



**UNIVERSIDAD CENTRAL DEL ECUADOR**



Facultad de Ingeniería en Geología, Minas, Petróleos y  
Ambiental

Escuela de Geología

**SOFTWARE APLICADO A LA GEOLOGÍA**

Descripción automática de Volcanes mediante Redes neuronales  
Convolucionales y Recurrentes.

**DOCENTE:** Ing. Cristian Mejía

**Integrantes:**

Caizaluisa Juan Pablo  
Noguera Dayanara  
Torres Joselyn  
Villacrés María José

**2021**

## **Contenido**

<b>1. INTRODUCCIÓN</b>	<b>3</b>
<b>2. OBJETIVOS</b>	<b>3</b>
<b>2.1. GENERAL</b>	<b>3</b>
<b>2.2. ESPECÍFICOS</b>	<b>3</b>
<b>3. TRABAJOS RELACIONADOS</b>	<b>4</b>
<b>4. RECURSOS COMPUTACIONALES</b>	<b>5</b>
<b>4.1. HARDWARE</b>	<b>5</b>
<b>4.2. SOFTWARE</b>	<b>6</b>
<b>5. METODOLOGÍA</b>	<b>8</b>
<b>5.1. DIAGRAMA DE FLUJO</b>	<b>8</b>
<b>5.2. DATASET</b>	<b>8</b>
<b>5.3. PRE-PROCESAMIENTO</b>	<b>12</b>
<b>5.4. DIVISIÓN</b>	<b>13</b>
<b>5.5. CREACIÓN DEL MODELO</b>	<b>15</b>
<b>5.5.1. Redes Neuronales Convolucionales</b>	<b>15</b>
<b>5.5.2. Redes Neuronales Recurrentes</b>	<b>18</b>
<b>5.6. COMPILACIÓN Y AJUSTE</b>	<b>20</b>
<b>5.7. GRÁFICAS DE PRECISIÓN Y PÉRDIDA</b>	<b>22</b>
<b>5.8. PREDICCIÓN</b>	<b>26</b>
<b>6. CONCLUSIONES Y RECOMENDACIONES</b>	<b>27</b>
<b>7. MODELOS</b>	<b>28</b>
<b>8. REFERENCIAS</b>	<b>28</b>

## **1. INTRODUCCIÓN**

Un Volcán es la expresión superficial de la dinámica interna de la Tierra, puede expulsar material fluido, rocoso y gases. La sucesiva acumulación de este material forma un relieve denominada edificio volcánico; el material puede ser expulsado de manera explosiva (bombas, lapilli, Ceniza y fragmentos juveniles) y no explosiva (lava), dependiendo de las características del magma. (Romero, 2018)

En volcanismo, al ser una rama de la geología que estudia los fenómenos volcánicos, permite conocer la formación y clasificación de volcanes, además de la distribución espacial del material expulsado (Romero, 2018). La importancia de analizar estos fenómenos se debe a que han aumentado los asentamientos humanos en las laderas o zonas cercanas a centros volcánicos, provocando que los habitantes se vuelvan vulnerables ante estos eventos (Gho, 2017), es así que, gracias a los avances tecnológicos se ha logrado dar solución a varios aspectos a través del uso de algoritmos de aprendizaje profundo o Deep Learning (Bagnato, 2018). Cabe recalcar que en la actualidad el uso de algoritmos de aprendizaje profundo en la industria para la optimización de procesos, ha causado gran impacto provocando que profesionales y estudiantes busquen el medio para aprender desde las bases de programación hasta lograr desarrollar algoritmos más avanzados, puesto que, de esa manera pueden mejorar sus habilidades y conseguir mejores propuestas laborales.

Por lo tanto, el desarrollo de pequeños proyectos como el presentado a continuación, permite poner en práctica el conocimiento adquirido durante toda la carrera y así poder reconocer temáticas donde la implementación de algoritmos de aprendizaje profundo sea necesario para mejorar y solventar ciertas incertidumbres. Por ello, el presente trabajo propone el uso de redes neuronales convolucionales y recurrentes, e imágenes satelitales, con el fin de generar una herramienta computacional confiable para caracterizar volcanes, optimizando tiempo y recursos involucrados en un análisis volcanológico convencional que implica el uso de instrumentos costosos, personal calificado y varias visitas de campo; este trabajo a su vez permitirá implementar planes de evacuación y contingencia para resguardar la vida de los habitantes.

## **2. OBJETIVOS**

### **2.1. GENERAL**

Aplicar Redes Neuronales Convolucionales y Recurrentes para la descripción automática de volcanes mediante imágenes satelitales.

### **2.2. ESPECÍFICOS**

- Elaborar un Dataset extenso y confiable de imágenes satelitales obtenidas de diversas fuentes.
- Diseñar e implementar un modelo de Deep Learning basado en redes neuronales y recurrentes para la descripción de volcanes.

- Obtener anotaciones automáticas de características en las imágenes satelitales analizadas y comparar con análisis convencionales.

### 3. TRABAJOS RELACIONADOS

La investigación de estas redes convolucionales y recurrentes juntas no se tienen muchos estudios, pero sin embargo por separado de estas redes se tiene un gran desarrollo en varios ámbitos, las redes convolucionales se tienen una gran aplicación en la clasificación de imágenes, mientras que las redes recurrentes son usadas actualmente en las aplicaciones de redes sociales donde se usan para la predicción de texto, o incluso estas son usadas en los traductores automáticos, ya en la aplicación en conjunto de CNN y RNN son muy escasos, pero se están desarrollando e investigando más ámbitos de aplicación, siendo uno de los principales la medicina donde se han implementado en el reconocimiento de imagen de rayos X o tomografías para la detección de posibles diagnósticos, también se tienen aplicaciones más sencillas siendo la identificación y descripción entre imágenes de gatos o perros, la implementación de estas redes se ven reflejadas en códigos como VGG16 e Inception V3, siendo ejemplos donde se pueden modificar sus bases de datos e implementar para nuevos estudios.

San Martín, C. (2020) realiza una compilación de videos, dichos videos han sido descomprimidos en fotogramas que contienen descripciones de los mismos. En el modelo creado por redes neuronales convolucionales y recurrentes se realiza una descripción de los videos en espacios de la outdoor e indoor que posteriormente serán transmitidos por voz a personas no videntes. El análisis de los fotogramas contiene el mismo principio el cual es, el reconocimiento de patrones en las imágenes que se usará en el proyecto en cuestión.

Loaiza, W. (2020) realiza una compilación de las diferentes técnicas de reconocimiento de objetos que utilizan inteligencia artificial, señala las ventajas, utilidades y las diferentes aplicaciones que tienen, entre estas tenemos las técnicas de redes convolucionales, este documento nos permite evidenciar los métodos existentes y acceder a información concentrada.

Rodríguez y Alesanco (2018), realizan una investigación para la determinación de diagnóstico de autismo en base a la programación y el trabajo en conjunto de las redes neuronales convolucionales y recurrentes, para el desarrollo de su investigación ellos trabajan con un Dataset de alrededor de 1000 imágenes de resonancia magnética, divididas en personas que presentan autismo y otras que no, mientras que las redes recurrentes en base a su patrón de secuencia determinan el diagnóstico de esta enfermedad con caracterización en ciertos índices de autismo.

Heras, D. (2017), realiza la aplicación de inteligencia artificial y análisis de imágenes para la automatización de procesos, mediante la implementación de algoritmos de clasificación automática de imágenes para la clasificación de frutas, los parámetros utilizados son el brillo, color y forma,

las imágenes utilizadas son de 500 píxeles de ancho, utilizaron un Dataset de 100 imágenes, que fueron clasificadas en 4 clases.

Los estudios previos mencionados han servido de base para poder desarrollar el estudio de descripción automática de volcanes, es así que se utilizarán principalmente las redes convolucionales no para clasificación como en los estudios previos, si no que se usarán para poder realizar una identificación y reconocimiento entre volcanes y paisajes, junto a este reconocimiento se usan las redes recurrentes para poder tener una descripción de estas imágenes, teniendo como primera característica identificada y la más importante si la imagen corresponde o no a un volcán, después la descripción será completada con otras características.

## 4. RECURSOS COMPUTACIONALES

### 4.1. HARDWARE

Cuando se trata de entrenar redes neuronales, el tipo de dispositivo que se use condicionará el tiempo de aprendizaje de dicha red. Los componentes más importantes se resumen en la siguiente tabla:

*Tabla 1 Especificaciones del computador*

DISPOSITIVO	DESCRIPCIÓN
<b>PROCESADOR</b>	Intel® Core™ i7-3610QM CPU @2,30 GHz 2301 MHz
<b>MEMORIA INSTALADA (RAM)</b>	8 GB tipo DDR3
<b>DISCO DURO</b>	TOSHIBA External USB 3.0 USB Device memoria 1,81TB
<b>PANTALLA</b>	Intel ® HD Graphics 4000

El computador local fue usado para la recopilación y preprocesamiento de las imágenes satelitales, y ejecución de las líneas de código en un entorno de programación, adicionalmente para un mejor manejo del Dataset, se almacenó en Google Drive (almacenamiento para ligar a Google Colaboratory) y OneDrive (almacenamiento principal) que son los componentes de un servicio integrado ligado a correos electrónicos Gmail y Outlook, respectivamente, que proporcionan un espacio en la nube con capacidad de 15Gb para almacenar, crear, modificar, compartir y acceder a documentos, archivos y carpetas (UPAEP, 2020), por esta razón se consideró el uso de estas 2 aplicaciones con el fin de evitar pérdidas y mejorar la accesibilidad a la información.

## 4.2. SOFTWARE

El Sistema operativo del dispositivo en el que se descargó, proceso, almaceno y permitió el acceso a un entorno de desarrollo para ejecutar las líneas de código, fue Windows 10 Pro de 64-bit, el código se desarrolló en la nube a través de la plataforma Google Colab.

El lenguaje de programación con el que se desarrolló el proyecto es Python, fue creado a principios de los noventa por Guido Van Rossum, es flexivo, sencillo de aprender y tiene grandes aportes para la comunidad ya que permite desarrollar un sin número de sitios web, desarrollo de software y juegos, además permite procesar fácilmente todo tipo de estructuras de datos, tanto numéricos como de texto.

Para el desarrollo del código se usó Google Colaboratory es un entorno gratuito de Jupyter Notebook que no requiere configuración y que se ejecuta completamente en la nube (De La Fuente, 2019). Para poder acceder al entorno se debe tener una cuenta Gmail, se digita en el navegador Google Colaboratory donde se abrirá el menú principal, se crea un nuevo cuaderno (Ilustración 1).

