



Alparking

Alparking

NullStorm: Gonzalo Cáceres - Jonay Temiño - Manuel Abos - Jordi Ortega

Keepcoding - IA Full Stack 2 - 2024/2025



1.

Definición del problema

- Elevados costos en contratación de personal o sistemas complejos.
- Pérdida de información relevante.

2.

Objetivos del proyecto

- Automatizar el acceso de vehículos.
- Reducir los costos operativos.
- Análisis de los datos para futuras mejoras.

3.

Claves del proyecto

- Preprocesamiento de las imágenes.
- Uso de modelos de detección de objetos y de OCR.
- Almacenar eficientemente los datos.
- Poder acceder a ellos de manera rápida y sencilla.



Servicio



BigQuery

*Almacenamos la información obtenida en el paso 1. Explotación y **Análisis de datos** mediante SQL.*



1

Computer Vision

2

EDW

3

LLM

Yolo + OCR

*Detección de objetos con Yolo y **extracción del texto** de las matrículas con OCR convirtiendo la conversión de contenido visual en **datos editables**.*



OpenAI

*Aplicamos **Prompt Engineering** para ejecutar consultas en castellano.*



Project Pipeline

A

Definición del Reto.
Control de accesos automatizado y analítica para parkings.

B

Recolección de Datos.
Imágenes cedidas por empresas del sector y fotografías propias.

C

Preprocesamiento de Datos. Funciones de tratamiento de imágenes y limpieza.

D

Análisis Exploratorio.
Inspección visual de imágenes, tamaños y colores.



E

Selección del Modelo.
YOLO (You Only Look Once) basado en redes neuronales convolucionales.
GPT-3.5-Turbo basado en transformers.

F

Reconocimiento de Caracteres. EasyOCR como biblioteca especializada.

G

Evaluación del Modelo.
Métricas de evaluación:
Precisión, Recall y F1-Score.

H

Optimización y Ajuste.
Funciones de limpieza y ejercicios de Prompt Engineering.





Project Pipeline

I

Producción.

Despliegue de una aplicación web integrada con BBDD desplegada en el entorno cloud de Google.



BigQuery



Quasar



Vue.js



FastAPI

J

Monitoreo y Mantenimiento.

[wip] Monitoreo de precisión y monitoreo de desempeño en producción. Se levanta log de errores.

K

Documentación.

"1. Fundamentos Teóricos y Exploración de los Datos.ipynb"



<https://github.com/nullstorm4/Project>

<https://drive.google.com/file/d/1vAe2Shmnngu3WBLGjtSS-Qw-u5ky8X2Fh/view?usp=sharing>



Producción



Desarrollo



1

Computer Vision

- **YOLOv5s**
Modelo de Ultralytics especializado en detección de objetos.
- **YOLOv5m-license-plate**
Versión 'medium' de v5 entrenada para detectar placas de vehículos.
- **Easyocr**
Librería OCR para reconocimiento de caracteres.





Preprocesamiento

- **Escala de grises:** reducción de la cantidad de información, enfocando el modelo en las formas.
- **Reducción de ruido con desenfoque Gaussiano:** suaviza la imagen para eliminar información irrelevante.
- **Detección de bordes:** ayuda a identificar los contornos de matrículas y vehículos.
- **Mejora de contraste:** resalta diferencias entre fondo y objeto.

Preprocesamiento aplicado	Precisión (%)
Sin preprocesamiento	40.40%
Escala de grises + Ruido	34.69%
Escala de grises + Bordes	2.04%
Escala de grises + Contraste	45.45%

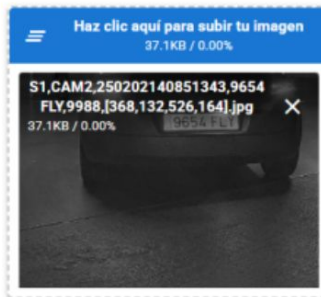


Modelo final

1

Computer Vision

- Formatos de las placas.
- Filtrar el texto para evitar caracteres no admitidos.
- Filtrar el texto basado en el tamaño de la región.
- Corregir errores comunes.





2

EDW

- **Mapa de entidades**
Se define la estructura del edw operacional, los esquemas de las entidades y la relación entre ellas.
- **Entorno Google Cloud**
Creamos el proyecto, habilitamos la **API de BigQuery** y configuramos credenciales.
- **Proceso - Proyecto BigQuery**
Importamos el módulo **bigquery** de la librería **google.cloud** para interactuar con gbq con SQL.



