



Guía Reto - Codefest Ad Astra 2023

1. RETO CODEFEST AD ASTRA 2023

Objetivo 1. Identificación objetos de interés en videos:

El objetivo 1 consiste en identificar de manera automatizada objetos de interés en videos capturados por sensores FLIR y Aeronaves Remotamente Pilotadas, en los cuales se evidencian situaciones de afectación ambiental a la Amazonía colombiana. Se requiere entonces una librería **open source** creada en python que genere un archivo con el resultado del análisis y recuadros de los objetos identificados.

La librería debe tener un método que reciba como parámetro la ruta del video y del archivo que contiene el resultado del análisis:

```
detect_objects_in_video(video_path, output_path)
```

El método debe generar un archivo con el resultado del análisis, en la ruta dada. El formato de este archivo debe ser CSV. Cada fila debe tener la siguiente estructura:

<ID>, <OBJECT_TYPE>, <TIME>, <COORDINATES_TEXT>

- <ID> := "Identificador del objeto encontrado. Puede ser un número o cadena única que servirá para identificar el archivo con el recuadro generado."
- <OBJECT_TYPE> := "Tipo de objeto identificado. Puede tener los siguientes: VEHICULO, CONSTRUCCIÓN, VIA, OTROS"
- <TIME> := "Tiempo en el video de aparición del objeto, en el formato HH:MM:SS (hora militar)"
- <COORDINATES_TEXT> := "Texto de coordenadas que aparece en la imagen mientras se ve el objeto"



Adicionalmente, dentro de la carpeta destino se debe crear un carpeta con nombre IMG, que incluirá los recuadros de los objetos identificados. Las imágenes con el recuadro deben tener como nombre el id del objeto reportado en el archivo de resultado.

Cada equipo contará con un conjunto de datos compuesto por 5 videos con una duración entre 11 mins y 46 mins.

Objetivo 2: Identificación de entidades en noticias

Este objetivo consiste en identificar entidades en noticias que describan afectaciones medio ambientales a la Amazonía colombiana. El conjunto de datos disponible está conformado por noticias desde 2010 hasta 2023 y cuenta con 187 registros, en los cuales se describen situaciones de afectación de la Amazonía colombiana relacionadas con contaminación, extracción ilícita de minerales y deforestación.

En este tipo de noticias es importante identificar de manera automatizada los actores que registran algún tipo de participación directa o indirecta en mencionada afectación. Para esta búsqueda y clasificación automatizada sobre textos existen herramientas NER que identifican en textos entidades tales como ubicaciones, organizaciones, personas y otros actores. Ejemplos de esas herramientas son: Spacy, BERT Google , NLP Stanford, Gate, OpenNLP, entre otras.

Para esto se requiere incluir en la librería los siguientes tres métodos que reciben como parámetro un string , que hace referencia al texto o su ubicación, y un string con el directorio del archivo de salida:

```
ner_from_str(text, output_path)
```

```
ner_from_file(text_path, output_path)
```

```
ner_from_url(url, output_path)
```

Los tres métodos deben generar un archivo JSON (en el output_path dado), con la siguiente estructura:



```
{  
  "text": "<here-goes-the-original-text>"  
  "org": ["<entity_1>", "<entity_2>", ...],  
  "loc" : ["<location_1>", "<location_2>", ...],  
  "per" : ["<person_1>", "<person_2>", ...],  
  "dates" : ["<date_1>", "<date_2>", ...],  
  "misc": ["<misc_1>", "<misc_2>", ...],  
  "impact": "<impact_type>"  
}
```

<impact_type>:= "Afectación al medio ambiente que puede ser MINERIA, CONTAMINACION, DEFORESTACION, NINGUNA"

TIP CODEFEST: los modelos de las herramientas NER pueden requerir re-entrenamiento para mejorar el entendimiento del contexto y reducir algunos de los problemas mencionados en las charlas CODEFEST. Para re-entrenar los modelos, en el caso de NER, es necesario contar con textos etiquetados. Algunas herramientas disponibles para esta tarea son Doccano, NER TEXT Annotator y Amazon Sage Maker.

