas y otra serie de estimaciones. Por lo tanto, si había que estimar valores de muchas variables para lugares concretos, para luego con esos datos crear un *dataset* para poder diseñar los datos sintéticos, el resultado fue un modelo muy impreciso con predicciones sin ningún tipo de sentido.

Debido a estos inconvenientes, este primer modelo para la Comunidad de Madrid era bastante malo e impreciso. En ocasiones predecía todo unos, es decir que todas las viviendas se iban a ocupar, o todo ceros, es decir ninguna. Otras veces se centraba en un municipio o en un distrito concreto y solo clasificaba como ocupadas las viviendas de ese lugar. En el apartado 5.1 Resultados.

Fue debido a esta serie de inconvenientes por lo que se optó por rechazar este modelo y se decidió crear uno nuevo restringido a Madrid Capital con datos más recientes, menos ruido y a la vez, más detallado. Al restringirlo a una región más reducida, permitía entrar más en detalle a ciertas variables interesantes que de la otra forma resultaba imposible.

El segundo cambio, no supuso una reestructuración del proyecto como el primero, que ya implicaba volver a buscar y obtener datos nuevos, crear datos sintéticos de una forma diferente, cambiar el preprocesado, programar un nuevo modelo, etc. Este segundo cambio fue una mejora del modelo restringido a Madrid Capital.

Cuando se desarrolló ese modelo centrado en Madrid Capital, había una variable considerada muy importante que era la calle de cada uno de los domicilios. El problema es que esa variable generaba tantas entradas que, a la hora de separarlas Google Colab se quedaba sin memoria RAM y por tanto el proceso se detenía. No era posible poder entrenar el modelo teniendo esta variable en cuenta. Además, no se había caído en que los datos de entrenamiento en ese modelo no estaban balanceados, había casi un 97% de datos que indicaban que no se iba a ocupar.

Estos dos problemas hacían que este modelo, aunque mejor que el de la Comunidad de Madrid, no fuera muy bueno. Se solucionaron de la siguiente forma:

- El balanceo de los datos se solucionó utilizando la técnica del *upsampling* (todo está recogido y explicado en detalle en el apartado 4.2 Descripción de la solución, metodologías y herramientas empleadas).
- Para poder utilizar la variable de la calle, estaba claro que lo que hacía falta era más potencia de cómputo, por lo que se solucionó creando una máquina virtual en Google Cloud Platform y desarrollando el modelo en la nube (todo está recogido y explicado en detalle en el apartado 4.2 Descripción de la solución, metodologías y herramientas empleadas).

Aunque puede parecer que estos dos cambios no son para tanto, supuso mucho trabajo darse cuenta del error del balanceo de los datos y poder conseguir una cuenta en GCP para poder crear una máquina virtual, a parte de la creación de la misma, ya que dio una gran cantidad de problemas. Y no solo eso, los resultados obtenidos con el último modelo son muy satisfactorios y fieles a la realidad, que era lo que se buscaba desde un primer momento.

Capítulo 6. CONCLUSIONES

6.1 Conclusiones del trabajo

El objetivo general del proyecto era el desarrollo de un modelo de predicción automática para estimar si una vivienda va a ser ocupada de forma ilegal.

Este objetivo se ha cumplido. Se ha conseguido desarrollar un modelo que presenta unos resultados muy satisfactorios, ya que es capaz de predecir la ocupación de una vivienda de una forma muy acertada. Por lo tanto, tal y como se estimó tras el estudio del contexto y del estado del arte, se puede concluir que efectivamente existe la tecnología necesaria para el desarrollo de modelos de este tipo. Modelos que en un primer momento puede parecer que tengan un objetivo inalcanzable, poco realista o incluso ideal, si se estudia el caso en profundidad y se trabaja de forma adecuada y constante se puede llegar a desarrollar.

Con esto, no se pretende dar a entender que se puede conseguir modelos que predigan cualquier cosa.

