UNIVERSIDAD POLITÉCNICA SALESIANA

SEDE CUENCA

CARRERA DE COMPUTACIÓN

DISEÑO, DESARROLLO E IMPLEMENTACIÓN DE UN MÓDULO DE DESCRIPCIÓN AUTOMÁTICA DE IMÁGENES PARA LA HERRAMIENTA DE ADAPTACIÓN DE OBJETOS DE APRENDIZAJE EN EL MARCO DEL PROYECTO EDUTECH

Trabajo de titulación previo a la obtención del título de Ingeniero en Ciencias de la Computación

AUTOR: CELSO LEONARDO ALVARADO TORRES

TUTOR: VLADIMIR ESPARTACO ROBLES BYKBAEV, Ph.D

Cuenca - Ecuador

2022

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Yo, Celso Leonardo Alvarado Torres con documento de identificación N. 0105652747 manifiesto que:

Soy autor y responsable del presente trabajo; y, autorizo a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Cuenca, 09 de marzo del 2022.

Atentamente,

_____

Celso Leonardo Alvarado Torres

0105652747

CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA

Yo, Celso Leonardo Alvarado Torres con documento de identificación Nº 0105652747, expreso mi voluntad y por medio del presente cedo a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que soy autor del Proyecto técnico: "Diseño, desarrollo e implementación de un módulo de descripción automática de imágenes para la herramienta de adaptación de objetos de aprendizaje en el marco del proyecto EduTech", el cual ha sido desarrollado para optar por el título de: Ingeniero en Ciencias de la Computación, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribo este documento en el momento que hago la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Cuenca, 09 de marzo del 2022.

Atentamente,

_____

Celso Leonardo Alvarado Torres

0105652747

CERTIFICADO DE DIRECCION DE TRABAJO DE TITULACIÓN

Yo, Vladimir Espartaco Robles Bykvaev con documento de identificación N° 0300991817, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: DISEÑO, DESARROLLO E IMPLEMENTACIÓN DE UN MÓDULO DE DESCRIPCIÓN AUTOMÁTICA DE IMÁGENES PARA LA HERRAMIENTA DE ADAPTACIÓN DE OBJETOS DE APRENDIZAJE EN EL MARCO DEL PROYECTO EDUTECH, realizado por Celso Leonardo Alvarado Torres con documento de identificación N° 0105652747, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Cuenca, 09 de marzo de 2022.

Atentamente,

_____

Ing. Vladimir Espartaco Robles Bykvaev, Ph.D.

0300991817

DEDICATORIA Y AGRADECIEMIENTO

El presenta trabajo de titulación lo dedico principalmente a Dios, por ser quien me permitió poder abrirme frente a las adversidades de la vida y lograr este anhelo tan deseado.

A mis padres Celso Alvarado y Rosario Torres, por no haber perdido la fe en mi aun cuando las situaciones eran desfavorables siendo ellos un ejemplo a seguir en busca de mis metas. A mi familia, en especial a mis hermanas Fabiola Alvarado y Soledad Alvarado quien siempre me brindaron su apoyo desde que estuve en la escuela hasta el último momento.

A mis maestros quienes gracias a su dedicación y paciencia lograron hacer de mí una mejor persona, no solo en el ámbito académico sino en la vida misma.

Celso Leonardo Alvarado Torres

RESUMEN

En el presente trabajo se realiza una investigación en el ámbito de la accesibilidad web, más objetivamente con los objetos de aprendizaje, estos temas se encuentran estrechamente relacionados a la educación virtual o e-learning. La investigación tiene como finalidad establecer una conexión entre la accesibilidad web y el acceso a los objetos de aprendizaje a personas con discapacidad, concretamente a personas con problemas visuales quienes no pueden acceder de la misma forma al contenido multimedia que forma parte de los recursos educativos, debido a los problemas que afrontan los estudiantes con discapacidad se opta por diseñar, desarrollar e implementar una solución que permita obtener la descripción de imágenes, de esta forma las herramientas web, como los navegadores, podrán hacer uso del texto generado para poder establecer una retroalimentación de lo que se está presentando en un determinado contenido multimedia. Para lograr este acometido se recurre al uso de redes neuronales convolucionales y redes neuronales recurrentes las cuales permiten clasificar imágenes dentro de categorías en el caso del primer ejemplo y describir las mismas de manera textual en el caso del segundo ejemplo. Para el proyecto se investigó diferentes técnicas y herramientas que pudieran servir de apoyo para mejorar los resultados, técnicas como por ejemplo ViT o Transfer Learning ya que las redes neuronales tradicionales sirven de punto de partida, pero no siempre generan los resultados esperados. Finalmente, este proyecto servirá como una herramienta para que tanto profesores como estudiantes puedan hacer uso objetos de aprendizaje más accesibles a través de la herramienta OerAdap.

Palabras clave:

Accesibilidad web, objetos de aprendizaje, redes neuronales, clasificación de imágenes, descripción de imágenes.

ABSTRACT

Web accessibility is considered a major issue in the context of educational websites because many educational learning objects used by professors are not prepared for students with disabilities. This research aims to establish a foundation for improving the accessibility of learning objects directly related to multimedia resources. In this context, different deep learning algorithms were analyzed to classify and caption digital images. As a result, we developed a tool for the assistance of educators and students with visual impairments. The function of the tool is to, in the first instance, classify images according to a chart in which we defined the main areas of educational digital images. After classifying, the image goes through an algorithm that generates textual captioning according to four main areas, real images, illustrations, mathematical formulas, and data tables. In the case of data tables, HTML code is generated ready to be encrusted in the learning object. Subsequently we discussed the importance of using specialized hardware to improve the results of the algorithm since there are some deep learning models which use specific, high-performance, libraries to achieve state-of-the-art results. The results showed that with the help of advanced neural networks techniques great improvements can be accomplished in the task of image classification and captioning. As a final point, this research will be part of a set of tools for the OerAdap project in an effort to make learning objects more accessible for people with disabilities.

Key words:

Web accessibility, learning objectives, neural networks, image classification, image captioning.

INDICE DE CONTENIDO

INDICE DE ILUSTRACIONES

INDICE DE TABLAS

1. Introducción

Sabemos que la accesibilidad en el ámbito de e-learning es de brindar la facilidad de que cada persona pueda acceder a los distintos contenidos educativos sin que afecte su comprensión o interacción, por ello la ausencia de accesibilidad en objetos de aprendizaje puede generar que estudiantes con necesidades de educación especial no logren percibir adecuadamente la información conllevando a que no tenga un correcto aprendizaje (Web AIM, 2020).

1.1. La accesibilidad web

La accesibilidad hace referencia a un diseño web que va a permitir que personas con diferentes tipos de discapacidad puedan acceder e interactuar con el web incluido sus contenidos (Henry, 2019).

La accesibilidad web busca que la información presentada dentro de sitios de internet pueda ser accedida por diferentes tipos de personas, sin importar si estas tienen conocimientos básicos dentro del área técnica o tienen discapacidad.

Dentro de la accesibilidad web existen ciertas barreras erróneas que la hacen difícil de implementar, como por ejemplo el costo de la implementación de esta, u otra en la cual se cree que únicamente las páginas web con información textual pueden ser accesibles. Estos problemas lejos de ser ciertos presentan un problema para las personas que utilizan los servicios web ya que al verse frenados por la dificultad que presenta acceder a esta optan por no hacerlo (Mora, 2006).

1.2. Metadatos

Dentro del ámbito web existe una gran cantidad de información, si bien la información está estructurada de dentro del sitio web no es el caso para los buscadores que deben indexarlos, por lo tanto, se debe describir de que trata un determinado sitio web junto con su contenido de forma singular si se hace uso de archivos multimedia como imágenes, ya que de esa forma se puede facilitar el proceso de búsqueda, uso y preservación de la información que está dentro de un sitio web (University of North Carolina Chapell Hill, 2021).

En base a lo anterior podemos concluir que los metadatos son presentados como información estructurada que permite describir determinada información de un recurso, en este caso de un recurso informático dentro del ámbito web.

"Los metadatos son la llave para asegurar que los recursos van a sobrevivir y continuar siendo accesibles en el futuro" (National Information Standards Organization, 2004).

1.3. Objetos de aprendizaje

Los objetos de aprendizaje forman parte de recursos educativos digitales que permiten apoyar al aprendizaje, los objetos de aprendizaje pueden ser de diferentes tipos de multimedia como texto, imágenes, videos u otros recursos que pueden ser reutilizados para otros fines. Los objetos de aprendizaje de acuerdo con Ministerio de Educación Colombiana deben contener metadatos de tal forma que sean fácilmente almacenados y accedidos (Universidad de Antioquia).

1.4. Inteligencia Artificial

La inteligencia artificial es un campo amplio dentro de las ciencias de la computación ya que busca comprender los principios que hacen a un agente natural o artificial 'inteligente', de tal forma que dicha inteligencia pueda ser sintetizada en otras áreas. En nuestro entorno, los humanos somos considerados como los agentes más inteligentes ya que podemos realizar tareas complejas o imposibles para otras especies, pero no se limita hasta ahí, ya que en sociedad los humanos somos aún más inteligentes, por ejemplo, somos capaces de diseñar y crear computadoras, cosa que una sola persona no podría hacerlo ya sea por su falta de conocimiento en el área o por la falta de experiencia, pero un grupo de personas, una sociedad, puede lograrlo (University Of British Columbia, 1998).

1.5. Machine Learning

Dentro del área de la computación se busca diseñar y desarrollar algoritmos que den solución a un determinado problema, pero existen ciertas áreas en las cuales estos problemas son demasiado complejos para ser representados por los algoritmos tradicionales, el problema es que un determinado algoritmo únicamente servirá para una única tarea, y en general eso es lo correcto, pero ¿Qué pasaría si se desea ejecutar el mismo algoritmo con información que nunca vio? Naturalmente el algoritmo no funciona ya que no fue diseñado para realizar otra tarea sino para la que fue prevista, por lo tanto, podemos concluir que un algoritmo tradicional no puede generalizar el proceso de aprendizaje.

Como un nuevo enfoque para superar las barreras de los algoritmos clásicos nace el aprendizaje de máquina, este enfoque forma parte de la inteligencia artificial, que a su vez busca simular la inteligencia de los seres humanos en general. Un algoritmo de aprendizaje de máquina se diferencia de un algoritmo clásico en el cual, el primero, puede adaptar su arquitectura conforme vaya apareciendo nueva información de tal forma que cuando aparezca información nunca vista este pueda generar una salida dentro de los parámetros establecidos.

Un ejemplo simple entre las limitaciones de un algoritmo clásico y un algoritmo basado en aprendizaje de máquina es la clasificación entre perros y gatos, en la cual un algoritmo normal requiere de establecer todos los parámetros necesarios para lograr diferenciar entre un objeto y otro, estos parámetros pueden ser (pero no se limitan) la forma del animal, el color, la altura, el peso, el tipo de pelaje. Para que un algoritmo tradicional logre clasificar correctamente debería establecer gran parte de las características de todas las especies de los dos animales, y aun así existe el riesgo de que no lograra realizar la tarea correctamente debido a que puede aparecer una especie que no está dentro de las especificaciones de esta. En conclusión, este tipo de algoritmo no puede generalizar entre cómo diferenciar un perro de un gato.

En base al mismo ejemplo anterior si aplicamos aprendizaje de máquina los resultados son completamente distintos, ya que un algoritmo de aprendizaje de máquina busca, como su nombre sugiere, **aprender** de las características únicas que define un determinado objeto de tal modo que cuando aparezca una nueva especie de cualquiera de las dos clases (gatos y perros) este pueda clasificarlos correctamente ya que este algoritmo ya tiene las bases necesarias para hacerlo (El Naqa, Li, & Murphy, 2015).

1.6. Deep Learning

Deep Learning es una subárea dentro del aprendizaje de máquina que busca dar solución a problemas en los que se requiere que la información que va utilizada venga de fuentes sin procesar, por ejemplo las técnicas convencionales de aprendizaje de máquina se basan en algoritmos matemáticos que permiten imitar la manera en la que los humanos aprendemos por lo que se requiere que los datos sean preprocesados a tal nivel que sean fáciles de manejar por el computador, por así decir, datos que hasta una persona pueda entenderlos, pero este no es el caso de muchos tipos de datos multimedia tales como imágenes, videos, sonidos, etc. Para estos tipos de datos se requiere un procesamiento diferente ya que no fueron diseñados para tareas de este tipo de complejidad, aunque si bien existen ciertos algoritmos y técnicas que permiten realizar este tipo de tareas su uso no es adecuado ya que existen otras estrategias en Deep Learning mucho más precisas y robustas como las redes neuronales

Los métodos y estrategias de Deep Learning buscan acercarse de forma más profunda al razonamiento humano a través de la representación multinivel yendo desde un nivel en la cual la entrada de datos es representada por conjuntos de datos 'en crudo' hasta llegar a un nivel de abstracción que representa los mismos datos, pero de una manera simple y concisa lista para realizar el proceso de aprendizaje.
En conclusión, la principal diferencia entre Machine Learning y Deep Learning es que Deep Learning no depende de un preprocesamiento manual complejo realizado por un desarrollador, mientras que Machine Learning requiere de información previamente procesada caso contrario no presenta buenos resultados o en el peor de los casos no funciona (LeCun, Bengio, & Hinton, 2015).

1.7. Neurona Artificial

Una neurona artificial es una representación de una neurona biológica que tiene la capacidad de procesar una entrada y generar una salida hacia otra neurona, las conexiones entre otras neuronas a través de nodos representan los pesos, estos pesos cambian acorde a la llegada de nueva información y de esa forma ocurre el aprendizaje (Tucci, 2018).

Cada vez que llega nueva información hacia una neurona, esta decide si enviarla o no hacia la siguiente neurona si cumple cierta condición sobre un valor dado, este proceso ocurre en conjunto con las funciones de activación (Tucci, 2018).

1.8. Redes Neuronales Artificiales

Las redes neuronales artificiales se definen como un conjunto de neuronas que, conectadas entre sí forman una red, esta red a su vez está compuesta por una o múltiples entradas, una parte oculta y una o varias salidas. Las conexiones entre las diferentes capas de redes neuronales contienen los pesos de las conexiones generadas, a su vez los pesos contienen la información que fue adquirida a través del entrenamiento y que será utilizada a posterioridad para predecir en base a nueva información (Tucci, 2018).

La lógica funcional detrás de las redes neuronales recae sobre las funciones de activación. Las funciones de activación se definen como la suma ponderada de las entradas es transformada en una salida ya sea para un nodo único (una neurona) o para una capa completa. Normalmente las funciones de activación son diferentes para las capas de entrada, ocultas y de salida, ya que dependiendo del problema la salida puede ser diferente, por ejemplo, si se desea hacer una clasificación binaria se utiliza la función **Sigmoid**

mientras que para el caso de clasificación multiclase se utiliza la función **Softmax** (Sharma, 2017)**.**

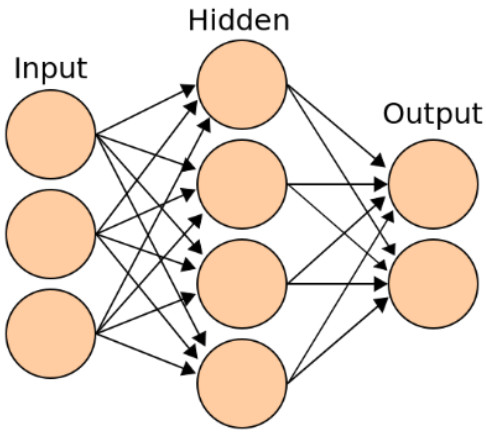


Ilustración 1 Ejemplo básico de una red neuronal

1.9. Visión por computador

La visión por computador es un campo de gran importancia dentro de la Inteligencia Artificial ya que es la encargada de manipular y obtener información importante de diferentes fuentes digitales como imágenes o videos.
"Si la AI permite a las computadoras pensar, la visión por computador permite ver, observar y entender." (IBM, n.d.).

El objetivo principal de la visión por computador es facilitar la automatización de tareas relacionadas al campo visual, por ejemplo:

a. Detección de objetos.
b. Segmentación de instancias.
c. Clasificación de objetos.
d. Reconstrucción de escenas.
e. Restauración de imágenes.

1.10. Procesamiento de lenguaje natural

El procesamiento de lenguaje natural es un enfoque que se centra en analizar texto en base a técnicas computaciones con el propósito de lograr un análisis y procesamiento semántico al nivel del ser humano en diferentes tareas o aplicaciones.

Un algoritmo de procesamiento de lenguaje natural completo debe ser capaz de lograr las siguientes tareas:

- **a.** Parafrasear un texto.
- **b.** Traducir un texto a otro lenguaje.
- **c.** Responder preguntas acerca de un texto.
- **d.** Obtener implicaciones de un texto.

Niveles del procesamiento de lenguaje natural (Liddy, 2001)**:**

a. Fonológico:

Interpreta los sonidos del habla en relación con las palabras.

b. Morfológico:

Interpreta los componentes naturales de las palabras desde su forma más básica con la finalidad de encontrar su significado a través de la separación de morfemas. Por ejemplo: Paraguas: Par (léxico) a (infijo) agu (léxico) a (sufijo de genero) s (morf. numero).

c. Léxico:

Interpreta el sentido individual de una palabra y cuál es su función dentro de un determinado contexto.

d. Sintáctico:

Este nivel se enfoca en analizar las palabras de una oración para determinar la estructura de una oración. Para que este proceso se lleve a cabo requiere de la gramática y del análisis de la oración.

e. Semántico:

Este nivel determina cuales son los posibles significados de la oración enfocándose únicamente en las interacciones entre palabras de manera que no exista ambigüedad en la información, ya que una palabra puede cambiar drásticamente a una oración dependiendo del contexto en la que sea utilizada.

f. Pragmático:

Se enfoca en entender el contenido del texto en base al contexto de este, ya que una palabra puede representar a un determinado objeto.

g. Discursivo:

Este nivel se encarga de determinar los componentes discursivos de un texto y la relación entre los mismos, por ejemplo, un artículo puede ser divido, en resumen, cuerpo, conclusiones y referencias.

1.11. Descripción de imágenes

Dentro del área de visión por computador y el procesamiento de lenguaje natural existe una importante tarea, la descripción de imágenes. En base a las diferentes instancias de una imagen se busca generar un texto lo suficientemente expresivo que describa el contenido de dicho objeto.

Una de las principales limitantes dentro de la descripción de imágenes es que no siempre se logra prestar atención a los aspectos más importantes dentro de la misma, por lo que se puede terminar generando un texto que no sea lo suficientemente conciso o no vaya acorde a lo que en realidad se quería describir. Por lo tanto, según (Quanzeng, Hailin, Zhaowen, Chen, & Jiebo, 2016) la atención visual a los detalles juega un rol esencial para seleccionar las partes más representativas de una imagen en lugar de prestar atención a partes irrelevantes.

2. Problema

Actualmente gran cantidad de contenido multimedia, específicamente imágenes digitales no contienen texto alternativo que permita determinar el contexto de dicho contenido, por lo que, herramientas que buscan ayudar en el ámbito de accesibilidad como lectores de pantalla no funcionan debido a que no existe los atributos necesarias para hacerlo, por ejemplo el atributo **alt** de la etiqueta **img** que permite establecer un texto alternativo para una determinada imagen de forma que si no se llegase a cargar la imagen aun así existe la posibilidad de mostrar un texto alternativo especificando el contexto de dicha imagen, esto será de mucha ayuda a personas con problemas visuales, ya que, a través de lectores de pantalla podrán acceder a la información que es presentada por una determinada imagen con contenido educativo. Esto resulta sumamente útil en el ámbito educativo ya que si determinado objeto de aprendizaje hace uso de contenido multimedia este podrá ser utilizado sin mayor dificultad. Sin embargo, el proceso de insertar texto alternativo dentro de los objetos de aprendizaje puede ser un proceso largo ya que esto se da en función a la cantidad de imágenes (Prabhu, 2021).

La cantidad de personas entre 4 a 24 años con discapacidad visual según (Consejo Nacional para la Igualdad de Discapacidades, 2021) es de aproximadamente 5122 hasta septiembre de 2021, se considera este rango de edad debido a que la mayoría de los estudiantes se encuentren en este rango de edades. Esta cifra es únicamente de Ecuador, mientras que mundialmente aproximadamente 2200 millones de personas en general sufren de esta discapacidad según (WHO, 2021), estas cifras nos indican la relevancia del proyecto debido a que varias de estas personas forman parte del cuerpo estudiantil independiente del grado de estudio en el que se encuentren.

Alcance del proyecto

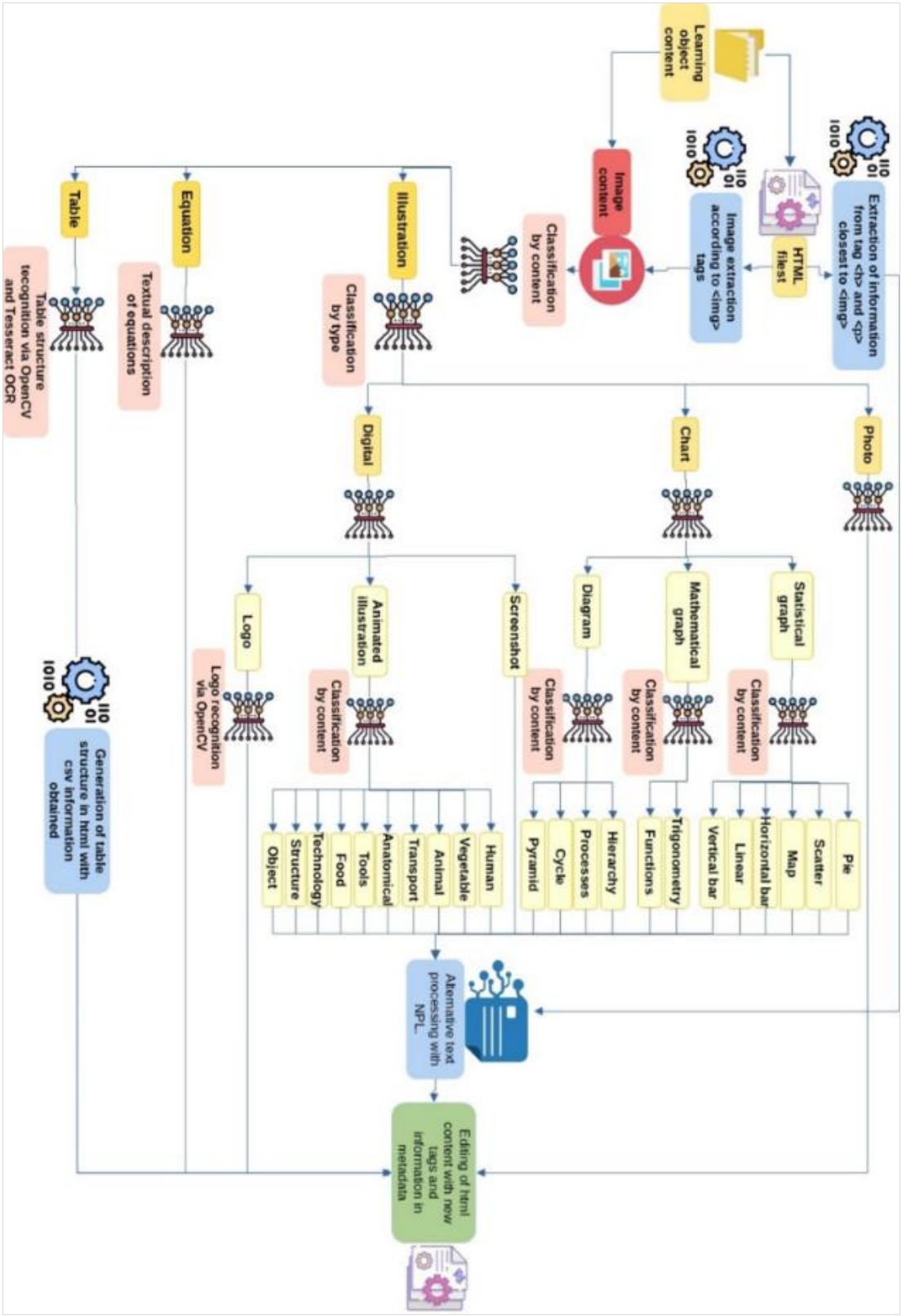


Ilustración 2 Diagrama de la propuesta para la descripción automática de imágenes.

EduTech de Erasmus+ busca implementar una solución para el manejo accesible de los objetos de aprendizaje que son fundamentales para el aprendizaje de los estudiantes sin importar el ámbito educativo, sin embargo existe una fuerte limitante que hace que los objetos de aprendizaje no puedan ser accedidos por todos, principalmente porque existen archivos multimedia tales como texto, imágenes o videos que no pueden ser analizados y evaluados tanto por docentes como por estudiantes al no cumplir con las normas de accesibilidad establecidas dentro del estándar W3.

Actualmente existen diferentes herramientas que permiten la creación de objetos de aprendizaje, por ejemplo, LOMPAD o ExeLearning, estas herramientas posibilitan la adaptación y distribución metadatos dentro del ámbito de los objetos de aprendizaje, no obstante, estas no se enfocan a la accesibilidad de la información multimedia.

Una solución óptima para habilitar la accesibilidad dentro de los objetos de aprendizaje es la creación de metadatos que posibiliten la interpretación fácil de un determinado tema, para lo cual se propone utilizar herramientas dentro del ámbito de la inteligencia artificial para mitigar estas limitaciones.

