

Avance #2: Creación de herramienta de aprendizaje automático basada en imágenes satelitales, como insumo para futuros casos de uso relacionados a penetración de paneles solares en Costa Rica

Autor: Carlos Andrés Rodríguez Trigueros
Máster universitario en computación.
Inteligencia Artificial.

Docente guía: Carlos Gaitán Potayos.
Diciembre de 2022.

Índice

1. Introducción.....	3
2. Descripción del avance del proyecto.....	3
2.1 Grado de cumplimiento de los objetivos y resultados previstos en el plan de trabajo.	3
2.2 Justificación de los cambios en caso necesario	6
3. Relación de las actividades realizadas.....	6
3.1 Actividades previstas en el plan de trabajo	6
3.2 Actividades no previstas y realizadas o programas.....	7
4. Relación de las desviaciones en la temporización y acciones de mitigación si procede y actualización del cronograma si procede	7
5. Listado de los resultados parciales obtenidos hasta el momento (entregables que se adjuntan).....	8
6. Comentarios de vuestro director particular si lo consideran necesario	8
7. Anexos	8

1. Introducción

El siguiente documento tiene como objetivo mostrar los avances finales obtenidos a fin de completar el trabajo final del máster, en el mismo se mostrarán cumplimiento de objetivos en base a cronograma original presentado, variaciones en planeamiento y recursos generados a ser entregados como parte del trabajo final de máster y por ende del avance.

2. Descripción del avance del proyecto

2.1 Grado de cumplimiento de los objetivos y resultados previstos en el plan de trabajo.

Con respecto al avance del proyecto y con respecto al cronograma presentado durante el primer avance, se detalla que los objetivos correspondientes a:

- Creación de modelo de aprendizaje automático

Como se mencionó en el avance anterior, este objetivo había dado inicio de forma temprana a fin de reducir riesgos asociados al proyecto, en el mismo se logra obtener un modelo de segmentación que debido a las imágenes amplias y con pocos paneles solares a nivel de métricas, se determina que para conocer qué tan preciso es el modelo se debe de analizar a nivel de decimales las variaciones en cuanto a pruebas generadas en base a un set de datos de prueba; cuestión que se analiza más a profundidad en el siguiente objetivo correspondiente al refinamiento.

Para el modelo de aprendizaje se utilizó Pytorch con una librería llamada segmentation models la cual facilita la rápida experimentación y Pytorch proporciona estabilidad ante variaciones de versiones comparado con librerías como tensorflow que es más famosa, esta decisión facilitó la creación del ambiente de trabajo y se contó con recursos que a pesar de su antigüedad funcionaban con pocas modificaciones, inclusive para el modelo de clasificación con el que se contó con menor tiempo para su creación.

- Refinamiento de modelo de aprendizaje automático

Este punto fue quizás de los más complejos pues el set de datos con el que se cuenta es bastante reducido (alrededor de 120 imágenes); a pesar de esta limitante se hizo uso de las capacidades de las librerías de machine learning y se incorporó además una herramienta online que permitiese hacer un rastreo de las diferentes variaciones que se aplicaran al modelo.

Se varió principalmente en cuanto a temas de arquitectura de red neuronal, cantidad de imágenes para batches y threshold de probabilidad de píxeles a fin

de generar máscaras de segmentación que se ajustaran mejor a lo que visualmente se observaba.

Todos estos experimentos, ya se encuentran disponibles en un reporte creado para ser incorporado en la página web a través de un iframe aprovechando las capacidades de este sitio web.

<https://costa-rica-solar.uc.r.appspot.com/info>

Es importante hacer notar que en link anterior se muestra también clasificación que se abarcara en objetivos posteriores y que también se le hizo refinamiento, pero no a la escala que se le hizo al modelo de segmentación que sirvió de guía para conocer igualmente las limitaciones y restricciones en cuanto a recursos que contaba.

