

Proyecto Semestral:

Clasificador de frutas en imágenes y Clasificador de noticias

Cristopher Acevedo, Sebastián Acevedo, Nicolás Ibarra
COM4402 – Introducción a Inteligencia Artificial
Escuela de Ingeniería, Universidad de O'Higgins
22, 12, 2023

Abstract— El informe presenta un análisis del desempeño de dos códigos considerando su eficacia y aplicabilidad, para ello se utilizarán los dos códigos disponibles: “Clasificación de frutas usando CNN” y “Clasificación de noticias textuales usando CNN y RNN”. Se ejecutarán los dos respectivos códigos para posteriormente realizar un análisis exhaustivo de los resultados del entrenamiento para su comparación y evaluación.

Palabras clave— Redes Neuronales Convolucionales, Redes Neuronales Recurrentes, Aprendizaje Profundo

I. INTRODUCCIÓN

A día de hoy, en la era de la inteligencia artificial, las redes neuronales desempeñan un rol muy importante, estas estructuras inspiradas en el funcionamiento del cerebro humano han demostrado ser muy versátiles en distintas aplicaciones, desde el reconocimiento de patrones, el procesamiento de imágenes, la generación de imágenes, etc.

A lo largo de este informe se llevará un análisis detallado, viendo el uso que tienen las redes neuronales en clasificación, para la realización del informe, se analizarán los resultados de dos códigos en específico:

1. **Clasificación de frutas usando CNN:** Se profundizará en la aplicación de las redes neuronales convolucionales con el fin de clasificar frutas y realizar un control de calidad de las mismas. El conjunto de datos que se utilizara para el entrenamiento del modelo es el conjunto **Fuit-360**, el cual contiene 82213 imágenes de frutas y verduras de 120 clases distintas (las imágenes son de 100x100 píxeles).
2. **Clasificación de noticias textuales usando CNN y RNN:** Se analizará el entrenamiento y funcionamiento de tanto las redes neuronales

convolucionales, como de las redes neuronales recurrentes, puesto que, si tan solo se utilizaran redes neuronales convolucionales, (a pesar de la gran capacidad de las mismas de reconocer patrones y características) la red no sería capaz de mantener un orden temporal a la hora de analizar los datos. Los datos a analizar serán del texto [Tann LeCun, 2016 \[1\]](#).

Para el análisis en estas distintas redes, se tomarán en cuenta los resultados de las mismas, para ello se analizarán sus parámetros y así analizar, interpretar y discutir la calidad de estos códigos a la hora de clasificar.

II. PROGRAMAS UTILIZADOS

Para la elaboración de este informe, se empleó [Google Colab \[2\]](#), una plataforma de Google Research que facilita la escritura y ejecución de código Python directamente en un navegador web. Esta herramienta permite a cualquier usuario llevar a cabo tareas de programación de manera ágil y sencilla. Será utilizado para ejecutar los dos códigos proporcionados para ver sus resultados y realizar su posterior análisis.

III. Marco teórico

A. Redes neuronales y entrenamiento.

Una red neuronal es un modelo computacional que se fundamenta en la emulación del funcionamiento del cerebro humano para el procesamiento de información. Está compuesta por una extensa cantidad de unidades de procesamiento interconectadas que representan a las neuronas del cerebro humano, desempeñando un papel

fundamental en los cálculos y la toma de decisiones en el contexto de una red neuronal.

El proceso de entrenamiento de modelos consta de varias etapas:

1. **Recopilación de datos:** En esta fase, se define la selección de la base de datos que será utilizada como fuente de información.
2. **Selección de un modelo:** En esta etapa, se elige un modelo de aprendizaje automático. Para el propósito de este informe, los modelos de aprendizaje seleccionados son las redes neuronales.
3. **Preprocesamiento de datos:** En esta etapa, se llevan a cabo tareas de procesamiento de datos con el objetivo de preparar la información para su utilización en el modelo de aprendizaje. Esto asegura que los datos estén en un formato adecuado para ser procesados por el modelo.
4. **Entrenamiento del modelo:** En esta fase, el modelo procesa los parámetros a lo largo de múltiples épocas, durante las cuales se procesan los datos de manera iterativa para minimizar la pérdida o el error.