El enfoque principal de esta solución es el planteamiento de una estrategia para la descripción de imágenes de manera automática en la cual, a través de procesamiento de lenguaje natural, visión por computador y Deep learning se logre generar un texto descriptivo para una determinada imagen, ya sea en base a texto complementario o únicamente en base a una imagen. A más de la descripción de imágenes la propuesta también se enfoca a la clasificación multinivel de imágenes de tal forma que se pueda determinar con qué tipo de imagen se está trabajando dentro del ámbito educativo.

3. Objetivos Generales y Específicos

Objetivo General

Desarrollar un módulo basado en visión por computador y procesamiento natural del lenguaje para la descripción de imágenes.

Objetivos Específicos

- OE1. Estudiar y conocer los principios de la accesibilidad web y metadatos aplicados a objetos de aprendizaje

- OE2. Diseñar y desarrollar un módulo basado en inteligencia artificial y procesamiento del lenguaje natural para generación de textos que permitan describir imágenes y la adaptación a contenido HTML
- OE3. Despliegue del módulo en la herramienta de adaptación que forma parte de la arquitectura del sistema Edutech.
- OE4. Diseñar y ejecutar un plan de experimentación que permita determinar la precisión del sistema con un dataset de objetos de aprendizaje
- OE5. Desarrollo de los manuales técnico y de usuario

4. Revisión de la literatura

Se realizo una investigación sobre diferentes herramientas de clasificación y descripción de imágenes que permitan trabajar sobre el conjunto de imágenes de una manera eficaz y a su vez que generen una buena precisión. Dentro de las herramientas más destacadas se encuentran:

Nombre	Fecha de creación	Novedad	Resultados
VGGNet19	2014	Reduce el número de parámetros en las capas convolucionales lo que mejora el rendimiento al entrenar la red.	92.5% de precisión en ImageNet.
ResNet50	2015	Resuelve el problema del desvanecimiento del gradiente a través del salto de conexiones.	80.67% de precisión en ImageNet.
Inception-v3	2015	Incrementa el tamaño de una red neuronal a lo ancho, es decir evita agregar más capas de esta forma combina capas convolucionales	78.8% de precisión en ImageNet.

		creando módulos Inception.	
InceptionResNetV2	2016	Reemplaza la concatenación de filtros usada en Inception por conexiones residuales basadas en ResNet.	95.1% de precisión en ImageNet.
ViT (Dosovitskiy, et al., An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale, 2020)	2021	Hace uso de los Transformers usados en el procesamiento de lenguaje natural, de esta forma busca mejorar las clasificaciones de grano fino.	90.72% de precisión en ImageNet.

Tabla 1 Herramientas para la clasificación de imágenes.

Las modelos de redes neuronales convolucionales antes presentadas proponen alternativas factibles para trabajar en las diferentes tareas de clasificación de imágenes, sin embargo, los ViT (Vision Transformers) son una alternativa mucho más eficiente al momento de clasificar imágenes muy similares, esta tarea se la conoce como clasificación de grano fino.

Como se estableció en la ilustración 2 se debe reconocer la estructura de una tabla con información, para esta tarea existen diferentes herramientas, algunas van desde simples librerías hasta artículos que tratan a profundidad este tema que puede llegar a ser complejo si se trata con tablas cuya estructura no contenga bordes. En la siguiente tabla se definen las herramientas más prominentes:

Nombre	Fuente	Fecha	Novedad
Image-based table recognition.	Articulo académico en arXiv	4 de marzo de 2019	Propone un modelo basado en atención codificador-doble-decodificador (EDD), logra convertir

			imágenes de tablas en código HTML.
CascadeTabNet	Articulo académico en arXiv.	27 de abril de 2020	Este modelo se basa en la técnica de transfer learning y aumento de datos para reconocer la estructura de las celdas de la tabla y obtener la estructura general de la tabla.
Open Intelligence	Tesis de grado Hogeschool Gent.	No se especifica	Propone una alternativa de código abierto ante API's de Google, Amazon, Microsoft, etc. Se basa en el algoritmo de CascadeTabNet y lo mejora a través de técnicas de visión por computador.

Tabla 2 Herramientas para el reconocimiento de las estructuras de tablas.

Como se ha puede evidenciar en la tabla 2, no existen muchas herramientas para la establecer la estructura de una tabla que sean de código abierto, sin embargo, existen otras alternativas de empresas como Microsoft que permiten definir la estructura de una tabla con excelentes resultados (Haiby, 2021), pero, esta herramienta es de propietario por lo que no puede ser usada en el presente proyecto.

La idea principal de este proyecto es la descripción de imágenes, por lo tanto, se pueden definir algunos modelos que han sido utilizados dentro de esta área, ya que, la descripción de imágenes es uno de los puntos principales dentro de la inteligencia artificial ya que une el área de visión por computador junto con el procesamiento de lenguaje natural de forma que se pueda generar una descripción de lo más acertada posible. En la siguiente tabla se muestran algunos de los trabajos más prominentes relacionados a esta área. Posterior, en este mismo trabajo se profundiza más sobre el tema.

Nombre	Fuente	Fecha	Novedad
Show, Attend and Tell: Neural Image Caption Generation with Visual Attention	Articulo académico en arXiv	10 de febrero de 2015	Permite describir imágenes utilizando mapas de atención de manera que se genera una oración con más sentido, el principal componente dentro de esta investigación se basa en modelos de atención.
Image Captioning with Semantic Attention	Articulo académico en ArXiv	12 de marzo de 2016	Este articulo académico busca solucionar los problemas que existían en el artículo anterior. En esta propuesta se une las técnicas Top-Down y Bottom-Up junto con modelos de atención logrando superar a modelos que tenían resultados state-of-the-art.

Tabla 3 Herramientas para la descripción de imágenes.

5. Marco metodológico

El proyecto consta de seis fases.

a. La **primera** es el estudiar y conocer los fundamentos de la accesibilidad web y cómo esta se relaciona como los metadatos educativos.

b. La **segunda** fase trata sobre adquisición de información multimedia, en concreto imágenes y datasets que formen parte de los delineamientos de la propuesta establecidos en la ilustración 2.

c. La **tercera** fase es el estudio de técnicas de clasificación de imágenes enfocadas al campo de Deep learning incluyendo las limitantes de esta como es el caso de la clasificación de imágenes de grano fino en las cuales se opta por estrategias avanzadas con la finalidad de mejorar la precisión de un modelo de machine learning.

d. La **cuarta** fase consta en el diseño de redes neuronales artificiales que logren clasificar correctamente las imágenes pasando capa por capa, desde la capa más general hasta la más específica, esta fase también establece un punto clave dentro de la descripción de contenidos, la descripción de tablas de información, ya que estos tipos imágenes normalmente están compuestas de información en forma de texto. Esta fase también comprende el desarrollo de una red neuronal artificial para la descripción de imágenes en base a un determinado contexto pudiendo ser este un texto o de manera directa en base a una imagen.

e. La **quinta** fase establece el proceso de validación del proyecto de Deep learning basándose en el uso de un conjunto de imágenes que no hayan sido utilizadas dentro del ámbito del entrenamiento de la red neuronal de tal manera que se pueda establecer cuán preciso es el modelo propuesto.

5.1. ¿Qué es la accesibilidad web?

La accesibilidad web es el permitir que la mayor cantidad posible de personas puedan navegar a través de internet sin depender de otras personas para hacerlo. Generalmente esto se asocia a personas que tienen discapacidades, pero no se limita a ellos si no a todos a quienes requieran de ayuda para acceder a los servicios que ofrece internet.

Un sitio web puede ser considerado accesible cuando cumpla con determinados lineamientos establecidos por la WAI (Web Accesibility Initiative), rama que forma parte de la W3C que se dedica a establecer reglas sobre cómo debe estar implementada una página web. Otro factor importante para determinar si una página es accesible es el que una persona con discapacidad pueda acceder a la información presentada en la página de la misma forma en la que lo haría una persona sin dicha discapacidad.

Se estima que aproximadamente el 20% de la población mundial tiene algún tipo de discapacidad, esto significa que las páginas web que no son accesibles excluyen desde el 5% al 20% de sus clientes potenciales debido a que no cumplen con las normas de accesibilidad establecidas. Este hecho no se limita únicamente a las empresas comerciales sino también a las instituciones educativas que hacen caso omiso de implementar la información en la web (Web AIM, 2020).

Principales categorías de tipos de discapacidad:

Ilustración 3 Principales tipos de discapacidad (Pathak, 2021).

Cada una de estas categorías requiere de una determinada adaptación dentro del contenido web, pero, si bien es cierto que este proceso puede ser largo, a la final no solo se beneficia a las personas con discapacidad sino a todos lo que requieran de acceder a la información. Por ejemplo, una página web correctamente estructurada, con información limpia, navegación clara, imágenes con texto alternativo es de mucha más utilidad que una página web que no cumple con eso. Claro que este es un ejemplo simple de accesibilidad y no está directamente ligado a una discapacidad en específico aun así es de mucha utilidad para cualquiera que quisiera acceder a dicho sitio web. Un ejemplo más específico que no solo beneficia a las personas con discapacidad sino a todos es los videos subtitulados y las imágenes con texto alternativo. Un video subtitulado puede ser de mucha utilidad para quienes no dispongan de un sistema de audio en su computador, pero deban de acceder a dicha información, mientras que una imagen con texto alternativo es de gran ayuda en el caso de que la imagen no se presente correctamente u ocurra cierto problema al cargarla.

Accesibilidad, objetos de aprendizaje y su relación con la educación.

Previo a determinar la relación que existe entre la accesibilidad web y los objetos de aprendizaje debemos definir este último.

Un objeto de aprendizaje es un conjunto de información interactiva, digital y reutilizable que permite el aprendizaje a través de contenido educativo. Los objetos de

aprendizaje son almacenados en metadatos, los metadatos permiten que un objeto de aprendizaje sea reutilizable y de fácil distribución.

Dentro de las características principales que un objeto de aprendizaje debe contener están:

- **Reusabilidad**:

  Como su nombre lo indica permite que un objeto de aprendizaje pueda ser utilizado como base para algo más, por ejemplo, un objeto de aprendizaje generado para el área de ciencias naturales puede ser utilizado dentro del área de biología. Esta característica facilita en gran medida la creación de nuevos objetos de aprendizaje ya que no se debe empezar desde cero o generar información redundante cuando existe otro que ya contiene dicha información.

- **Durabilidad y de fácil actualización:**

  Un objeto de aprendizaje debe ser durable, esto quiere decir que puede ser almacenado dentro de repositorios para que luego cualquier otra persona que quiera acceder a este pueda hacerlo de una manera fácil, esto se logra gracias a los metadatos, que son fundamentales para almacenar información de este tipo. Mientras que la durabilidad se enfoca en la persistencia del objeto de aprendizaje, la fácil actualización hace mención de que un objeto de conocimiento podrá ser fácilmente editado y mejorado conforme pase el tiempo sin que este proceso presente mayor dificultad tanto para su autor como para otros usuarios.

- **Autocontenibilidad**:

  La autocontenibilidad hace referencia a que la información presente en un objeto de conocimiento deberá contener toda la información necesaria para que se pueda cumplir con los objetivos propuestos, por lo que el objeto de aprendizaje propuesto será útil tanto para entornos presenciales como virtuales (Universidad de Antioquia).

En base al conocimiento previo de accesibilidad web y objetos de aprendizaje se puede establecer la relación que tienen estos dos.

Como se definió anteriormente un objeto de aprendizaje es una herramienta educativa digital, por lo que al ser digital está presente dentro de sitios web que pueden ser accedidos de manera directa por los estudiantes y profesores. Al ser recursos educativos virtuales estas presentan las limitaciones a las que están sujetos todos los sitios web, los problemas

de accesibilidad. Previamente se dijo que la accesibilidad afecta principalmente a las personas con discapacidad, esto a su vez representa una gran cantidad de personas.

Según WHO (World Health Organization) más de 1 billón (mil millones) de personas sufren de algún tipo de discapacidad. Dentro de esa cantidad, entre 93 a 150 millones son niños, estas dificultades duplican el chance de que esa persona nunca entre a estudiar debido al reto de adaptarse a un entorno complejo por sus limitaciones (Their world, 2021).

En respuesta a la problemática del difícil acceso a los recursos de aprendizaje diferentes organizaciones han propuesto distintas soluciones entre las más importantes se encuentran:

- **IMS Accessibility Specifications:**

  El consorcio global de aprendizaje IMS busca mejorar la enseñanza a través de medios digitales permitiendo el fácil acceso a los recursos educativos para todas las personas. Dentro de IMS existe un grupo de actividades relacionadas a la accesibilidad que se enfocan en la adaptación y personalización de recursos educativos para satisfacer las necesidades de todos, de esa manera se evita la segregación en la forma en la que un estudiante aprende con respecto a sus capacidades físicas (IMS Global Learning Consortium, 2021).

- **WCAG 2.0** (W3C, 2020)**:**

  La WCAG en su versión 2.0 establece 4 principios claves para que un recurso de aprendizaje sea accesible:

  h. Perceptible:

  La información y los componentes de la interfaz de usuario deberán ser fácilmente perceptibles.

  i. Operable:

  Los componentes de la interfaz de usuario y la navegación deberán ser navegables, de forma que no exista determinada información o componente que no pueda ser accedido a través de la interacción tradicional.

  j. Entendible:

  Tanto la información presentada en pantalla como los componentes que forman parte de la interfaz de usuario deben ser entendibles y de fácil comprensión evitando la ofuscación y la mala interpretación del contenido presentado.

k. Robusto:

El contenido deberá ser lo suficientemente robusto para soportar diferentes tecnologías de apoyo, además de que el contenido pueda adaptarse a nuevas tecnologías de manera relativamente fácil.

5.2. ¿Qué son los metadatos y como se aplican dentro de la web?

Los metadatos son fundamentales para la educación, ya sea en ambientes presenciales como virtuales, ya que estos permiten principalmente el almacenamiento y tratamiento de datos, en palabras simples los metadatos son **información acerca de información**, esto quiere decir que los metadatos almacenan información importante sobre un determinado archivo.

La importancia de los metadatos yace en que permite encontrar, usar, preservar y reutilizar los datos en un futuro. Esto es fundamental en diferentes áreas, ya que debido a la gran cantidad de información que existe en internet esta debe estar estructurada de manera que no represente un reto organizarla o utilizarla.

Los metadatos están organizados en tres tipos:

l. Descriptivos:

Contiene información sobre el contenido y el contexto en relación con los datos almacenados, por ejemplo, el título, su descripción, etc.

m. Estructurales:

Contiene información de cómo está compuesto un determinado archivo, por ejemplo, el formato de archivo o su relación con otros archivos.

n. Administrativos:

Los metadatos administrativos tienen como objetivo almacenar información sobre cómo es administrado cierto archivo, por ejemplo, su fecha de creación, su autor, el software requerido para que funcione, etc.

A partir de lo anterior podemos deducir una clara relación entre los metadatos y la información contenida en la web, ya que la web contiene prácticamente toda la información que pueda ser imaginada, debido a esto los datos contenidos en internet deben estar relacionados con sus debidos metadatos, a través de los metadatos podemos indexar, buscar a acceder a los diferentes portales web que contienen la información que nosotros estamos buscando.

El internet sin metadatos sería un conjunto inmenso de datos a los cuales sería casi imposible acceder debido a su gran volumen y falta de estructuración.

Importancia de los metadatos dentro de la educación.

Como se definió anteriormente un objeto de aprendizaje contiene información relevante para el ámbito educativo, este tipo de información al ser digital debe ser contenido dentro de metadatos, ya que a través de los metadatos se puede crear, organizar, utilizar y reutilizar los objetos de aprendizaje conforme sea necesario.

En la relación entre metadatos y objetos de aprendizaje existe un modelo que es utilizado para describir un objeto de aprendizaje, IEEE LOM (IEEE Learning Object Metadata) permite definir cómo está compuesto un determinado objeto de aprendizaje, todos los datos generados dentro de este modelo son almacenados en forma de XML que representa al metadato relacionado al recurso de aprendizaje.

5.3. Adquisición de información multimedia.

Para la adquisición de las imágenes multimedia se optaron por diferentes datasets, las fuentes varían dependiendo del tipo de objeto de aprendizaje con el que se desee trabajar, en la siguiente tabla se enumeran los datasets usados junto con su respectiva fuente.

Nombre	Tamaño	Fuente	Descripción
Flickr8k	8.000 imágenes.	https://www.kaggle.com/adityajn105/flickr8k	Conjunto de imágenes para la descripción de imágenes.
Flickr30k	~30.000 imágenes.	https://www.kaggle.com/adityajn105/flickr30k	Conjunto de imágenes para la descripción de imágenes.
Coco Dataset 2017	118.000 imágenes.	https://cocodataset.org/#home	Conjunto de imágenes de la vida real utilizadas para benchmarking en la descripción de imágenes.

OpenLogo	352 clases con ~27.000 imágenes.	https://qmul-openlogo.github.io/	Conjunto de logos sin preprocesar que requieren un tratamiento con técnicas de visón por computador para ser entrenadas.
DomainNet	Contiene 345 categorías de diferentes imágenes.	https://paperswithcode.com/dataset/domainnet	Conjunto de imágenes para la clasificación por dominio. De aquí se derivan ciertas imágenes para más de una categoría.
Office-Home	Contiene 65 categorías con ~15.500 imágenes.	https://paperswithcode.com/dataset/office-home	Conjunto de imágenes para la clasificación de dominio.
WebScreenshots	20.000 imágenes.	https://www.kaggle.com/aydosphd/webscreenshots	Conjunto de imágenes generadas por computador que caen en la categoría Screenshot.
Dataset generado usando Google.	50.000 imágenes tomadas de Google.	Dataset no disponible.	Se obtuvieron imágenes utilizando Google Image debido a que no existen determinados datasets para ciertas áreas de clasificación.

Tabla 4 Datasets utilizados para el entrenamiento de los modelos.

5.4. Estudio de técnicas de clasificación de imágenes.

Fundamentándonos en lo que ya se ha descrito sobre las diferentes arquitecturas de redes neuronales y el objetivo de estas dentro de este trabajo es correcto decir que el proceso de entrenamiento y pruebas puede llegarnos a tomar una gran cantidad de tiempo esto independientemente de los recursos que tengamos disponibles tanto en hardware como software. Sin embargo, el tiempo no es el único factor determinante dentro del entrenamiento de redes neuronales, sino también la precisión que estas generen, ya que a fin de cuentas lo que se busca es que un modelo genere la precisión más alta al momento de clasificar imágenes. En consecuencia, se estudió diferentes técnicas que permitan solventar los anteriores problemas, entre las más destacadas se encuentran:

5.4.1. **Transfer Learning** (Marcelino, 2018)**:**

Dentro de las redes neuronales se suele trabajar con conjuntos de datos considerables, esto varia acorde a la tarea que vayamos a realizar, pero, comúnmente se suele tener entre miles a millones de registros de información con los cuales se va a trabajar, sin embargo, que pasaría si ¿El conjunto de datos es demasiado pequeño?, en el caso de que se llegue a presentar este problema se suele llegar a generar un sobre entrenamiento u overfitting, este problema hace que una red neuronal, en este caso convolucional, genere resultados iguales para entradas diferentes, por ejemplo, si se desea clasificar diferentes categorías establecidas en la ilustración 2 la mayoría de veces devolverá una categoría X cuando en realidad debía devolver una categoría Y, como ejemplo se puede decir que el 50% de veces devolvió X en lugar de la categoría W, Y o Z.

Sin embargo, el tamaño del dataset no es el único problema sino también el tiempo de entrenamiento que esto conlleva, por ejemplo, en el mejor de los casos si existe un dataset de Ilustraciones (Fotografías, Diagramas e Imágenes digitales) con cientos de miles de imágenes la red neuronal no tendrá mayor problema en lograr entrenar y clasificar dichas imágenes, no obstante, este proceso puede llegar a tomar grandes cantidades de tiempo en relación con los parámetros que hayan sido seleccionados, pero en el mejor de los casos el entrenamiento tardaría entre 12 horas a más de un día. Si bien parece que el tiempo empleado no es mucho, existen casos en los cuales forma un factor determinante en el éxito de una investigación.

I. Historia:

En 1976 se publicó un artículo científico en el cual se hablaba sobre una técnica que podría beneficiar al aprendizaje de las redes neuronales y reducir el tiempo de

entrenamiento, pero para ese momento la técnica propuesta no tenía tanta relevancia debido a que no se demostraba sus beneficios en un nivel técnico (Bozinovski, 2020).

Desde 1976 no ocurrieron grandes avances hasta que en 1993 se publicó un artículo en el cual se demostró que efectivamente la técnica de Transfer Learning funcionaba a través del traspaso de los pesos de una red neuronal entrenada a otra cuando existían tareas relacionadas de por medio, esto dio paso a que las redes neuronales nuevas puedan ser mucho más rápidas que las tradicionales que iniciaban con pesos previamente establecidos (Karim Barznji, 1997).

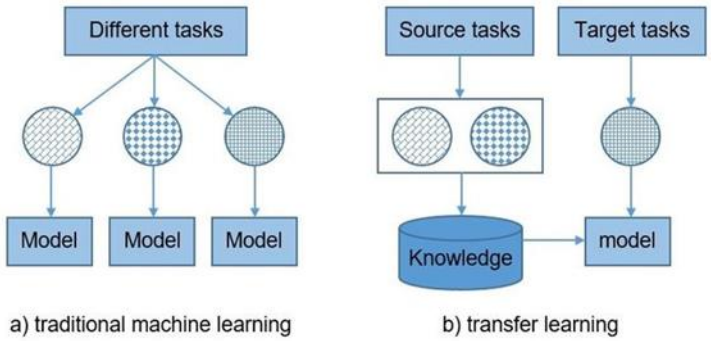


Ilustración 4 Entrenamiento tradicional vs. Transfer Learning (Duong, Tran, & Xuan, 2019).

II. Proceso general de entrenamiento:

Antes de iniciar con la descripción del entrenamiento de los modelos de redes neuronales se debe especificar cuáles son las características del equipo que fue utilizado, cabe recalcar que el equipo es propiedad de la Universidad Politécnica Salesiana.

Memoria RAM:	**15.5GB**
Procesador:	**AMD Ryzen 7 3700 x 8 núcleos.**
Gráficos:	**NVIDIA GeForce RTX 2060 Super.**
Capacidad de SSD:	**512GB.**

Es también necesario establecer las librerías que se utilizaron dentro del proyecto para el entrenamiento de la red, así como para el preprocesamiento de información:

a. **Python (Como lenguaje de programación)**
b. **Tensorflow**

c. **Keras**
d. **OpenCV**

Para entrenar una red neuronal primero se requiere de disponer de un conjunto de datos.

Consecuente al punto anterior se selecciona un modelo previamente entrenado, para esto disponemos de diferentes alternativas dentro de la librería Keras, como, por ejemplo, ResNet50, este modelo es ampliamente usado dentro del estudio de transfer learning debido a su versatilidad y facilidad de uso. Un punto importante dentro de este proceso es que debemos seleccionar una red neuronal que haya sido usada en una tarea relacionada con la que vamos a trabajar. En la siguiente matriz de similitud nos explica en base a que parámetros debemos seleccionar nuestra red neuronal.

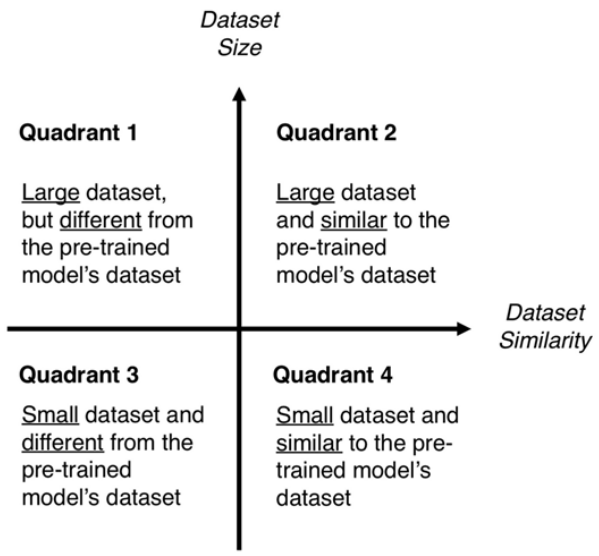


Ilustración 5 Matriz de similitud Tamaño vs Similitud de modelo (Marcelino, 2018).

En la ilustración 5 se explica cómo se debe seleccionar una red neuronal acorde a los cuatro cuadrantes.

a. **Cuadrante 1:**

   Cuando se trabaje con un dataset grande no importa si la red neuronal para la transferencia no tenga mucha relación con el trabajo que se vaya a realizar igual va a ser útil.

b. **Cuadrante 2:**

Al trabajar con un dataset grande y una red neuronal pre entrenada con un dataset similar entonces, en teoría, se puede hacer lo que se quiera, ya que ambas redes son similares y además trabajan con información parecida, por lo que no existe un riesgo mayor de overfitting.

c. Cuadrante 3:

Este es el peor caso en el cual se trabaja con un dataset pequeño y se tiene una red neuronal que no es casi en nada similar con la que trabajamos, por lo que, si se desea obtener buenos resultados es mejor optar por usar otras técnicas como Data Augmentation (esto se da en el preprocesamiento), a través de esto se puede mejorar los resultados dentro de este cuadrante.

d. Cuadrante 4:

El cuarto cuadrante es un equilibrio entre la un dataset pequeño y un modelo de red neuronal que trabajo sobre un problema similar, este es el caso más común en el cual se aplica transfer learning, para hacer esto se debe eliminar la última capa del modelo y congelar el resto de las capas indica (Marcelino, 2018) en su artículo web.

Cuando ya se haya determinado cual es la red neuronal adecuada para el entrenamiento se debe preprocesar el conjunto de datos, imágenes en este caso, de manera que puedan ser entrenados en la red, esto se debe a que cada modelo de red neuronal tiene diferentes parámetros de entrada, ya sean el tamaño de las imágenes o la cantidad de canales.