En el caso de este proyecto, la idea de la que parte puede parecer un tanto ideal y puede que hasta irrealizable, incluso después de estudiar en profundidad y documentarse. Y todo esto debido a un factor primordial: para crear un modelo de inteligencia artificial son imprescindibles los datos y en este caso no hay. Hay mucha información y muchas noticias sobre ocupación, pero no datos concretos, por lo que era necesario construirlos.

Actualmente hay grandes avances en el campo de la inteligencia artificial. Existen algoritmos realmente potentes con los que se pueden obtener resultados inimaginables. Volviendo a lo anterior, puede parecer imposible llegar a predecir la ocupación o cualquier otra cosa en la que influyan una gran cantidad de variables, pero en las acciones humanas y en la naturaleza hay patrones. Patrones matemáticos que, aunque una persona no sea capaz de identificar, un computador puede llegar a hacerlo.

Al fin y al cabo, así es como funcionamos los humanos. Desde que nacemos procesamos una gran cantidad de información y vamos aprendiendo con ello. En el campo del machine learning se denomina entrenamiento, en el nuestro se conoce como experiencia.

6.2 Conclusiones personales

A lo largo del desarrollo de este proyecto, se han llegado a una serie de conclusiones tanto de la experiencia personal como relacionadas con lo aprendido sobre el tema tratado.

En primer lugar, el tema del Trabajo de Fin de Grado llamó la atención y me resultó muy atractivo porque suponía un gran reto, y además la ocupación ilegal es un tema que me ha resultado siempre muy interesante y que he seguido de cerca. Aunque nunca he comprendido la situación en la que se encuentra a nivel legal, ni por qué puede ser un problema la contratación de empresas cuyo servicio es el desalojo de personas que se adueñan de hogares

que no son suyos. Es un tema sobre el que hay mucho desconocimiento y miedo al respecto. Por tanto, ese fue el primer motivo que me empujó a hacer este proyecto, para ver si de alguna forma podía aportar mi granito de arena y encontrar una solución al problema que os afecta a todos.

Tras la realización del proyecto, mis conocimientos sobre la situación actual de la ocupación han aumentado significativamente. He llegado a comprender la ocupación desde el punto de vista legal, las leyes que hay detrás del delito de ocupación, por qué no se puede desahuciar a un presunto ocupa, así como así y el problema de contratar empresas dedicadas al desalojo. Es un tema muy delicado y sensible socialmente hablando, porque realmente nadie ocupa por gusto

El otro motivo fue la relación de un tema tan interesante con la inteligencia artificial. El campo del *machine learning* y del desarrollo de software para otorgar cualidades meramente humanas a los sistemas informáticos, es un tema realmente apasionante porque es capaz de relacionar las matemáticas, la informática y el ingenio humano. Son herramientas que permiten alcanzar objetivos inimaginables.

La lucha contra la ocupación viene de muchos años atrás y lejos de solucionarse, cada año aumenta. La esperanza es la inteligencia artificial, capaz de predecir si una vivienda es propensa a ocuparse y actuar en consecuencia para poder evitarlo. Pero no basta con eso. No hay que quedarse ahí. Hay que tener en cuenta que, si este tipo de modelos tienen éxito, la ocupación no tiene por qué erradicarse, sino que tenderá a evolucionar. Cambiarían los patrones de ocupación y los factores influyentes, por lo que estos modelos deben estar en constante cambio y evolución para poder combatirla.

Cuando se comenzó a redactar el anteproyecto, el interés y entusiasmo se acrecentó, ya que empezaba a ver el potencial que podía tener el proyecto. Una idea innovadora que podría suponer un punto de inflexión en la lucha contra la ocupación.

El problema surgió cuando se empezó a trabajar más en serio y en profundidad en el proyecto. La motivación comenzó a decaer. El motivo de ello fue el choque de realidad al ver la falta, y en muchos casos, inexistencia de determinados datos, que se consideraban cruciales, para el desarrollo del modelo. Aunque surgían algunas ideas su implementación no tenía ningún éxito. Cuanto más se indagaba, más difícil llegar a conseguir y tener acceso a datos con los que poder entrenar el modelo.

La solución a este problema surgió con el descubrimiento de los datos sintéticos.