B. Redes Neuronales Convolucionales (CNN).

Las redes neuronales convolucionales o CNN por sus siglas en inglés “*Convolutional Neural Network*” es un tipo de red neuronal las cuales han demostrado ser especialmente eficaces a la hora de trabajar en el procesamiento de imágenes y el reconocimiento de patrones. Estas tienen variadas características distintivas, entre ellas se encuentran las capas convolucionales, capas de agrupación o *pooling*, etc.

Las redes neuronales convolucionales han revolucionado el campo del aprendizaje profundo y han sido fundamentales para avances en reconocimiento facial, detección de objetos, segmentación semántica y más.

C. Capa Convolutiva.

Estas capas son las encargadas de aplicar filtros a regionales locales dentro de la entrada (la imagen), permitiendo a la red neuronal extraer características

o patrones específicos de la imagen de entrada. Es particularmente eficaz y útil a la hora de identificar formas, bordes y texturas.

D. Capas de agrupación (*pooling*).

Las capas de agrupación o *pooling*, son las capas que vienen luego de las capas convolucionales, estas son las encargadas de reducir la dimensión en los datos de la entrada, esto con el propósito de reducir el tamaño del mismo, lo que implica que la red neuronal debe entrenar con menos parámetros, evitando así el sobreajuste y acelerando el entrenamiento. Con esta reducción también se busca extraer solo las características más relevantes y representativas de la imagen de entrada, permitiendo así que se retenga la información esencial de la imagen.

E. Redes Neuronales Recurrentes (RNN).

Las redes neuronales recurrentes o RNN por sus siglas en inglés “*Recurrent Neural Network*” son un tipo de red neuronal la cual está diseñada para trabajar con datos de una manera secuencial, con una relación temporal. A diferencia de las redes neuronales convolucionales, las RNN poseen conexiones que forman ciclos dentro de la estructura de la misma, lo cual les permite manejar y utilizar información sobre eventos anteriores en la secuencia. El orden de los datos es importante en este tipo de redes neuronales.

Posee varias características claves, que definen una RNN, entre ellas: las conexiones recurrentes, la memoria a corto y largo plazo y el manejo de datos secuenciales. Todas las anteriores herramientas esenciales para la red, con el fin de modelar la dependencia temporal en los datos.

IV. METODOLOGÍA

En la metodología se incluirán las arquitecturas principales de las dos redes neuronales vistas en el transcurso del informe, siendo estas, la red neuronal convolutiva y la red neuronal recurrente.

1. Red Neuronal Convolucional

La arquitectura principal de una red neuronal convolucional está compuesta por los siguientes componentes:

- **Entrada de datos:** Se inicia un preprocesamiento de los datos, con el fin de limpiar, normalizar y transformar los datos para su posterior uso.
- **Capa de Entrada:** Representa los datos de entrada, que pueden ser imágenes, texto o cualquier otro tipo de datos.
- **Capa convolucional:** Capa que realiza operaciones de convolución en los datos, mediante la aplicación de filtros, esto se hace con el fin de extraer las características de la entrada y realizar mapas de características.
- **Capa de pooling:** Capa de agrupación encargada de reducir la dimensión de la entrada de datos mediante sub-muestreo
- **Capa de aplanado (flatten):** Capa encargada de transformar la salida multidimensional de la capa anterior a una matriz unidimensional, preparando los datos para las entradas siguientes.
- **Capa totalmente conectada:** Capa en la cual cada neurona de la red está conectada a neuronas de las anteriores y posteriores capas. Su función es captar los patrones y relación entre los datos.
- **Capa de salida:** Esta capa representa la salida final de la red neuronal, es la responsable de generar los resultados deseados.
- **Función de Pérdida:** Mide la diferencia entre los datos de entrada y los datos reconstruidos. Función común para pérdida es la entropía cruzada binaria usada para tareas de clasificación binaria.
- **Optimizador:** Se utiliza un algoritmo de optimización, para ajustar los pesos de la red, con el propósito de minimizar la función de pérdida total.
- **Entrenamiento:** Durante el entrenamiento de una CNN, el proceso se lleva a cabo en lotes de datos para mejorar la generalización del modelo. En cada iteración, conocida como época, los datos se propagan a través de la red desde la capa de entrada hasta la capa de salida. Durante este proceso, se calcula la pérdida comparando las

predicciones del modelo con las etiquetas reales de los datos de entrenamiento.