Abordando este tema, durante el refinamiento aspectos reconocidos en el análisis de riesgo jugaron un gran factor, las imágenes al contener una cantidad inmensa de detalles , era importante tomar en cuenta factores como resolución de la imagen, pues se denoto que a mayor resolución el modelo siempre se comportaba mejor, pero el tener imágenes de alta resolución hicieron que el modelo de segmentación se basara en batches muy reducidos, lo cual no significa que el modelo presenta una precisión negativa importante , sino que se trató de balancear y aprovechar una estrategia de entrenamiento, donde un batch-size de 1 puede lograr buenos resultados, los cuales se detallan con más especificidad en la minuta del trabajo final.

El aspecto más relevante que limito la experimentación fue la memoria de la GPU, muchísimos experimentos fallaron pues Pytorch no era capaz de asignar recursos cuando los batches eran grandes (la GPU contaba con 8GB), lo único que se podía lograr era variar arquitectura o resolución y así se hizo, sacándole máximo uso a la GPU con la que contaba y generando resultados satisfactorios que se observan en el reporte y se analizan en la minuta.

También por recomendación del profesor guía se realizaron experimentos en plataformas cloud, utilizando Google Colab para ello y se lograron ejecutar nuevos entrenamientos en una K80 GPU de 12 GB, todos los experimentos igualmente se envían a WandB para su rastreo. En los experimentos con una mejor GPU no se notó mejoría, por ende, se deduce que los problemas están más relacionados a la cantidad de imágenes con las que se dispone, pero que fueron igualmente suficientes para generar un modelo de segmentación aceptable.

- Creación de sitio web.

Para la creación del sitio web, en el avance #1 o PEC 2, se había iniciado la página web con un diseño sencillo y rápido para generar una primera versión que sirviera de prueba de concepto, a partir de esta versión el sitio web fue mejorando en aspectos de usabilidad, portabilidad e interfaz de usuario sin querer ser perfecto pues el objetivo del trabajo final del máster que se planteo es abarcar las diferentes

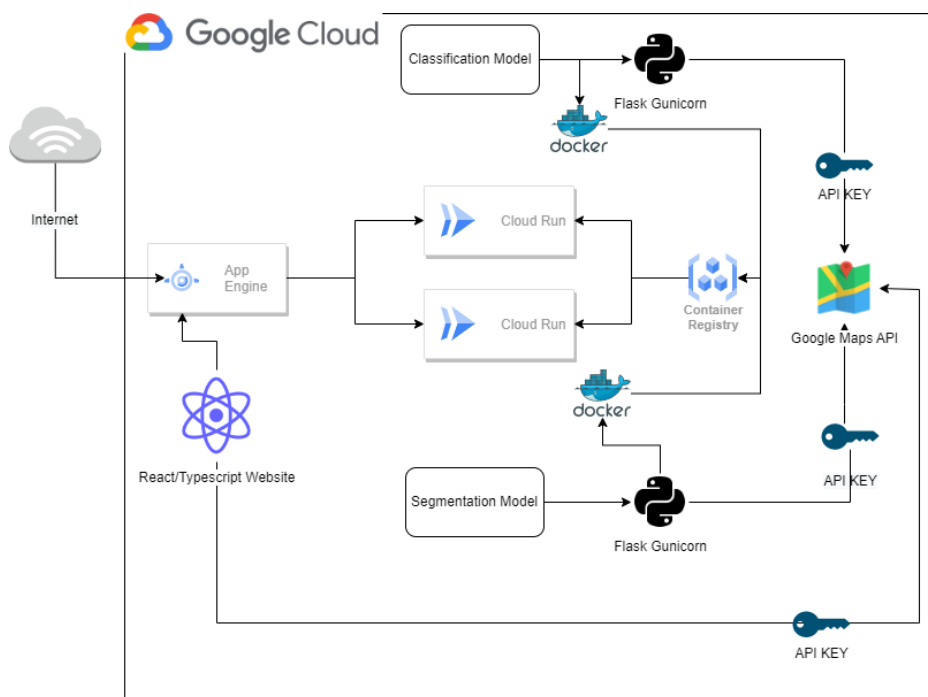
áreas que conlleva la creación de una producto web con funciones de machine learning.

A continuación, se muestra una muestra del sitio web y se incluye nuevamente la URL en su página principal. <https://costa-rica-solar.uc.r.appspot.com/>

En la minuta, se incluirán diferentes imágenes, mostrando el sitio web en sus diferentes versiones para móvil y desktop, algo importante a destacar es que como parte de la creación del sitio web existe un componente de arquitectura software importante y significativo.