Finalmente, cuando se haya preprocesado las imágenes se procede a entrenar el modelo, con el tiempo se logrará observar que el entrenamiento a través de transfer learning conlleva a mejores resultados tanto en precisión como rendimiento en comparación a entrenar desde cero un modelo de red neuronal.

5.4.2. Vision Transformers (ViT):

Una de las habilidades del ser humano es el proceso cognitivo de la atención, este proceso se trata de seleccionar y enfocarse en un punto específico de una determinada tarea. Sin duda esta habilidad es sumamente útil dentro del campo visual ya que nos enfocamos únicamente en lo que nos interesa y dejamos de lado a la información irrelevante.

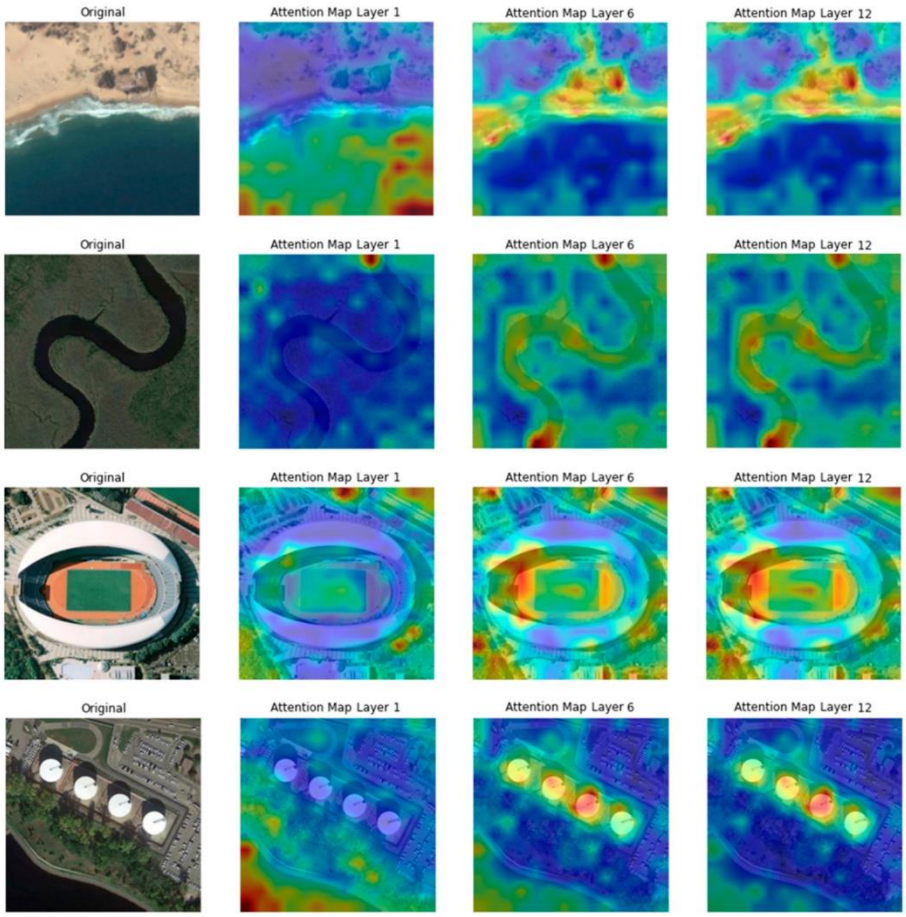


Ilustración 6 Mapa de atención en ViT (Bazi, Bashmal, Al Rahhal, Al Dayil, & Al Ajlan, 2021).

En la ilustración 6 se muestra cómo funcionan los mapas de atención dentro de los transformadores de visión, en la primera columna se muestra la imagen original, desde la segunda columna hasta la cuarta columna con sus respectivos resultados en base a la capa en la que se encuentra. Los resultados por cada capa se llaman mapas de atención, cada mapa de atención muestra cual es la parte más representativa de la imagen, en el ejemplo anterior es más fácil apreciar que la cuarta imagen en su cuarta columna nos muestra cuatro puntos de color amarillo o rojo, esto nos indica que esa parte de la imagen es la que mayor importancia tiene mientras que lo que este marcado de azul no tiene mayor relevancia.

I. Historia:

Hasta el año 2017 en el procesamiento de lenguaje natural se usaban redes neuronales recurrentes y redes de largo-corto plazo al ser estas las que mejores resultados presentaban en esta área, sin embargo, en el año 2017 se presentó el artículo

(Vaswani, et al., 2017), en este artículo se presenta una nueva alternativa hacia las RNN, los Transformers. Los Transformers presentan una alternativa prometedora frente a las RNN o CNN tradicionales dentro del área de NLP, en primer lugar, debido a que los Transformers permiten un nivel mayor de paralelización, esto quiere decir que se pueden utilizar múltiples GPU's o CPU's para reducir el tiempo de entrenamiento. En segundo lugar, los Transformers miden la relación entre pares de tokens (palabras) independiente de cuán lejos estén las palabras.

Debido al impacto de los transformers dentro del área de procesamiento de lenguaje natural se procedió a implementar una versión de los transformers para el área de visión por computador, a esta nueva técnica se la conoce como ViT o Vision Transformers. La técnica de ViT fue introducida en el año 2020 por lo que puede decirse que es relativamente nueva. En el artículo (Dosovitskiy, et al., An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale, 2020) se explica en cómo las imágenes pueden ser tratadas de manera similar a las palabras para el uso con transformadores, en general la idea detrás de ViT es dividir una imagen en pedazos pequeños, luego de dividir la imagen calcula la relación que tienen los 'pedazos' de imágenes de la misma forma que hiciera con vectores de palabras. No obstante, la principal diferencia entre los transformadores originales y los ViT es que los transformadores visuales no hacen una comparación a nivel de píxeles sino a nivel de 'pedazo' de imagen ya que si se optara a nivel de píxel el costo computacional sería demasiado alto, por el otro lado los transformadores originales calculan la similitud a por cada vector que representa a una palabra.

II. Proceso general de las ViT *(Dosovitskiy, et al., An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale, 2020)*:

1. Se divide la imagen en pedazos de igual tamaño, a cada pedazo se lo conoce como token.

2. Al conjunto de tokens se los aplana (flattening) para pasarlos de 2-D a 1-D.

3. Se vincula cada token con su posición y la clase a la que pertenecen. Se podría decir que se genera una tupla entre el token, la posición y la clase.

4. Los tokens pasan a través del transformer, el transformer es el encargado de determinar la similitud entre los tokens de una imagen, dentro del transformer se realiza el siguiente proceso:

    1. **Encoder attention:**
       Se realiza el proceso de atención sobre el token actual.

    2. **Decoder attention:**
       Se realiza el proceso de atención sobre el token objetivo (token siguiente).

3. **Encoder-Decoder attention:**
   Se realiza el proceso de atención sobre el token de ingreso (token anterior).

El proceso de atención dentro del transformer se realiza de forma paralela sobre todo los tokens a través del Multi-Headed Self-Attention.

III. Desventajas de las ViT sobre las Redes Neuronales Convolucionales:

- Las ViT requieren de un conjunto de imágenes relativamente grande en comparación a las CNN tradicionales, para que una ViT iguale o supere a las mejores CNN debe trabajar con al menos 14 millones de imágenes. Sin embargo, esto puede ser solapado a través de Transfer Learning o Fine Tuning sea el caso respectivo (Khan, et al., 2021).
- Las CNN al ser un tipo de red neuronal artificial relativamente antigua (apareció aproximadamente en 1980) es lo suficientemente madura como para ser utiliza por diferentes frameworks y a su vez puede ser optimizada de diferentes formas, ya sea a través de sus hiper parámetros, optimizadores como SGD, data augmentation, entre otros. Por el otro lado los ViT al ser una técnica reciente (aparece en 2020) no es tan fácil de afinar sus hiper parámetros, esto se debe a que el primer proceso dentro de una ViT es el dividir una imagen en patches o tokens por lo que, al estar subdividida en múltiples bloques, generalmente de 16x16 se hace complejo el tratar de optimizar por el tamaño del bloque y la cantidad de esta. Para solucionar este problema en el artículo (Xiao, et al., 2021) se propone reemplazar la capa de patching por capas convolutivas de forma que sea relativamente fácil optimizar y mejorar los resultados.

5.5. Artículos de interés en relación con la investigación:

Dentro del área de aprendizaje profundo existen diferentes herramientas y técnicas que van apareciendo en conjunto con el tiempo, el objetivo de las nuevas técnicas es, por su puesto, generar los mejores resultados superando a técnicas anteriores que son conocidas como *'state of the art'* que en palabras simples significa una investigación que incorpora las técnicas más novedosas junto a los mejores resultados.

En los siguientes artículos se presentan técnicas novedosas que fueron consideradas dentro de este trabajo, sin embargo, solo pocas herramientas fueron implementadas, ya sea debido a la falta de tiempo o a su baja relación resultado-complejidad.

5.5.1. Multi-path Convolutional Neural Networks for Complex Image Classification

(Wang, Mingming;University, Dalhousie, 2015) **:**

En este artículo se menciona como las redes neuronales convolucionales tradicionales no puede trabajar con imágenes muy complejas. En grosso modo se podría decir que una red neuronal convolucional obtiene las características principales de una imagen y en base a eso logra identificar la diferencia entre las diferentes clases de imágenes disponibles. Sin embargo, este proceso no siempre se puede llevar a cabo cuando existen imágenes que tienen múltiples características que solapan a la característica que se busca, por ejemplo, en un campo de flores se busca clasificar cuales son rosas, pero debido a que existen otros elementos a lado de esa flor el proceso no puede llevarse a cabo. En la siguiente imagen se ejemplifica mejor la misma clase de un pez tomado del dataset ImageNet.

Ilustración 7 Pez tenca (Imagen simple) (Deng, 2009).

Ilustración 8 Pez tenca (Imagen compleja) (Deng, 2009).

En la ilustración 7 y ilustración 8 se muestra a la misma clase de pescado (tenca), sin embargo, en el primer caso es fácilmente diferenciable ya que, se puede decir, solo se encuentra el pez y ningún otro objeto o ruido que altere el proceso de clasificación de una red neuronal. En cambio en la ilustración 8 ya no es tan fácil determinar el pez dentro de la imagen, al menos no para una red neuronal, ya que en la izquierda el color del pez se solapa con el del entorno, en el caso de la imagen de la derecha existen dos características principales que confunden a la red neuronal, la primera es una persona y la segunda son los pescados, estas dobles características hacen que la red convolucional no logre sacar las características correctas llevando a un proceso de aprendizaje fallido.

Una vez establecido las limitaciones de las CNN se puede decir que se requiere de una alternativa para solventar dicha limitación. Para solucionar este problema el presente articulo propone lo siguiente.

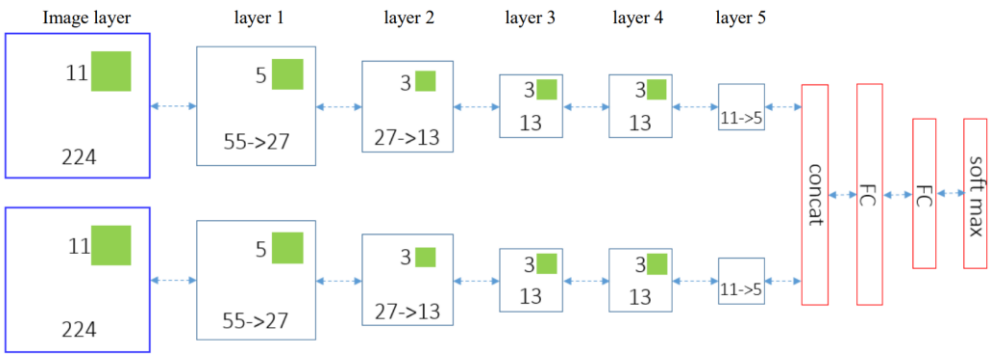


Ilustración 9 Arquitectura de red neuronal convolucional multi-path (Wang, Mingming;University, Dalhousie, 2015).

En general la idea de solución para evitar que características sin importancia oculten a las características principales es optar por una arquitectura de red de doble entrada. En la ilustración 9 se establece brevemente como está formado esta arquitectura.

En la entrada de arriba se usa la imagen original, sin ningún cambio ni alteración y luego se trabaja con ella a través de con conjunto de capas de filtrado y pooling.

En la entrada de abajo se trabaja de manera distinta, primero la imagen procesada y pasa a través de un filtro bilateral, este filtro tiene la ventaja de que preserva los bordes y reduce el ruido lo que lo hace ideal para extraer características que sean notorias en el primer plano dejando de lado el fondo de la imagen. Luego de esto la imagen es tratada de la misma manera que en la parte superior.

Finalmente, cuando se completa tanto el proceso de los dos caminos se concatenan las conexiones de ambos caminos y se termina el proceso con capas completamente conectadas (fully conected layer).

Según los autores del presente artículo, esta nueva técnica reduce el rango de error top-1 de 66.5% a 64.2% dentro reto Larga Scale Visual Recognition Challenge 2013.

5.5.2. **Squeeze and Excitation Networks** (Hu, Shen, Albanie, Sun, & Wu, 2017)**:**

Si bien en el artículo anterior se estableció una técnica novedosa para lograr clasificar imágenes complejas en este artículo se busca obtener mejores resultados sobre cualquier dataset en general. En este caso se propone el agregar un bloque llamado SE (Squeeze and Excitation), este bloque ha demostrado que puede mejorar el porcentaje resultante en 25% en la competencia ImageNet con respecto a los resultados del año 2016.

La idea detrás de esta técnica es agregar un único parámetro por cada canal de bloque convolucional de tal forma que la red pueda ajustarse a los pesos del mapa de

características. En palabras simples lo que hace el bloque SE es agregar un nuevo parámetro a cada canal convolutivo diciendo que tan importante es este.

Para entender mejor el enunciado anterior primero se debe establecer que son los bloques convolutivos. Estos bloques extraen información de las imágenes, las capas inferiores compuestas por estos bloques son los encargados de extraer información trivial de una imagen como sus bordes, frecuencias, etc. Mientras que las capas iniciales compuestas por los bloques convolutivos se encargan de sacar información significativa, figuras complejas, rostros, texto, etc. Sin embargo, la red no puede saber que tan importante es la característica que acaba de obtener, por lo que el bloque de SE es el encargado de determinar el 'peso' de esa característica.

Los autores de este articulo optaron por aplicar el bloque de SE dentro de la arquitectura ResNet obteniendo mejores resultados que la arquitectura original. Ellos aseguran que aplicando los bloques SE dentro de la variante ResNet-50 logra aproximarse al resultado obtenido en la variante ResNet-101 con la mitad de costo computacional de esta última variante.

En la siguiente imagen se demuestra cómo está formado un bloque del modelo ResNet-50 y a su derecha un bloque SE adaptado al bloque original. Nótese que el cambio no es complejo, pero, pese a eso, genera mejores resultados.

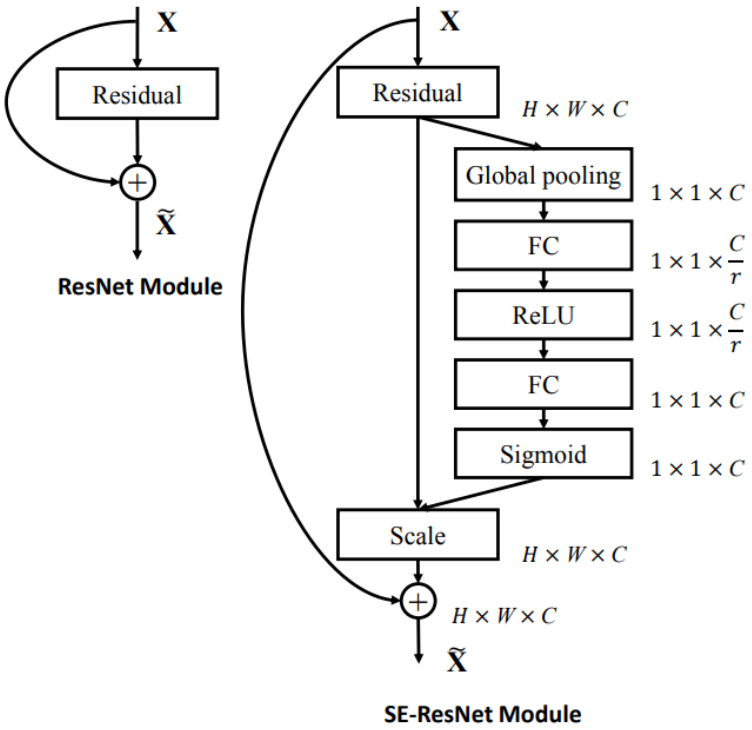


Ilustración 10 Bloque de ResNet-50 original vs bloque ResNet-50 con Squeeze-and-Excitation (Hu, Shen, Albanie, Sun, & Wu, 2017).

5.5.3. Looking for the Devil in the Details: Learning Trilinear Attention Sampling Network for Fine-Grained Image Classification (Zheng, Fu, Zha, & Luo, 2019):

El algoritmo propuesto en este articulo busca solventar los problemas de categorización de imágenes de grano fino (FGVC Fine Grained Visual Categorization), los problemas presentados por los algoritmos de FGVC son los siguientes:

1. El número de módulos de atención son limitados y predefinidos lo que restringe la flexibilidad de los modelos de clasificación.
2. En los modelos de atención implementados hasta la fecha (del articulo) no trabajan con anotaciones sobre las partes de las imágenes con las que se trabajó por lo que no se aprende de manera consistente.
3. Los modelos tradicionales de atención entrenan cada pedazo de imagen en una CNN lo que lleva a cuellos de botella.

Para resolver los problemas mencionados en la parte anterior se propone un método conocido como Muestreo de Atención Trilineal (Trilinear Attention Sampling) que funciona de la siguiente manera:

a. Localización de detalles usando atención trilineal:

Dada una imagen I se extrae los mapas de características de la imagen a través de la exposición de esta en varias capas convolucionales. Para lograr esto se usa el modelo ResNet-18 cambiando ciertos parámetros de este.

Los mapas de características son transformados en mapas de atención a través del módulo de atención trilineal. La fórmula de la función trilineal es:

$$M(X) \coloneqq N(N(X)X^T)X$$

Los elementos de la función anterior son los siguientes:

$N(\cdot) \rightarrow$ Denota la normalización Softmax sobre la segunda dimensión de una matriz de valores.

$X \rightarrow$ Indica un mapa de características obtenido de una capa convolucional.

$XX^T \rightarrow$ Denota la relación entre canales de mapas de características.

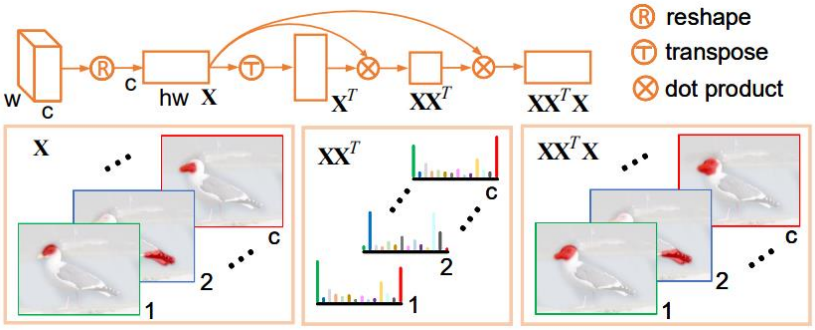


Ilustración 11 Producto Trilineal (Zheng, Fu, Zha, & Luo, 2019).

b. Extracción de detalles usando muestreo de atención:

En la segunda subsección se utilizada el modelo de atención propuesto en el cual se ingresan el mapa de atención trilineal junto con la imagen original. El resultado de este proceso son dos imágenes:

1. El primer resultado es una imagen que mantiene la estructura de la imagen.
2. El segundo resultado es una imagen que mantiene los detalles.

La diferencia entre los dos resultados (dos imágenes obtenidas por el proceso de atención) es que la imagen que mantiene la estructura captura los detalles importantes dentro de la imagen, es decir esta imagen elimina regiones que no tienen detalles de grano fino.

La segunda imagen se enfoca en únicamente en una parte de la imagen intentando preservar la mayor cantidad de detalles en una determinada región en lugar de hacer sobre toda la imagen.

c. Optimización de detalles usando destilación de conocimiento:

En base al proceso anterior que retorna dos imágenes (estructural y basada en detalles) se puede utilizar transfer learning para enlazar el conocimiento de ambas imágenes un una sola, para esto primero se pasa cada uno de los dos resultados a una red backbone como, por ejemplo, ResNet50, el modelo de red neuronal devuelve las características de ambas imágenes y en base a los resultados de los pesos se hace transfer learning de manera en que se vinculen las características de ambas imágenes en una especie de enseñanza maestro-estudiante.

Finalmente, el resultado pasa a través de un clasificador Softmax para determinar la probabilidad sobre cada una de las clases entrenadas.

Para las pruebas se trabajaron sobre diferentes datasets, pero en general se optó por uno de los más complejos dentro del área de clasificación de grano fino, el dataset iNaturalist 2017 y se obtuvieron los siguientes resultados.

Super Class	# Class	Resnet [8]	SSN [22]	TASN
Plantae	2101	60.3	63.9	**66.6**
Insecta	1021	69.1	74.7	**77.6**
Aves	964	59.1	68.2	**72.0**
Reptilia	289	37.4	43.9	**46.4**
Mammalia	186	50.2	55.3	**57.7**
Fungi	121	62.5	64.2	**70.3**
Amphibia	115	41.8	50.2	**51.6**
Mollusca	93	56.9	61.5	**64.7**
Animalia	77	64.8	67.8	**71.0**
Arachnida	56	64.8	73.8	**75.1**
Actinopterygii	53	57.0	60.3	**65.5**
Chromista	9	57.6	57.6	**62.5**
Protozoa	4	78.1	79.5	**79.5**
Total	5089	59.6	65.2	**68.2**

Ilustración 12 Resultados de TASN sobre el dataset iNaturalist 2017 (Zheng, Fu, Zha, & Luo, 2019).

5.5.4. Look Closer to See Better: Recurrent Attention Convolutional Neural Network for Fine-Grained Image Recognition (Fu, Zheng, & Mei, 2017):

Este articulo presenta un método novedoso para la clasificación de imágenes a través de técnicas del uso de técnicas de atención de forma que en base a una imagen original se busca el mapa de atención, posteriormente se hace acercamiento hacia la parte más significativa del mapa y recorta la imagen en forma de que queda únicamente la región significativa mostrando mejores resultados que un entrenamiento clásico sobre toda la imagen.

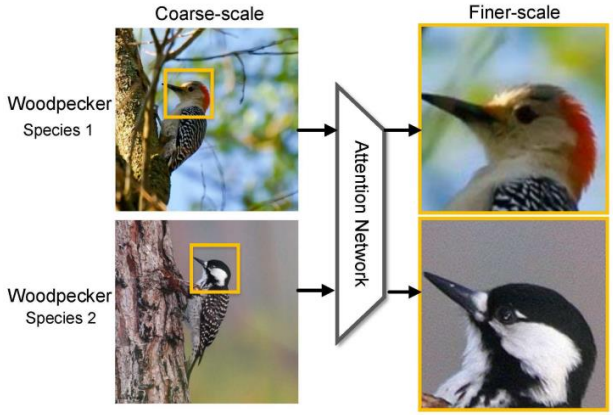


Ilustración 13 Clasificación de imágenes basada en mapas de atención y acercamiento de imagen (Fu, Zheng, & Mei, 2017).

El proceso de este método se divide en tres partes en las cuales se divide el proceso de escalado, entrenamiento y clasificación de las clases en un conjunto de datos:

- **Red de atención:**
  a. **Formulación multitarea:**

     Se propone una APN (Red de Atención Propuesta) la cual dada una imagen X primero extrae características utilizando capas convoluciones pre entrenadas. El resultado del proceso anterior son dos salidas:

     La primera salida del modelo es la probabilidad sobre las categorías de grano fino.

     La segunda salida provee de un conjunto de coordenadas que representan la región de atención para el escalamiento, esto también se conoce como el mapa de atención de la región más prominente.

  b. **Localización de atención y amplificación:**

     Una vez ubicada la región de atención hipotética se procede a cortar la imagen y se hace un acercamiento. Para lograr un corte aproximado más preciso se usa la función Boxcar.

     Las ventajas de la función Boxcar es en primer lugar que, permite aproximar de manera precisa el corte hacia las regiones significativas en base a los resultados de la red neuronal. En segundo lugar, la ventaja de esta función es que permite la representación entre regiones de atención y coordenadas las cuales son necesarias para el proceso de backward-propagation.

- **Clasificación y ranking:**

  Como siguiente punto dentro de la clasificación de imágenes se propone la creación de una red neuronal convolucional con módulos de atención. Esta red es optimizada por dos métodos de supervisión:

  a. Perdida de clasificación intra-escala.
  b. Perdida de clasificación por pares entre-escalas.

  El método anterior asegura la habilidad discriminativa en cada escala, lo que se traduce en una mejor predicción de las etiquetas de categorías.

- **Representación conjunta multi-escala:**

  Finalmente, cuando la RA-CNN ha sido entrenada en las escalas calculadas se puede normalizar cada descriptor de obtenido por escala y concatenarlos en una capa completamente conectada. A través de la función

de activación Softmax se procede a predecir la clase que fue calculada en base a los pasos anteriores, sin embargo, este proceso también puede ser realizado utilizando SVM lineales generando resultados muy similares.