BigQuery



00_aiparking

abonados

accesos

parking

tarifa

tipo_vehiculo

accesos

CONSULTA

ABRI

ESQUEMA

DETALLES

VISTA PREV

Filtro Ingresar el nombre o el valor de la p

<input type="checkbox"/>	Nombre del campo	Tipo
<input type="checkbox"/>	matricula_id	STRING
<input type="checkbox"/>	parking_id	STRING
<input type="checkbox"/>	fecha_registro_dt	DATETIME
<input type="checkbox"/>	camara_in_id	STRING
<input type="checkbox"/>	camara_out_id	STRING
<input type="checkbox"/>	tarifa_id	STRING
<input type="checkbox"/>	marca_ds	STRING
<input type="checkbox"/>	modelo_ds	STRING
<input type="checkbox"/>	color_ds	STRING
<input type="checkbox"/>	tipo_vehiculo_id	STRING

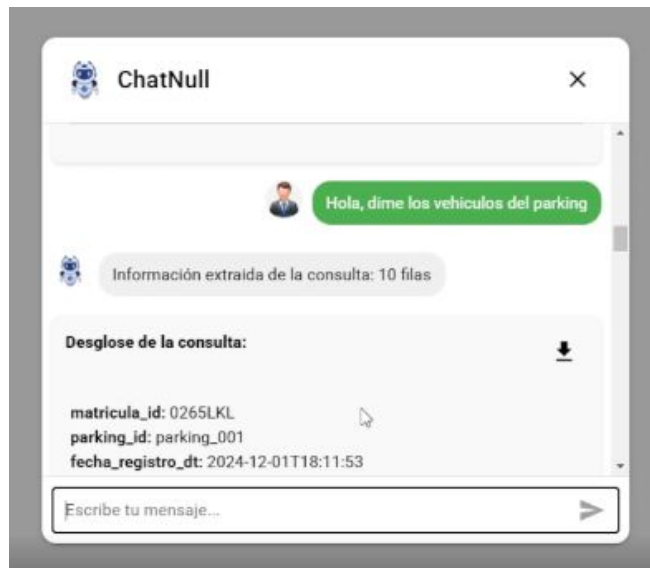


Large Language Model

3

LLM

- **gpt-3.5-turbo**
Importamos el módulo `openai` de la librería `openai` y trabajamos con la versión optimizada de gpt-3 para tareas de lenguaje natural.
- **Prompt Template**
Entregamos una instrucción clara, definimos un contexto, damos ejemplos clarificadores y planteamos parámetros configurables con el objetivo de interactuar en español con la base de datos.





Métricas y Evaluación

A

Evaluación

Métricas - 2.242 imágenes

- Precisión 87%
- Recall 79%
- F1-Score 81%

Errores más comunes

Carácter	Lectura	#
W	I	20
4	0	20
W	H	18
G	6	17
T	I	16

Nombre del archivo	Placa	Placa detectada	Errores	Similitud
E1,CAM1,250211093523443,5361KNT,9962.jpg	5361KNT	5361LNT	['K->L']	85,71%
E1,CAM1,250211100504491,6822HXW,9999.jpg	6822HXW	6822HXH	['W->H']	85,71%
E1,CAM1,25021111823386,7715KLT,9996.jpg	7715KLT	7715KLH	['T->H']	85,71%
E1,CAM1,250211102223608,6598HSH,9730.jpg	6598HSH	6590HSH	['8->0']	85,71%
E1,CAM1,250211113127033,9958GVD,9737.jpg	9958GVD	9958GVB	['D->B']	85,71%
E1,CAM1,250211084626206,5767GXJ,9998.jpg	5767GXJ	6767GXJ	['5->6']	85,71%
E1,CAM1,250211101308215,3773CXY,10000.jpg	3773CXY	8773CXY	['3->8']	85,71%
E1,CAM1,250211083428634,4980CBL,9957.jpg	4980CBL	2980CBL	['4->2']	85,71%
S1,CAM2,250210090620585,5017LKP,9991.jpg	5017LKP	5017LKP	[]	100%
S1,CAM2,250210084922545,6662HTL,9948.jpg	6662HTL	6662HTL	[]	100%
S1,CAM2,250210090236785,6968MLH,9999.jpg	6968MLH	6968MLH	[]	100%
S1,CAM2,250210084635210,5839GXB,9993.jpg	5839GXB	5839GXB	[]	100%
S1,CAM2,250210084557009,8485MSR,9978.jpg	8485MSR	8485MSR	[]	100%
S1,CAM2,250210085639741,0228MSZ,9960.jpg	0228MSZ	0228MSZ	[]	100%
S1,CAM2,250210090700662,8123KTG,9999.jpg	8123KTG	8123KTG	[]	100%
S1,CAM2,250210075314986,3133DRL,9974.jpg	3133DRL	3133DRL	[]	100%
S1,CAM2,250210063809315,3802GVZ,9910.jpg	3802GVZ	3802GVZ	[]	100%
S1,CAM2,250210072412936,7983FPC,10000.jpg	7983FPC	7983FPC	[]	100%