Para lanzar el sitio web y basado en experiencias anteriores, se decidió optar por Google cloud platform en lugar de AWS que fue la plataforma utilizada para captura de datos. En términos de usabilidad fue mucho más sencillo de utilizar y se utilizaron diferentes fuentes que se mencionaran en la minuta para lanzar el sitio web.

Figura #1, Arquitectura software,



Nota: Creación propia, utilizando Drawio.

En la figura #1, se observan diferentes componentes, se observa el sitio web basado en React y Typescript y que también utiliza material UI para facilidad en cuanto a aspectos visuales. En lo que corresponde a lanzamiento de los modelos de machine learning, se siguieron varios procesos, entre ellos el lanzamiento de servidores utilizando Flask dado que se utilizó Python para el entrenamiento de los

modelos que posteriormente se lanzan utilizando Docker y se envían al Container Registry de Google cloud, desde el cual se hizo un lanzamiento de servicios “cloud run” los cuales permiten la aplicación funcional en todo momento a un bajo costo (es sobre demanda o llamadas)

Se exploró y se denotó que se puede utilizar un Google Kubernetes Engine, que facilitaría un mejor costo-rendimiento en caso de lanzar un proceso exhaustivo (por ejemplo, un código que permita analizar regiones grandes); se hizo una prueba y es bastante sencillo de migrar pero se decidió no proceder a esta opción, pues a razón de manejo de riesgos en cuanto a recursos, los cuales también incluyen económico, se decidió optar por una arquitectura que redujera costos a sabiendas que puede ser fácilmente escalable, inclusive con cloud run, pero a un costo mayor.

A nivel de funcionamiento, el sitio web utiliza un API Key, con el cual se comunica con Google Maps API, que proporciona acceso al mapa satelital, también se utiliza este API Key en Google Cloud Run, pues se optó por solo capturar coordenadas en el cliente de React, pues es más sencillo y rápida la transferencia de coordenadas únicamente, una vez dadas las coordenadas a Cloud Run, este se encargará de traer la imagen de Google Map y proceder con el análisis.

El análisis cuenta con dos puntos, en el primero se procederá a llamar a la función que se encarga de la clasificación, y basado en el resultado de esta se procederá o no a la ejecución de la segunda función (cloud run). Si el resultado es positivo, el proceso es el mismo, se pasan las coordenadas, se descarga la imagen y se procede a la generación de la máscara de segmentación para su posterior visualización en el cliente.

2.2 Justificación de los cambios en caso necesario

No se denotó ningún cambio al cronograma del proyecto luego de ajuste de este en avance #1.

3. Relación de las actividades realizadas

3.1 Actividades previstas en el plan de trabajo

- Creación de modelo de segmentación de imágenes satelitales para detección de paneles solares
- Refinamiento de modelo de segmentación
- Creación de sitio web

- Actualización de capítulos de minuta.

3.2 Actividades no previstas y realizadas o programas

- Se ha invertido tiempo significativo en el ajuste visual del sitio web a fin de generar un mejor producto y ser de mayor agrado para los usuarios, también se ha optado por realizar ajustes a fin de que se pueda utilizar desde un celular.

4. Relación de las desviaciones en la temporización y acciones de mitigación si procede y actualización del cronograma si procede

Se detectaron complicaciones previamente detectadas durante el análisis de riesgos, sin embargo, debido a la planificación y consideración previa de estas, no hubo repercusiones a nivel de cronograma. Se detalla cuales fueron los puntos del análisis de riesgo que se presentaron y se les aplicó mitigación:

- Métricas de calidad satisfactorias y Uso de recursos computacionales

Se observó que estos dos riesgos estuvieron altamente conectados y por ende para su mitigación se experimentó con variaciones en cuanto a modelo principalmente de segmentación que se aplicaron como parte del objetivo de refinamiento, debido a la limitante de recursos computacionales, se asumió en cierto punto que debido a este factor no sería posible la ejecución de algunos experimentos por lo tanto se procedió a estudiar la posibilidad de ejecutar los procesos en un ambiente diferente al local con el que se trabajaba, pero al considerar la carga de trabajo que esto generaba vs otro de los puntos de mitigación que consiste en aceptar las métricas, se optó por el último punto a fin de continuar con los demás objetivos y reconocer que se obtuvieron métricas muy impresionantes a pesar de la poca cantidad de imágenes de entrenamiento.