En un principio, puede parecer que los datos sintéticos son una especie de datos pseudoaleatorios para poder simular, de una forma no muy realista, una situación. Pero no es así. Como se explica detalladamente en el apartado 4.2 Descripción de la solución, metodologías y herramientas empleadas, los datos sintéticos son datos que simulan de una forma muy fiel la realidad, ya que utiliza información real y distribuciones estadísticas de los datos reales.

Además, a parte del conocimiento sobre el tema tratado, se han incrementado mis aptitudes técnicas en programación y tratamiento de datos. He conocido más en detalle el mundo del *Deep Learning* y las posibilidades que tiene esta rama de la inteligencia artificial.

Una conclusión muy interesante, a la que se ha llegado realizando la documentación, es que todo el miedo y temor que hay respecto a este tema, es un miedo infundado. Es decir, estamos sujetos a un sensacionalismo producido por los medios de comunicación. No es un hecho inventado o especulativo, realmente en Madrid Capital hay menos de 3.000 viviendas ocupadas de un total de más de un millón y medio de viviendas [37]. Es un hecho puramente estadístico, la proporción es muy baja, aunque pueda parecer que 3.000 viviendas sean muchas. [38, 12]. Es verdad que, en un mundo ideal, el número de viviendas ocupadas debería ser cero porque nadie merece vivir con miedo a que se ocupe una casa o a sufrir como su hogar es maltratado, y es por eso por lo que se ha desarrollado este modelo.

En lo relativo a si se va a seguir trabajando en este proyecto, hay una serie de ideas, detalladas en el apartado Capítulo 7 FUTURAS LÍNEAS DE TRABAJO, que son los siguientes pasos a seguir para que este trabajo y este modelo pueda realmente en algún momento poder ayudar a personas que lo necesitan, que viven con miedo, que no cuentan con los recursos necesarios para poder solucionar una situación de ocupación y llegar a ser útil para la sociedad.

Capítulo 7. FUTURAS LÍNEAS DE TRABAJO

Una vez desarrollado el proyecto, se han descubierto líneas de trabajo con las que se puede mejorar y ampliar el espectro del problema. A continuación de enumeran las futuras líneas de trabajo:

- La línea futura de trabajo más clara es la ampliación de la región geográfica a la que restringir el modelo. Para este proyecto se decidió focalizar en Madrid capital por dos motivos principales:
 - El primero es que Madrid es una de las ciudades de España más afectadas por la ocupación ilegal.
 - En segundo lugar, Madrid es la capital de España y a su vez, la ciudad más grande. Con esto se refiere a que hay más cantidad de personas, de sucesos, de noticias... lo que convierte a Madrid en una ciudad muy rica en información y, en consecuencia, es una ciudad donde se realizan muchos estudios y la cantidad de datos que se pueden obtener es mayor que otros municipios.

Entonces, el objetivo sería ampliar el modelo y para ello habría que seguir un cierto orden. Aunque el objetivo final sería el desarrollo de un modelo de predicción automática para comprobar si una vivienda va a ser ocupada de forma ilegal en cualquier municipio o pueblo de España, es importante ir escalando poco a poco. Una vez establecido un modelo para Madrid capital, lo correcto sería, antes de pasar a la Comunidad de Madrid, pasar a implementar el modelo en Barcelona, ya que esta ciudad es muy similar a Madrid en cuanto a los requisitos por los que se escogió Madrid. Es una de las ciudades más afectadas por la ocupación ilegal y es la segunda ciudad más grande de España.

Tras Barcelona, adaptarlo al resto de capitales de provincia, ya que son ciudades grandes y que suelen estar afectadas por la ocupación.

Pero es importante realizar buenos estudios de los lugares que se quiere implementar, ya que invertir tiempo y recursos en desarrollar el modelo para predecir en lugares donde no hay ninguna vivienda ocupada y que el riesgo es casi nulo, contradeciría el objetivo del modelo de optimizar costes.