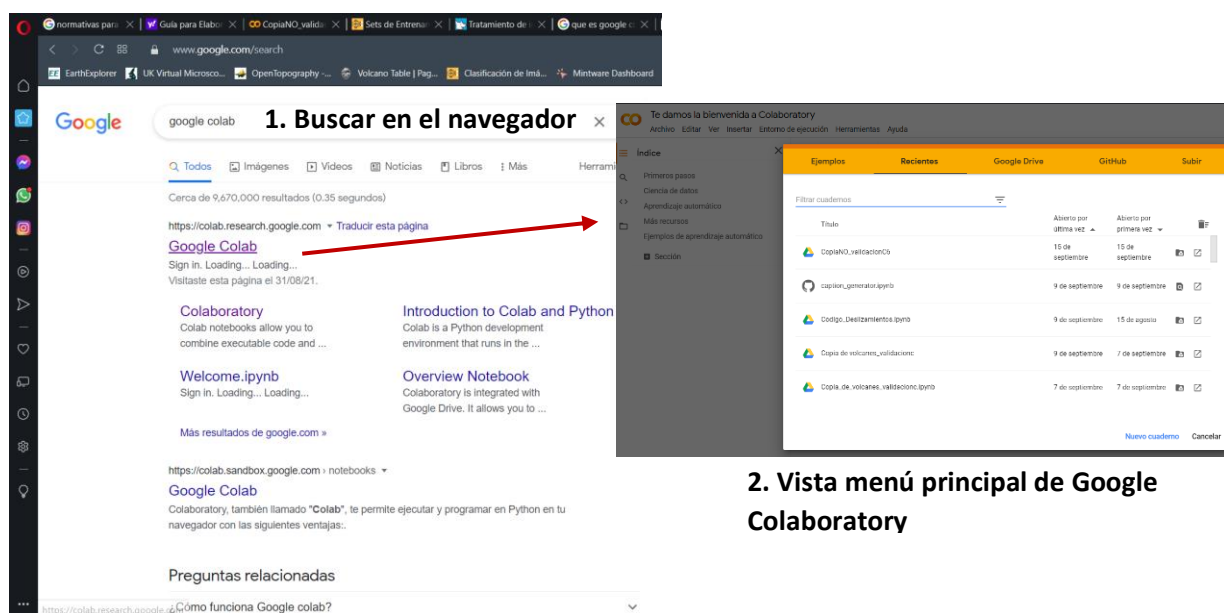
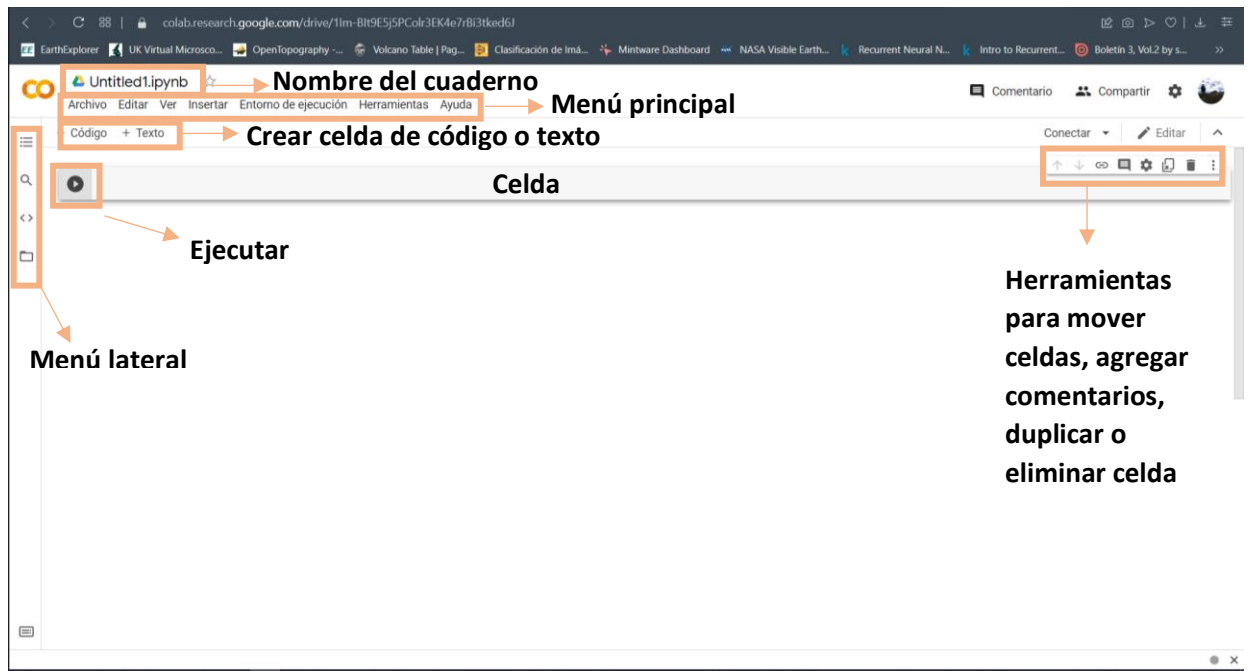


Ilustración 1 Pasos para ingresar a Google Colaboratory y componentes de su interfaz



*Ilustración 1 Pasos para ingresar a Google Colaboratory y componentes de su interfaz*

Las librerías usadas para la realización del código en la descripción automática de los volcanes se detallan en la Tabla 2, las primeras librerías se usan para cargar imágenes y obtener un resumen de la cantidad de imágenes, seguido se tienen librerías ya netamente para la creación de redes convolucionales y recurrentes como el TensorFlow y Keras.

*Tabla 2 Descripción de las principales librerías de Python usadas en el proyecto*





TIPO	NOMBRE	DESCRIPCIÓN
Librerías de Python para visualización	Matplotlib	Generación de gráficos como: series temporales, histogramas, espectros de potencia, diagramas de barras etc.
	Seaborn	Basada en matplotlib, especializada en la visualización de datos estadísticos.
Librerías de Python para cálculo numérico y análisis de datos	NumPy	Proporciona funciones matemáticas, análisis de datos y el intercambio de datos entre distintos algoritmos.
Librerías de Python para Machine Learning	scikit-learn	Presenta técnicas de aprendizaje supervisado y no supervisado, utilizado en clasificación y regresión.
Librerías de Python para Deep Learning	TensorFlow	Usada para realizar cálculos numéricos mediante diagramas de flujo de datos
	Keras	Contiene un sin número de funciones para el procesamiento de imágenes, redes convolucionales, Deep learning en general.

## 5. METODOLOGÍA

### 5.1. DIAGRAMA DE FLUJO

Para la elaboración del diagrama de flujo que resumirá la metodología que se aplicó para la detección y obtención de anotaciones automáticas de volcanes, se consideró la simbología dictada por la norma ANSI (American National Standard Institute), que se muestran en la siguiente tabla:

Tabla 3 Significado de la simbología según norma ANSI

Símbolo	Significado	Símbolo	Significado
	<b>Inicio/Fin</b> Inicio y final del diagrama		<b>Decisión</b> Indica posibles alternativas
	<b>Proceso/Operación</b> Representa un procedimiento		<b>Database</b> Almacenamiento de información

Entonces se tiene como resultado el diagrama de flujo de la Ilustración 2. Cabe recalcar que el símbolo de Decisión asociado a la precisión de los modelos, se considera aceptable cuando la descripción de una imagen arroja su respectiva descripción completa, moderadamente aceptable si arroja una descripción que se relacione con la categoría (Paisaje o Volcán), es decir que la descripción arrojada no corresponda con la imagen pero que por lo menos incluya la palabra volcán o paisaje, y no aceptable si la descripción no tiene relacionan alguna con la imagen mostrada.

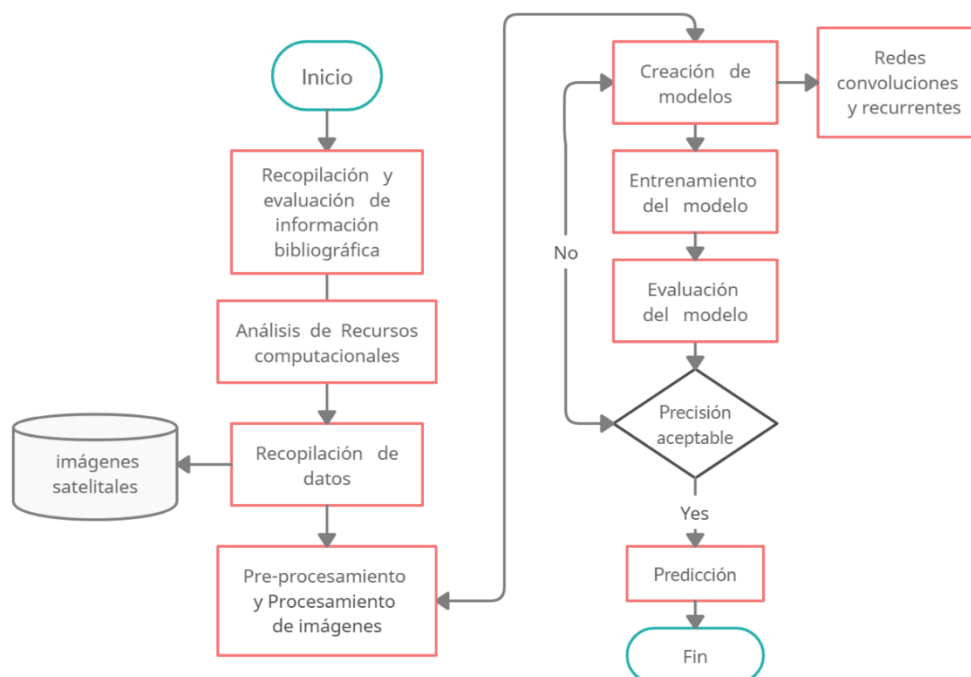


Ilustración 2 Diagrama de flujo de la metodología usada en descripción automática de Volcanes. Según norma ANSI



## 5.2. DATASET

La creación del Dataset consiste en la recopilación de imágenes correspondientes a volcanes y paisajes. Las imágenes de volcanes se obtuvieron de Google Earth Pro, cada imagen satelital cuenta con una resolución de 1023\*728 pixeles en formato .jpg, en total se descargaron 800 imágenes, los pasos para descargar las imágenes son:

1. Seleccionar la zona deseada, en este caso el volcán Altar.
2. Se va a la opción guardar imagen que se encuentra en la barra de herramientas.