B

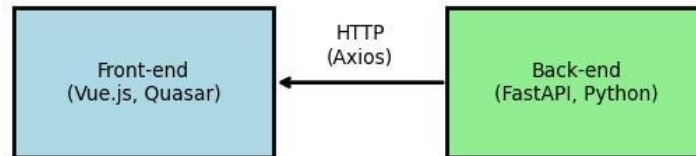
Despliegue

Arquitectura General y Tecnologías Utilizadas

Aquí se muestra la visión global del sistema, destacando la separación entre el front-end (responsable de la interacción con el usuario) y el back-end (donde se realiza el procesamiento de imágenes y se gestionan los endpoints).

Seleccione estas tecnologías por los siguientes motivos:

- **Interfaz de usuario moderna:** Se ha empleado Vue.js junto a Quasar para desarrollar interfaces dinámicas y responsive, ofreciendo una experiencia de usuario fluida y de alta calidad.
- **Despliegue eficiente de modelos:** Los modelos se han desplegado con FastAPI, una herramienta ideal para integrar y exponer modelos de forma eficiente, aprovechando también rutas estáticas para alojar la web en el mismo servicio.
- **Comunicación robusta:** La librería Axios se utiliza para gestionar la comunicación entre el frontend y el backend, garantizando un intercambio de datos fiable y sencillo.





Next Steps

Computer Vision

1

- Levantar procesos de monitoreo para asegurar la precisión y la eficiencia en el desempeño.
- Identificar otros componentes de las fotografías: marca, modelo y color.
- Añadir más formatos de placa de otros países.

EDW

2

- La POC se ha centrado en la tabla 'accesos' y el proceso debería alimentar la estructura operacional definida.
- Desnormalizar los datos para mejorar el rendimiento de consultas analíticas.
- Orquestación de los flujos de datos con otros sistemas del cliente.

LLM

3

- Levantar procesos de monitoreo para asegurar la precisión y la eficiencia en el desempeño.
- Utilizar un modelo LLM que consuma menos recursos
- Añadir memoria al modelo para recordar consultas pasadas.

Despliegue

B

- Mejorar diseño de interfaz como por ejemplo mostrar un acumulado del resto de pruebas.
- Convertir en un archivo ejecutable para poder desplegarlo más fácilmente.
- Añadir la opción de subir un video y poder procesarlo con los modelos.



AI Parking

Project

servidor_nullStorm_python

D:\Bootcamp IA\Project\servidor_nullStorm_python

dist

env library root

alparking-451016-e10ace861ca1.json

coche.jpg

main.py

requirements.txt

volov5s.pt

External Libraries

Scratches and Consoles

main.py

coche.jpg

Download grammar and spelling checker for Spanish?

insertar_matricula

```
466 #
467 @app.get("/")
468 def index():
469     return FileResponse("dist/index.html")
470
471
472 # ruta SPA para controlar las rutas de la web
473 # (full_path:path)
474 @app.get("/{full_path:path}")
475 def spa_fallback(full_path: str):
476     file_path = os.path.join("dist", full_path)
477     if os.path.exists(file_path) and not os.path.isdir(file_path):
478         return FileResponse(file_path)
479     # si ninguna ruta existe, volvera a archivo raíz de la web
480     return FileResponse("dist/index.html")
481
482
483 # uvicorn main:app --reload --host 0.0.0.0 --port 5000
```

Terminal

Local

Windows PowerShell

Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma <https://aka.ms/pscore6>

(env) PS D:\Bootcamp IA\Project\servidor_nullStorm_python> uvicorn main:app --reload --host 0.0.0.0 --port 5000

servidor_nullStorm_python

main.py

483:3 (52 chars) CRLF UTF-8 4 spaces Python 3.12 (servidor_nullStorm_python)

NullStorm - Brave

servidor_nullStorm_p...

20:13