- El proyecto, a priori, está desarrollado y pensado como un modelo de negocio B2B, es decir *business to business*. El público objetivo en el que se centra son empresas propietarias de un gran número de inmuebles, como pueden ser los bancos, o para empresas de alarmas para poder establecer sus precios y dónde establecer sus campañas de promoción. Por lo que otra futura línea de trabajo sería la adaptación del modelo para que lo usasen particulares que tengan miedo de que su casa o su segunda vivienda pueda ser usurpada. La mejor forma de adaptar el modelo para que lo usasen particulares sería desarrollando una aplicación móvil, la cual sea accesible de una forma sencilla y que cualquier persona pueda tener acceso a ella.

Para ello, lo más apropiado sería la contratación de una empresa externa especializada en el desarrollo de aplicaciones móviles, ya que, si se quiere hacer de una forma seria y con fines comerciales, lo adecuado sería que la aplicación sea lo más profesional posible. Un primer paso para poder establecer un modelo de negocio B2C (*business to consumer*), antes de contratar una empresa de desarrollo de software, sería un estudio estadístico para ver el promedio de personas interesadas, que pagarían por la aplicación y que están preocupadas porque su vivienda pueda ser ocupada. Por lo tanto, con un estudio de este estilo podría llevarse a delante la idea o evitar cometer el error de invertir dinero en un producto que no tendría ningún éxito.

- Por último, como punto de mejora y que además podría ayudar con la primera línea de trabajo descrita, sería muy interesante la creación de un modelo de IA explicable para analizar el modelo creado. La IA explicable se utiliza para describir un modelo de inteligencia artificial. Esta IA consistiría, como su propio nombre indica, en un modelo de inteligencia artificial el cual es capaz de analizar el modelo y establecer cuáles son las variables más importantes e imprescindibles y poder de esta forma optimizarlo [39, 40].

Capítulo 8. REFERENCIAS

- [1] "Radiografía de la okupación en Madrid: casi cuatro usurpaciones al día y el 15%, asaltos violentos", *ELMUNDO*, 2021. [Online]. Available: <https://www.elmundo.es/madrid/2020/09/13/5f5d10fcfc6c837b0e8b45a2.html>. [Accessed: 15- Jan- 2021].
- [2] "Mapa de la okupación en España distrito a distrito - Brainsre news España", Brainsre news España, 2021. [Online]. Available: <https://brainsre.news/mapa-de-la-okupacion-en-espana-distrito-a-distrito/>. [Accessed: 15- Jan- 2021].
- [3] C. Castro, "La okupación como problema legal y social - Hay Derecho", Hay Derecho, 2021. [Online]. Available: <https://hayderecho.expansion.com/2020/09/10/la-okupacion-como-problema-legal-y-social/#:~:text=Seg%C3%BAn%20el%20Ministerio%20del%20Interior,que%20se%20acab%C3%B3%20en%202019.> [Accessed: 17- Jan- 2021].
- [4] Interior.gob.es, 2021. [Online]. Available: <http://www.interior.gob.es/prensa/balances-e-informes>. [Accessed: 18- Jan- 2021].
- [5] "Protocolo de actuación de las Fuerzas y Cuerpos de Seguridad del Estado ante la ocupación ilegal de viviendas", Iberley.es, 2021. [Online]. Available: <https://www.iberley.es/temas/protocolo-actuacion-fuerzas-cuerpos-seguridad-estado-okupacion-ilegal-viviendas-64874>. [Accessed: 18- Jan- 2021].
- [6] "6 formas de solucionar el problema de las viviendas ocupadas - pisos Al día - pisos.com", pisos Al día - pisos.com, 2021. [Online]. Available: <https://www.pisos.com/aldia/6-formas-de-solucionar-el-problema-de-las-viviendas-ocupadas/1623016/>. [Accessed: 20- Jan- 2021].
- [7] "Los riesgos de recurrir a las empresas 'desokupa': propietarios que acaban investigados - El Independiente", El Independiente, 2021. [Online]. Available: <https://www.elindependiente.com/politica/2020/08/23/los-riesgos-de-recurrir-a-las-empresas-desokupa-propietarios-que-acaban-investigados/>. [Accessed: 18- Jan- 2021].
- [8] "Moody's advierte a España: "Las políticas que protegen la ocupación ilegal son una amenaza para la vivienda"", *ELMUNDO*, 2021. [Online]. Available: <https://www.elmundo.es/economia/vivienda/2021/02/10/60241978fdddf9d908b4643.html>. [Accessed: 21- Jan- 2021].
- [9] *Economiadigital.es*, 2021. [Online]. Available: https://www.economiadigital.es/inmobiliario/una-casa-okupada-se-deprecia-un-42_184656_102.html. [Accessed: 21- Jan- 2021].
- [10] 2021. [Online]. Available: https://elpais.com/ccaa/2018/11/09/madrid/1541777305_325010.html. [Accessed: 23- Jan- 2021].