Ilustración 3 Capturas de Google Earth Pro para evidenciar los pasos para descargar imágenes satelitales

3. Se desactivan las opciones de Leyenda, Escala, Brújula, Título y Leyenda.
4. Se selecciona la resolución de imagen.

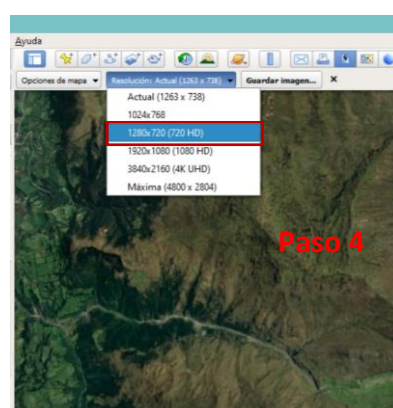
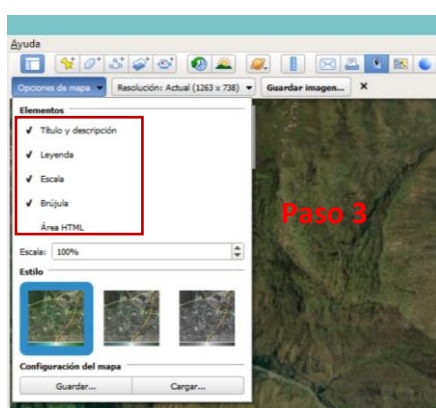
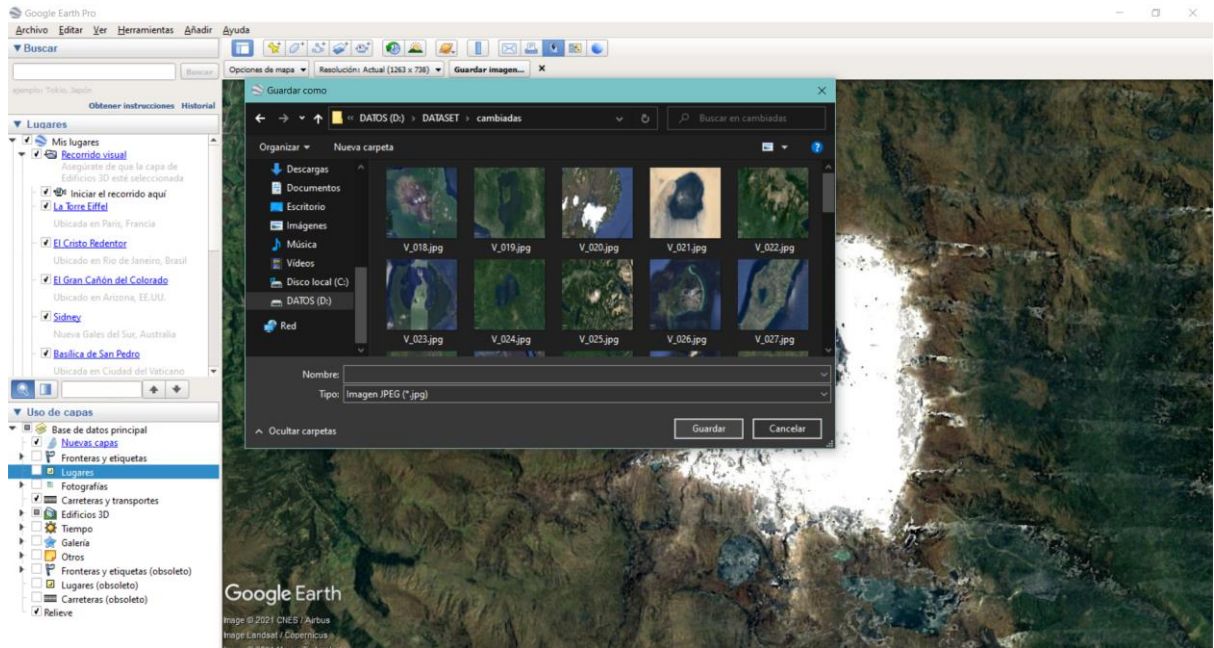


Ilustración 4 Capturas de Google Earth Pro para evidenciar los pasos para descargar imágenes satelitales

5. Finalmente se da click en Guardar imagen, se abre el explorador de archivos y se asigna nombre.



*Ilustración 5 Capturas de Google Earth Pro para evidenciar los pasos para descargar imágenes satelitales*

El Dataset de paisajes se descargó de la Web, la carpeta contiene 800 imágenes en formato .jpg con resolución de 1023\*728 pixeles.

Para el almacenamiento de estas imágenes se ha creado 2 carpetas en OneDrive (Ilustración 6, a) la primera llamada Morfología y la segunda paisajes, cada una de estas contiene alrededor de 800 imágenes por carpeta (1600 imágenes en total). Mientras que en Google Drive se utilizó una carpeta llamada DATASET que contiene a su vez 4 subcarpetas divididas en Dataset completo, textos (descripciones y códigos), test y train que contienen las imágenes que serán utilizadas en el código (Ilustración 6, b).

a)

Mis archivos > PROYECTO\_SOFTWARE > DATASET <sup>Ⓜ</sup>

	Nombre ▾	Modificado ▾
	MORFOLOGIA	1 de agosto
	PAISAJES	24 de julio
	DESCRIPCION.xlsx	Hace 6 días


b)










Archivos

- drive
  - MyDrive
    - Colab Notebooks
    - Convolucion
    - DATASET**
      - Completo
      - Textos
        - Ctest.txt
        - Ctrain.txt
        - DataTotal.txt
        - TEST.txt
        - TRAIN.txt
      - test
        - PAISAJES
        - VOLCANES
      - train
        - PAISAJES
        - VOLCANES

*Ilustración 6 Carpetas para el almacenamiento principal en a) OneDrive y b) Google Drive*

La organización del Dataset se basó en la codificación de cada imagen con las siguientes siglas: morfología con V seguida de la primera letra del tipo de morfología por ejemplo V\_n\_001 que significa volcán y paisaje con la sigla P\_001, seguido de una cifra de 3 dígitos, como se muestra en la Ilustración 7.

Mis archivos > PROYECTO\_SOFTWARE > Dataset completo 

 Nombre ▾	Modificado ▾
 V_l_099.jpg	5 de septiembre
 V_l_100.jpg	5 de septiembre
 V_l_057.jpg	5 de septiembre
 V_l_058.jpg	5 de septiembre
 V_l_093.jpg	5 de septiembre
 V_l_021.jpg	5 de septiembre
 V_l_074.jpg	5 de septiembre
 V_l_002.jpg	5 de septiembre

*Ilustración 7 Ejemplificación de la codificación asignada a las imágenes.*

Finalmente se creó un archivo de extensión CSV, donde consta la descripción de los principales parámetros para que la imagen sea considerada un volcán, los cuales son: morfología (estratovolcán, cono, cráter), presencia o ausencia de nieve, cobertura vegetal presencia de flujos de lava, poblados cercanos y actividad; fue realizada manualmente por cada integrante del grupo y registrada inicialmente en un documento Excel. Esta base consta de dos filas, la primera con código y la segunda con descripción. Las descripciones de Volcanes son más extensas que la de Paisajes debido a los parámetros usados para su descripción, puesto que la descripción de paisajes consta de tres parámetros, mientras que la de volcanes con seis parámetros, esto debido a que el objetivo de estudio se enfoca en la descripción de volcanes.

Para un mejor manejo de la base, se han suprimido signos de puntuación y palabras monosílabas dentro de la descripción ya que según la bibliografía encontrada esto podría ocasionar que el código arroje un error. (Ilustración 8)

V_789	estratovolcán presenta drenajes sin actividad con cobertura vegetal	Descripción para volcanes
V_790	estratovolcán presenta drenajes con actividad con cobertura vegetal	
V_791	estratovolcán presenta drenajes sin actividad con cobertura vegetal	
V_792	estratovolcán presenta drenajes sin actividad sin cobertura vegetal	
V_793	estratovolcán presenta drenajes sin actividad sin cobertura vegetal	
V_794	estratovolcán no presenta drenajes sin actividad sin cobertura vegetal	
V_795	estratovolcán no presenta drenajes sin actividad con cobertura vegetal	
V_796	estratovolcán no presenta drenajes sin actividad sin cobertura vegetal	
V_797	estratovolcán no presenta drenajes sin actividad con cobertura vegetal	
V_798	estratovolcán presenta drenajes sin actividad sin cobertura vegetal	
V_799	estratovolcán presenta drenajes sin actividad con cobertura vegetal	Descripción para paisajes
V_800	estratovolcán presenta drenajes sin actividad con cobertura vegetal	
P_001	no es volcán paisaje montañas verdes de alta pendiente	
P_002	no es volcán paisaje montañas verdes moderada pendiente	
P_003	no es volcán paisaje parque con árboles	
P_004	no es volcán paisaje zona de playa con mar	
P_005	no es volcán paisaje montaña con árboles	
P_006	no es volcán paisaje llanura verde con arboles	
P_007	no es volcán paisaje camino en una llanura café	
P_008	no es volcán paisaje llanura con presencia de agua	
P_009	no es volcán paisaje pajonal con mar	
P_010	no es volcán paisaje llanura con animales silvestres	
P_011	no es volcán paisaje laguna	
P_012	no es volcán paisaje llanura café	

Ilustración 8 Ejemplificación de la descripción asignada a cada imagen.

Link de acceso Dataset:

[https://drive.google.com/drive/folders/12MICpeTUG\\_14qIJYUDUJG0-iK04uJSiT?usp=sharing](https://drive.google.com/drive/folders/12MICpeTUG_14qIJYUDUJG0-iK04uJSiT?usp=sharing)

### 5.3. PRE-PROCESAMIENTO

El preprocesamiento de las imágenes satelitales se llevó a cabo en el software GIMP 2.14, donde se redujo el tamaño de cada imagen correspondiente a volcanes para normalizar la base de datos a un tamaño de 500\*500 pixeles y con una profundidad de RGB. (Ilustración 9)

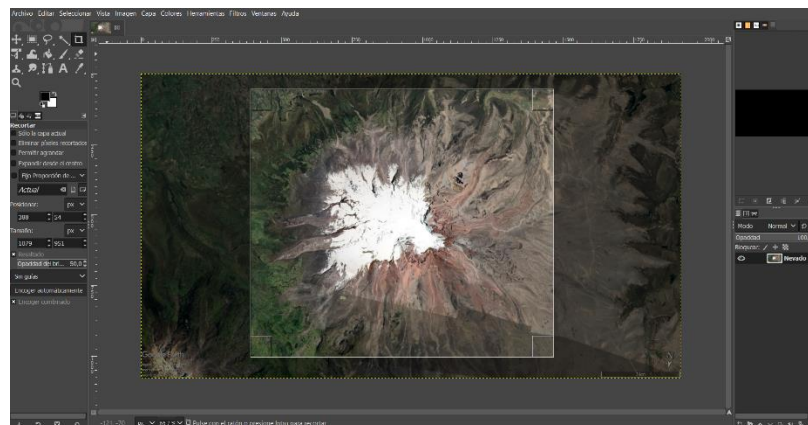


Ilustración 9 Captura de pantalla del preprocesamiento de imágenes

En el cuaderno de Google Colab, posterior a la importación de librerías, vinculación de directorios y gráficas del Dataset, se implementa una línea de código dedicada al re-escalamiento de las imágenes y el uso la técnica de Data augmentation mediante la función ImageDataGenerator que pertenece a la librería keras, este proceso es necesario debido a que el procesar imágenes con resoluciones de 500x500 requiere de mayor memoria.