5.5.5. **Domain Adaptative Transfer Learning with Specialist Models** (Ngiam, et al., 2018)**:**

En base al conocimiento previo de transfer learning se logra establecer bases sobre como este permite desarrollar modelos de visión por computador de alto rendimiento, sin embargo, existen ciertas limitantes dentro de transfer learning en las cuales esta técnica no es el mejor método de solución directa, sino que antes se debe analizar el problema. En el presente artículo se muestra los hallazgos realizados en un estudio minucioso hacia esta técnica junto con un método novedoso que permite obtener excelentes resultados dentro de la clasificación de grano fino.

Entre los hallazgos encontrados utilizando la técnica de transfer learning se encuentran:

- **Mas datos de preentrenamiento no siempre ayudan:**

  No siempre trabajar con modelos que fueron entrenados con grandes volúmenes de datos van a generar los mejores resultados en transfer learning.

- **Encontrar la distribución correcta del dataset objetivo mejora el transfer learning:**

  Se demuestra que se debe optar por un método que calcule la importancia de los pesos en base a un modelo pre entrenado de manera que se obtiene resultados state-of-the-art en dataset de clasificación de grano fino.

- **Tareas de clasificación de grano fino requieren de entrenamiento con información de grano fino:**

  Se ha demostrado que el rendimiento de un modelo con transfer learning depende directamente de si los datos de entrenamiento tienen características discriminativas relacionadas al modelo original con el que fue entrenado, ya que si esto no ocurre no existen beneficios significativos de utilizar transfer learning en datasets de grano fino.

De manera general el funcionamiento de la técnica presentada en este artículo es evaluar las imágenes con las que se va a trabajar contra un modelo pre entrenado en un dataset determinado, en este caso se usa el dataset JFT que contiene 300M de imágenes y un poco más de mil millones de etiquetas (más de una etiqueta por imagen). Para cada imagen que se pruebe contra el modelo se obtiene el valor de predicción sobre las etiquetas de JFT, con el valor de predicción obtenido se procede a calcular la importancia del peso de la siguiente manera:

$$P_t(y)/P_s(y)$$

El valor de P_t representa al resultado de predicción obtenido evaluando la imagen sobre todas las etiquetas JFT.

El valor de P_s representa a la distribución de etiquetas al momento de entrenamiento sobre el dataset JFT.

Finalmente, el resultado de la formula anterior permite determinar la importancia de los pesos sobre etiqueta, de forma que, se puede entrenar nuevamente el modelo sobre JFT estableciendo la importancia de los pesos. Luego de entrenar nuevamente el modelo con los nuevos pesos se puede utilizar la técnica de fine tuning para ajustar los pesos con el nuevo dataset mejorando considerablemente los resultados y rendimiento que al entrenar una red neuronal desde cero.

Pre-training Method	Target Dataset					
	Birdsnap	Oxford-IIIT Pets	Stanford Cars	FGVC Aircraft	Food-101	CIFAR-10
Entire JFT Dataset	74.2	92.5	94.0	88.2	88.6	97.6
JFT - Bird	80.7	86.4	88.1	74.9	87.5	96.9
JFT - Animal	77.8	96.7	89.1	78.2	89.2	98.1
JFT - Car	73.4	79.8	**96.0**	82.1	86.1	93.0
JFT - Aircraft	73.4	78.7	88.2	91.1	87.1	96.1
JFT - Vehicle	74.2	79.6	95.8	86.8	81.6	96.4
JFT - Transport	74.4	78.4	95.9	88.4	86.9	96.2
JFT - Food	74.9	81.1	90.3	85.6	93.5	96.4
JFT - Adaptive Transfer	**81.7**	**97.1**	95.7	**94.1**	**94.1**	**98.3**
ImageNet - Entire Dataset	77.2	93.3	91.5	88.8	88.7	97.4
ImageNet - Adaptive Transfer	76.6	94.1	92.1	87.8	88.9	97.7
Random Initialization	75.2	80.8	92.1	88.3	86.4	95.7

Ilustración 14 Resultados utilizando el método propuesto en base al dataset JFT y aplicándolo a diferentes datasets de clasificación de grano fino (Ngiam, et al., 2018).

5.5.6. **Techniques for Detecting and Extracting Tabular Data from PDFs and Scanned Documents: A Survey** (Kekare, Jachak, Gosavi, & Hanwate, 2020)**:**

Debido a la gran cantidad de imágenes tabulares generadas por diferentes fuentes como, por ejemplo, bancos, universidades, escuelas, áreas financieras, etc. Se ha visto en la imperativa necesidad de desarrollar herramientas que permitan

extraer la información contenida en dichas imágenes. En el presente artículo se establecen las herramientas más prominentes que han logrado obtener información con una baja tasa de perdida.

- **Tabula:**

  Tabula es una herramienta de código abierto desarrollado originalmente para Java, pero actualmente cuenta con una implementación en Python. Esta herramienta es sumamente poderosa debido a que puede detectar tablas dentro de archivos PDF y luego extrae la tabla para luego procesar y obtener los datos de manera estructurada, sin embargo, la extracción de datos basadas en tablas sin bordes no es precisa ya que comete muchos errores aun así funciona muy bien con tablas bien estructuradas.

- **Camelot:**

  Esta herramienta trabaja de manera similar a Tabula, pero se diferencia de la anterior en que esta presenta mejores resultados obteniendo información de imágenes en lugar de archivos PDF. Aun así, todavía presenta el inconveniente de obtener datos de tablas que carecen de una estructura bien definida.

- **DeepDeSRT** (Schreiber, Agne, Wolf, Dengel, & Ahmed, 2017)**:**

  Esta técnica es diferente a las dos anteriores en relación con que se hace uso de técnicas de Deep learning para el reconocimiento de la estructura de una tabla. Para el entrenamiento usan el dataset Marmot, debido a la falta de datasets públicos para el entrenamiento de redes neuronales para la detección de estructuras de tablas solo se pudo entrenar en base a ese dataset, a pesar de esta limitante, el modelo presente buenos resultados utilizando datos provistos por una empresa europea.

- **TableNet** (Paliwal, D, Rahul, Sharma, & Vig, 2020)**:**

  En el caso anterior del modelo DeepDeSRT donde se generan dos modelos, uno para la detección de la tabla y otro para definir su estructura, en este caso se presenta un único modelo que, en primera instancia, obtiene la tabla y luego en base a su tipo (con bordes o sin bordes) procede a obtener los datos manteniendo su estructura, una ventaja de este modelo es que presenta énfasis en el uso de técnicas novedosas como transfer learning y parameter tuning.

5.5.7. **Show, Attend and Tell: Neural Image Caption Generation with Visual Attention** (Xu, et al., 2015)

Como se menciona en el título del presente trabajo, el objetivo técnico principal del proyecto de titulación es la descripción de imágenes, por lo que, para esta tarea existen diferentes enfoques que pueden ser utilizados, sin embargo, en el presente artículo se establece una técnica novedosa para la época ya que hace uso de modelos de atención que permiten realizar descripción de imágenes con una mayor precisión y una contextualización mejor que un modelo encoder-decoder tradicional.

El objetivo principal de esta técnica se basa en el uso de modelos de atención, que, como ya se mencionó anteriormente en otros artículos, a través de un mapa de atención se puede determinar qué áreas son las más importantes dentro de una imagen y de esa manera se puede trabajar con resultados más precisos. En este caso, se utiliza el mapa de atención para determinar que palabra predomina dentro de un vector de palabras y se relaciona con la imagen de forma que se pueda determinar cuál es la parte más importante por describir. Un ejemplo en la siguiente imagen:

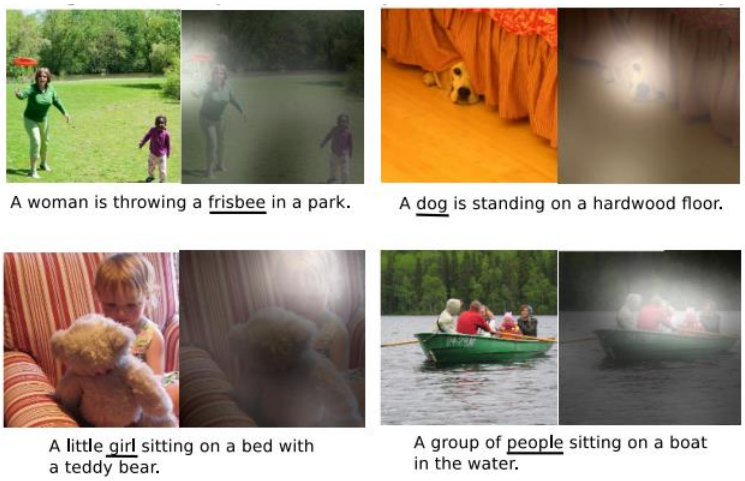


Ilustración 15 Mapa de atención junto con la descripción de la imagen (Xu, et al., 2015).

El presente articulo presenta dos enfoques relacionados pero diferentes, el primero se trata de un aprendizaje ‘duro’ o aprendizaje estocástico, mientras que el segundo enfoque se trata de un aprendizaje ‘suave’ o aprendizaje determinístico. Sin embargo, ambos modelos se basan en la siguiente estructura de red neuronal:

- **Encoder:**

  El codificador o encoder es el encargado de definir un modelo de red convolutiva que se encarga de extraer las características más prominentes de una imagen en base a un mapa de atención, el resultado de este proceso es un vector que contienen posibles palabras en relación con la imagen, sin embargo, estos resultados aun no son los finales ya que todavía deben pasar por un segundo filtro en el cual se selecciona las partes más representativas.

  Una característica digna de mención dentro de este artículo es que el modelo genera un mayor peso sobre determinados elementos del vector cuando calcular una mayor importancia a dicha palabra para describir a la imagen con la que trabajo.

- **Decoder:**

  El decodificador o también conocido como decoder es el encargado de trabajar con el valor resultante del codificador de manera que, en primera instancia pueda generar un vector de contexto. El vector de contexto hace referencia a seleccionar a los elementos más significativos del vector original que representarían a diferentes ubicaciones de la imagen.

Una vez entendido el proceso concreto del funcionamiento de una red neuronal para la descripción de imágenes se puede explicar los dos enfoques que diferencian a este modelo de los modelos tradicionales.

- **Atención estocástica:**

  La atención estocástica se basa en que el modelo determine en donde va a prestar atención, para lograr esto se usa la distribución multinoulli o distribución de Bernoulli generalizada junto con una aproximación de gradiente basada en el método Monte Carlo.

- **Atención determinística:**

  La atención determinística se basa en la técnica estándar de back-propagation en la que se busca aproximar la probabilidad marginal sobre los lugares en los que se estableció el mapa de atención.

Como se mencionó al antes, para medir la capacidad de describir la información de una imagen se utiliza la métrica BLEU que viene siendo el equivalente a la métrica Accuracy dentro de las tareas clásicas de las redes

neuronales. En la siguiente ilustración se muestra los resultados del modelo propuesto utilizando atención estocástica y atención determinística.

		BLEU				
Dataset	Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR
Flickr8k	Google NIC(Vinyals et al., 2014)†Σ	63	41	27	—	—
	Log Bilinear (Kiros et al., 2014a)°	65.6	42.4	27.7	17.7	17.31
	Soft-Attention	**67**	44.8	29.9	19.5	18.93
	Hard-Attention	**67**	**45.7**	**31.4**	**21.3**	**20.30**
Flickr30k	Google NIC†°Σ	66.3	42.3	27.7	18.3	—
	Log Bilinear	60.0	38	25.4	17.1	16.88
	Soft-Attention	66.7	43.4	28.8	19.1	**18.49**
	Hard-Attention	**66.9**	**43.9**	**29.6**	**19.9**	18.46
COCO	CMU/MS Research (Chen & Zitnick, 2014)a	—	—	—	—	20.41
	MS Research (Fang et al., 2014)†a	—	—	—	—	20.71
	BRNN (Karpathy & Li, 2014)°	64.2	45.1	30.4	20.3	—
	Google NIC†°Σ	66.6	46.1	32.9	24.6	—
	Log Bilinear°	70.8	48.9	34.4	24.3	20.03
	Soft-Attention	70.7	49.2	34.4	24.3	**23.90**
	Hard-Attention	**71.8**	**50.4**	**35.7**	**25.0**	23.04

Ilustración 16 Resultados BLEU y METEOR sobre tres dataset de descripción de imágenes (Xu, et al., 2015).

5.5.8. **Image Captioning with Semantic Attention** (You, Jin, Wang, Fang, & Luo, 2016)**:**

Como se ha venido hablando en los artículos anteriores, y como es de interés para el redactor, la inteligencia artificial es un área sumamente amplia que abarca diferentes ámbitos, sin embargo el área más desafiante para los investigadores sigue siendo la visión por computador junto con el procesamiento de lenguaje natural, ya que si bien se ha logrado avances significativos no se ha logrado hacer que una red neuronal pueda describir exactamente con un alto nivel de precisión que es lo que muestra una imagen, tarea que es fácil para los humanos.

En el artículo anterior se presentó una solución para la descripción de imágenes utilizando mecanismos de atención, pero, los autores de dicho modelo hacia uso del enfoque conocido como Top-Down, que, si bien es un enfoque sumamente robusto no es lo suficientemente preciso para lograr los mejores resultados bajo determinadas condiciones, a continuación, se presenta las dos técnicas que son utilizadas dentro de la descripción de imágenes junto con las limitaciones de estos dos enfoques:

- **Enfoque Top-Down:**

  Se basa en mecanismos de atención que permiten un entendimiento profundo sobre la imagen. También se asocia en ciertas investigaciones con el razonamiento multi-paso.

  Las limitaciones que tiene el enfoque Top-Down es que no presenta atención a detalles pequeños o los deja fuera del modelo de atención

debido a esto generaliza a puntos que pueden ser clave para la contextualización de imágenes.

- **Enfoque Bottom-Up:**

  Este enfoque se basa en generar un conjunto de palabras que describan varios aspectos de una imagen, luego los combina generando la descripción, una gran ventaja de este enfoque es que no requiere de imágenes de alta calidad.

  La gran limitación de Bottom-Up sobre el enfoque Top-Down es que carece de un proceso de formulación de aspectos en oraciones lo que lo hace más sensible a aspectos sin relación a la imagen con la que se encuentra trabajando.

La diferencia con la solución del artículo anterior en el cual se establecía el uso de modelos de atención es que en este paso primero se une el proceso de aprendizaje Top-Down con Bottom-Up en la búsqueda de solapar sus debilidades entre si crean un modelo más robusto.

Otro enfoque que diferencia este nuevo modelo de es el proceso de feedback o retroalimentación, este proceso hace que no sea necesario el uso de datos externos para entrenar conceptos visuales o aprender relaciones semánticas entre palabras.

Entre los resultados de este modelo basado en la unión de tres enfoques se encuentran:

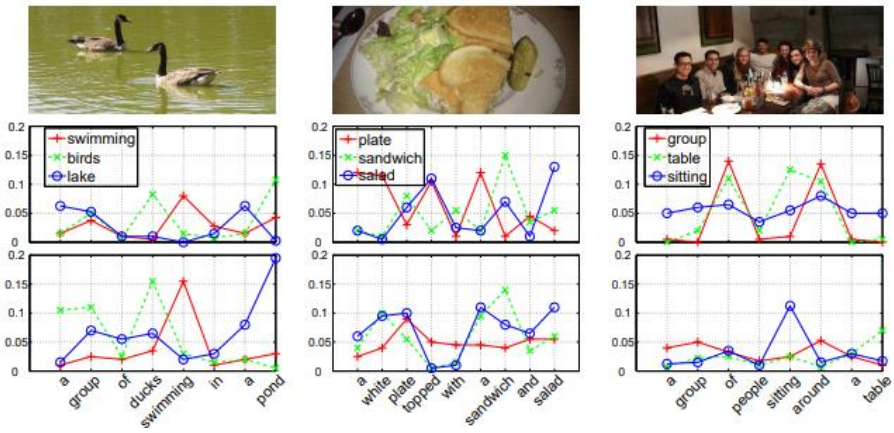


Ilustración 17 Palabras que mayor peso dentro de la oración en base al modelo de atención y el proceso de retroalimentación (You, Jin, Wang, Fang, & Luo, 2016).

En la imagen anterior se establece como las palabras toman mayor importancia en base al resultado de los mapas de atención, por ejemplo, en la primera imagen de un par de patos nadando en, lo que parece una laguna, las palabras más importantes son Nadar, Pájaros, Lago, como se puede observar esto

va de acorde a lo que se buscaba determinar. Sin embargo, el modelo también se equivoca generando oraciones cuya relación semántica es equivoca con lo que en realidad muestra la imagen.

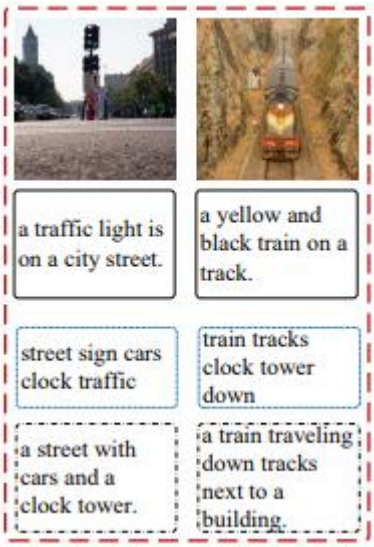


Ilustración 18 Resultados equívocos del modelo. De arriba hacia abajo (Google NIC, Top-5 Visual atributes, ATT-FCN).

Debido a la atención a los detalles se puede observar en la ilustración 18 como los resultados del modelo ATT-FCN producen salidas cuyo valor no tiene relación directa a lo que en verdad se quiere mostrar, pese a esto, este modelo ha generado resultados altamente confiables siendo el ganador en la precisión basada en métricas como BLEU 1, 2, 3 y 4. Así también ha logrado superar a Google NIC dentro de la métrica METEOR dentro del dataset Microsoft Coco y Flickr30.

5.5.9. **ClipCap: CLIP Prefix for Image Captioning** (Modaky, Hertz, & H. Bermano, 2021)**:**

En este artículo se presenta un enfoque nuevo basado en la herramienta de OpenAI CLIP junto con GPT-2 de la misma empresa. El objetivo de este artículo es presentar una manera de describir imágenes de manera novedosa llegando a presentar resultados que logren igual a los obtenidos por modelos state-of-the-art. La ventaja que presenta este modelo sobre los modelos de anteriores años es que, al estar tanto el modelo de CLIP como el modelo GPT-2 congelados (no pueden ser entrenados ni tampoco pueden ser modificados sus pesos) se requiere una menor cantidad de tiempo al momento de entrenar.

Una ventaja superior que muestra este modelo es el de no requerir de un dataset adicional con más anotaciones para mejorar la precisión de este, sino que

únicamente se basa en el conocimiento existe para generar un conjunto de palabras que, una vez enlazadas, generen un resultado entendible acorde al contexto de la imagen, cabe recalcar que el único entrenamiento que hace es el de un pequeño transformador que permite enlazar los resultados de CLIP hacia el modelo de GPT2.

Para entender el funcionamiento de este modelo primero se requiere un conocimiento previo sobre las herramientas/modelos que se hacen uso dentro de este, como primer punto tenemos a CLIP que será descrito a continuación.

- **CLIP:**

  Esta herramienta fue desarrollada por OpenAI en el año 2021 por lo que es relativamente nueva a comparación de otras técnicas y herramientas. Las ventajas de esta nueva herramienta son las siguientes:

  a. **Disminuye el costo sobre datasets:**

     Normalmente cuando se desea entrenar un modelo de Deep Learning se requiere de datasets con conjuntos de datos inmensos para obtener buenos resultados. Sin embargo, en el caso de este modelo se utilizan pares de datos (texto-imagen) disponibles en internet por lo que se carece del uso de un dataset especializado, pero este proceso lleva a una mala descripción con objetos especializados o que pueden ser únicamente encontrados dentro de datasets específicos.

  b. **Adaptable:**

     Al entrenar un modelo de Deep Learning se establece una única tarea en la cual se va a enfocar dicho modelo, por lo que se si se desea que realice una tarea diferente difícilmente podrá ser adaptado directamente o se deberá utilizar técnicas más complejas como Transfer Learning. Este no es el caso de CLIP ya que este modelo generaliza de sumamente bien por lo que puede ser adaptada a diferentes datasets con conjuntos de datos nuevos obteniendo de esa manera resultados alentadores a diferencia de entrenar un modelo desde cero.

  c. **Altamente eficiente:**

     El modelo CLIP entrena sobre un conjunto de datos altamente variado con datos algunas de las veces tiene ruido lo que lo hace excelente para trabajar con técnicas como Zero-Shot Learning, sin embargo para lograr resultados sin comprometer la eficiencia del entrenamiento se hace uso de optimizaciones algorítmicas

como, por ejemplo, la adopción de una técnica conocida como Objetivo Contrastivo, esta técnica novedosa se trata de tomar las características de alto nivel ignorando las pequeñas o que poca importancia tienen (trabaja de manera similar a los mapas de atención). La técnica anterior reduce entre 4x a 10x el tiempo de entrenamiento en experimentos de mediana escala.

- **GPT-2 (Generative Pre-Training):**

  GTP-2 es un modelo de lenguaje creado por OpenAI. En palabras simples, un modelo de lenguaje es un modelo de aprendizaje de maquina dedicado a predecir que palabra va a continuación de otra de manera que se genere un texto lo suficiente verosímil. Las aplicaciones de GTP-2 son variadas, desde la traducción de texto, resúmenes, hasta la aplicación de modelos ajenos a su propósito general usando técnicas como Fine-Tunning.

  En general el proceso y uso de GPT-2 dentro de los modelos de descripción de imágenes se basa en la generación de texto utilizando un prefijo. Por ejemplo, el modelo CLIP genera las representaciones visuales de una imagen, posterior a esto se procede a usar un decodificador de texto, que si bien puede ser una red LSTM en este caso se utiliza GPT-2 debido a su gran precisión al generar texto.

El proceso general del funcionamiento de esta herramienta s resume en la siguiente imagen en la se muestra como a través de CLIP, una red de mapeo basada en transformadores y GPT-2 se genera la descripción de una imagen.

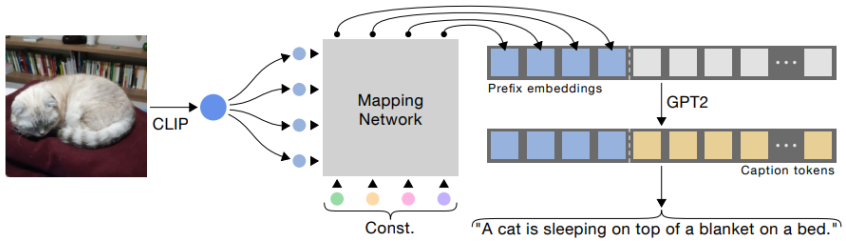


Ilustración 19 Proceso general del funcionamiento del presente modelo (Modaky, Hertz, & H. Bermano, 2021).

Como se mencionó en los párrafos anteriores, el modelo en general se basa en gran medida en CLIP y GPT-2 por lo que, en sí, el principal aporte de este

modelo es el uso de los transformadores, los transformadores son modelos de Deep Learning que permiten establecer atención, en este caso, sobre determinadas palabras de forma que el resultado final no contenga, por así decir ruido. Una ventaja de los transformadores es que no depende de redes neuronales recurrentes como GRU o LSTM, sino que únicamente trabaja con modelos de atención. El uso de los transformadores es importante ya que CLIP devuelve diferentes salidas por lo que es importante 'procesar' dichas salidas a través de un transformador de manera que se eliminen los resultados innecesarios para que luego la herramienta GPT-2 logre establecer una oración como resultado.

Algunos ejemplos de los resultados de este modelo se presentan en las siguientes imágenes, cabe recalcar que el modelo fue entrenado originalmente sobre sobre el dataset COCO por parte de CLIP, pero también se hizo Fine-Tunning sobre el dataset Conceptual Captions.

			
Ground Truth	A man with a red helmet on a small moped on a dirt road	A young girl inhales with the intent of blowing out a candle.	A man on a bicycle riding next to a train.
Oscar	a man riding a motorcycle down a dirt road.	a woman sitting at a table with a plate of food.	a woman riding a bike down a street next to a train.
Ours; MLP + GPT2 tuning	a man riding a motorcycle on a dirt road.	a woman is eating a piece of cake with a candle.	a man is standing next to a train.
Ours; Transformer	a man is riding a motorbike on a dirt road.	a young girl sitting at a table with a cup of cake.	a man is standing next to a train.

Ilustración 20 Resultados en base al entrenamiento en el dataset Microsoft COCO.

			
Ground Truth	A life in photography – in pictures.	Photograph of the sign being repaired by brave person.	Globes : the green 3d person carrying in hands globe.
VLP	Actors in a scene from the movie.	The sign at the entrance.	Templates: green cartoon character holding the earth globe.
Ours; MLP + GPT2 tuning	Actor sits in a hotel room.	The sign at the entrance.	3d render of a man holding a globe.
Ours; Transformer	person sitting on a chair in a room.	a sign is seen at the entrance to the store.	stock image of a man holding the earth.

Ilustración 21 Resultados en base al entrenamiento en el dataset Conceptual Captions (Modaky, Hertz, & H. Bermano, 2021).

5.5.10. Textual Description for Mathematical Equations (Mondal & Jawahar, 2019):

En este artículo se propone una nueva técnica para la descripción de imágenes matemáticas basado en técnicas de aprendizaje profundo, la motivación de este artículo es generar una solución que permita generar texto a partir de una fórmula matemática, si bien esto puede ser considerado como Optical Character Recognition (OCR) pero no es lo mismo, ya que este caso se busca manejar diferentes símbolos de ecuación y números en lugar de letras.

El modelo que se propone dentro de este artículo se encuentra basado en la siguiente forma:

2. Una red neuronal convolucional que sirve de Encoder.
3. Una red neuronal recurrente con mecanismos de atención que trabaja como Decoder.