Especificaciones adicionales:

Note que se pide una librería mas no un notebook, es decir, la librería debe poder ser consumida e importada por código cliente. Al ser open source, debe estar alojada en github, con una licencia permisiva de código abierto (e.g., MIT, APACHE) declarada en el repositorio. Esta licencia no debe entrar en conflicto con las licencias de las librerías usadas. En estos enlaces pueden encontrar más información respecto a diferentes tipos de licencias:

<https://choosealicense.com>
<https://fossa.com/blog/all-about-permissive-licenses/>



Adicionalmente, al ser una librería se busca que ésta siga buenas prácticas de documentación, estilo y organización de código. En esa medida la librería debe incluir un archivo README que explique el uso de la librería y documentación detallada de los métodos y clases en el código. El README debe tener instrucciones claras sobre la creación y uso (i.e., import) de la librería a partir del código fuente en el repositorio, es decir, se deben explicar las técnicas usadas, pipeline, etc. En el README se debe describir detalladamente cómo se resolvió cada objetivo, es decir, se deben explicar las técnicas usadas, pipeline, etc.

Las funciones deben contar con comentarios “inline”, es decir, comentarios/documentación en el código que explique la estrategia usada. Esta es una guía de ejemplo sobre documentación de código en python: <https://realpython.com/documenting-python-code/>.

La legibilidad del código es un aspecto importante, por lo tanto se espera que se usen identificadores de clases, métodos, variables, archivos expresivos y que faciliten el entendimiento del código. La organización del código (e.g, layout, indentación) debe facilitar el entendimiento del código.

2. ESQUEMA DE EVALUACIÓN

La evaluación del reto CODEFEST AD ASTRA 2023, se realizará en fases eliminatorias que buscan medir la calidad de las soluciones generadas por los equipos con un enfoque multi-dimensional.

Fase 1: Repositorio

El primer paso de la evaluación consiste en la validación de los requerimientos relacionados con el repositorio y código de la solución. La rúbrica de evaluación para esta fase es la siguiente:

Criterio de pre-evaluación	Cumple
Enlace a un repositorio github público	
La librería tiene una licencia open source	



permisiva que no entra en conflicto con las librerías usadas	
El archivo README describe el uso de la librería	
Al seguir las instrucciones del README es claro cómo usar la librería	
La librería se puede importar	
El README explica las estrategias usadas para resolver el reto	
Durante el evento el equipo no presentó violación alguna al código de honor	
Los comentarios de código de las funciones en el repositorio es clara	
La Librería al ser ejecutada no genera errores de compilación	
La librería al ser ejecutada en el conjunto de pruebas genera los archivos solicitados	

Si se cumplen todos los criterios anteriores, se procederá a la segunda fase de evaluación.

Fase 2: Calidad de la solución

Durante esta fase se evaluará la calidad de la solución desde el punto de vista del desempeño funcional de las funciones para cumplir con los dos objetivos del reto. La evaluación estará a cargo de miembros de la FAC.

Criterio	Valor de la métrica
NOTICIAS: Calidad en la detección de las entidades (Precision, Recall, F1-score, etc)	
NOTICIAS: Calidad en la detección del tipo de afectación (Precision, Recall, F1-score, etc)	
VIDEOS: Calidad en la detección de objetos en los videos (Precision, Recall, F1-score, etc)	
VIDEOS: Calidad en la detección de coordenadas (True Positive Rate)	



VIDEOS: Calidad en la identificación del tiempo de detección del objeto (True Positive Rate modificada)	
VIDEOS: Porcentaje de concordancia acumulado entre tipo de objeto y objeto en el recuadro de imagen. En caso de no existir la imagen el porcentaje es cero (0) para cada caso	
BONO: se darán puntos adicionales a las soluciones que puedan detectar objetos diferentes a vehículos en los videos	
BONO: se darán puntos adicionales a las soluciones que puedan detectar entidades diferentes a ORG, LOC, PER en los textos	

Fase 3: Elevator pitch

Los seis equipos con los puntajes más altos de las fases anterior serán seleccionados para esta fase. Cada equipo contará con 10 minutos para presentar/sustentar su solución (6 minutos de presentación + 4 minutos de preguntas por parte del jurado). Se espera que cada equipo describa brevemente la infraestructura y estrategias usadas para solucionar cada objetivo, las limitaciones de la solución, y ejemplos de los resultados en los videos/noticias de prueba. El jurado estará compuesto por personal FAC y expertos de las empresas patrocinadoras.

Criterio	No hay claridad (0)	Parcialmente (3)	Totalmente (6)
Claridad a la hora de explicar las estrategias usadas para la solución del objetivo 1			
Claridad a la hora de explicar las estrategias usadas para la solución del reto 2			
Calidad del material usado para la presentación			



El total de puntos de esta fase corresponde al promedio de puntos dados por los jurados.

3. SELECCIÓN DE GANADORES

Los 3 equipos ganadores se seleccionarán con base en el total de puntos obtenidos en las fases 2 y 3. Los 3 equipos con la puntuación más alta. En el caso de empates estos se resolverán con base en criterios adicionales definidos por los expertos.