Para el Re-escalamiento el método más usado consiste en dividir los valores de pixel para 255, obteniendo imágenes con un valor entre 0 y 1, permitiendo que el programa pueda trabajar de una manera más eficiente. Por otro lado, ImageDataGenerator divide el proceso de entrenamiento en bloques (batches) de menor tamaño de imágenes y adicionalmente, considerando que el número de imágenes del Dataset original es relativamente pequeño, la técnica de Data augmentation permite aumentar el número de imágenes originales con las funciones shear, zoom y flip. Cabe recalcar esta técnica solo se usa en las imágenes de entrenamiento.

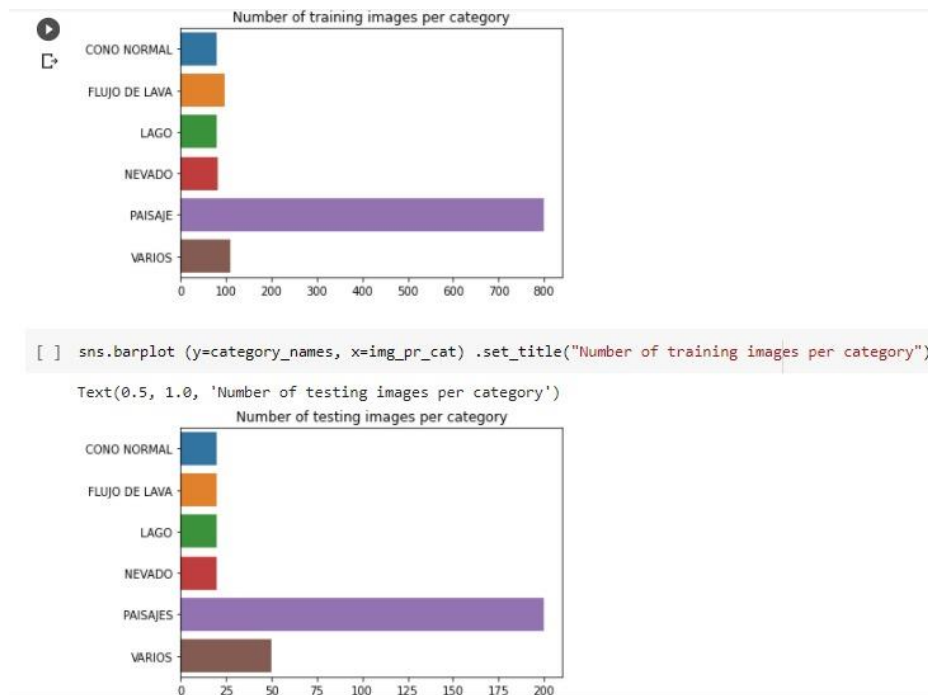
#### **5.4. DIVISIÓN**

La división de Dataset se fundamenta en la necesidad de verificar el funcionamiento de un modelo y evitar un overfitting (sobreajuste) o un underfitting (subajuste), ya que, los algoritmos intentan definir patrones de un Dataset que permitan predecir un resultado, pero usando un Dataset diferente al que se usó para el aprendizaje del modelo, es decir, si el Dataset que uso para el aprendizaje de mi modelo contiene solo imágenes de volcanes tipo cono y para verificar uso una imagen de volcán tipo cráter, la predicción falla porque en el entrenamiento no incluí imágenes de volcanes tipo cráter (Underfitting), por lo tanto no cuento con los parámetros para predecir; ahora si el Dataset contiene imágenes con todos los tipos de volcanes, pero todas las imágenes tienen cobertura vegetal y para verificar el modelo ingreso una imagen pero sin cobertura vegetal, la predicción falla porque a pesar de cumplir con los parámetros de volcán el hecho de que no presente cobertura vegetal lo desecha ya que en su aprendizaje esa característica no existe (Overfitting). Por esta razón el Dataset siempre debe ser variado, robusto y contener todos los parámetros existentes relacionados al objetivo de estudio.

Bajo este contexto, para evitar estas complicaciones se recomienda dividir el Dataset en entrenamiento y prueba. El Dataset de entrenamiento son usados para entrenar el modelo, de esto dependerá la calidad de predicción, puesto que el aprendizaje es directamente proporcional a la calidad de los datos. El Dataset de prueba son los datos que se guardan para comprobar si el modelo predice correctamente.

La división del Dataset se realizó manualmente, la selección fue aleatoria y se usó dos carpetas: entrenamiento y prueba; se dividió el 80% de imágenes para entrenamiento y el 20% restante para la prueba final de la red que permitirán verificar la eficacia y certeza de la red programada para la descripción automática de volcanes. Estos porcentajes fueron seleccionados en base a la bibliografía, puesto que se consideran los más óptimos.

Como se puede observar en la Ilustración 10 el Dataset estaba subdividido en 6 categorías y la categoría Paisajes era la que contenía mayor cantidad de imágenes.



*Ilustración 10 Gráfico de barras de cada carpeta del Dataset en Google Colab*

Al momento de realizar la predicción en el modelo, se evidencio errores debido a la manera de categorizar, esto como consecuencia de que los volcanes presentan más de una característica entre ellos, por ejemplo, un volcán tipo cono puede ser también un nevado, por lo tanto, puede ser clasificado de las dos formas, por este motivo se optó por equilibrar el Dataset y evitar complicaciones, para esto se reorganizó las imágenes de una manera más sencilla, las correcciones realizadas fueron:

- Agrupar solo dos carpetas con nombres de: Volcanes y Paisajes
- Cambiar el código de cada fotografía y en la descripción de las imágenes. (Volcanes: V\_001 – V\_800; Paisajes: P\_001 – P\_800)
- Incrementar el Dataset en cada categoría cargada en la nube
- Comprobación de que se tengan 1600 descripciones en el archivo con extensión .txt

Al finalizar todos estos cambios se tiene una mejor organización del Dataset, descartando las posibles complicaciones en el desarrollo del código para las redes recurrentes y las descripciones en .txt. En la Ilustración 11 se puede observar los gráficos en barra corregidos, mostrando la división para entrenamiento y prueba donde los porcentajes correspondientes son: para el entrenamiento (train) el 80% de todo el Dataset (1200 imágenes), mientras que para la prueba contiene el 20% del Dataset (320 imágenes).



Ilustración 11 Gráfico de barras del Dataset. (Nueva Dataset) Google Colab.

## 5.5. CREACIÓN DEL MODELO

El modelo fue creado paso a paso en base a recopilación bibliográfica (personalizada), tomando en consideración para las redes convolucionales modelos de predicción y para las redes recurrentes el código caption\_generator.ipynb, tomados de la plataforma GitHub.

### 5.5.1.Redes Neuronales Convolucionales

Las redes neuronales convolucionales es un algoritmo de Deep Learning que está diseñado para trabajar con imágenes, tomando estas como input, asignándole importancias (pesos) a ciertos elementos en la imagen para así poder diferenciar unos de otros. Este es uno de los principales algoritmos que ha contribuido en el desarrollo y perfeccionamiento del campo de Visión por computadora.

Para el código de las redes recurrentes se han importado otras librerías que nos ayudan a generar las convoluciones y sus parámetros, en base a un kernel de (3\*3), stride=1 y padding=1, con la generación de primeros 8 mapas de características.

Para contar con un panorama claro la función que cumple el hiperparámetro kernel conecta cada neurona de la capa oculta con la neurona que le corresponda de la capa de entrada; strides indica el avance de las convoluciones, es decir el número de pasos en que se mueve la ventana de kernel; padding agrega un borde a la imagen de entrada para que

la imagen de salida tenga las mismas dimensiones; Relu es una función de activación que activa un solo nodo si la entrada está por encima de un umbral; Maxpooling es una función que reduce el tamaño de la próxima capa de neuronas pero manteniendo las características más importantes que detectó cada filtro; flatten es una función que devuelve una matriz unidimensional convirtiendo cada capa en una entrada; dense son capas que conectan cada neurona con todas las salidas de la capa anterior; softmax es una función de activación que devuelve una distribución de probabilidad sobre clases de salida excluyente.

Con estos hiperparámetros y funciones se realizaron 2 modelos de redes convolucionales, la primera solo se hace con 3 convolucionales y para 6 convoluciones se tiene el sumario en la Ilustración 12, y una visualización gráfica del proceso de convolución en las Ilustraciones 13 y 14.