El presente articulo busca, de manera similar a este proyecto, reducir las limitaciones de acceso a la educación para personas con discapacidad visual a través de un MED (Mathematical Equation Descriptor), ya que hasta ahora todas las herramientas de procesamiento de texto a voz no pueden determinar de manera correcta el contenido de una imagen, en especial cuando esta es una fórmula matemática. Otra limitante que se presenta a la hora de trabajar con imágenes matemáticas es la sensibilidad de las imágenes, esto quiere decir que una imagen puede tomar diferentes sentidos acordes al orden de los factores, tamaño de los factores, operadores, entre otros aspectos. Un ejemplo simple de esto se presenta a continuación:

3^x

En este ejemplo se muestra ‘3 *elevado a la potencia x’*

X^3

En este ejemplo se muestra ‘x elevado al cubo’

$3x$

En este ejemplo se muestra ‘3 veces x’

En base al ejemplo anterior se puede ver cuan fácil puede llegar a ser la mala interpretación de una fórmula matemática debido a la variabilidad de esta con respecto a los factores antes mencionados.

Este modelo si bien no es muy novedoso en relación con los modelos antes investigados si forma parte importante en un avance en el ámbito de la accesibilidad web para personas con problemas de la visión, en especial a personas que se encuentran estudiando.

Cabe recalcar que el presente modelo ha sido entrenado bajo determinadas limitantes de fórmulas entre las que se encuentran:

4. **Ecuaciones Lineales.**
5. **Inecuaciones.**
6. **Limites.**
7. **Derivadas.**
8. **Integrales.**
9. **Integrales finitas.**

Eso si bien es un buen inicio aun deja un largo conjunto de fórmulas por tratar debido a que existen muchas más combinaciones y elementos matemáticos por ser tratados. A continuación, se muestran ciertos resultados de este modelo:

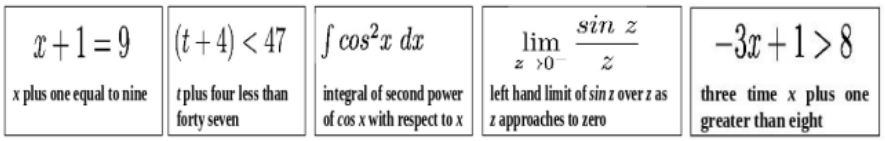


Ilustración 22 Ejemplos de descripción de fórmulas matemáticas (Mondal & Jawahar, 2019).

5.6. Diseño e implementación de las redes neuronales.

Antes de proceder con el desarrollo de las redes neuronales se debe primero establecer como está organizado el grafico de la ilustración 2 para que sea más fácil de

comprender como se va a estructurar la clasificación, para esto se establecen diferentes niveles en los cuales se va a clasificar las imágenes multimedia, una explicación más clara de cómo están formados los niveles es la siguiente:

Nivel 1:

El nivel 1 está formado por la clasificación de las primeras tres categorías, Ilustraciones, Ecuaciones y Tablas.

En el caso de las ecuaciones se procede a realizar una descripción textual sobre el contenido de las ecuaciones.

Por el otro lado, en el caso de las tablas se procede de otra manera, primero se preprocesa la tabla y luego se procede utilizar Visión por Computador y Reconocimiento Óptico de Caracteres, OCR, por sus siglas en inglés, de forma que se pueda obtener el texto contenido dentro de una tabla.

Nivel 2 de Ilustraciones:

En el nivel 2 se encuentran 3 subcategorías de las ilustraciones, Fotografías, Gráficos, e Imágenes Digitales.

En el caso de las fotografías se hace referencia a imágenes de la vida real, ya sean estas capturadas por cámaras, dibujadas a mano, etc.

En el caso de los gráficos nos referimos a gráficos generados por computadora pero que formen parte de gráficos estadísticos, matemáticos, o diagramas.

Finalmente, en el caso de las imágenes digitales se hace referencia a imágenes generadas por computadora o imágenes artificiales, como, por ejemplo, capturas de pantalla, ilustraciones animadas como cartoons, o logos de empresas.

Nivel 3 de la subcategoría Gráficos de la categoría Ilustraciones:

Dentro de la categoría Gráficos se encuentran las siguientes subcategorías, Gráficos Estadísticos, Gráficos Matemáticos, Diagramas (Flujo, Procesos, Jerárquicos, etc.).

En la subcategoría de gráficos estadísticos se encuentran las siguientes categorías: Gráficos circulares, de dispersión, mapas de calor, gráficos de barras horizontales y verticales, gráficos lineales.

En la subcategoría de gráficos matemáticos se encuentran: Gráficos trigonométricos y gráficos de funciones.

Finalmente, en la subcategoría de gráficos tipo diagrama se encuentran: Gráficos jerárquicos, de proceso, cíclicos, y piramidales.

Nivel 4 de la subcategoría Digital de la categoría Ilustraciones:

Dentro de la categoría Digital se encuentran los siguientes tipos de imágenes, Capturas de pantalla, Ilustraciones animadas, logos.

En la subcategoría de capturas de pantalla se trata de, como su nombre lo indica, una captura de pantalla de un dispositivo electrónico ya sea este celular, computador, tableta, etc.

En la subcategoría ilustraciones animadas nos referimos a todo tipo de imagen que haga relación a imágenes generadas por computador y que forme parte de la educación, tal es el caso de partes del cuerpo humano, la naturaleza, comida, estructuras, etc.

Finalmente, en la subcategoría logo se trata de imágenes de logos de empresas, en este punto se puede obtener el texto del logo según sea el caso.

Modelos utilizados dentro del proyecto:

Como se determinó en la parte anterior de la definición de los datasets se estableció que cada dataset está formado por una determinada cantidad de clases que representarían la salida de la clasificación. Por ejemplo, en el nivel uno existe tres diferentes categorías, por lo tanto, la salida del modelo para esta clasificación seria de tres salidas una por cada categoría.

En general se ha utilizado un modelo estándar de red neuronal, este modelo se llama InceptionResNetV2. Dicho modelo ha sido entrenado sobre el dataset ImageNet, que, como ya se ha venido tratando en puntos anteriores, este dataset contiene una gran variedad de imágenes de propósito general lo que la hace excelente para el presente proyecto debido a que los datasets con los que se trabaja son de ámbito general permitiendo adaptar los pesos de InceptionResNetV2 para las tareas de clasificación. Una ventaja que sobre lleva las limitaciones de las redes neuronales tradiciones es que se requiere de una cantidad muy baja de épocas (periodos de entrenamiento) en la mayoría de los casos reduciendo el tiempo empleado en el entrenamiento.

En la siguiente ilustración se muestra el modelo InceptionResNetV2 de manera comprimida debido a que el modelo a detalle es sumamente complejo y con una gran cantidad de capas.

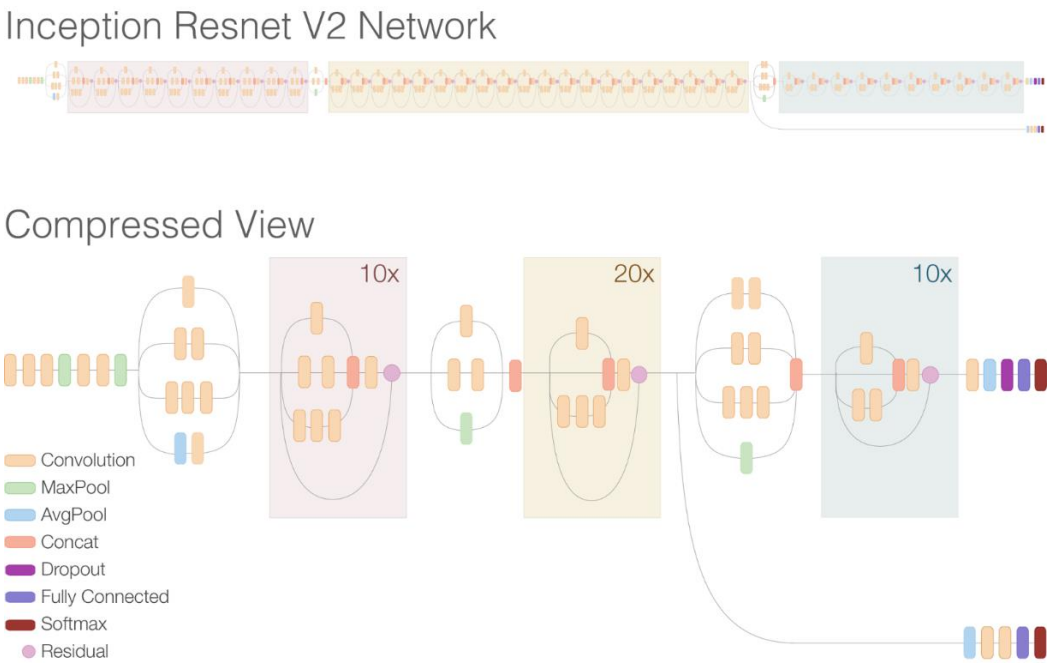


Ilustración 23 Arquitectura del modelo InceptionResNetV2 (Alemi, 2016)

El presente modelo de la ilustración 23 muestra como está formado la arquitectura del modelo base para la clasificación, sin embargo, para hacer uso de transfer learning basado en los pesos del modelo base se requiere bloquear o congelar los pesos de entrenamiento de este por lo que para poder 'adaptar' los pesos se requiere de un modelo simple que sirva de salida para la clasificación, este modelo este compuesto de la siguiente manera:

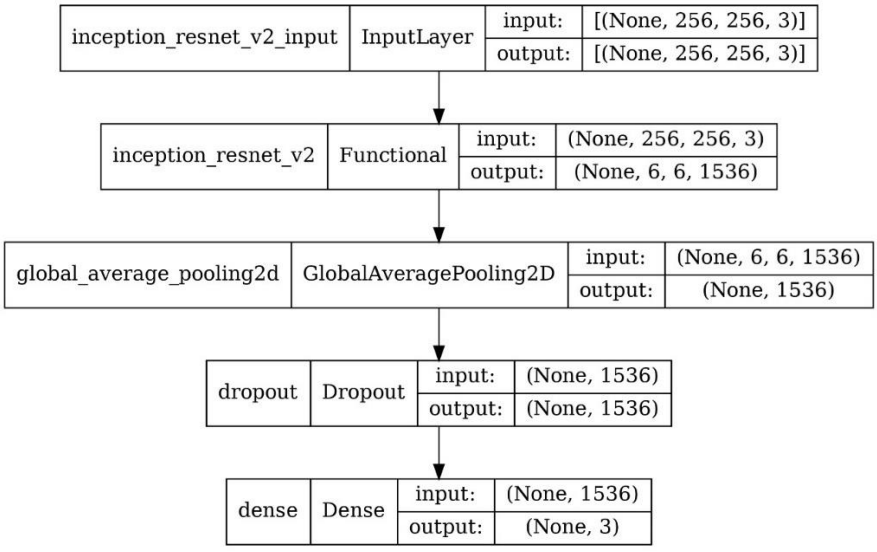


Ilustración 24 Modelo general para transfer learning

En el modelo de la ilustración 24 se muestra el modelo general para hacer uso del transfer learning, este modelo se puede considerar como el modelo general para todos los datasets, únicamente lo que cambia entre los datasets es la cantidad de salidas, a continuación, se ahonda más sobre la arquitectura propuesta definiendo el porqué de su uso.

Como se estableció anteriormente el modelo que se está utilizando como base es InceptionResNetV2 debido a sus excelentes resultados en el dataset ImageNet estableciendo valores de predicción sumamente altos comparados con otros modelos state-of-the-art. Otra razón de su uso es debido a su fácil acceso ya que el modelo se encuentra disponible dentro de la librería Tensorflow. Si bien este modelo es sumamente útil para clasificación general de imágenes requiere de imágenes de un tamaño de 256x256 pixeles y de 3 capas de color (RGB), esto se puede considerar como un traspié debido a que con datasets grandes se requiere de un mayor tiempo de entrenamiento en contraste con modelos que trabajen con imágenes a escala de grises y menores tamaños.

En base al párrafo anterior se podría asumir que únicamente con el modelo de InceptionResNetV2 se podría realizar la tarea de clasificación ya que genera excelentes resultados, y si bien en parte eso es cierto, el entrenamiento de esa red requeriría de datasets grandes y, por su puesto, una gran cantidad de tiempo para lograr buenos resultados por lo que no es recomendable hacerlo sino con fines investigativos. Para sobrellevar la limitante de entrenar el modelo desde cero se puede usar el conocimiento de ese modelo y únicamente ajustarlo al dataset con el que queremos trabajar.

InceptionResNetV2 originalmente produce 1000 salidas (basado en la tarea de clasificación del dataset ImageNet), pero las 1000 salidas no es lo que se busca obtener en este presente trabajo, sino que buscamos clasificar acorde al dataset actual con el que se trabaje, para esto se elimina la capa de salida del modelo InceptionResNetV2 y se pasa el resultado de la última capa a través de una capa llamada GlobalAveragePooling2D. Esta capa se utiliza para realizar operaciones de convertir objetos multidimensionales en objetos de 1 dimensión. Posterior al proceso de pooling se requiere agregar una capa de Dropout, esta capa de regularización elimina neuronas basadas en la distribución de Bernoulli de manera que se reduzca la posibilidad de llegarse a producir un sobre-entrenamiento de la red neuronal generando salidas inesperadas. Finalmente se agrega una capa Dense, esta capa es una de las más comunes y ‘básicas’ dentro de los modelos de redes neuronales, a la final lo que permite esta capa es calcular la probabilidad de salida en base a una función de activación. Al ser la capa Dense la última capa esta también se la conoce como la capa de salir, esta capa recibe como parámetro la cantidad de neuronas de salida y una función de activación.

Cabe recalcar que la ilustración 24 es el modelo base y en general lo que cambia entre el resto de los modelos es únicamente la cantidad de salidas que representan a las clases de clasificación.

Una vez definido el modelo es imperativo establecer los parámetros de compilación para todos los modelos. Estos parámetros son los siguientes:

- **Clase de perdida:**

  Sparse Categorical Crossentropy

  Al ser una clasificación de múltiples salidas la tarea que se realiza se la conoce como clasificación categórica (por el contrario, si fueran únicamente dos posibles salidas se consideraría clasificación binaria).

  Otro factor que liga hacia el uso de esta clase de perdida es que la estructura de salida de las clases, en el siguiente ejemplo se muestra la diferencia entre CategoricalCrossentropy y SparseCategoricalCrossentropy:

  - Cuando los datos están estandarizados utilizando la técnica OneHotEncoder ya sea utilizando sklearn u otra librería se deberá utilizar CategoricalCrossentropy debido a que trabaja con este tipo de datos, por ejemplo, si se tuviera 3 clases el OneHotEncoding quedaría:
    [1, 0, 0], [0, 1, 0], [0, 0, 1] (Clase 1, 2, y 3 respectivamente).

  - En el caso de que las categorías sean valores enteros simples (o valores dispersos) se debe utilizar SparseCategoricalCrossentropy, por ejemplo, si se tuviera 3 clases quedaría:
    [0], [1], [2] (Clase 1, 2 y 3 respectivamente).

- **Optimizador:**

  Adam

  Para el caso del optimizador se ha optado por el uso de Adam, este optimizador es un método del descenso del gradiente estocástico que es altamente efectivo en términos computacionales, además es el optimizador de facto para la clasificación multi categoría lo que lo hace adecuado para uso dentro de este proyecto.

- **Métrica:**

  Accuracy

La métrica con la que medimos que tan bien está entrenando nuestro modelo es la exactitud o accuracy, esta métrica mide que tan cerca está el valor predicho del valor real. Muchas de las veces se suelen confundir entre Accuracy y Precision, que son dos métricas para medir los modelos de aprendizaje de máquina. En la siguiente ilustración se muestra la diferencia entre estas dos métricas.

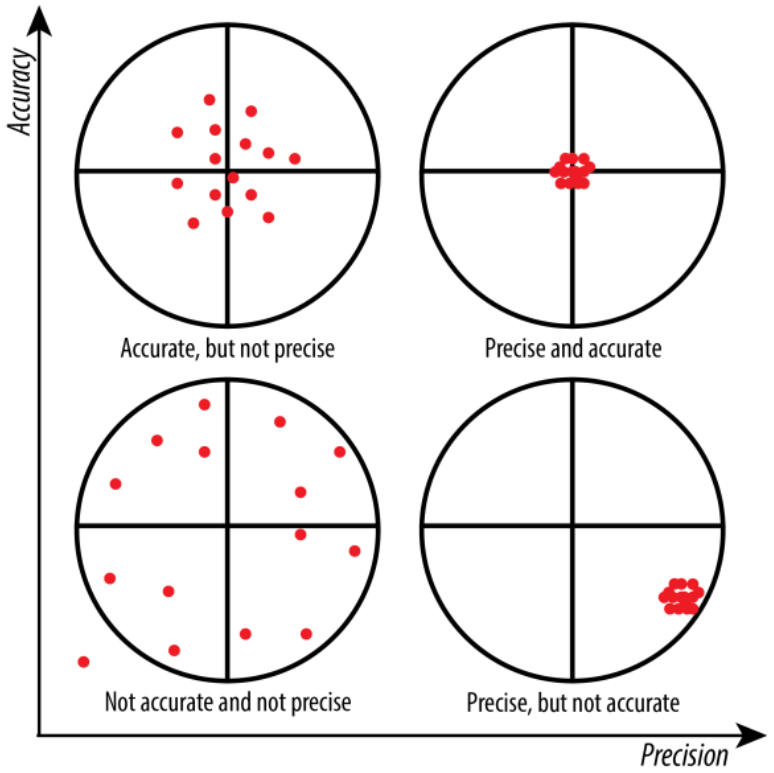


Ilustración 25 Accuracy vs. Precision (St. Olaf College, 2022)

Relación entre el modelo original y los datasets utilizados:

Al inicio del punto de seis, se estableció como estaba formado el dataset de imágenes que sirven para el entrenamiento y validación de este proyecto, sin embargo, debido a la gran cantidad de clases y salidas que se muestran tanto en el diagrama como en la explicación de los dataset se ve la necesidad de generar un diagrama más simple que condense en capas la relación entre las capas que existen de forma que sea más simple la definición de la tarea de clasificación de imágenes.

En la ilustración 26 se muestra de manera condensada como la relación entre capas y se detalle también el alcance de clasificación de cada capa.

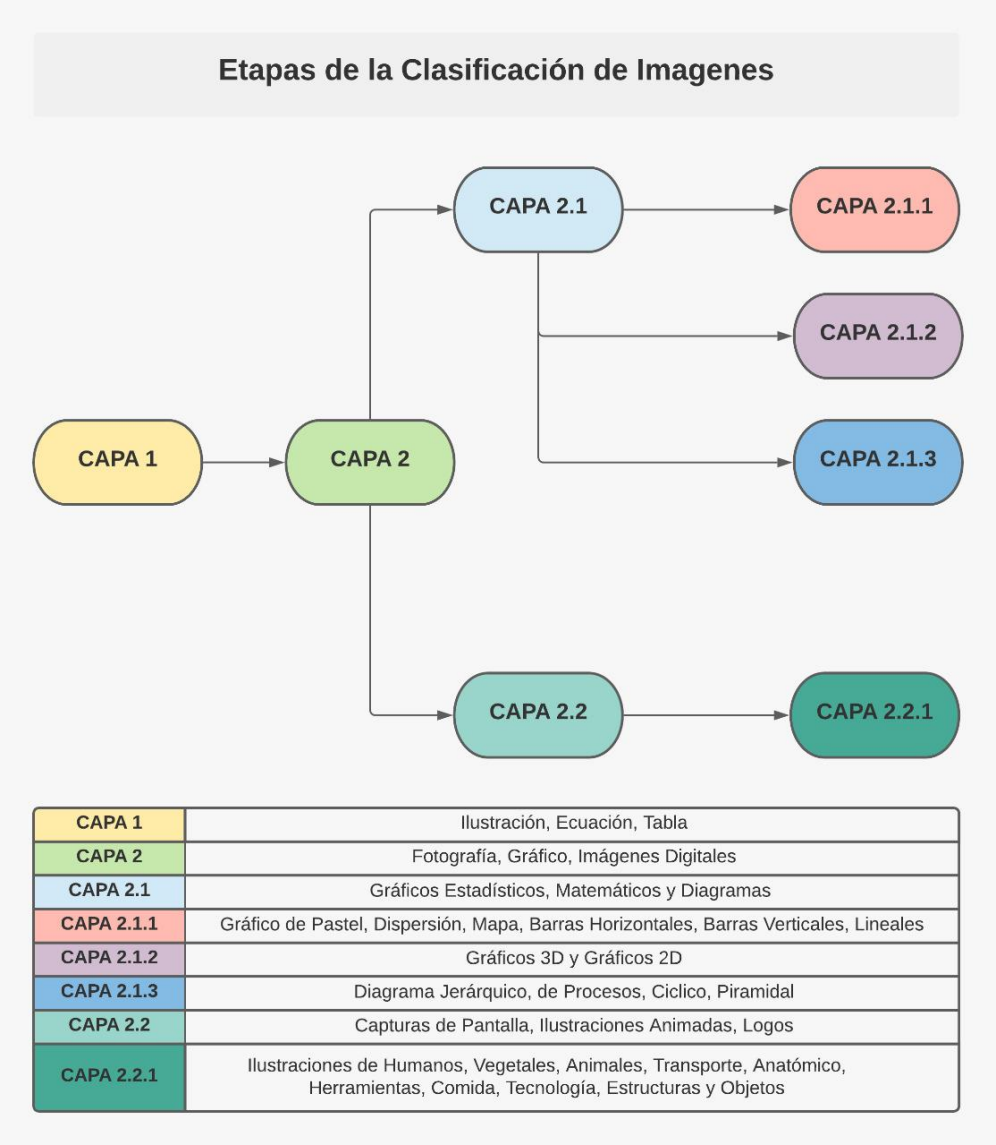


Ilustración 26 Relación del Diagrama y Capas para la Clasificación de Imágenes

Una vez establecido la relación entre capas en la ilustración anterior se puede explicar cómo está formada cada capa junto con su número de salidas y épocas de entrenamiento. Es necesario recalcar que el número de salidas es directamente proporcional al número de clases del entrenamiento.

Capa	Numero de Salidas	Numero de Épocas
Capa 1	3	2
Capa 2	3	1

Capa 2.1	3	4
Capa 2.1.1	6	10
Capa 2.1.2	2	15
Capa 2.1.3	4	20
Capa 2.2	3	10
Capa 2.2.1	10	20

Tabla 5 Relación Capa-Salidas-Épocas.

En el caso de la capa 2.2 y 2.2.1 se requirió de una configuración especifica al momento de la compilación del modelo debido a que se buscaba evitar el sobre entrenamiento de esta, por lo tanto, se utilizó dos técnicas conocidas como CallBacks, la primera técnica se llama ReduceLROnPlateau, esta técnica permite reducir la velocidad de aprendizaje para evitar saltos muy grandes en el accuracy.

El segundo callback que se utilizo es EarlyStopping que permite detener el entrenamiento de un modelo de manera que no se sobre entrene la red neuronal llegando a generarse un overfitting.

Capa	Parámetro Especifico	Valor del Parámetro
Capa 2.2	**ReduceLROnPlateau**	**Monitor:** Validation loss.
		Factor: 0.0005
		Patience: 2 épocas.
	EarlyStopping	**Monitor:** Loss
		Patience: 3 épocas.
Capa 2.2.1	**ReduceLROnPlateau**	**Monitor:** Validation loss.
		Factor: 0.0005
		Patience: 2 épocas.

Tabla 6 Parámetros específicos en determinadas capas.

Modelos de investigación y herramientas que forman parte del proyecto:

- **Manejo de Tablas con librería table_ocr:**

  Para el manejo y obtención de datos dentro de las tablas de datos se hizo uso de una herramienta OpenSource que se llama table_ocr, esta herramienta permite la detección y obtención de datos de imágenes de tablas.

  En la ilustración 26 se muestra el funcionamiento general de la herramienta table_ocr, en la cual en base a una imagen de una tabla se la preprocesa y obtiene cada una de las celdas de la tabla, posteriormente se envía cada celda hacia a Tesseract OCR para obtener su valor en forma de texto, finalmente se organiza los resultados y se genera una tabla HTML en representación a la tabla original.

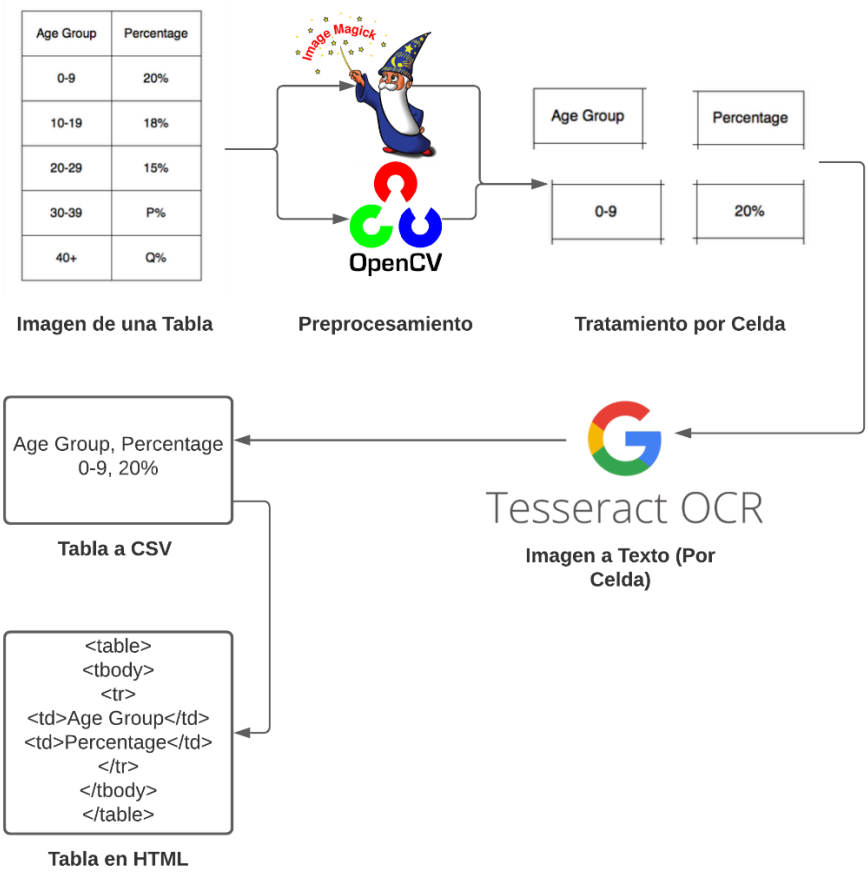


Ilustración 27 Funcionamiento de table_ocr.