- [11] "Web Scraping with Python", Google Books, 2021. [Online]. Available: https://books.google.es/books?hl=es&lr=&id=V_I_CwAAQBAJ&oi=fnd&pg=PP1&dq=web+scraping&ots=G0xo4tRyXo&sig=8Zezyb5c7pgQeAV0H80OkwUkhXU#v=onepage&q&f=false. [Accessed: 25- Mar- 2021].
- [12] ""Realmente el número de viviendas particulares ocupadas en España es muy pequeño" - El Faradio | Periodismo que cuenta", El Faradio | Periodismo que cuenta, 2021. [Online]. Available: <https://www.elfaradio.com/2020/09/02/realmente-el-numero-de-viviendas-particulares-ocupadas-en-espana-es-muy-pequeno/>. [Accessed: 30- Jan- 2021].
- [13] "Policía Municipal. Datos estadísticos actuaciones Policía Municipal - Portal de datos abiertos del Ayuntamiento de Madrid", Datos.madrid.es, 2021. [Online]. Available: <https://datos.madrid.es/portal/site/egob/menuitem.c05c1f754a33a9fbe4b2e4b284f1a5a0/?vgnextoid=bffff1d2a9fdb410VgnVCM2000000c205a0aRCRD&vgnextchannel=374512b9ace9f310VgnVCM100000171f5a0aRCRD&vgnextfmt=default>. [Accessed: 02- Apr- 2021].
- [14] "Indicadores de renta media y mediana (31097)", INE, 2021. [Online]. Available: <https://www.ine.es/jaxiT3/Datos.htm?t=31097>. [Accessed: 06- Apr- 2021].
- [15] "Total viviendas familiares y total viviendas principales por municipios (lista completa)", INE, 2021. [Online]. Available: <https://www.ine.es/jaxi/Datos.htm?path=/t20/e244/viviendas/p06/l0/&file=9mun28.px>. [Accessed: 06- Apr- 2021].
- [16] "Practical Synthetic Data Generation", Google Books, 2021. [Online]. Available: https://books.google.es/books?id=XWnnDwAAQBAJ&printsec=frontcover&dq=synthetic+data&hl=es&sa=X&redir_esc=y#v=onepage&q=synthetic%20data&f=false. [Accessed: 10- Apr- 2021].
- [17] "Alarma anti okupas: cómo echarlos y precios (2021)", Selectra, 2021. [Online]. Available: <https://selectra.es/alarmas/seguridad-anti-okupas>. [Accessed: 26- Apr- 2021].
- [18] "La Comunidad de Madrid contabiliza más de 10.500 viviendas sociales ocupadas ilegalmente", Fuenlabrada Noticias, 2021. [Online]. Available: <https://fuenlabradanoticias.com/art/99857/la-comunidad-de-madrid-contabiliza-mas-de-10500-viviendas-sociales-ocupadas-ilegalmente>. [Accessed: 27- Apr- 2021].
- [19] E. País, "El perfil real del okupa que no aparece en los debates: familias en pisos propiedad de bancos", Verne, 2021. [Online]. Available: https://verne.elpais.com/verne/2019/05/16/articulo/1558015569_606214.html. [Accessed: 28- Apr- 2021].
- [20] "TensorFlow", TensorFlow, 2021. [Online]. Available: <https://www.tensorflow.org/>. [Accessed: 10- May- 2021].
- [21] "¿Qué es el Deep Learning?", SmartPanel, 2021. [Online]. Available: <https://www.smartpanel.com/que-es-deep-learning/>. [Accessed: 17- May- 2021].