#### a) Modelo de 3 convoluciones

Model: "sequential\_1"

Layer (type)	Output Shape	Param #
conv2d_3 (Conv2D)	(None, 500, 500, 8)	224
max_pooling2d_3 (MaxPooling2)	(None, 250, 250, 8)	0
conv2d_4 (Conv2D)	(None, 250, 250, 16)	1168
max_pooling2d_4 (MaxPooling2)	(None, 125, 125, 16)	0
conv2d_5 (Conv2D)	(None, 125, 125, 32)	4640
max_pooling2d_5 (MaxPooling2)	(None, 62, 62, 32)	0
flatten_1 (Flatten)	(None, 123008)	0
dense_2 (Dense)	(None, 32)	3936288
dense_3 (Dense)	(None, 2)	66

Total params: 3,942,386  
 Trainable params: 3,942,386  
 Non-trainable params: 0

#### b) Modelo de 6 convoluciones

Model: "sequential\_1"

Layer (type)	Output Shape	Param #
conv2d_6 (Conv2D)	(None, 500, 500, 8)	224
max_pooling2d_6 (MaxPooling2)	(None, 250, 250, 8)	0
conv2d_7 (Conv2D)	(None, 250, 250, 16)	1168
max_pooling2d_7 (MaxPooling2)	(None, 125, 125, 16)	0
conv2d_8 (Conv2D)	(None, 125, 125, 32)	4640
max_pooling2d_8 (MaxPooling2)	(None, 62, 62, 32)	0
conv2d_9 (Conv2D)	(None, 62, 62, 64)	18496
max_pooling2d_9 (MaxPooling2)	(None, 31, 31, 64)	0
conv2d_10 (Conv2D)	(None, 31, 31, 128)	73856
max_pooling2d_10 (MaxPooling2)	(None, 15, 15, 128)	0
conv2d_11 (Conv2D)	(None, 15, 15, 256)	295168
max_pooling2d_11 (MaxPooling2)	(None, 7, 7, 256)	0
flatten_1 (Flatten)	(None, 12544)	0
dense_2 (Dense)	(None, 256)	3211520
dense_3 (Dense)	(None, 2)	514

Total params: 3,605,586  
 Trainable params: 3,605,586  
 Non-trainable params: 0

Ilustración 12 Sumario de a) 3 y b) 6 convoluciones

Para una mayor comprensión del cálculo de parámetros se desglosó paso a paso en la Tabla 4. A simple vista se observa que el modelo de 3 convoluciones cuenta con más parámetros que el modelo de 6 convoluciones, esto es debido a que en cada convolución se extrae las características más relevantes de la imagen, por lo tanto se reducen paulatinamente los parámetros analizados ya que excluyen parámetros repetitivos o innecesarios, esto se evidencia en dimensiones de la última capa y el valor de kernel; el modelo de 3 convoluciones finaliza con una capa de 62x62 y un kernel de 32 dando como resultado 123008 parámetros de entrada, mientras que en el modelo de 6 convoluciones la última capa es de 7x7 y un kernel de 256 dando como resultado 12544 parámetros de entrada.



Tabla 4 Calculo de los parámetros en cada convolución realizada

CONVOLUCIÓN	CÁLCULO DE PARAMETROS	
<b>1 era Convolución</b>	Kerner: 3*3 Prof: 8 Bias: 8	#param= kerner*prof+bias= #param #param= (3*3) *8+8=80 param
<b>2 da Convolución</b>	Kerner: 3*3 Prof: 16 Bias: 16	#param= kerner*prof+bias= #param #param= (3*3) *8*16+16= 1168 param
<b>3 era Convolución</b>	Kerner: 3*3 Prof: 32 Bias: 32	#param= kerner*prof+bias= #param #param= (3*3)*16 *32+32=4640 param
<b>4ta da Convolución</b>	Kerner: 3*3 Prof: 64 Bias: 64	#param= kerner*prof+bias= #param #param= (3*3) *32*64+64= 18496 param
<b>5ta Convolución</b>	Kerner: 3*3 Prof: 128 Bias: 128	#param= kerner*prof+bias= #param #param= (3*3) *64*128+128= 73856 param
<b>6ta Convolución</b>	Kerner: 3*3 Prof: 256 Bias: 256	#param= kerner*prof+bias= #param #param= (3*3)*128*256+256= 295168 param

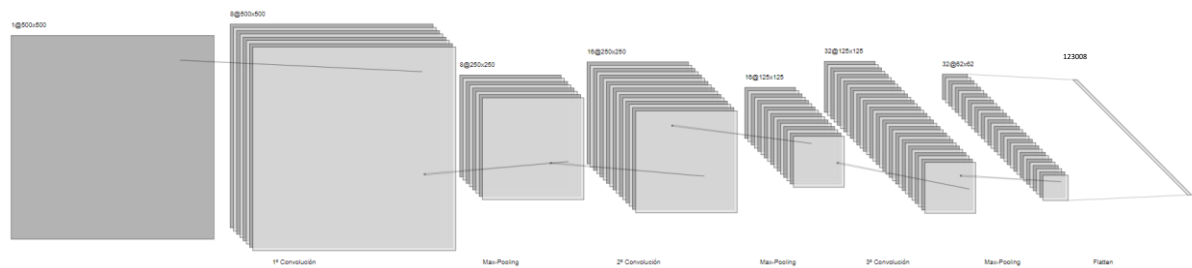
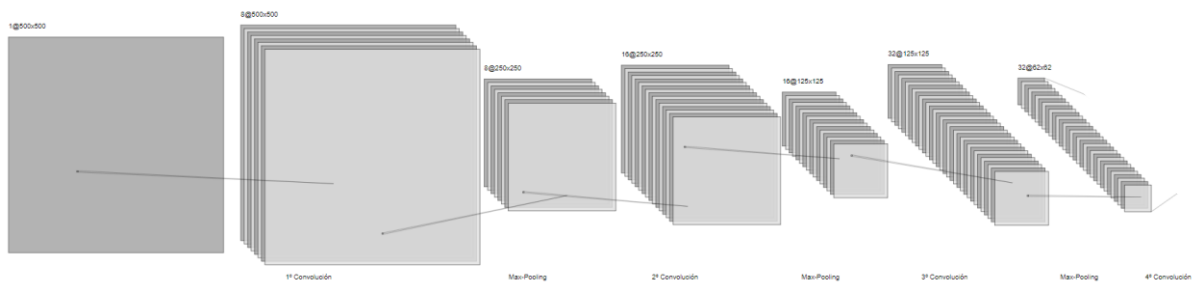
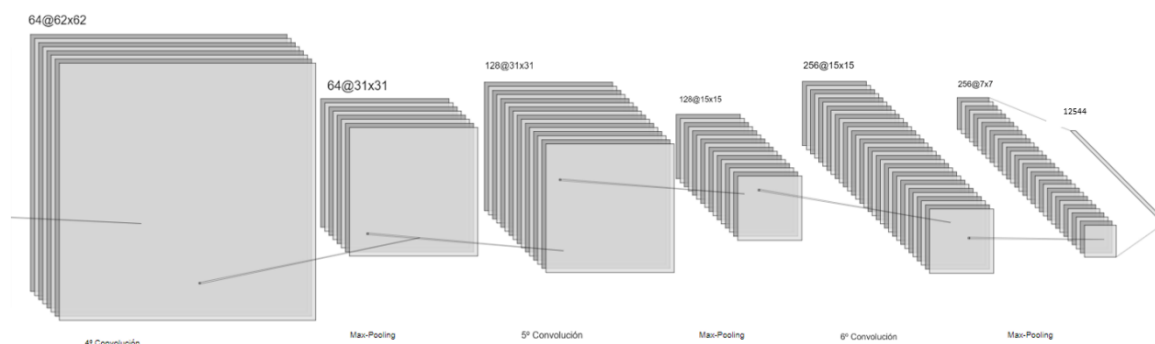


Ilustración 13 Modelo de 3 convoluciones





*Ilustración 14 Modelo de 6ta convolución.*

### 5.5.2. Redes Neuronales Recurrentes

Las redes recurrentes son capaces de procesar cualquier tipo de secuencias de datos, además pueden generar nuevas secuencias. Es usado en la predicción de clima, procesamiento de audio y video y el procesamiento de texto.

En nuestro caso fueron utilizadas para identificar y generar descripciones automatizadas de volcanes y paisajes, dentro del vocabulario se obtuvieron 164 palabras de 1600 descripciones de imágenes ingresadas, los datos fueron ingresados de forma progresiva para evitar la saturación de la memoria. Los datos fueron almacenados en 256 neuronas para posteriormente ingresar en el modelo de la red neuronal recurrente LSTM, para estos después ser utilizados en el modelo final. (Ilustración 15)

Las LSTM son un tipo especial de redes recurrentes. La característica principal de las redes recurrentes es que la información puede persistir introduciendo bucles en el diagrama de la red, por lo que, básicamente, pueden «recordar» estados previos y utilizar esta información para decidir cuál será el siguiente. Esta característica las hace muy adecuadas para manejar series cronológicas. Mientras las redes recurrentes estándar pueden modelar dependencias a corto plazo (es decir, relaciones cercanas en la serie cronológica), las LSTM pueden aprender dependencias largas, por lo que se podría decir que tienen una «memoria» a más largo plazo. (Garzón, 2018)

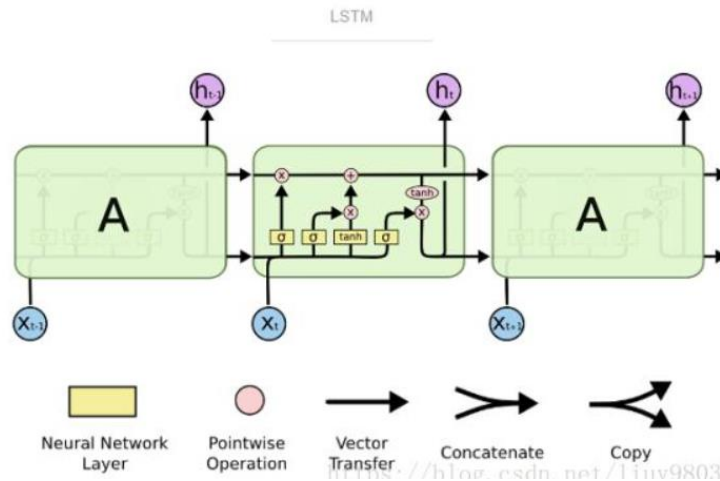


Ilustración 15 Arquitectura de las redes LSTM.

Se prosiguió a realizar dos modelos con iguales datos relacionados a las distintas arquitecturas es decir uno para el modelo previo de 3 convoluciones y otro para el de 6 convoluciones, teniendo así los parámetros e hiperparámetros de los modelos en sus respectivos sumarios.

a) Modelo de 3 convoluciones

b) Modelo de 6 convoluciones

Model: "model_3"				Model: "model_3"			
Layer (type)	Output Shape	Param #	Connected to	Layer (type)	Output Shape	Param #	Connected to
input_6 (InputLayer)	[(None, 16)]	0		input_2 (InputLayer)	[(None, 16)]	0	
input_5 (InputLayer)	[(None, 123008)]	0		input_1 (InputLayer)	[(None, 12544)]	0	
embedding_2 (Embedding)	(None, 16, 256)	39424	input_6[0][0]	embedding (Embedding)	(None, 16, 256)	39424	input_2[0][0]
dropout_4 (Dropout)	(None, 123008)	0	input_5[0][0]	dropout (Dropout)	(None, 12544)	0	input_1[0][0]
dropout_5 (Dropout)	(None, 16, 256)	0	embedding_2[0][0]	dropout_1 (Dropout)	(None, 16, 256)	0	embedding[0][0]
dense_8 (Dense)	(None, 256)	31490304	dropout_4[0][0]	dense (Dense)	(None, 256)	3211520	dropout[0][0]
lstm_2 (LSTM)	(None, 256)	525312	dropout_5[0][0]	lstm (LSTM)	(None, 256)	525312	dropout_1[0][0]
add_2 (Add)	(None, 256)	0	dense_8[0][0] lstm_2[0][0]	add (Add)	(None, 256)	0	dense[0][0] lstm[0][0]
dense_9 (Dense)	(None, 256)	65792	add_2[0][0]	dense_1 (Dense)	(None, 256)	65792	add[0][0]
dense_10 (Dense)	(None, 154)	39578	dense_9[0][0]	dense_2 (Dense)	(None, 154)	39578	dense_1[0][0]
Total params: 32,160,410 Trainable params: 32,160,410 Non-trainable params: 0				Total params: 3,881,626 Trainable params: 3,881,626 Non-trainable params: 0			