- **Descripción de Ecuaciones con MED:**

Para el manejo de ecuaciones matemáticas se hace uso del modelo definido en el artículo académico (Mondal & Jawahar, 2019), en la siguiente ilustración se establece el funcionamiento de esta herramienta.

En la ilustración 27 se muestra el funcionamiento a detalle del módulo de descripción de tablas. En la cual en base a un Encoder – Decoder se genera la descripción de la formula, cabe recalcar que se realizaron determinadas adaptaciones para utilizar únicamente el CPU en lugar del GPU.

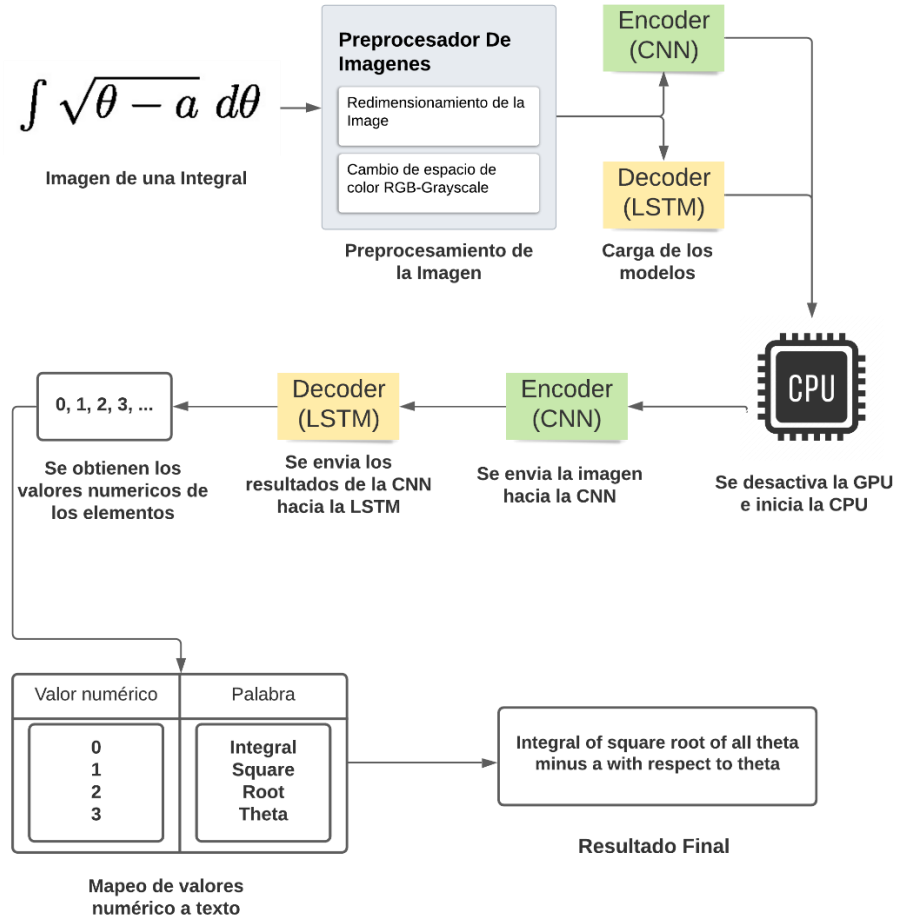


Ilustración 28 Diagrama del funcionamiento para la descripción de Ecuaciones.

- **Descripción de Imágenes con CLIPCap:**

Para el módulo de descripción de imágenes se ha utilizado el modelo del articulo (Modaky, Hertz, & H. Bermano, 2021), en este modelo permite obtener excelentes resultados tanto para imágenes reales como para ilustraciones

generadas por computadora, en la siguiente ilustración se presenta el funcionamiento general de esta herramienta.

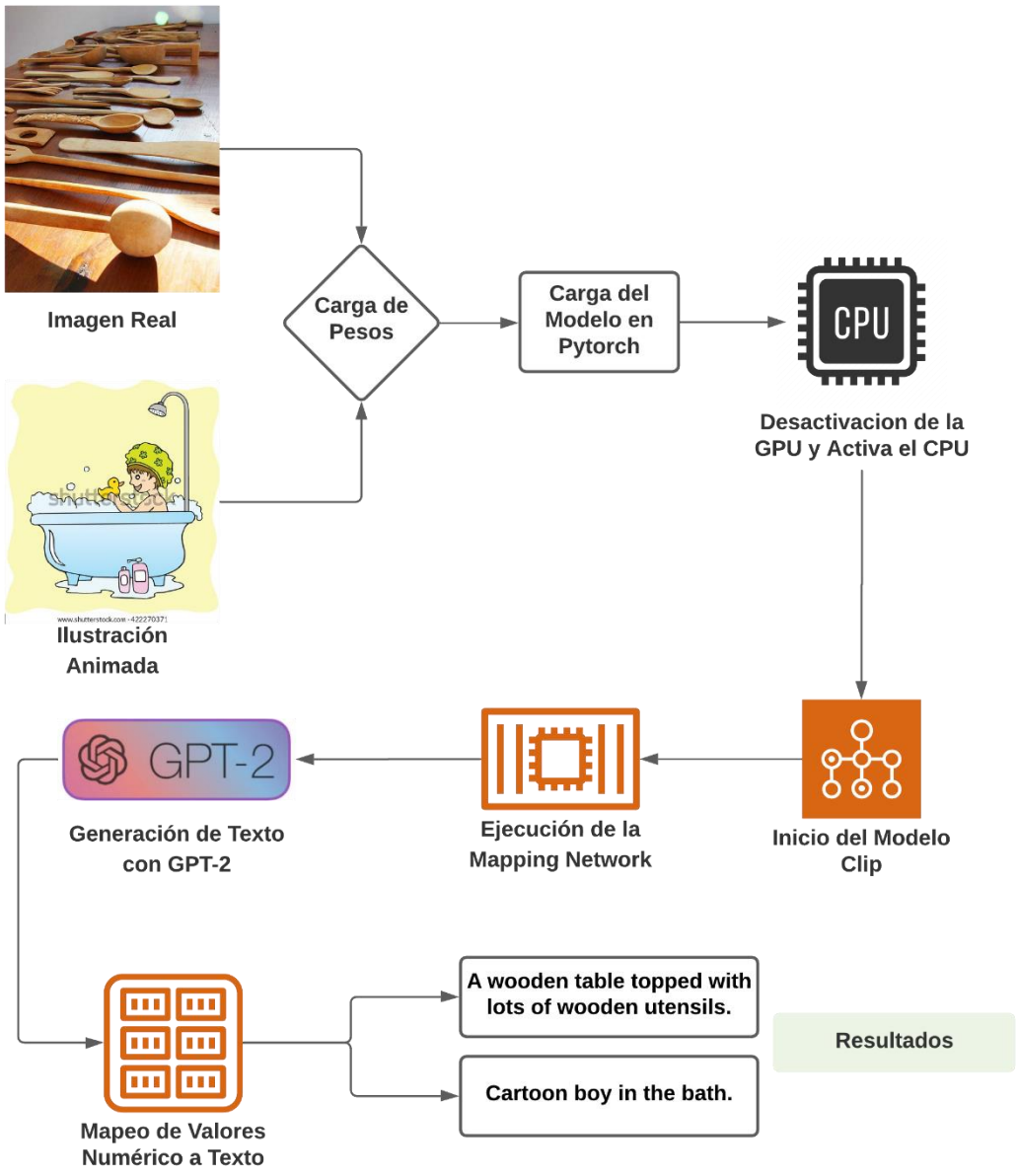


Ilustración 29 Funcionamiento del modelo CLIPCap.

6. Resultados

Para esta sección de resultados se establece la siguiente estructura en la cual se presentarán los resultados, la primera sección se contará con los resultados de la clasificación de imágenes que forma la primera parte de este proyecto, como segundo punto se mostrará los resultados de la descripción de contenido multimedia tanto a un nivel de imágenes generales como a un nivel más específico como ecuaciones matemáticas y tablas, como tercer punto se establecerá como el proyecto se implementa como un módulo del proyecto EduTech (OerAdap). Finalmente se contará con una sección para las limitaciones que a su vez contará con subsecciones para cada uno de los temas antes mencionados.

6.1. Resultados de la clasificación de imágenes:

6.1.1. Capa 1:

Gráfica de la capa 1:

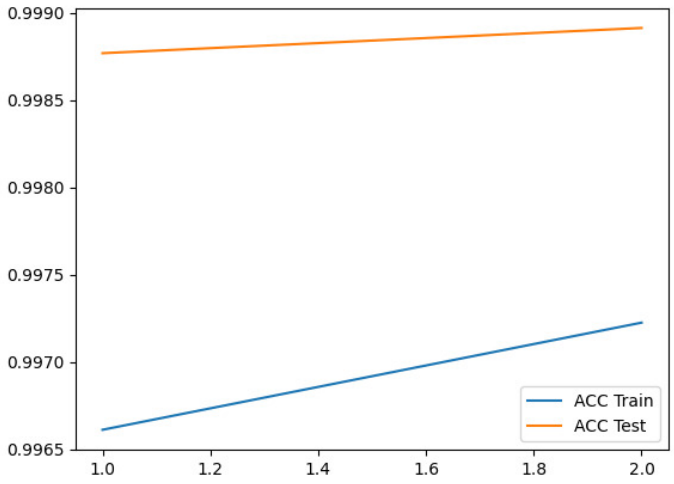


Ilustración 30 Resultados Capa 1.

Tabla de resultados capa 1:

Época	Exactitud en Entrenamiento	Exactitud en Testing
1	99.6610%	99.8769%
2	99.7225%	99.8914%

Tabla 7 Resultados Capa 1

Explicación de la cantidad de épocas y resultados de exactitud:

Debido a la gran diferencia que existen entre clases para esta capa se establece una fácil diferenciación entre las tres clases por lo que con tan solo dos épocas se logró excelentes resultados.

6.1.2. Capa 2:

Tabla de resultados capa 2:

Época	Exactitud en Entrenamiento	Exactitud en Testing
1	92.1888%	94.0812%

Tabla 8 Resultados Capa 2.

Explicación de la cantidad de épocas y resultados de exactitud:

En la capa 2 del diagrama para el presente proyecto se debe clasificar entre imágenes Digitales, Gráficos y Fotografías reales. Debido a la relativa similitud entre las imágenes Digitales y Gráficos se ha presentado un claro problema al momento de la clasificación de estas, por tanto, se ha optado entrenar la red únicamente con una época ya que con una cantidad mayor de épocas no se han obtenido mayores resultados, en la sección de discusión se profundiza más sobre las limitaciones del modelo.

6.1.3. Capa 2.1:

Gráfica de la capa 2.1:

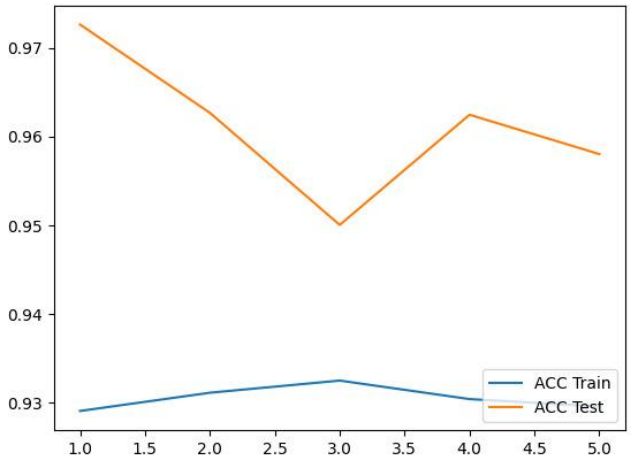


Ilustración 31 Resultados Capa 2.1

Tabla de resultados capa 2.1:

Época	Exactitud en Entrenamiento	Exactitud en Testing
1	92.9098%	97.2633%
2	93.1151%	96.2688%
3	93.2520%	95.0005%
4	93.0435%	96.2470%
5	92.9595%	95.8042%

Tabla 9 Resultados Capa 2.1

Explicación de la cantidad de épocas y resultados de exactitud:

En este caso la diferencia entre los distintos tipos de diagramas es relativamente alta, por lo que algoritmo base con transfer learning puede generalizar correctamente en la mayoría de los casos. Después de la época 5 no se mejoraron los resultados por lo que se optó entrenar hasta este punto.

6.1.4. Capa 2.1.1:

Gráfica de la capa 2.1.1:

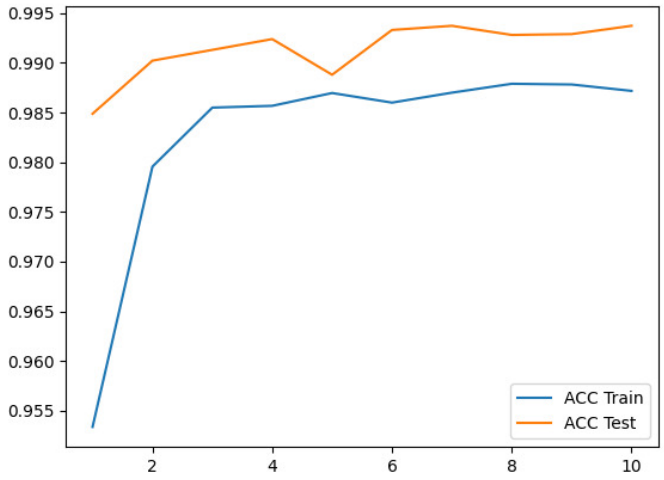


Ilustración 32 Resultados Capa 2.1.1

Tabla de resultados capa 2.1.1:

Época	Exactitud en Entrenamiento	Exactitud en Testing
1	95.3353%	98.4882%
2	97.9558%	99.0227%
3	98.5501%	99.1313%
4	98.5680%	99.2399%
5	98.6969%	98.8808%
6	98.6002%	99.3318%
7	98.7005%	99.3773%
8	98.7900%	99.2817%
9	98.7828%	99.2900%
10	98.7184%	99.3735%

Tabla 10 Resultados Capa 2.1.1

Explicación de la cantidad de épocas y resultados de exactitud:

Este caso es una razón evidente de la ventaja de transfer learning sobre el entrenamiento tradicional ya que como se muestra desde la primera época se obtiene un buen resultado considerando la cantidad de clases que se tienen (6 clases) y su baja cantidad de imágenes por clase.

6.1.5. Capa 2.1.2:

Gráfica de la capa 2.1.2:

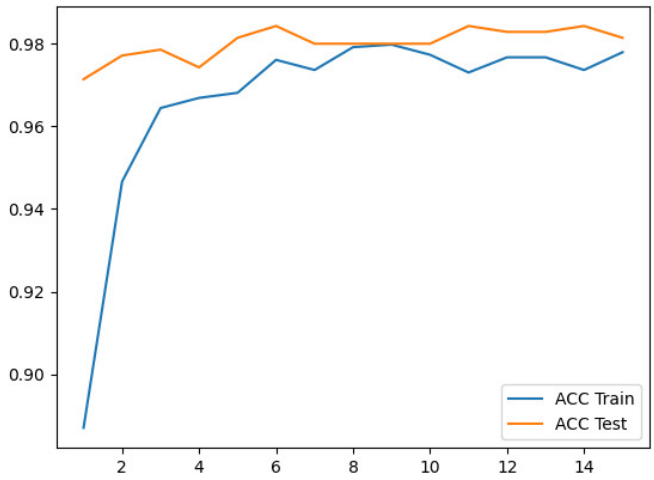


Ilustración 33 Resultados Capa 2.1.2

Tabla de resultados capa 2.1.2:

Época	Exactitud en Entrenamiento	Exactitud en Testing
1	88.7116%	97.1387%
2	94.6625%	97.7110%
3	96.4417%	97.8540%
4	96.6871%	98.1401%
12	97.6687%	98.2832%
13	97.6687%	98.2832%
14	97.3619%	98.4263%
15	97.7914%	98.1401%

Tabla 11 Resultados Capa 2.1.2

Explicación de la cantidad de épocas y resultados de exactitud:

De entre los modelos anteriores esta es la primera capa que muestra un resultado con relativamente bajo en una primera época aun cuando se hace uso de transfer learning, sin embargo, se muestra una clara mejoría acorde se va entrenando el modelo. Debido a variaciones pequeñas no se presentan los resultados de épocas intermedias.

6.1.6. Capa 2.1.3:

Gráfica de la capa 2.1.3:

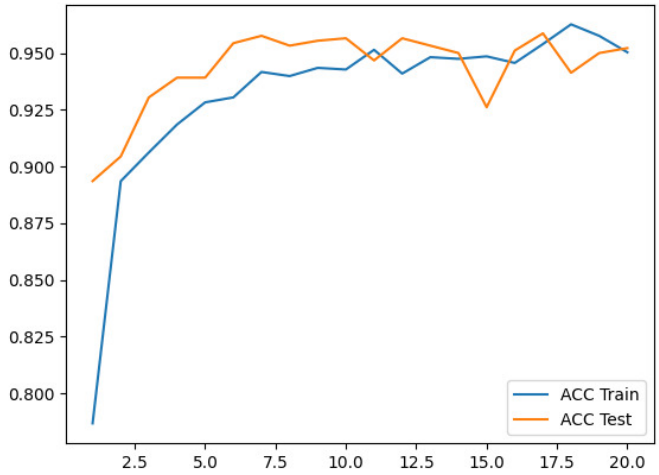


Ilustración 34 Resultados Capa 2.1.3

Tabla de resultados capa 2.1.3:

Época	Exactitud en Entrenamiento	Exactitud en Testing
1	78.6748%	89.3593%
2	89.3555%	90.4451%
3	90.6227%	93.0510%
4	92.8312%	93.9196%
17	95.4018%	95.8740%
18	96.2708%	94.1368%
19	95.7639%	95.0054%
20	95.0398%	95.2225%

Tabla 12 Resultados Capa 2.1.3

Explicación de la cantidad de épocas y resultados de exactitud:

De manera similar al caso anterior la primera época nos muestra una exactitud baja en comparación a las capas anteriores, sin embargo, conforme se va entrenando el modelo el resultado va mejorando, esto es debido a que los pesos de transfer learning deben adaptarse al nuevo dataset en cuestión.

6.1.7. Capa 2.2:

Gráfica de la capa 2.2:

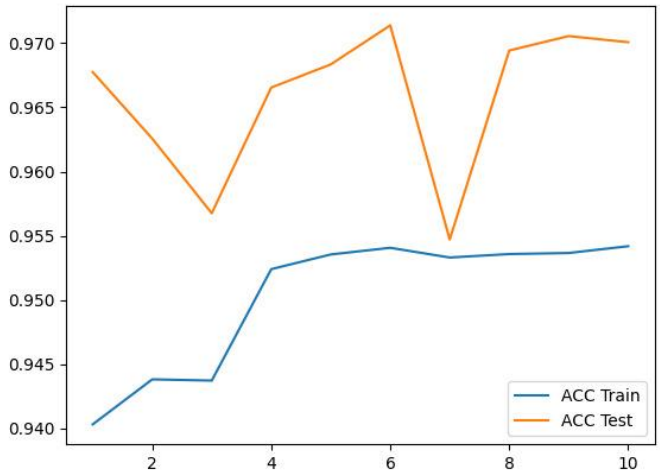


Ilustración 35 Resultados Capa 2.2

Tabla de resultados capa 2.2:

Época	Exactitud en Entrenamiento	Exactitud en Testing
1	94.0322%	96.7740%
2	94.3824%	96.2559%
9	95.3658%	97.0543%
10	95.4188%	97.0066%

Tabla 13 Resultados Capa 2.2

Explicación de la cantidad de épocas y resultados de exactitud:

En este modelo se presenta nuevamente un bloque de entrenamiento bajo la época 10, ya que, a partir de esta época no se ha mostrado un aumento significativo en la exactitud del modelo, por lo que se estableció como límite esta época.

6.1.8. Capa 2.2.1:

Gráfica de la capa 2.2.1:

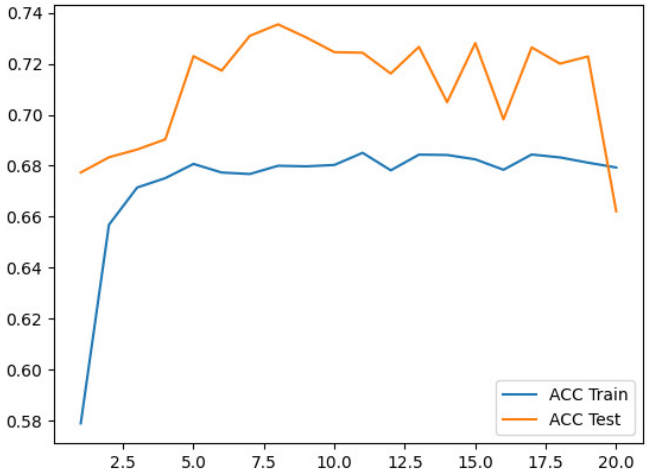


Ilustración 36 Resultados Capa 2.2.1

Tabla de resultados capa 2.2.1:

Época	Exactitud en Entrenamiento	Exactitud en Testing
1	57.9089%	67.7355%
2	65.6887%	68.3328%
3	67.1472%	68.6366%
4	67.5154%	69.0348%
17	68.4406%	72.6396%
18	68.3283%	72.0004%
19	67.1218%	72.2833%
20	67.9331%	66.2160%

Tabla 14 Resultados Capa 2.21

Explicación de la cantidad de épocas y resultados de exactitud:

Si bien en la tabla se muestra un incremento acorde el tiempo desde la primera época hacia la época numero 20 el incremento se detiene ahí, desde ese punto comienza los resultados comienzan a oscilar entre un valor aproximado de ~67% para el entrenamiento y ~66% para pruebas, por lo que, para evitar un sobre entrenamiento se optó por detener el proceso.

6.2. Resultados de la descripción de imágenes:

6.2.1. Descripción de imágenes generales:

Para la descripción de imágenes generales o de la vida real se utiliza el modelo CLIPCap

Imagen:		
Descripción original:	A river with trees and a rainbow in the background.	A group of people standing around a table.
Descripción traducida:	Un rio con árboles y un arcoíris en la parte de atrás.	Un grupo de personas paradas alrededor de una mesa.
Imagen:		
Descripción:	A woman with a laptop in her lap.	A group of people sitting around a table eating food.
Descripción traducida:	Una mujer con su computador en su regazo.	Un grupo de personas sentadas alrededor de una mesa comiendo comida.
Imagen:		
Descripción:	A small dog standing in a field of flowers.	A man in a classroom with a chalkboard.
Descripción traducida:	Un perro pequeño a lado de un campo de flores.	Un hombre en un aula de clases con una pizarra.

Imagen:	
Descripción:	A family is walking down a shopping mall aisle.
Descripción traducida:	Una familia camina por un pasillo de un centro comercial.

Tabla 15 Resultados de la descripción de imágenes con ClipCap y Coco

6.2.2. Descripción de ilustraciones animadas:

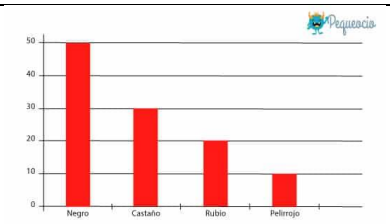
Imagen:		
Descripción:	The number of people who have been diagnosed with autism.	Young couple cooking in the kitchen.
Descripción traducida:	El número de personas quienes han sido diagnosticadas con autismo.	Una joven pareja cocinando en la cocina.
Imagen:		
Descripción:	A vector illustration of a band performing in a concert.	School building in the village.
Descripción traducida:	Una ilustración vectorial de una banda tocando en un concierto.	Una construcción de una escuela en el pueblo.

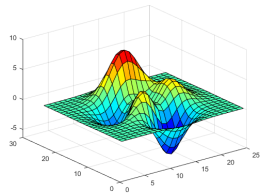
Imagen:		
Descripción:	Group of people dancing on a white background.	A plot of the number of particles in the sample.
Descripción traducida:	Un grupo de personas bailando en un fondo blanco.	Una gráfica del número de partículas en el ejemplo.
Imagen:		
Descripción:	Group of people running in the marathon.	
Descripción traducida:	Un grupo de personas corriendo en una maratón.	