- [22] Arxiv.org, 2021. [Online]. Available: <https://arxiv.org/pdf/1904.05526.pdf>. [Accessed: 03-Jun- 2021].
- [23] "Handling imbalanced datasets in machine learning", Medium, 2021. [Online]. Available: <https://towardsdatascience.com/handling-imbalanced-datasets-in-machine-learning-7a0e84220f28>. [Accessed: 30- Jun- 2021].
- [24] "pandas.get_dummies — pandas 1.3.0 documentation", Pandas.pydata.org, 2021. [Online]. Available: https://pandas.pydata.org/docs/reference/api/pandas.get_dummies.html. [Accessed: 30- Jun- 2021].
- [25] "Explicación Funciones de activación y práctica con Python.", Medium, 2021. [Online]. Available: <https://rubialesalberto.medium.com/explicaci%C3%B3n-funciones-de-activaci%C3%B3n-y-pr%C3%A1ctica-con-python-5807085c6ed3>. [Accessed: 04- May- 2021].
- [26] "Before you continue to YouTube", Youtube.com, 2021. [Online]. Available: <https://www.youtube.com/watch?v=HKPE3mrXOPo>. [Accessed: 04- May- 2021].
- [27] Diegocalvo.es, 2021. [Online]. Available: <https://www.diegocalvo.es/funcion-de-activacion-redes-neuronales/>. [Accessed: 04- May- 2021].
- [28] "Cloud Computing Services | Google Cloud", Google Cloud, 2021. [Online]. Available: <https://cloud.google.com/>. [Accessed: 01- Jul- 2021].
- [29] "Colab+GCP Compute — how to link them together", Medium, 2021. [Online]. Available: <https://medium.com/@senthilnathangautham/colab-gcp-compute-how-to-link-them-together-98747e8d940e>. [Accessed: 02- Jul- 2021].
- [30] "Building Web Apps with Python and Flask", Google Books, 2021. [Online]. Available: https://books.google.es/books?id=gtwiEAAQBAJ&printsec=frontcover&dq=flask+python&hl=es&sa=X&redir_esc=y#v=onepage&q=flask%20python&f=false. [Accessed: 03- Jul- 2021].
- [31] "Pricing | Cloud Functions | Google Cloud", Google Cloud, 2021. [Online]. Available: <https://cloud.google.com/functions/pricing>. [Accessed: 30- Jun- 2021].
- [32] "Pricing | Cloud Storage | Google Cloud", *Google Cloud*, 2021. [Online]. Available: <https://cloud.google.com/storage/pricing>. [Accessed: 30- Jun- 2021].
- [33] "Pricing | Compute Engine: Virtual Machines (VMs) | Google Cloud", Google Cloud, 2021. [Online]. Available: <https://cloud.google.com/compute/all-pricing>. [Accessed: 30- Jun- 2021].
- [34] "Classification: Accuracy | Machine Learning Crash Course", Google Developers, 2021. [Online]. Available: <https://developers.google.com/machine-learning/crash-course/classification/accuracy>. [Accessed: 30- Jun- 2021].
- [35] 2021. [Online]. Available: <https://pixabay.com/es/vectors/gr%C3%A1fica-smiley-emoticon-llorar-3643600/>. [Accessed: 05- Jul- 2021].
- [36] 2021. [Online]. Available: <https://pixabay.com/es/vectors/smiley-feliz-cara-sonrisa-suerte-559124/>. [Accessed: 05- Jul- 2021].

[37] 2021. [Online]. Available: <https://www.larazon.es/local/madrid/vallecas-el-distrito-con-mas-casas-okupadas-MF25469126/>. [Accessed: 24- Apr- 2021].

[38] 2021. [Online]. Available: <https://elpais.com/espana/2020-09-05/una-dudosa-alarma-sobre-los-okupas.html>. [Accessed: 19- Mar- 2021].

[39] "Inteligencia artificial explicable", Ibm.com, 2021. [Online]. Available: <https://www.ibm.com/es-es/watson/explainable-ai>. [Accessed: 08- May- 2021].