Ilustración 1616 Parámetros e Hiperparámetros de a) modelo de 3 convoluciones y b) modelo de 6 convoluciones

### 5.5.2.1. Diagrama de la arquitectura

Un diagrama de arquitectura es una representación esquemática que permite transmitir, mostrar y aclarar el desarrollo de cada proceso involucrado en el modelo. En la ilustración 16, se ilustra tanto el proceso que sigue la red recurrente como la convolucional, en la parte superior se presenta la red recurrente desde el input que consiste en la introducción de los datos, pasa al embedding el cual guarda los datos de longitud de la descripción y numero de filtros para pasarlos al dropout y la red LSTM, la salida pasa a las capas

densas, mientras que en la parte inferior se tiene la parte convolucional, el modelo inicia con un input o ingreso de las imágenes o nuestro dataset, los modelos presentan una variación en el número de convoluciones, el modelo de la ilustración presenta las 6 convoluciones con los respectivos maxpooling, posteriormente se realiza un flatten con el número de características que también es usado en el dropout de la red recurrente. El modelo decodificador se encargará de unir las características que se obtuvieron en la secuencia (256 capas densas) y por último se concatena las imágenes con la secuencia y se almacenan en una variable condicionada por las funciones While y For.

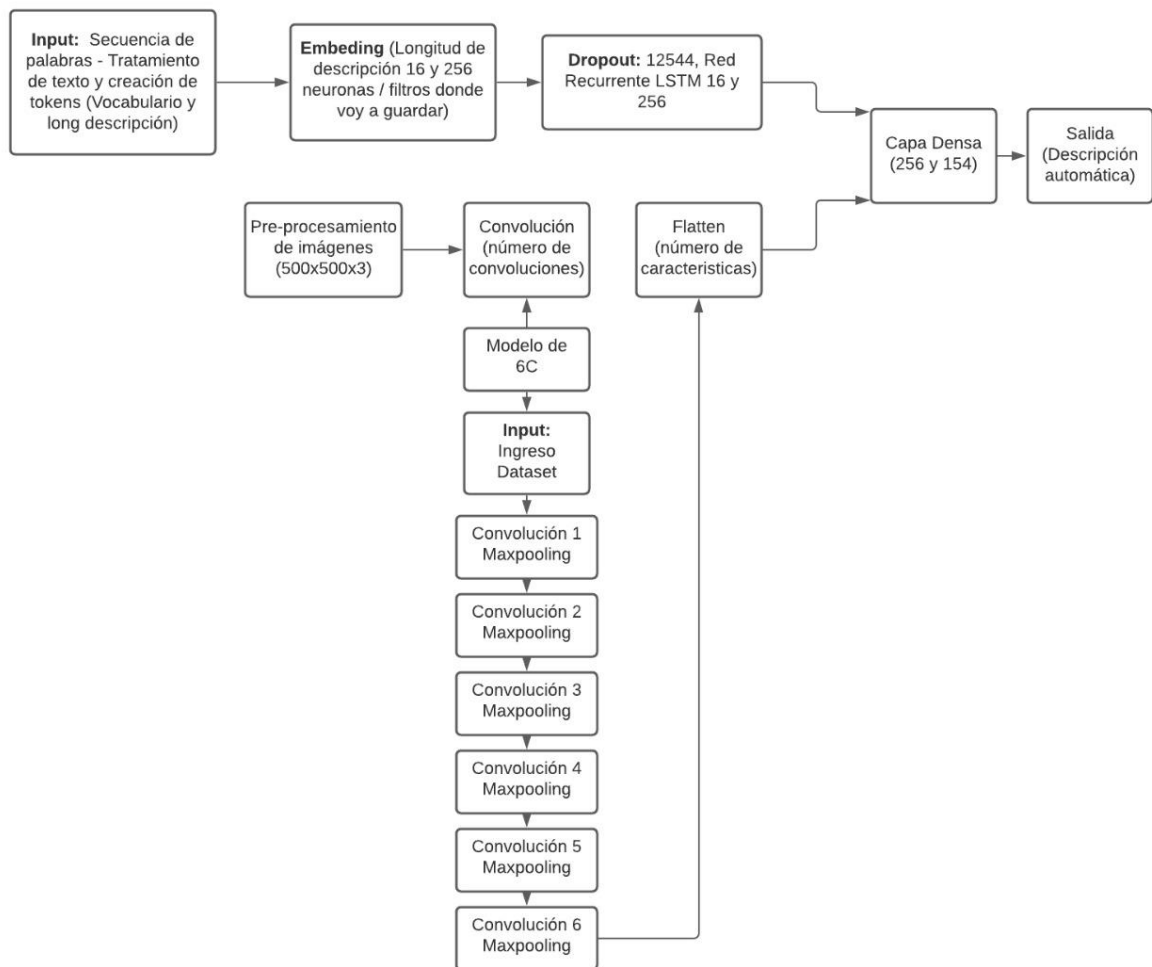


Ilustración 17 Diagrama de arquitectura de la Red neuronal Recurrente

## 5.6. COMPILACIÓN Y AJUSTE

Para la compilación de los modelos convolucionales se utilizó la métrica Accuracy, la precisión es la fracción de predicciones que nuestro modelo ha acertado. Mientras este valor se Accuracy tiene que ser lo más próximo al valor de la unidad. Para las tres convolucionales

se tiene un valor de 0.94 (Ilustración 18, a), y para las seis convolucionales se tiene un valor más próximo a la 1 de 0.97 (Ilustración 18, b).

Para el modelo se utilizaron 20 épocas y un batch\_size de 32, las diferencias entre los dos modelos de convolución fueron el tiempo de entrenamiento mostrado en la tabla 4 y el valor de val\_accuracy que guarda el modelo final.

a)

```
Epoch 00017: val_accuracy did not improve from 0.94687
Epoch 18/20
40/40 [=====] - 77s 2s/step - loss: 0.1018 - accuracy: 0.9664 - val_loss: 0.1625 - val_accuracy: 0.9469

Epoch 00018: val_accuracy did not improve from 0.94687
Epoch 19/20
40/40 [=====] - 77s 2s/step - loss: 0.0672 - accuracy: 0.9758 - val_loss: 0.2197 - val_accuracy: 0.9375

Epoch 00019: val_accuracy did not improve from 0.94687
Epoch 20/20
40/40 [=====] - 77s 2s/step - loss: 0.0694 - accuracy: 0.9750 - val_loss: 0.1364 - val_accuracy: 0.9438

Epoch 00020: val_accuracy did not improve from 0.94687
```

b)

```
Epoch 00017: val_accuracy improved from 0.96250 to 0.96562, saving model to volcanes.h5
Epoch 17/20
40/40 [=====] - 276s 7s/step - loss: 0.0425 - accuracy: 0.9859 - val_loss: 0.1865 - val_accuracy: 0.9344

Epoch 00018: val_accuracy did not improve from 0.96562
Epoch 18/20
40/40 [=====] - 274s 7s/step - loss: 0.0527 - accuracy: 0.9789 - val_loss: 0.1033 - val_accuracy: 0.9750

Epoch 00019: val_accuracy improved from 0.96562 to 0.97500, saving model to volcanes.h5
Epoch 19/20
40/40 [=====] - 276s 7s/step - loss: 0.0420 - accuracy: 0.9797 - val_loss: 0.1489 - val_accuracy: 0.9531

Epoch 00020: val_accuracy did not improve from 0.97500
Epoch 20/20
40/40 [=====] - 275s 7s/step - loss: 0.0445 - accuracy: 0.9852 - val_loss: 0.1263 - val_accuracy: 0.9594

Epoch 00020: val_accuracy did not improve from 0.97500
```

*Ilustración 18 Valor de Accuracy a) 3 convoluciones y b) 6 convoluciones*

Para el modelo de descripción automática se utilizan el optimizador adam que permite adaptar las épocas a la función de pérdida, la cual está definida por el parámetro categorical\_crossentropy, la métrica utilizada para evaluar los modelos es Accuracy, se realizó el entrenamiento para 20 épocas y un batch\_size de 32, la diferencia entre los dos procesos fue el tiempo y el valor de pérdida evidenciados en la tabla 4. Se realizó un análisis de pérdida después de ejecutar el modelo, como resultado se obtuvo pérdidas de 0.4539 y 0.4876 en los modelos de 3 y 6 convoluciones, respectivamente. Estos datos indican que el manejo de la creación de texto se ve afectada posiblemente porque para la cantidad de vocabulario ejecutado se necesitaría un Dataset más amplio o a su vez se necesitaría disminuir el número de palabras involucradas en la descripción de imágenes.