Tabla 16 Resultados de la descripción de imágenes con ClipCap y Contextual Dataset

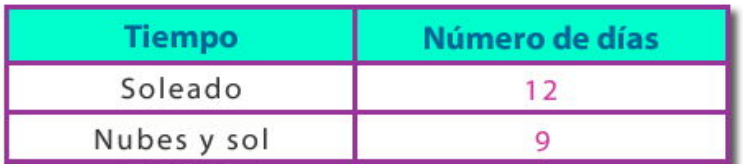
6.2.3. Descripción de fórmulas matemáticas:

Imagen:	$\frac{d}{dx}\left(\sqrt{x^2-81}\right)$	$\int_{\frac{\pi}{4}}^{\frac{\pi}{2}} sin\ y\ dy$
Descripción:	Differentiation of second root of all x square minus eighty-one as x approaches to nine.	Integral of y and sin y with respect to y from lower limit pi by four to upper limit pi by two.
Descripción traducida:	Diferenciación de la segunda raíz de todo x cuadrado menos ochenta y uno mientras x se acerca a nueve.	Integral de Y y seno de Y con respecto a Y de un limite inferior pi para cuatro y un limite superior pi para dos.

Imagen:	$2x < 9$	$\int sin^{-1}x \, dx$
Descripción:	Nine less than two times x.	Integral of inverse cos x with respect to x.
Descripción traducida:	Nueve menor que dos veces x.	Integral de coseno de x inverso con respecto a x.
Imagen:	$\lim_{x \to 8} \frac{3x^2 - 5x + 7}{8x^2 + 3x - 7}$	$3x - 10 = -2$
Descripción:	Left hand limit of eight times x square plus three times x minus seven all over minus three minus five times x plus ten times x square as x approaches to eight.	Minus three times x minus two equals to minus ten.
Descripción traducida:	Limite de mano izquierda de ocho veces x cuadrado mas tres veces x menos siete todo sobre menos tres menos cinco veces x mas diez veces x cuadrado cuando x se aproxima a ocho.	Menos tres veces x menos dos igual a menos 10.
Imagen:	$-8x - 6y = -9$ $3x - 3y = 4$	
Descripción:	Minus eight times x minus six times y equal to minus four and minus three times x equal to minus nine.	
Descripción traducida:	Menos ocho veces x menos seis veces e igual a menos cuatro y menos tres veces x igual a menos nueve.	

Tabla 17 Resultados de la descripción de fórmulas matemáticas usando MED

6.2.4. Descripción de tablas:

Imagen:	
Descripción:	<table border="1" class="dataframe"> <thead> <tr style="text-align: right;"> <th></th> <th>Tiempo</th> <th>Numerodedias</th> </tr> </thead> <tbody> <tr> <th>0</th> <td>Soleado</td> <td>12</td> </tr> <tr> <th>1</th> <td>Nubes y sc</td> <td>9</td> </tr> </tbody> </table>
Imagen:	
Descripción:	<table border="1" class="dataframe"> <thead> <tr style="text-align: right;"> <th></th> <th>Matricula</th> <th>Nombre</th> <th>Carrera</th> </tr> </thead> <tbody>

	<tr> <th>0</th> <td>1</td> <td>Garcia, Juan</td> <td>13</td> </tr> <tr> <th>1</th> <td>25</td> <td>Lucero, Mercedes</td> <td>1</td> </tr> </tbody> </table>

Tabla 18 Resultados de la descripción de tablas utilizando table_ocr

6.3.Integración del proyecto con la herramienta EduTech (OerAdap):

El proyecto en el que se ha trabajado forma parte de un proyecto más grande, el proyecto OerAdap del marco EduTech. El proyecto antes mencionado busca mejorar la accesibilidad a estudiantes con discapacidad a través de la adaptación de objetos de aprendizaje. Mencionado lo anterior se puede definir el alcance del presente proyecto en relación con el proyecto principal.

Como se mencionó, este proyecto se enfoca en la descripción de imágenes a cuatro niveles, imágenes generales, imágenes contextuales, fórmulas matemáticas y tablas, mientras que otros proyectos permiten, entre otras cosas, la lectura fácil a través del uso de procesamiento de lenguaje natural, el subtitulado automático de videos en tiempo real y la interacción entre usuario – objeto de aprendizaje.

El último proyecto mencionado está a cargo de Edwin Márquez y Claudio Maldonado, dicho proyecto permite hace uso de diferentes proyectos, entre el que se encuentra el presente proyecto. Como es de esperarse, se debe establecer un canal de comunicación que permita el consumo de los servicios que fueron definidos en este trabajo, por lo tanto, se optó por dos alternativas, la primera alternativa es a través del uso de un archivo tipo script escrito bash de Linux. Este script permite la comunicación directa entre el módulo principal y el módulo de descripción de imágenes. El segundo canal de comunicación es a través de una API que se encuentra definida usando FastAPI, sin embargo, la finalidad de este segundo canal de comunicación es permitir el uso del módulo para pruebas, mas no en un ambiente de producción.

A continuación, se detalla de una manera más precisa el alcance y funcionamiento de cada uno de los canales de comunicación mencionados anteriormente.

2. Script en bash:

Basado en los párrafos anteriores se dispuso que el módulo que este trabajo define forma parte de un proyecto más grande, por lo que, es imperativo permitir una comunicación confiable y rápida hacia el módulo de interacción para que este a su vez pueda utilizar los modelos de redes neuronales antes definidos. Para lograr lo anterior se propuso un script que en base a la dirección absoluta de una imagen hace uso de los servicios del proyecto.

En general el script llama de manera automática a la clasificación de la primera capa y según sus resultados procede realizar tres subrutinas, la primera es clasificar a través de las otras capas y describir la imagen o bien hacer uso de la descripción de fórmulas matemáticas o la extracción de datos de una tabla.

Al finalizar el proceso del script se genera un archivo que se llama ‘output.txt’, este archivo contiene información acorde a lo que haya determinado el modelo general de red neuronal, por lo que puede tener tres posibles salidas. La primera es una descripción de una imagen real o contextualizada, la segunda es la descripción de una fórmula matemática y la tercera es un código HTML que representa a una tabla extraída de una imagen.

A continuación, se expone un ejemplo del funcionamiento del script en mención:

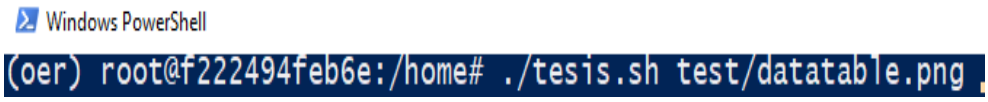


Ilustración 37 Llamada al script para la descripción de imágenes.

Ilustración 38 Resultados de la ejecución del script con una tabla.

3. API utilizando FastAPI:

Debido a que se requiere de una interfaz gráfica para poder demostrar el funcionamiento correcto del proyecto en el que se trabajó se optó por la creación de una API que permita la interacción con los modelos de redes neuronales, dicha API funciona de manera similar al script mencionado en el punto anterior, sin embargo, para este caso se utiliza Python mientras que para la parte visual o frontend se utiliza swagger.

A continuación, se presenta un ejemplo de la interfaz gráfica generada con swagger para el consumo de los modelos de clasificación y descripción:

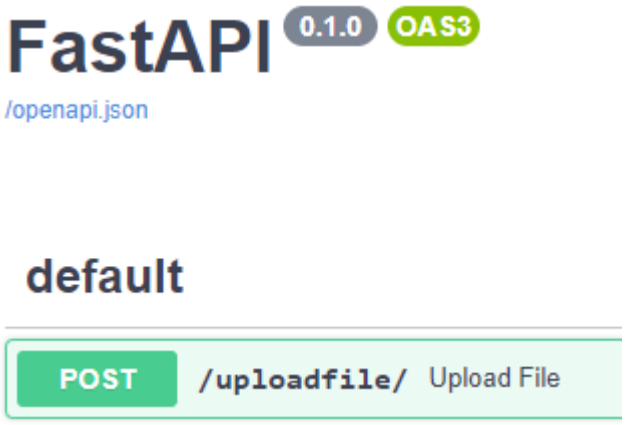


Ilustración 39 Método de la API en FastAPI.

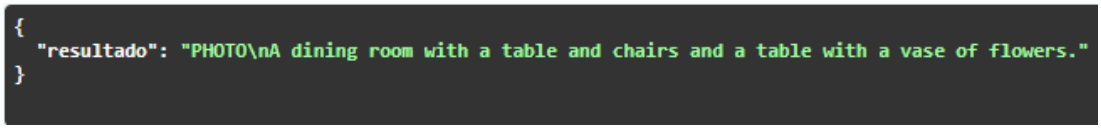


Ilustración 40 Resultado de la llamada a la API en formato JSON.

6.4.Limitaciones del presente proyecto:

Si bien en el presente proyecto se presentaron diferentes modelos y herramientas para la clasificación y de descripción de imágenes, estos modelos pueden ser mejorados de manera que se pueda obtener mejores resultados, sin embargo, debido a las siguientes limitaciones no se ha podido generar modelos cuyos resultados puedan ser considerados como state-of-the-art, aun así se han logrado tener grandes avances en relación a la mejora de accesibilidad a personas con problemas de visión a través de esta herramienta, a continuación se listan las limitaciones las cuales influyeron el desarrollo:

- **Tiempo:**

  Como se está consciente, dentro del cronograma de actividades se establece una cantidad de 300 horas en las cuales se dividieron todas las tareas delimitadas en el mismo cronograma, sin embargo, al ser este proyecto investigativo sumamente amplio, ya que existen muchas más áreas de interés, el tiempo, de cierta manera, puede ser considera insuficiente ya que no se han podido estudiar a fondo las investigaciones previas que se realizó en los artículos de interés relacionados a la investigación (punto V de metodología).

Aun así, se ha logrado hacer uso de dos técnicas, la primera Clipcap para la descripción de imágenes y la segunda MED para la descripción de tablas.

Aun así, el tiempo al jugar un rol importante, no ha permitido que se pueda mejorar tanto el rendimiento como otras métricas en función de obtener mejores resultados, por lo que existen ciertas limitantes a la hora de describir determinados gráficos. Estas limitaciones se mostrarán a continuación de las limitaciones.

- **Recursos computacionales:**

Como es de conocimiento dentro del área de computación se requiere de recursos computaciones altamente eficientes a la hora de trabajar con modelos de aprendizaje de máquina, y más aún cuando los conjuntos de datos son relativamente grandes, esto también se ve ligado al tamaño de la imagen con el que se trabaje y los canales de color.

Como se estableció en la sección de metodología, la computadora utilizada para el presente proyecto posee de prestaciones moderadas para el entrenamiento de los modelos, aun así, el tiempo de entrenamiento tomaba tiempo, por lo que no se podía subdividir el uso de los recursos para otra tarea. Los recursos computaciones de la computadora fueron esenciales para poder trabajar con modelos de Deep learning, ya que, gracias a la GPU (NVIDIA RTX 2060 Super) se aceleró el proceso de aprendizaje de forma radical comparado con el entrenamiento sobre CPU. Aun con dicha tarjeta gráfica de gama alta, el tiempo se pudo haber visto reducido de haberse trabajado con otros tipos de tarjetas de nivel empresarial como el modelo RTX 3060 que gracias a sus capacidades de memoria dedicada más amplia y su mayor número de procesadores CUDA hubieran sido sumamente útiles para lograr ampliar el rango de entrenamiento y poder hacer uso de técnicas como Hyperparameter tuning, que permite la mejora de resultados a través de la búsqueda de hiper parámetros óptimos.

Cabe recalcar que los recursos computacionales están directamente relacionados a la limitante de tiempo, ya que, debido a que los recursos eran utilizados completamente para el entrenamiento de un único modelo no se podía entrenar otro modelo.

- **Recursos de datos:**

Una de las tareas principales dentro del desarrollo de modelos de aprendizaje de maquina es la adquisición de los datos, esta tarea es fundamental ya que mientras haya una mayor cantidad de datos mejor serán los resultados,

aunque esto también se ve ligado a la calidad de los datos. Una de las limitaciones presentes en este.

Debido a que el presente proyecto busca definir una herramienta innovadora para la accesibilidad web no existe mucha información directamente relacionada al respecto, por lo tanto, a la hora de obtener datasets con información existieron determinados problemas, como, por ejemplo, la falta de imágenes para determinadas áreas, la baja calidad de determinadas imágenes y el sesgo de imágenes por clase o categoría.

Dentro del proyecto se ha presentado principalmente dos limitantes de los anteriores listados, a continuación, se determinará cuales fueron.

- **Falta de imágenes para determinadas categorías:**

  En base al diagrama en el que se establecen las capas del proyecto existe un punto del que trata de ilustraciones animadas, dicho punto genero una limitante grande a la hora de definir un modelo de clasificación de imágenes. Esto se debe a que no existe un dataset propiamente sobre este tipo de imágenes, sino que se tuvo que optar por obtenerlas de manera manual a través del uso de herramientas en línea, sin embargo, la cantidad de imágenes por categoría (10 categorías en total) fueron relativamente bajas (~1.000 por categoría), este problema, junto a la difícil tarea de discriminar entre imágenes sumamente similares llevo a que la falta de datos represente una grave limitante.

  Otro punto en donde la falta de imágenes jugó un factor determinante fue la clasificación de logos, ya que, nuevamente al no existir un dataset especializado para dicha tarea se generó uno, sin embargo, la cantidad de imágenes por categoría eran sumamente bajas (~10 a ~15 imágenes por cada una de más de 300 categorías) lo que hacía que la tarea de clasificación sea casi imposible.

- **Desequilibrio de imágenes por clase:**

  Para las tareas de aprendizaje de maquina rara ves se puede encontrar un dataset cuyos datos sean balanceados de manera correcta, con esto se hace referencia a que exista una cantidad similar de, en este caso, imágenes por categoría. Por ejemplo, en un dataset de animales de 10 categorías, se busca que cada una de las categorías tenga una cantidad similar de imágenes, de esta forma el modelo de redes neuronales puede generalizar sin problema.

  Si bien en el ejemplo del escenario anterior se muestra una situación en la cual se tiene conjuntos de datos balanceados, no es el caso de la

vida real. Dentro de los datasets en los que se ha trabajado existieron muchas veces categorías en las cuales las cantidades de imágenes se diferenciaban en gran medida de las otras categorías, esto surgió una necesidad imperante de solventar, ya que cuando se entrena un modelo de redes neuronales de esta manera se terminara dando más importancia a la clase cuya cantidad de imágenes sea superior al resto.

Para equilibrar la falta de balanceo de imágenes por categoría se utilizó de la librería sklean, a través de esta librería se utilizó una técnica conocida como Class Weights, esta técnica permite balancear el entrenamiento de un modelo acorde a la cantidad de imágenes disponibles. Por ejemplo, en una clasificación binaria en la cual se deban clasificar entre perros y gatos, en el primer caso se tienen 20.000 imágenes mientras que en el segundo caso se tienen únicamente 8.000 imágenes, claramente la segunda clase se ve en desventaja. Para esto la herramienta Class Weights lo que hace es brindarle un peso mayor de entrenamiento a la clase de menor cantidad de datos, esto quiere decir que, la red neuronal al momento de entrenar dará más importancia a la clase de gatos que a la de perros debido al desequilibrio de datos.

A continuación, se presentará los resultados en las cuales tanto el modelo de red neuronal propuesta como las herramientas de descripción de imágenes cometen errores relacionados con las limitaciones antes mencionadas.

6.4.1. Limitación en la descripción de imágenes generales:

Imagen:		
Descripción original:	A train traveling down train tracks next to a factory.	A person is holding a plant in their hand.
Descripción traducida:	Un tren viajando a través de rieles a lado de una empresa.	Una persona sosteniendo una planta en sus manos.

Imagen:		
Descripción:	A pair of scissors sitting on top of a grass covered field.	A black and white fire hydrant leaking water.
Descripción traducida:	Un par de tijeras asentadas encima de un campo cubierto de hierba.	Una boca de incendios en blanco y negro con una fuga de agua.

Tabla 19 Limitación de la descripción de imágenes generales.

Como se puede observar dentro de los resultados de la descripción de imágenes reales existen determinadas imágenes en las cuales el modelo ClipCap no funciona de manera adecuada, si bien a simple vista podría decirse que la imagen está relacionada a la descripción que provee, no sería correcto asumir que no existe una manera más precisa de hacerlo. Entre las posibles soluciones que se pueden proveer para mitigar la descripción incorrecta de estos tipos de imágenes se encuentran dos métodos principalmente:

Utilizar otros conjuntos de datos con mayor cantidad de información:

Si bien el conjunto de datos Microsoft COCO es sumamente extenso y contiene una gran cantidad de información, este no es lo suficientemente descriptivo en determinadas imágenes, ya es el caso de la primera imagen en la cual se establece que es un tren lo que se presente en la imagen, sin embargo, es un conjunto de fábricas. Entonces, para mejorar la descripción de este tipo de imagen se puede apoyar en el uso de conjuntos de datos como Flickr que tienen una cantidad de imágenes variadas que sirven como un punto de partida para obtener mejores resultados.

También se debe considerar la creación de conjuntos de datos propios ya que de esa forma se puede mitigar problemas de manera directa en donde se sabe que se tiene malos resultados.

Utilizar técnicas de clasificación de grano fino:

Como se encuentra en la sección de investigaciones, existen ciertos modelos de aprendizaje profundo en el cual se aplican técnicas para obtener características altamente descriptivas de imágenes que son complejas de entender a simple vista, estos modelos pueden servir como apoyo para el modelo principal de ClipCap mejorando la contextualización de las imágenes.

6.4.2. Descripción de ilustraciones animadas:

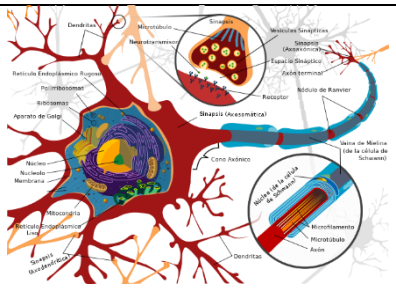
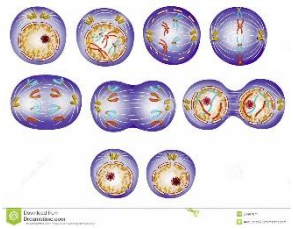
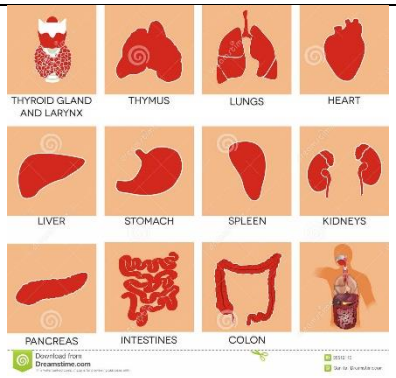
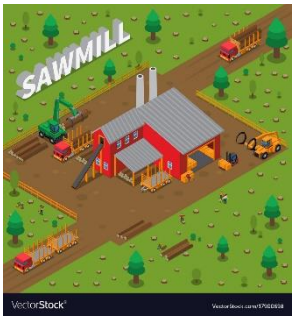
Imagen:		
Descripción:	The brain is the organ that processes information.	Illustration of the planets in the solar system.
Descripción traducida:	El cerebro es el órgano que procesa la información.	Ilustración de los planetas en el sistema solar.
Imagen:		
Descripción:	Set of vector illustrations of vegetables and fruits in a flat style.	Isometric illustration of a farm.
Descripción traducida:	Un grupo de ilustraciones de vegetales y frutas en un estilo plano.	Ilustración isométrica de una granja.

Tabla 20 Limitación de la descripción de imágenes contextuales.

De igual manera que en el caso de la descripción de imágenes reales se puede hacer uso de las dos técnicas anteriores para mejorar la precisión de la descripción de imágenes, sin embargo, en este tipo de imágenes se debe tener cuidado ya que no existen conjuntos de datos complemente especializados para esta tarea, por lo que es recomendable optar por crear un dataset propio que permita trabajar con soluciones para problemas que se conozcan que se pueden llegar a dar.

6.4.3. Descripción de fórmulas matemáticas:

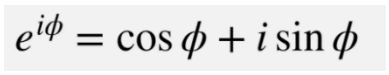
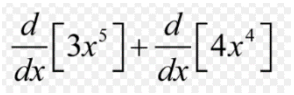
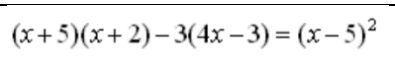
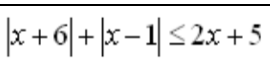
Imagen:		
Descripción:	Minus eight times x minus four greater than minus nine times x plus nine	Right hand limit of minus five times z plus three all over minus eight times z square plus five times z plus one as z approaches to eight
Descripción traducida:	Menos ocho veces x menos cuatro mayor que menos nueve veces x más nueve.	Límite de mano derecha de menos cinco veces z más tres todo sobre menos ocho veces z cuadrado más cinco veces z más uno cuando z se acerca a ocho.
Imagen:		
Descripción:	Integral of one minus inverse hyperbolic tan x with respect to x.	Integral of one plus inverse hyperbolic cos theta with respect to theta
Descripción traducida:	Integral de uno menos la tangente hiperbólica inversa de x con respecto a x.	Integral de uno más el coseno hiperbólico inverso de theta con respecto a theta.

Tabla 21 Resultados de la descripción de fórmulas matemáticas usando MED

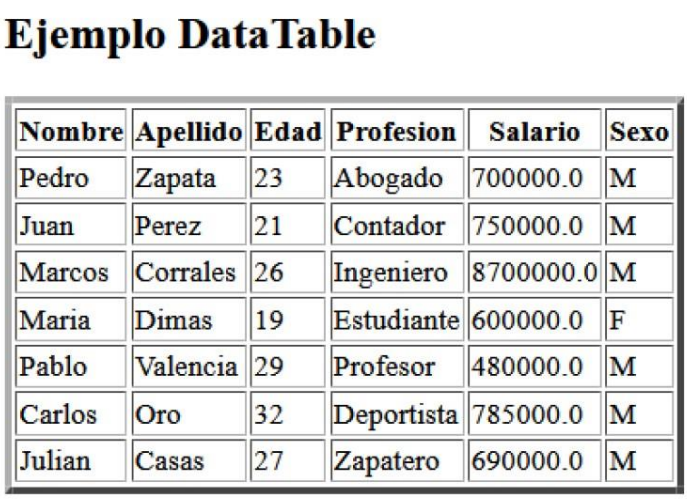
Si bien en el caso de la descripción de imágenes tanto reales como ilustraciones se habló sobre el uso de conjuntos de datos de apoyo que permitan mejorar la precisión de las descripciones, en el caso de las fórmulas existe un problema superior más difícil de solucionar.

En el presente proyecto se investigó diferentes fuentes de datos para la descripción de fórmulas, sin embargo no existen datasets específicamente para esta

tarea, sino que la mayoría se enfoca en la clasificación en lugar de la descripción de las mismas, por lo que en este caso se debe optar directamente por generar un conjunto de datos de manera manual, aun cuando esto puede llegar a ser tedioso y consumir tiempo el objetivo final ayudara a que trabajos futuros se puedan especializar en esta área ampliado más el alcance de la descripción de fórmulas más allá de los seis tipos que están propuesto en este trabajo.

6.4.4. Descripción de tablas:

Imagen:



Ejemplo DataTable

Nombre	Apellido	Edad	Profesion	Salario	Sexo
Pedro	Zapata	23	Abogado	700000.0	M
Juan	Perez	21	Contador	750000.0	M
Marcos	Corrales	26	Ingeniero	8700000.0	M
Maria	Dimas	19	Estudiante	600000.0	F
Pablo	Valencia	29	Profesor	480000.0	M
Carlos	Oro	32	Deportista	785000.0	M
Julian	Casas	27	Zapatero	690000.0	M

Descripción:

```
<table border="1" class="dataframe">
  <thead>
    <tr style="text-align: right;">
      <th></th>
      <th>Nombre |Apellido|Edad||Profesion| Salario |Sexo</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <th>0</th>
      <td>5st soneee</td>
    </tr>
    <tr>
      <th>1</th>
      <td>|Marcos|Corrales |26|ngeniero |8700000.0|M</td>
    </tr>
    <tr>
      <th>2</th>
      <td>NaN</td>
    </tr>
  </tbody>
</table>
```

Imagen:	
Descripción:	No se detecta correctamente la imagen, por lo tanto, no se puede obtener los datos de esta.

Nombre	Apellido	ID Depto.	Departamento
Alicia	Villareal	2	Mercadotecnia
Blanca	Díaz	2	Mercadotecnia
Daniel	Palacios	2	Mercadotecnia
Víctor	Lemus	5	Informática
Karen	Sánchez	5	Informática

Tabla 22 Resultados de la descripción de tablas utilizando table_ocr

Para el caso de la descripción de tablas se debe abordar de una manera diferente, el problema principal con las tablas es la falta de procesamiento de imágenes que permita mejorar la detección de celdas.

Como se mencionó en las investigaciones de técnicas disponibles para la descripción de tablas la herramienta table_ocr específicamente utiliza OpenCV para realizar preprocesamiento de las imágenes, sin embargo, el tratamiento que da en las imágenes es relativamente simple, por lo que no se preocupa por mejorar los bordes horizontales o verticales haciendo que estos sean más fáciles de detectar por la herramienta. Por lo tanto, una solución que puede ser aplicada para mejorar este tipo de clasificación es el uso de técnicas de visión por computador más avanzadas como detección de bordes, el refinamiento de las palabras, etc., con la finalidad de facilitar el trabajo de herramientas OCR.

Una solución más sofisticada para la detección y extracción de datos de las tablas es el uso de la herramienta CascadeTabNet, este modelo de red neuronal clasifica entre tablas con bordes o sin bordes y posteriormente extrae la información de estas, sin embargo, esta herramienta no pudo ser utilizada debido a que hace uso intensivo de la GPU, y debido a que el servidor en el cual será implementado el presente proyecto no tiene GPU se optó por lo utilizar dicho modelo.

7. Cronograma

Descripción de Actividades

- OE1. Estudiar y conocer los principios de la accesibilidad web y metadatos aplicados a objetos de aprendizaje.