- Posible Inclusión de modelo de clasificación.

En este riesgo se detectó, que las variaciones observadas en el avance 1 y cambios propuestos fueron exitosos y se logró la incorporación de un modelo de clasificación que resultó ser altamente preciso en su funcionalidad.

5. Listado de los resultados parciales obtenidos hasta el momento (entregables que se adjuntan)

- Se hace entrega de un repositorio, que contiene todo el código fuente relacionado con la aplicación. Ver anexo #4
 - A. Código fuente de servidores que contienen modelos.
 - B. Código fuente de cliente web basado en React/Typescript
 - C. Código fuente de modelos de segmentación y clasificación.
- Actualización de minuta de trabajo final de máster.

Entregables previos enviados (Avance #1):

- Un set de datos comprimido, que contiene dentro, dos carpetas llamadas test y train, compuesto en su totalidad por alrededor de 120 imágenes.
- Un set de datos comprimido, que contiene dentro, una carpeta llamada noPanel, compuesto por alrededor de 400 imágenes,
- Jupyter Notebooks, que contienen código fuente utilizado para entrenamiento de modelos, en etapa avanzada y de refinamiento o tuning.
- Cronograma y análisis de riesgos actualizado en anexos.

6. Comentarios de vuestro director particular si lo consideran necesario

Se recomendó explorar usar recursos de la UOC para entrenamiento, se tomó la sugerencia, pero en lugar de solicitarlos a la UOC, se utilizaron recursos de Google Cloud Platform.

Igualmente, existieron dudas con respecto a la usabilidad del sitio web, por lo que se incluyeron mejores instrucciones.

7. Anexos

Anexo #1: Análisis de riesgos (Sin variaciones, para Avance 2):

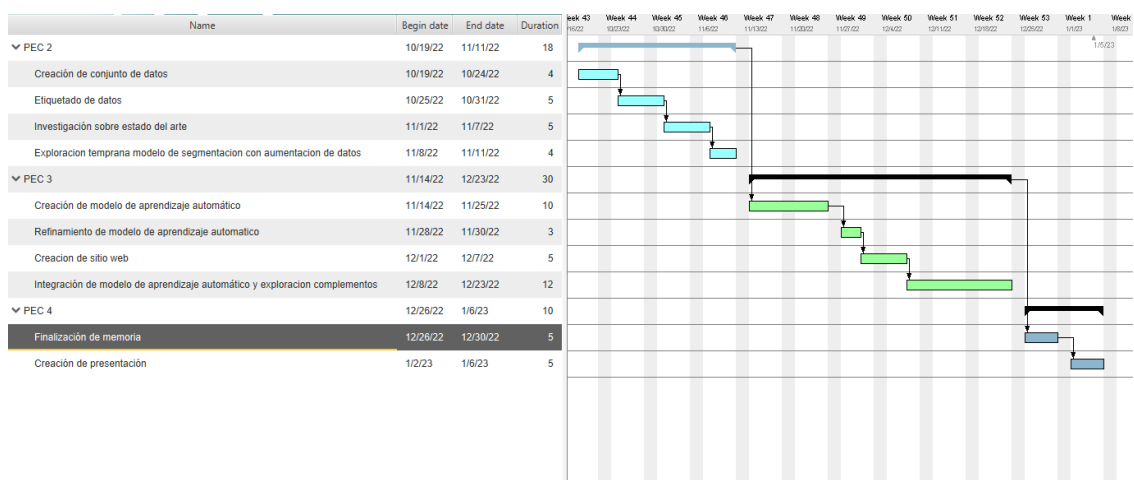
- Métricas de calidad satisfactorias:
 - Factores: Calidad de conjunto de datos recolectado
 - Descripción: Datos recolectados no sean suficientes como para producir buenas métricas en el modelo de aprendizaje automático.