## a) Modelo de 3 convoluciones

## b) Modelo de 6 convoluciones

```

/usr/local/lib/python3.7/dist-packages/keras/engine/training.py:1972: UserWarning:
warnings.warn("`Model.fit_generator` is deprecated and ")
1280/1280 [=====] - 276s 212ms/step - loss: 6.8821
/usr/local/lib/python3.7/dist-packages/keras/utils/generic_utils.py:497: CustomM
category=CustomMaskWarning)
1280/1280 [=====] - 268s 209ms/step - loss: 0.8942
1280/1280 [=====] - 267s 209ms/step - loss: 0.6540
1280/1280 [=====] - 267s 209ms/step - loss: 0.5831
1280/1280 [=====] - 266s 208ms/step - loss: 0.5377
1280/1280 [=====] - 268s 209ms/step - loss: 0.5524
1280/1280 [=====] - 265s 207ms/step - loss: 0.4848
1280/1280 [=====] - 265s 207ms/step - loss: 0.4693
1280/1280 [=====] - 266s 208ms/step - loss: 0.4615
1280/1280 [=====] - 266s 207ms/step - loss: 0.4525
1280/1280 [=====] - 269s 210ms/step - loss: 0.4377
1280/1280 [=====] - 267s 208ms/step - loss: 0.4373
1280/1280 [=====] - 266s 208ms/step - loss: 0.4377
1280/1280 [=====] - 267s 208ms/step - loss: 0.4428
1280/1280 [=====] - 263s 205ms/step - loss: 0.7379
1280/1280 [=====] - 263s 205ms/step - loss: 0.4544
1280/1280 [=====] - 265s 207ms/step - loss: 0.4432
1280/1280 [=====] - 265s 207ms/step - loss: 0.4423
1280/1280 [=====] - 267s 208ms/step - loss: 0.4665
1280/1280 [=====] - 266s 208ms/step - loss: 0.4539

/usr/local/lib/python3.7/dist-packages/keras/engine/training.py:1972: UserWarni
warnings.warn("`Model.fit_generator` is deprecated and ")
1280/1280 [=====] - 108s 81ms/step - loss: 4.2246
/usr/local/lib/python3.7/dist-packages/keras/utils/generic_utils.py:497: Cus
category=CustomMaskWarning)
1280/1280 [=====] - 102s 80ms/step - loss: 1.6622
1280/1280 [=====] - 102s 80ms/step - loss: 0.9749
1280/1280 [=====] - 102s 80ms/step - loss: 0.7684
1280/1280 [=====] - 102s 80ms/step - loss: 0.6640
1280/1280 [=====] - 102s 80ms/step - loss: 0.8230
1280/1280 [=====] - 104s 81ms/step - loss: 0.5903
1280/1280 [=====] - 108s 84ms/step - loss: 0.5555
1280/1280 [=====] - 105s 82ms/step - loss: 0.5561
1280/1280 [=====] - 105s 82ms/step - loss: 0.5363
1280/1280 [=====] - 104s 82ms/step - loss: 0.5229
1280/1280 [=====] - 105s 82ms/step - loss: 0.5258
1280/1280 [=====] - 104s 81ms/step - loss: 0.5310
1280/1280 [=====] - 104s 81ms/step - loss: 0.5102
1280/1280 [=====] - 103s 80ms/step - loss: 0.4945
1280/1280 [=====] - 103s 80ms/step - loss: 0.4981
1280/1280 [=====] - 103s 80ms/step - loss: 0.4913
1280/1280 [=====] - 102s 80ms/step - loss: 0.4868
1280/1280 [=====] - 103s 80ms/step - loss: 0.4719
1280/1280 [=====] - 103s 80ms/step - loss: 0.4876

```

Ilustración 19 19 Épocas de entrenamiento del modelo de descripción automática

### 5.6.1.Experimentos realizados y tiempos de ejecución

Se realizaron 2 modelos para la comparación entre procesos, las redes constan de la misma arquitectura, sin embargo, el uno presenta 3 redes convolucionales y un segundo modelo 6 redes convolucionales, los datos presentados en la Tabla 5 corresponden a la comparación de estos.

Tabla 55 Comparación entre los modelos propuestos

PARÁMETRO	MODELO CON 3 REDES CONVOLUCIONALES	MODELO CON 6 REDES CONVOLUCIONALES
TIEMPO DE ENTRENAMIENTO DE LA RED CONVOLUCIONAL	1h:30	2h:40
TIEMPO DE ENTRENAMIENTO DE LA RED RECURRENTE	2h:00	3h:20
VAL_ACURRACY CONVOLUCIONAL	0,9718	0,9750
LOSS RECURRENTE	0,4539	0,4876
MATRIZ DE CONFUSIÓN	Mayor precisión en datos de Volcanes	Mayor precisión en datos de Paisajes
PREDICCIÓN RED CONVOLUCIONAL	0 Volcanes y 1 Paisajes, realiza la categorización correcta	0 Volcanes y 1 Paisajes, realiza la categorización correcta
DESCRIPCIÓN FINAL	Termina con una sola descripción automática para las imágenes	Termina con una sola descripción automática para las imágenes

### 5.7. GRÁFICAS DE PRECISIÓN Y PÉRDIDA

La ilustración 20 se evidencia las gráficas de precisión y pérdida correspondientes a los modelos convolucionales de 3 y 6 convoluciones, se observó que las gráficas no presentan un overfitting, y cabe recalcar la relación inversa entre el número de convoluciones con la aproximación del modelo teórico con el experimental, esto se evidencia claramente que mientras mayor precisión

tenemos entre el modelo de seis convoluciones disminuye la pérdida. En el caso de las gráficas correspondientes a la precisión podemos ver que el valor se acerca a 1 además la curva de validación y los puntos de entrenamiento son muy cercanos, en cambio para las gráficas de pérdida los valores se encuentran entre 0 – 0.4 y a su vez la curva y los puntos son cercanos. También se puede notar que a partir de los gráficos de precisión el modelo cumple satisfactoriamente con el entrenamiento durante las épocas de formación establecidas.

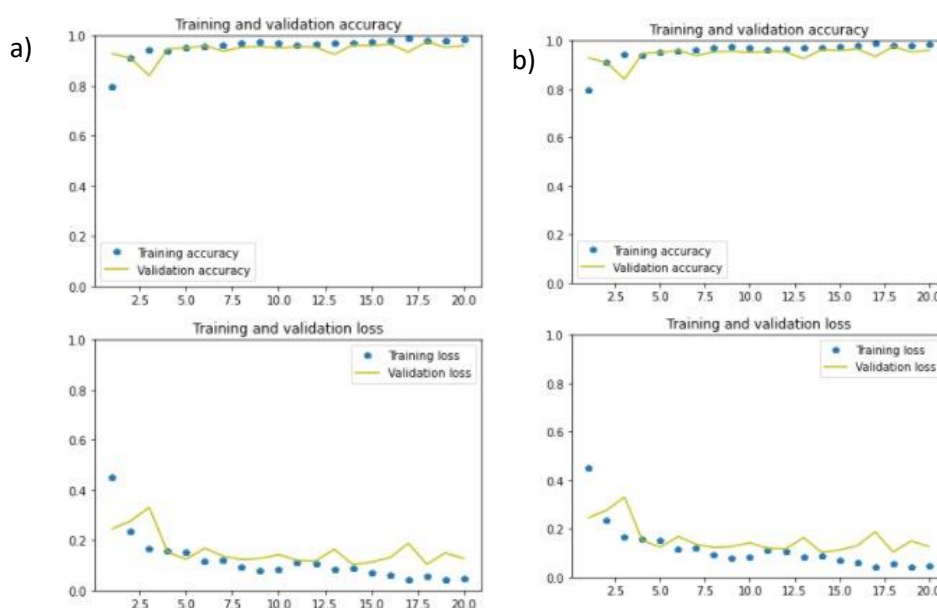


Ilustración 20 20 Gráfica de Precisión: a) Modelo de tres convoluciones. b) Modelo de seis convoluciones.

### 5.7.1. Matriz de confusión (Red Neuronal Convocional)

Una matriz de confusión evalúa el rendimiento de un modelo de clasificación, predice valores correctos o incorrectos comparando los siguiente variables (Tabla 6): VP representa un valor verdadero positivo (El valor real fue positivo y el modelo predijo un valor positivo), FP (Falso positivo), el valor real fue negativo pero el modelo predijo un valor positivo), FN (Falso negativo), el valor real fue positivo, pero el modelo predijo un valor negativo y finalmente VN (Verdadero negativo), el valor real fue negativo y el modelo predijo un valor negativo. (IArtificial.net, 2021).

Tabla 6 6 Variables de Matriz de Confusión

Valores reales
----------------

		Positivo	Negativo
Valores predichos	Positivo	VP	FP
	Negativo	FN	VN

Para obtener los valores numéricos de la Matriz de Confusión se deben calcular las métricas Recall, Precisión, Acurracy y Medida F, mediante las siguientes fórmulas:

Tabla 7 7 Formulas de métricas de la Matriz de confusión

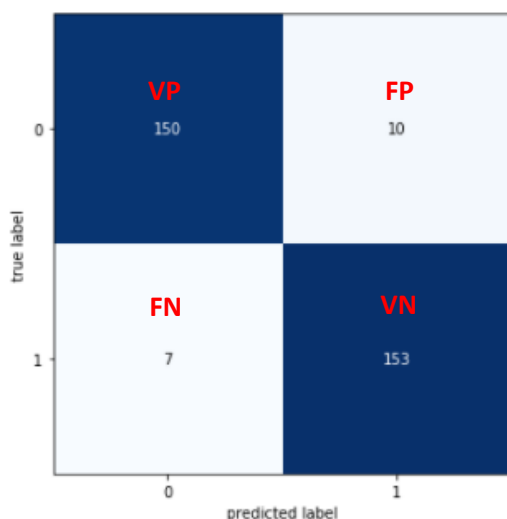
Métrica	Ecuación	Métrica	Ecuación
<b>Recall</b>	$\frac{VP}{VP + FN}$	<b>Acurracy</b>	$\frac{VP}{VP + FP}$
<b>Precisión</b>	$\frac{VP + VN}{TOTAL}$	<b>Medida F</b>	$\frac{2 * Recall * Precisión}{Recall + Precisión}$

En base a esta información se analizó las siguientes ilustraciones:

- Modelo de 3 convoluciones

La matriz de confusión definida para el modelo de 3 convoluciones, cuenta un total de 320 datos de los cuales 303 son verdaderos (150 positivos y 153 negativos) y 17 son falsos (7 negativos Y 10 positivos).

Todas las métricas presentan porcentajes superiores al 90%, lo que implica que las predicciones en imágenes de Paisajes y Volcanes es óptima.



Métrica	Ecuación	Resp.
<b>Recall</b>	$\frac{150}{150 + 7}$	<b>0.9554</b>
<b>Precisión</b>	$\frac{150 + 153}{320}$	<b>0.9468</b>
<b>Acurracy</b>	$\frac{150}{150 + 10}$	<b>0.9375</b>
<b>Medida F</b>	$\frac{2 * 0.9554 * 0.9468}{0.9554 + 0.9468}$	<b>0.9511</b>

Ilustración 21 21 Matriz de confusión para el modelo de 3 convoluciones y cálculo de métricas

La Ilustración 20 desglosa los valores de las Métricas para cada una de las categorías, donde 0 representa Paisajes y 1 Volcanes, se corrobora que los valores calculados



manualmente son muy aproximados a las métricas calculadas por el Modelo. La métrica de Precisión refleja mayor porcentaje en la clase 0 (96%), por lo tanto, la categoría de Paisajes presenta mayor cantidad de interacciones correctas; Recall muestra un 96% en la clase 1, lo que implica que las imágenes fueron identificadas y asignadas correctamente a la categoría Volcanes. Por última Acurracy indica que el 95% de las interacciones arrojaron una clase correctamente.

	precision	recall	f1-score	support
Class 0	0.96	0.94	0.95	160
Class 1	0.94	0.96	0.95	160
accuracy			0.95	320
macro avg	0.95	0.95	0.95	320
weighted avg	0.95	0.95	0.95	320

Ilustración 22 22 Métricas detalladas obtenidas del Modelo de 3 convoluciones

- Modelo de 6 convoluciones

La matriz de confusión definida para el modelo de 3 convoluciones, cuenta un total de 320 datos de los cuales 301 son verdaderos (152 positivos y 149 negativos) y 19 son falsos (11 negativos Y 8 positivos). En comparación con el modelo anterior la variación entre los porcentajes es mínimos, se encuentran aún sobre el 90% lo que implica que a pesar de aumentar las redes convolucionales la predicción, aunque bajó un poco sigue siendo optima.