No.	Actividad
1	Estudiar los fundamentos de accesibilidad web dentro del ámbito educativo.
2	Establecer la relación entre la accesibilidad y los objetos de aprendizaje y su relación dentro de la educación.
3	Estudiar los fundamentos básicos de los metadatos y como se aplican dentro de la web.
4	Determinar la importancia de los metadatos dentro de la educación.
5	Redacción de la información relevante en relación con la accesibilidad web y los metadatos.

- OE2. Diseñar y desarrollar un módulo basado en inteligencia artificial y procesamiento del lenguaje natural para generación de textos que permitan describir imágenes y la adaptación a contenido HTML.

No.	Actividad
1	Adquisición de conjuntos de imágenes (datasets) que formen parte de la estructura descrita en la ilustración 2.
2	Estudio diferentes propuestas para la clasificación de imágenes complejas como parte del estudio de trabajo previo, así como propuestas novedosas y sus aplicaciones relacionadas con el tema propuesto.
3	Diseño un modelo de Deep Learning factible para la clasificación de imágenes en base a modelos previamente establecidos.
4	Diseño un modelo de Deep Learning para la descripción de imágenes utilizando técnicas avanzadas como transfer learning.
5	Diseño un algoritmo de visión por computador para el reconocimiento de tablas de información y la obtención de información de esta.

6	Redacción de los métodos y estrategias utilizados dentro del área de visión por computador y Deep Learning en relación con el trabajo realizado.

- OE3. Despliegue del módulo en la herramienta de adaptación que forma parte de la arquitectura del sistema Edutech.

No.	**Actividad**
1	Adaptación del módulo de clasificación de imágenes en conjunto con el módulo de descripción de imágenes.
2	Redacción de los resultados en la comunicación entre los dos módulos.
3	Diseño de una API básica con FastAPI y Swagger.

- OE4. Diseñar y ejecutar un plan de experimentación que permita determinar la precisión del sistema con un dataset de objetos de aprendizaje.

No.	**Actividad**
1	Adquisición de un conjunto de imágenes que no formen parte de los datasets utilizados dentro del entrenamiento de la red neuronal.
2	Diseño un plan de pruebas con el nuevo conjunto de imágenes.
3	Ejecución de un plan de pruebas y obtener los resultados de esta en base a métricas de calidad.
4	Redacción de los resultados, recomendaciones y conclusiones.

- OE5. Desarrollo de los manuales técnico y de usuario.

No.	**Actividad**
1	Recopilación de los resultados e información obtenida de las actividades anteriores.
2	Estructuración del manual técnico acorde a las actividades desarrolladas.
3	Redacción del manual técnico y de usuario.
4	Revisión del manual técnico y de usuario.

El cronograma en base a la metodología SCRUM está formada por 4 Sprint, cada uno tiene una duración de 4 semanas, que representa aproximadamente a 1 mes de trabajo.

Sprint	Objetivo Especifico	Actividad por desarrollar
Uno	**OE. 1**	**Ac. 1, 2, 3, 4, 5**
	OE. 2	**Ac. 1, 2, 3, 4, 5**
Dos	**OE. 2**	**Ac. 6**
	OE. 3	**Ac. 1, 2, 3**
Tres	**OE. 4**	**Ac. 1, 2, 3**
Cuatro	**OE. 4**	**Ac. 4**
	OE. 5	**Ac. 1, 2, 3, 4**

Cronograma de Actividades

Objetivo	Actividad	Semana	Fecha Inicio	Fecha Fin	Días	Horas
OE. 1	Ac. 1	1	04/10/2021	04/10/2021	1	2
	Ac. 2	1	04/10/2021	04/10/2021	1	4
	Ac. 3	1	05/10/2021	05/10/2021	1	6
	Ac. 4	1	06/10/2021	06/10/2021	1	4
	Ac. 5	1	06/10/2021	07/10/2021	2	8
OE. 2	Ac. 1	1	08/10/2021	09/10/2021	2	10
	Ac. 2	2	11/10/2021	13/10/2021	3	14
	Ac. 3	2-3	14/10/2021	20/10/2021	5	22
	Ac. 4	3-4	21/10/2021	03/11/2021	10	42
	Ac. 5	4	04/11/2021	05/11/2021	5	22
	Ac. 6	5	08/11/2021	11/11/2021	4	18
OE. 3	Ac. 1	5-6	12/11/2021	17/11/2021	4	18
	Ac. 2	6	18/11/2021	19/11/2021	2	8
	Ac. 3	6	19/11/2021	19/11/2021	1	4
OE. 4	Ac. 1	7	22/11/2021	24/11/2021	3	14
	Ac. 2	7	25/11/2021	26/11/2021	2	10
	Ac. 3	8	29/11/2021	02/12/2021	4	14
	Ac. 4	9-10	03/12/2021	09/12/2021	5	22
OE. 5	Ac. 1	10-11	10/12/2021	14/12/2021	3	14
	Ac. 2	11	15/12/2021	17/12/2021	3	14
	Ac. 3	12-13	03/01/2022	11/01/2022	7	26
	Ac. 4	13	12/01/2022	12/01/2022	1	4

Cantidad de semanas: 13

Total, de Días: 66

Total, de Horas: 300

Fecha de Inicio: 04/10/2021

Fecha de Finalización: 12/01/2022

8. Presupuesto

DENOMINACION	CANT.	COSTO UNITARIO	COSTO TOTAL
	Unidades	Dólares	Dólares
1. Bienes			
Papel bond A-4	2	1.00	1.00
Copias	100	0.05	5.00
2. Tecnológico			
Computador de escritorio (AMD Ryzen 7 – 512 SSD – 16GB RAM – Geforce RTX 2060 Super)	1	2700.00	2700.00
3. Servicios			
Servicio de internet (investigación de artículos académicos)	3	35.00	105.00
Material bibliográfico.	2	14	28
4. Personal			
Estudiante investigador	1	1,300.00	1,300.00
5. Otros			
Imprevistos	1	100.00	100.00
TOTAL		4,136.05	4,239.00

9. Conclusiones

En el presente trabajo se ha logrado determinar la importancia que tiene la accesibilidad web dentro del ámbito educativo ya que gran cantidad de estudiantes cuyas discapacidades ameritan el uso de herramientas especializadas para hacer uso de objetos de aprendizaje no logran acceder a la información presente en internet, esto debido a que no existen instrumentos de aprendizajes completamente adaptados ante el software disponible.

Tal y como se ha podido comprobar en la sección de resultados, se puede observar que los modelos de aprendizaje profundo son de gran utilidad debido a su versatilidad de uso, esto los hace idóneos para el uso en herramientas de aprendizaje educativo en las cuales otras técnicas no son completamente adecuadas para su uso. Tal es el caso de la descripción de imágenes, si bien se podría hacer de forma manual, esta tarea requiere de tiempo, por lo que normalmente se opta por no realizarla sino obviarla lo que lleva a que los objetos de aprendizaje se vean incompletos e inaccesibles por los navegadores web.

A pesar de las limitaciones que se han presentado en el proyecto, se ha logrado establecer un delineamiento base para trabajos futuros gracias al apoyo abnegado que existe dentro de la universidad y el trabajo realizado por el grupo de investigación GI-IATA por lo que gracias a ellos se podrá avanzar hacia el crecimiento educativo inclusivo y abierto a través del uso de tecnologías emergentes generando nuevas fuentes de conocimiento para los estudiantes de la Universidad Politécnica Salesiana.

Trabajo futuro

Es importante aclarar que este proyecto es la base de proyectos futuros en los cuales a través de nuevas técnicas y herramientas se pueda mejorar y adaptar una mayor cantidad de capas de información, llegando a reducir el limitado acceso de la educación a estudiantes con discapacidad. Entre los temas de interés que quedan como trabajo futuro se encuentran, pero no se limitan a, clasificación de imágenes en mayor rango, obtención de conjuntos de datos más grandes para mejorar las métricas de calidad de los modelos, mejora de la descripción de imágenes utilizando procesamiento de lenguaje natural a mayor escala, aumento de categorías dentro de la descripción de fórmulas matemáticas, y desarrollo de una herramienta propia para la transformación de tablas a HTML utilizando técnicas establecidas en el trabajo relaciono.

Dentro del marco EduTech se planean realizar trabajos futuros a fin con el presente proyecto, a continuación, se presentan dos líneas principales de trabajo directamente relacionadas al proyecto.

Generación de mapas HTML a través de visión por computador:

Como primera área de trabajo futuro se encuentra la generación de diagramas tipo mapa utilizando HTML y visión por computador. El objetivo de esta herramienta es facilitar la explicación de la imagen que represente a un diagrama. Para lograr este objetivo se requiere de utilizar técnicas de visión por computador e inteligencia artificial.

Entre las finalidades de este proyecto se encuentran mejorar la accesibilidad a un nivel más específico, en este caso, a nivel de diagrama, de esta forma a partir de una imagen estática se generará código HTML que permita a los usuarios navegar entre nodos haciendo más interactivo el proceso de manipulación.

Generación de un conjunto de datos para la descripción de imágenes de manera manual:

Como se estableció en la subsección de limitaciones del proyecto, uno de los mayores problemas que se presentó conforme se desarrolló el módulo de descripción fue la falta de imágenes para determinadas tareas, por ejemplo, para la descripción de fórmulas se limita únicamente a 6 categorías, mientras que para el resto no existe información. Debido a lo anterior se establece como una línea de trabajo futuro la obtención de conjuntos de datos manuales que permitan el entrenamiento y mejoramiento de la descripción de fórmulas matemáticas e imágenes en general.

Recomendaciones

La sección de recomendaciones se divide en tres subsecciones entre las cuales se enlista cuáles son las sugerencias de implementación de este proyecto, así mismo se establece como este trabajo presenta un punto de partida y abre las puertas a mejoras y nuevos trabajos en áreas relacionadas en función de la investigación que se realizó en puntos anteriores.

Recomendaciones en el ámbito de desarrollo:

Como parte de las recomendaciones para la sección de desarrollo de la aplicación existen diferentes puntos que van a ser abordados, entre las secciones que forman parte de esta sección de recomendaciones se encuentran:

- **Recomendaciones relacionadas al hardware empleado:**

  Los proyectos de Deep Learning trabajan con grandes volúmenes de datos por lo que, para obtener resultados de manera efectiva se requiere del uso de hardware especializado, en el caso de este proyecto se requiere de contar con una GPU. Este componente de hardware permite conducir el entrenamiento de redes neuronales de una forma más rápida debido a que una GPU tiene una gran cantidad de núcleos y tensores los mismos que permiten realizar cálculos más rápido que una CPU.

- **Recomendaciones relacionadas al uso de librerías:**

  El proyecto en el que se trabajó se basó en el uso de librerías como Tensorflow, Keras, Sklearn, Matplotlib, Pytorch, entre otras. Para utilizar las librerías antes mencionadas, y de manera general cualquier librería, es recomendable que se lea la documentación de estas, ya que de esa manera se puede evitar malentendidos de problemas que puedan surgir al momento de entrenar o probar una red neuronal.

  Uno de los errores más comunes al trabajar con librerías como Tensorflow y/o Keras es el mal manejo de los parámetros de entrada y de salida, por ejemplo, cuando se desea trabajar con una imagen de 3 canales entonces la variable que representa la imagen debe estar formado por el tamaño columnas, filas y finalmente la cantidad de canales, sin embargo, este parámetro vario acorde a los canales que sean empleados. Otro problema común dentro de la definición de capas en Keras es el manejo incorrecto de los parámetros de perdida y la cantidad de neuronas de salida, lo que lleva a que no se ejecute el código correctamente, o peor aún, que se ejecute el código, pero genere valores de entrenamiento sin sentido.

Con respecto al resto de librerías los errores son más específicos y dependen de su uso, por lo que, como recomendación general es, verificar el origen del error y cuál es el mensaje de salida, ya que muchas veces el error se genera en líneas adyacentes en lugar de la línea que marca error.

- **Recomendaciones relacionadas al manejo del tiempo:**

El tiempo juega un factor importante independiente del proyecto que se esté realizando, pero para el caso de proyectos investigativos este factor tiene mayor importancia, por lo que, es recomendable utilizar metodologías agiles como SCRUM para manejar de mejor manera el tiempo, a través de esta metodología se puede dividir el proyecto en 'pedazos' más pequeños que puedan ser tratables y estimables dentro de un rango de tiempo. Una ventaja que tiene SCRUM es también la flexibilidad que provee ya que se puede ajustar a casi cualquier tipo de proyecto por lo que, para este caso es recomendable su uso, de esa forma se evita contratiempos.

Recomendaciones en el ámbito de producción:

- **Uso de recursos computacionales avanzados:**

Como ya se estableció anteriormente, el servidor de despliegue del proyecto cuenta con recursos limitados, entre las limitaciones se conlleva la falta de una GPU, debido a que no se tiene una tarjeta gráfica no se han podido utilizar modelos de redes neuronales especializados para determinadas tareas, como es el caso de CascadeTabNet. Por lo tanto, es recomendable de que cuando se despliegue un proyecto de esta clase en la cual se haga uso de modelos de investigación se tenga acceso a una GPU, de esta forma se abre camino a procesos de inferencia con mejores resultados a comparación del uso de técnicas tradicionales a través de librerías.

- **Manejo de versiones de librerías:**

Las librerías que fueron empleadas, tanto para el entrenamiento como para el despliegue del módulo requirieron de determinadas versiones, dichas versiones están en constante cambio, si bien los cambios no son muy significativos, estos pueden llegar a generar desde mensajes de advertencia hasta errores graves que no permitan ejecutar el código. Entonces es imperativo manejar un archivo de 'requirements.txt' que contenga las librerías necesarias

junto con sus versiones específicas para que el proyecto pueda funcionar correctamente, y de darse el caso, modificar el código para actualizarlo acorde a los cambios de las librerías.

Otra recomendación con respecto a este tema es el de mantener un historial de versiones dentro de un ambiente de desarrollo como Anaconda, a través de estos ambientes se puede mantener versiones específicas de librerías que se sepan que están funcionando, y cuando se requieran de estas simplemente se puede exportar dicho ambiente evitando el problema de volver a instalar todo desde cero.

- **Optimización de recursos:**

Cuando se desarrolla código normalmente no se utiliza al pie de la letra las recomendaciones de desarrollo o buenas prácticas, esto a la final termina generando código basura que a su vez genera problemas de rendimiento o mal uso de recursos. Un ejemplo de esta recomendación es el verificar que se utilicen las librerías estrictamente necesarias ya que muchas de las veces quedan librerías que son importadas, pero nunca usadas, otro ejemplo es la ejecución de código muerto que no a la final no sirve para nada o que su función puede ser simplificada a través de código existente que es más optimo.

El objetivo de esta recomendación es que se evite malgastar recursos del servidor en el que se aloja el módulo haciendo que la ejecución sea más lenta y menos productiva.

Recomendaciones en el ámbito investigativo:

- **Investigación de trabajo relacionado:**

Como se puede evidenciar en el presente trabajo, gran parte del trabajo se basa en modelos o técnicas previamente existentes por lo que, una parte considerable del tiempo fue empleado en investigar diferentes fuentes de información que tengan material relacionado al proyecto de investigación.

Existen diferentes fuentes en las cuales se puede extraer información, sin embargo, hay unas cuantas fuentes de interés que permiten obtener información de manera precisa y adjuntan, en la mayoría de los casos, el código fuente del proyecto realizado. A continuación, se presente los enlaces en los cuales se puede empezar una investigación de temas de aprendizaje de maquina y Deep Learning.

https://paperswithcode.com/

Este primer enlace contiene información sobre las últimas técnicas aplicada en el campo de la investigación, además adjuntan sus resultados y el código implementado en frameworks famosos como Tensorflow o Pytorch.

https://arxiv.org/list/cs.AI/recent

La página arXiv provee de un gran conjunto de artículos académicos con gran relevancia en diferentes áreas de investigación como matemática, física, estadística, entre otras. Sin embargo, el área más interesante para el proyecto es el área de inteligencia artificial. Cabe recalcar que todos los artículos publicados en arXiv son de acceso libre y gratuito por lo que permite que cualquier persona pueda utilizar su información.

- **Investigación de datasets:**

El pilar inicial dentro de un proyecto de investigación es la obtención de datos, este proceso a su vez presenta una gran limitante para obtener buenos resultados o completar determinada tarea, por lo tanto, a continuación, se enlista un conjunto de fuentes en las cuales se puede buscar información de manera libre y de uso educativo.

https://paperswithcode.com/datasets

De la misma manera que en el punto anterior, este sitio web permite acceder tanto a artículos académicos como a un gran conjunto de datasets de dominio público, lo que lo hace idóneo para tener una idea de por dónde empezar por una investigación ya que la mayoría de los conjuntos de datos ya tienen un modelo que los utiliza permitiendo establecer un ejemplo de la carga de datos y su manipulación.

https://datasetsearch.research.google.com/

Este motor de búsqueda es relativamente nuevo, pero aun así permite navegar entre diferentes alternativas de conjuntos de datos, una ventaja de esta herramienta es que informa sobre el nivel de acceso a los datos, por ejemplo, determinados datasets permiten el uso únicamente educativo, mientras que otros permiten el uso comercial de los mismos.

https://www.kaggle.com/datasets

Finalmente, una gran base de datos y de desarrollo es Kaggle, este sitio web a través de competencias permite que cualquier persona suba sus conjuntos de datos de manera libre, permitiendo así tener un conjunto de datos más amplio y libre. Aun así, la ventaja de acceso libre también representa una desventaja ya que se puede subir conjuntos de datos sin preprocesar o con datos que no tienen relación con el dataset.

Referencias

Alemi, A. (2016, Agosto 31). *Improving Inception and Image Classification in Tensorflow*. Obtenido de Google AI Blog: https://ai.googleblog.com/2016/08/improving-inception-and-image.html

Bazi, Y., Bashmal, L., Al Rahhal, M., Al Dayil, R., & Al Ajlan, N. (2021). Vision Transformers for Remote Sensing Image Classification. *MDPI Open Access Journals*.

Bozinovski, S. (2020). Reminder of the first paper on transfer learning in neural networks, 1976. *Research Gate*.

Consejo Nacional para la Igualdad de Discapacidades. (2021, Septiembre 01). *Estadisticas de Discapacidad*. Obtenido de CONADIS Estadisticas de Discapacidad: https://www.consejodiscapacidades.gob.ec/estadisticas-de-discapacidad/

Deng, J. D.-J.-F. (2009). Imagenet: A large-scale hierarchical image database. *IEEE*.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., . . . Houlsby, N. (2020). An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv*.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., . . . Houlsby, N. (2021). An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale. *ICLR*, 1-22.

Duong, N., Tran, T., & Xuan, H. (2019). Automated Pneumonia Detection in X-Ray Images via Depthwise Separable Convolution Based Learning. *ResearchGate*, 2.

El Naqa, I., Li, R., & Murphy, M. J. (2015). What Is Machine Learning? En I. El Naqa, R. Li, & M. J. Murphy, *Machine Learning in Radiation Oncology* (págs. 3-4). Montreal: Springer Nature.

Fu, J., Zheng, H., & Mei, T. (2017). Look Closer to See Better: recurrent Attention Convolutional Neural Networks for Fine-Grained Image Recognition. *IEEE Xplore*.

Haiby, N. (2021). Enhanced Table Extraction from documents with Form Recognizer. *Azure AI Blog*.

Henry, S. L. (2019, July 11). *Introducción a la Accesibilidad Web*. Obtenido de W3C: https://www.w3.org/WAI/fundamentals/accessibility-intro

Hu, J., Shen, L., Albanie, S., Sun, G., & Wu, E. (2017). Squeeze-and-Excitation Networks. *arXiv*.

IBM. (s.f.). *What is computer vision?* Obtenido de IBM: https://www.ibm.com/topics/computer-vision

IMS Global Learning Consortium. (2021). *Accessibility*. Obtenido de IMS Global Learning Consortium: https://www.imsglobal.org/activity/accessibility

Karim Barznji, H. (1997). Transfer Learning as New Field in Machine Learning. *Academia*, 204-211.

Kekare, A., Jachak, A., Gosavi, A., & Hanwate. (2020). Techniques. *Techniques for Detecting and Extracting Tabular Data from PDFs and Scanned Documents: A Survey*.

Khan, S., Naseer, M., Hayat, M., Waqas Zamir, S., Shahbaz Khan, F., & Shah, M. (2021). Transformers in Vision: A survey. *ArXiv*, 20-25.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep Learning. *Nature*, 1-2-3.

Liddy, E. (2001). Natural Language Processing. En E. Liddy, *Encyclopedia of Library and Information Science* (págs. 1-11). New York: Marcel Dekker.

Marcelino, P. (2018, October 23). *Transfer learning from pre-trained models*. Obtenido de Towards Data Science: https://towardsdatascience.com/transfer-learning-from-pre-trained-models-f2393f124751

Modaky, R., Hertz, A., & H. Bermano, A. (2021). ClipCap: CLIP Prefix for Image Captioning. *arXiv*.

Mondal, A., & Jawahar, C. (2019). Textual Description for Mathematical Equations. *ICDAR*.

Mora, S. L. (2006). *Accesibilidad Web*. Obtenido de Universidad de Alicante: http://accesibilidadweb.dlsi.ua.es/?menu=mitos

Morales, F. (2021, Octubre). Attention Map in Vit-Keras. Houston, Texas, United States of America.

National Information Standards Organization. (2004). Understanding Metadata. En N. Press, *Understanding Metadata* (págs. 1-2). Bethesda: NISO Press.

Ngiam, J., Peng, D., Vasudevan, V., Kornblith, S., V. Le, Q., & Pang, R. (2018). Domain Adaptative Transfer Learning with Specialist Models. *arXiv*.

Paliwal, S., D, V., Rahul, R., Sharma, M., & Vig, L. (2020). TableNet: Deep Learning model for end-to-end Table detection and Tabular data extraction from Scanned Document Images. *ArXiv*.

Pathak, A. (2021). *Website accessibility*. Obtenido de Geekflare: https://geekflare.com/es/test-web-accessibility

Prabhu, D. (2021, October 01). *The importance of ALT attributes in images*. Obtenido de Band the Table: https://helpdesk.bangthetable.com/en/articles/3654115-the-importance-of-alt-attributes-in-images

Prasad, D., Gadpal, A., Kapadni, K., Visave, M., & Sultanpure, K. (2020). CascadeTabNet: An approach for end to end table detection and structure recognition from image-based documents. *arXiv*.

Quanzeng, Y., Hailin, J., Zhaowen, W., Chen, F., & Jiebo, L. (2016). Image Captioning with Semantic Attention. *Computer Vision Foundantion*, 1-8.

Schreiber, S., Agne, S., Wolf, I., Dengel, A., & Ahmed, S. (2017). DeepDeSRT: Deep Learning for Detection and Structure Recognition of Tables in Document Images. *IEEE Xplore*.

Sharma, S. (2017, September 6). *Activation Functions in Neural Networks*. Obtenido de Towards Data Science: https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6

St. Olaf College. (2022). *Stolaf*. Obtenido de Precision vs. Accuracy: https://wp.stolaf.edu/it/gis-precision-accuracy/#:~:text=Precision%20is%20how%20close%20measure,end%20of%20a%20given%20measurement.&text=Accuracy%20is%20how%20close%20a,are%20both%20precise%20and%20accurate.

Their world. (2021). *Explainer Children with disabilities*. Obtenido de Their World: https://theirworld.org/explainers/children-with-disabilities

Tucci, L. (2018, May). *Artificial Neuron*. Obtenido de SearchCIO: https://searchcio.techtarget.com/definition/artificial-neuron

Universidad de Antioquia. (s.f.). *Introduccion a objetos de aprendizaje (OA).* Obtenido de Universidad De Antioquia: https://ingenieria2.udea.edu.co/multimedia-static/aemtic/unidad_4/descargas/objetos_aprendizaje.pdf

University Of British Columbia. (1998). Computational Intelligence A Logical Approach. En D. Poole, A. Mackworth, & R. Goebel, *Computational Intelligence A Logical Approach* (págs. 1-2). New York: Oxford University Press.

University of North Carolina Chapell Hill. (2021, June 29). *Metadata for Data Management: A Tutorial*. Obtenido de UNC University Libraries.: https://guides.lib.unc.edu/metadata/importance

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., N. Gomez, A., . . . Polosukhin, I. (2017). Attention Is All You Need. *arXiv*.

W3C. (2020). *Introduction to Understanding WCAG 2.0*. Obtenido de https://www.w3.org/TR/UNDERSTANDING-WCAG20/intro.html#introduction-fourprincs-head

Wang, Mingming;University, Dalhousie. (2015). Multi-path Convolutional Neural Networks for Complex Image Classification. *arXiv*.

Web AIM. (2020, April 14). *Introduction to Web Accessibility.* Obtenido de WebAIM: https://webaim.org/intro/

WHO. (2021, Febrero 26). *WHO Ceguera y discapacidad visual*. Obtenido de Ceguera y discapacidad visual: https://www.who.int/es/news-room/fact-sheets/detail/blindness-and-visual-impairment#:~:text=A%20nivel%20mundial%2C%20se%20estima,tienen%20m%C3%A1s%20de%2050%20a%C3%B1os.

Xiao, T., Singh, M., Mintun, E., Darrel, T., Dollar, P., & Girshick, R. (2021). Early Convolutions Help Transformers See Better. *arXiv*.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhutdinov, R., . . . Bengio, Y. (2015). Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. *arXiv*.

You, Q., Jin, H., Wang, Z., Fang, C., & Luo, J. (2016). Image Captioning with Semantic Attention. *arXiv*.

Zheng, H., Fu, J., Zha, Z.-J., & Luo, J. (2019). Looking for the Devil in the Details: Learning Trilinear Attention Sampling Network for Fine-Grained Image Classification. *arXiv*.