- Nivel: Medio
 - Mitigación: Modificar variables de modelo de aprendizaje automático, expansión de set de datos de forma artificial.
- Imágenes satelitales, en país con clima tropical y en desarrollo.
 - Factores: Popularidad de paneles solares en áreas residenciales
 - Descripción: Se espera que no se puedan llegar a encontrar muchos paneles solares y que existan muchas imágenes, donde cuestiones relacionadas a economía y clima afecten negativamente al modelo de aprendizaje automático, afectando por lo tanto predicciones.
 - Nivel: Medio
 - Mitigación: Expansión de set de datos de forma artificial, delimitar o reconocer variables que afectan negativamente el modelo, por ende, asumir casos donde predicción no sea exitosa.
 - Actualización: Se ha recolectado y se ha encontrado que no muchas edificaciones cuentan con paneles solares, se ha hecho uso de librerías de aumentación de datos y ha resultado exitoso.
- Entendimiento y uso de estado del arte:
 - Factores: Investigación, conocimiento e implementación de las técnicas
 - Descripción: Generar solución que se alinee al estado del arte con respecto al modelo de aprendizaje automático, donde el estado del arte consuma mucho tiempo y sea muy difícil de implementar.
 - Nivel de riesgo: Alto
 - Mitigación: Lectura y exploración de diferentes alternativas y hacer pequeños experimentos en caso de ser necesario.
 - Actualización: Mitigado de forma efectiva, no se espera variaciones altas.
- Uso de recursos computacionales:
 - Factores: Recursos computacionales y propiedades de imágenes.
 - Descripción: Dependiendo de conjunto de datos y calidad de imágenes, tiempo para entrenamiento puede ser significativo y requerir uso de recursos computacionales con GPU, que poseen un costo significativo si se llega a extender por varias horas.
 - Nivel de riesgo: Medio
 - Mitigación: Aceptar posibles repercusiones a nivel de métricas o revisar planes gratuitos u opciones para casos académicos. Explorar posibilidad de generar entrenamiento en computadora propia por múltiples días para así disminuir costo.
- Posible Inclusión de modelo de clasificación.
 - Factores: Recursos computacionales y propiedades de imágenes.
 - Descripción: Estado del arte recomienda la incorporación de modelo de clasificación, aparte de modelo de segmentación.
 - Nivel de riesgo: Alto
 - Mitigación: Captura de imágenes sin paneles solares a fin de dar posibilidad a esta opción, adelanto de actividades de cronograma e inclusión de modelo de clasificación para actividades de segundo PEC en caso de contar con disponibilidad de tiempo.
 - Actualización: Se capturaron imágenes extras que serán parte de entrega parcial de resultados y se planea iniciar modelo de clasificación como parte

de la tarea actualizada llamada: “integración de modelo de aprendizaje automático y exploración complementos”

Anexo #2: Cronograma (Sin variaciones, para Avance 2)

Name	Begin date	End date	Duration
▼ PEC 2	10/19/22	11/11/22	18
Creación de conjunto de datos	10/19/22	10/24/22	4
Etiquetado de datos	10/25/22	10/31/22	5
Investigación sobre estado del arte	11/1/22	11/7/22	5
Exploracion temprana modelo de segmentacion con aumentacion de datos	11/8/22	11/11/22	4
▼ PEC 3	11/14/22	12/23/22	30
Creación de modelo de aprendizaje automático	11/14/22	11/25/22	10
Refinamiento de modelo de aprendizaje automatico	11/28/22	11/30/22	3
Creacion de sitio web	12/1/22	12/7/22	5
Integración de modelo de aprendizaje automático y exploracion complementos	12/8/22	12/23/22	12
▼ PEC 4	12/26/22	1/6/23	10
Finalización de memoria	12/26/22	12/30/22	5
Creación de presentación	1/2/23	1/6/23	5



Anexo #3. Sitio web

<https://costa-rica-solar.uc.r.appspot.com/>

Anexo #4. Repositorio publico con todo el codigo fuente.

<https://github.com/carlosan1708/MasterTFM>