[40] "Interpretable Machine Learning", Google Books, 2021. [Online]. Available: https://books.google.es/books?id=jBm3DwAAQBAJ&printsec=frontcover&dq=what+is+explainable+ai&hl=es&sa=X&redir_esc=y#v=onepage&q=what%20is%20explainable%20ai&f=false. [Accessed: 04- Jul- 2021].

Capítulo 9. ANEXOS

9.1 Uso de la aplicación web

Como se ha explicado previamente en el apartado 4.6.3 Aplicación *web*, se ha creado una aplicación web muy simple, con una interfaz muy sencilla de utilizar e intuitiva.

Según se carga se ve la una interfaz con cuatro campos para completar (*Figura 9*)

- Calle
- Distrito (Distrito de Madrid donde se encuentra el domicilio)
- Propietario (Banco, Particular o Ayuntamiento)
- Alarma (Sí/No)

Simplemente se autocompletan:

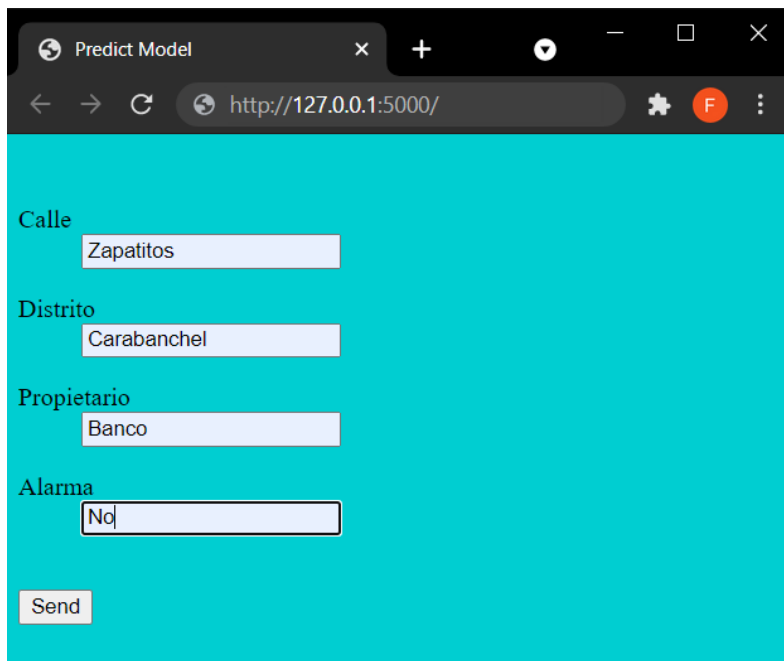


Figura 13. Interfaz de aplicación web - Ejemplo de campos completados

Se clicla en 'Send' y se realiza la predicción.

- En caso de que se prediga que va a ser ocupada devuelve: Figura 10.
- En caso contrario, si se predice que no va a ser ocupada devuelve: Figura 11.

9.2 Repositorio de GitHub

9.2.1 Extracción de datos

El código de los *scripts* desarrollados en la extracción de datos y la información recopilada, pueden revisarse en el siguiente [REPOSITORIO](#) de GitHub.

9.2.2 Datos sintéticos

El código de la creación de datos sintéticos junto con los datos utilizados para su desarrollo, pueden revisarse en el siguiente [REPOSITORIO](#).

9.2.3 Modelo de predicción

El notebook de Python donde se ha programado el modelo, junto con los datos de entrenamiento y el modelo obtenido, puede revisarse en el siguiente [REPOSITORIO](#).

9.2.4 Aplicación web

Todo el código de la aplicación web junto con todos los archivos que utiliza, puede revisarse en el siguiente [REPOSITORIO](#).

9.2.5 Resultados de modelos descartados

Los resultados de pruebas realizadas a modelos descartados pueden consultarse en el siguiente [REPOSITORIO](#).

9.2.6 Conjuntos de datos

Todos los conjuntos de datos que se han utilizado y que se han creado, pueden consultarse en el siguiente [REPOSITORIO](#).

[PÁGINA INTENCIONADAMENTE EN BLANCO]