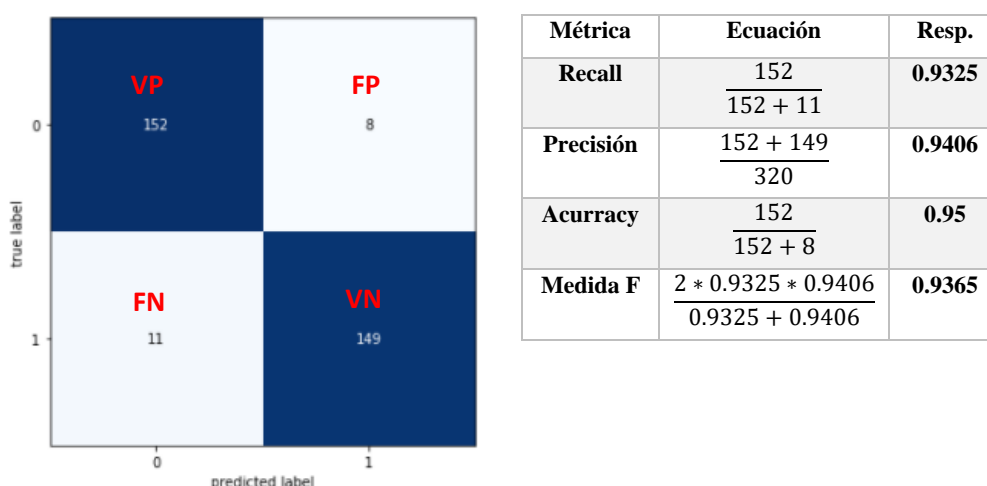


Ilustración 23 23 Matriz de confusión para el modelo de 6 convoluciones y cálculo de métricas

La métrica de Precisión refleja mayor porcentaje en la clase 1 (95%), por lo tanto, la categoría de Volcanes presenta mayor cantidad de interacciones correctas; Recall muestra un 95% en la clase 1, lo que implica que las imágenes fueron identificadas y asignadas correctamente a la categoría Volcanes. Por último, Acurracy indica que

el 94% de las interacciones arrojaron una clase correctamente. Con esta Ilustración se corrobora la variación ya detectada en los cálculos manuales, además se podría suponer que al aumentar las convoluciones las características detectadas son más claras en la categoría de Volcanes. (Ilustración 24)

	precision	recall	f1-score	support
Class 0	0.93	0.95	0.94	160
Class 1	0.95	0.93	0.94	160
accuracy			0.94	320
macro avg	0.94	0.94	0.94	320
weighted avg	0.94	0.94	0.94	320

*Ilustración 24 Métricas detalladas obtenidas del Modelo de 6 convoluciones*

## 5.8. PREDICCIÓN

El proceso de predicción se utilizó en la parte convolucional en los dos modelos las predicciones entre Volcanes y Paisajes fue acertada, sin embargo, al momento de terminar los modelos se evidencia que tanto para el modelo de 6 como de 3 convoluciones los resultados no son adecuados ya que la descripción automática no es la que corresponde, cada modelo tomó una sola descripción para todas las imágenes, por lo cual no se pudo realizar una descripción automatizada adecuada, este problema puede deberse a los múltiples problemas generados al cargar los datos de entrenamiento y prueba para las descripciones, ya que al cargar una sola codificación al decodificar pueden causar inconvenientes. A su vez aún cuando el modelo es generado paso a paso, al ser modificado de varios códigos establecidos puede haber problemas con el tipo de imágenes utilizadas, ya que tienen varias características que pueden afectar el reconocimiento. (Ilustración 25)

### a) Modelo de 3 convoluciones

```
prueba = '/content/drive/MyDrive/DATASET/test/PAISAJES/P_648.jpg'
img = cv2.imread(prueba)
img_cvt=cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
plt.imshow(img_cvt)
plt.show()
# cargar y preparar imagen
photo = extract_features(prueba)
# generar descripcion
description = generate_desc(model, tokenizer, photo, max_length)
print(description)
```



startseq no es volcan paisaje llanura endseq

### b) Modelo de 6 convoluciones

```
#Prueba del código - descripciones
prueba = '/content/drive/MyDrive/DATASET/test/PAISAJES/P_648.jpg'
img = cv2.imread(prueba)
img_cvt=cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
plt.imshow(img_cvt)
plt.show()
photo = extract_features(prueba)
description = generate_desc(model, tokenizer, photo, max_length)
print(description)
```



startseq no es volcan paisaje sembrío endseq

Ilustración 25 25 Resultado de descripción automática de imágenes

## 6. CONCLUSIONES Y RECOMENDACIONES

Se puede evidenciar que los dos modelos generados son bastante precisos, teniendo valores de precisión altos tanto para Paisajes como para Volcanes, el modelo de 3 convoluciones tiene mayor precisión para los datos de Volcanes mientras que el de 6 convoluciones presenta mayor precisión para los datos de Paisajes. Debido a estos datos se optó por continuar los dos modelos para aplicar la red neuronal recurrente.

Una vez que se ha propuesto la arquitectura del modelo que se va a utilizar, procedimos a la selección de la base de datos con la que se entrena para adquirir el conocimiento necesario para las descripciones automáticas de volcanes. A la hora de realizar la primera prueba de training pudimos determinar que la base de datos no se ajustaba correctamente al modelo implementado por lo cual fue necesaria una etapa de depuración para la adaptación al modelo, ya que cada base de datos requiere de un previo tratamiento de datos. Es recomendable realizar con mucha precisión este proceso para no tener errores en el test y training.

Es recomendable que para una futura ejecución del Modelo se amplíe considerablemente el Dataset debido a que este tipo de modelos requieren más información para un mejor entrenamiento del Modelo, además como dato adicional se podría mejorar la predicción de texto disminuyendo el número de palabras en la descripción de las imágenes.

## 7. MODELOS

Modelo de 3 convolucionales. Link de acceso:

[https://colab.research.google.com/drive/1u8q7wkUPimDrov2uhllyC\\_RUTJPglhwh?usp=sharing](https://colab.research.google.com/drive/1u8q7wkUPimDrov2uhllyC_RUTJPglhwh?usp=sharing)

Modelo de 6 convolucionales. Link de acceso:

[https://colab.research.google.com/drive/1eBVPlIKXi4Xo-52y\\_wa4XiU93N3ElzEq?usp=sharing](https://colab.research.google.com/drive/1eBVPlIKXi4Xo-52y_wa4XiU93N3ElzEq?usp=sharing)

Modelos convolucionales y recurrentes requeridos en los modelos principales:

[https://drive.google.com/drive/folders/18JTWG80r8c2-feHr9ZX9NM6OGxVInSJ\\_?usp=sharing](https://drive.google.com/drive/folders/18JTWG80r8c2-feHr9ZX9NM6OGxVInSJ_?usp=sharing)

## 8. REFERENCIAS

Bagnato, J. (2018). *Aprendiendo Machine Learning*. Obtenido de

<https://www.aprendiendomachinelearning.com/clasificacion-de-imagenes-en-python/>

CAmino, G. L. (2020). *Desarrollo de una red neuronal convolucional para la clasificación de botellas plásticas en una empresa envasadora de bebidas gaseosas de la ciudad de Milagro*. Milagro: Universidad Estatal de Milagro.

Fuente, O. D. (2019). Google Colab: Python y Machine Learning en la nube. *Tutoriales de Machine Learning*.

Garzón, I. (2018). Cómo usar redes neuronales (LSTM) en la predicción de averías en las máquinas. Link: <https://blog.gft.com/es/2018/11/06/como-usar-redes-neuronales-lstm-en-la-prediccion-de-averias-en-las-maquinas/>

Gho, R. (2017). *La volcanología Chilena en el contexto latinoamericano*. Chile: ALVO, Jóvenes vulcanólogos .

*Github*. (junio de 2019). Obtenido de

<https://gist.github.com/bhaveshwadhvani/acf84ec887391c53caf5ff24ca87bb07>

*Github*. (noviembre de 2020). Obtenido de

<https://gist.github.com/Priyankadiddi/241b9c9f949c7bfb66b21b8b0930dd8e>

*Google Colaboratory*. (Mayo de 2019). Obtenido de

[https://colab.research.google.com/gist/vybhav72954/187ab5fde841584756b6c5bcc1ee2fa6/caption\\_generator.ipynb](https://colab.research.google.com/gist/vybhav72954/187ab5fde841584756b6c5bcc1ee2fa6/caption_generator.ipynb)

Heras, D. (2017). *Clasificador de imágenes de frutas basado en inteligencia artificial*. Killkana Técnica.

IArtificial.net. (2021). *Librerías de Python para Machine Learning*. Obtenido de

[https://www.iartificial.net/librerias-de-python-para-machine-learning/#Librerias\\_de\\_Python\\_para\\_Visualización](https://www.iartificial.net/librerias-de-python-para-machine-learning/#Librerias_de_Python_para_Visualización)

Llamas, J. M. (2020). *Reconocimiento de imágenes mediante redes neuronales convolucionales*. Madrid: Universidad Politécnica de Madrid.

Loaiza, W. (2020). *Reconocimiento de objetos usando técnicas de inteligencia artificial*. Pereira: Universidad tecnológica de Pereira.

MIDEPLAN. (2020). Guía para la elaboración de diagramas de flujo. 21.

- Peláez, R. A. (2010). Simbología ASME y ANSI. *Slideshare*, 12.
- Rodriguez, P. (2018). *Aplicación de redes neuronales convolucionales y recurrentes al diagnóstico de autismo a partir de resonancias magnéticas funcionales*. Madrid: Industriales.
- Romero, J. (2018). *Volcanología: El estudio de los Volcanes*. Chile: ALVO, Jóvenes Volcanólogos .
- UPAEP. (2020). Obtenido de <http://gapps.upaep.mx/inicio/googledocs/google-drive/que-es-google-drive>
- Utrera, J. (2019). Tratamiento de imágenes usando ImageDataGenerator en Keras. *Deep Learning para principiantes*.