2. Red Neuronal Recurrente

La arquitectura principal de una red neuronal recurrente está compuesta por los siguientes componentes:

- **Entrada de datos:** el proceso de entrada de datos se inicia con preprocesamiento de los datos, con la función de limpiar, normalizar y transformar los datos brutos. Este paso es crucial para asegurar que los datos estén en un formato adecuado para la RNN, involucrando prácticas como la eliminación de datos irrelevantes, la normalización de escalas y la aplicación de transformaciones específicas según la naturaleza de los datos.
- **Capa de Entrada:** Responsable de recibir los datos de entrada, estos pueden ser secuencias de puntos de datos como datos de series temporales u oraciones.
- **Capa recurrente:** Componente principal y central de la red neuronal recurrente, es la encargada de introducir bucles o conexiones entre los distintos nodos para permitir que la información persista en el tiempo, permitiendo a la red capturar dependencias temporales.
- **Entrenamiento:** Durante el entrenamiento de una RNN, se lleva a cabo un proceso de iteración de datos con el fin de mejorar la generalización del modelo. En cada iteración, conocida como época, los datos se propagan a través de la red desde la capa de entrada hasta la capa de salida. Durante este proceso, se calcula la pérdida comparando las predicciones del modelo con las etiquetas reales de los datos de entrenamiento.
- **Pesos y sesgos:** Parámetros que la red aprende durante el entrenamiento con el propósito de capturar las relaciones entre los datos de entrada y las predicciones de la salida
- **Capa de salida:** Esta capa representa la salida final de la red neuronal, es la responsable de generar los resultados deseados.
- **Función de Pérdida:** Mide la diferencia entre los datos de entrada y los datos reconstruidos.

Durante el entrenamiento se tiene como propósito minimizar esta función de perdida.

- **Retropropagación:** Algoritmo utilizado para entrenar la RNN, se calculan gradientes con respecto a la perdida y se actualizan los pesos para minimizar la misma.
- **Optimizador:** Determina como se ajustan los pesos de la red, con el propósito de minimizar la función de pérdida total.
- **Módulos de preprocesamiento y postprocesamiento:** Manejan el preprocesamiento de los datos, antes de alimentarlos a la red y el postprocesamiento de las salidas del modelo.

V. RESULTADOS

Los resultados obtenidos luego de haber realizado el entrenamiento y la evaluación de las tres estructuras y modelos de redes utilizadas, es decir, el modelo con una red neuronal convolucional para el conjunto de datos *Fruit-360*, y el modelo donde se combinaron una red neuronal convolucional junto con una red neuronal recurrente para el subconjunto de datos *AG News*, fueron los siguientes:

A. Red Neuronal Convolucional utilizando el conjunto de datos *Fruit-360*

TABLA I
RENDIMIENTO DE LA RED EN EL CONJUNTO DE ENTRENAMIENTO Y DE VALIDACIÓN

# of epochs	Train loss	Validation Loss	Train accuracy	Validation Accuracy
10	0.0334	0.0333	1.0000	1.0000

VISUALIZACIÓN DEL LOSS Y ACCURACY EN EL CONJUNTO DE ENTRENAMIENTO Y VALIDACIÓN POR ÉPOCAS

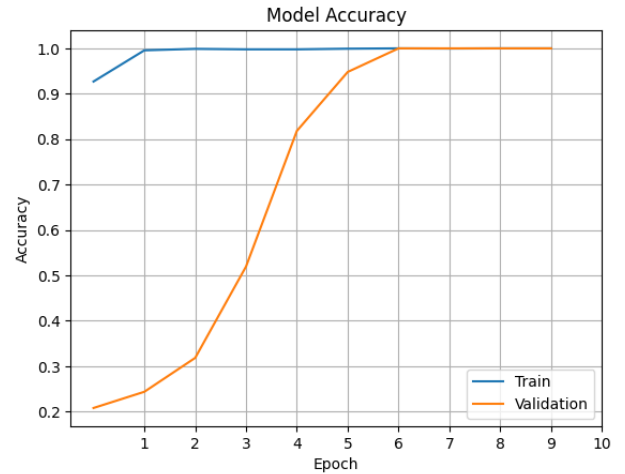
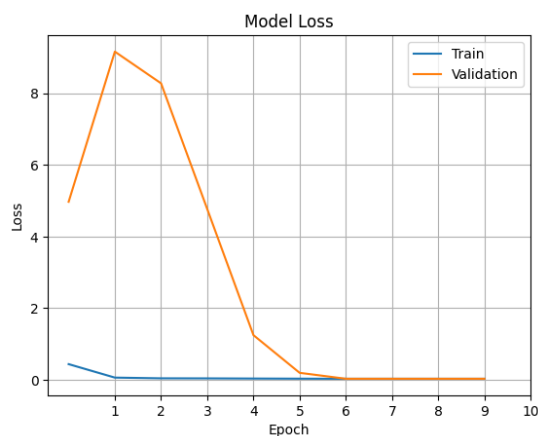
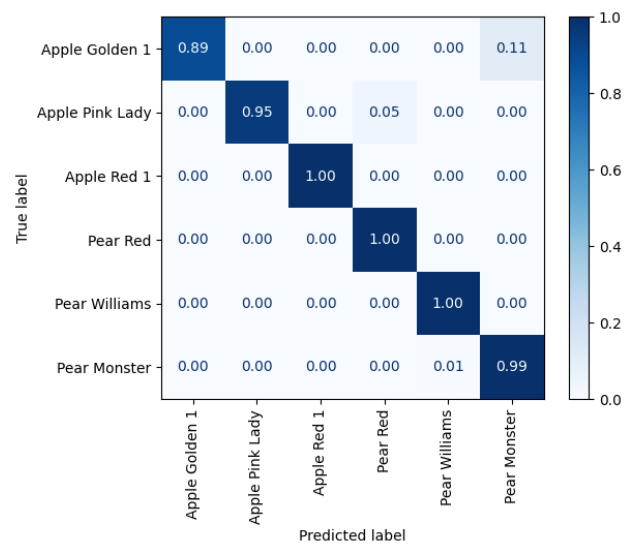


TABLA II
RENDIMIENTO DE LA RED EN EL CONJUNTO DE PRUEBA

Test Accuracy	Test Precision Macro Avg.	Test Recall Macro Avg.
0.9784	0.97	1030

MATRIZ DE CONFUSIÓN DEL CONJUNTO DE PRUEBAS



B. Red neuronal Convolucional y Red Neuronal Recurrente utilizando el conjunto de datos *AG News*

TABLA III
RENDIMIENTO DE LA RED EN EL CONJUNTO DE ENTRENAMIENTO Y DE VALIDACIÓN UTILIZANDO SOLO UNA RED NEURONAL CONVOLUCIONAL

# of epochs	Train loss	Validation Loss	Train accuracy	Validation Accuracy
25	0.3382	0.2506	0.8494	0.9161

VISUALIZACIÓN DEL LOSS Y ACCURACY EN EL CONJUNTO DE ENTRENAMIENTO Y VALIDACIÓN POR ÉPOCAS UTILIZANDO SOLO UNA RED CONVOLUCIONAL

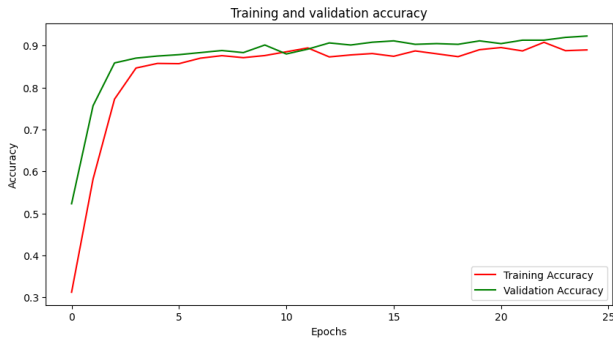
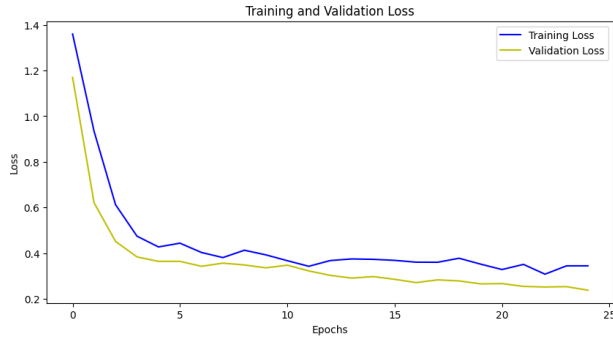
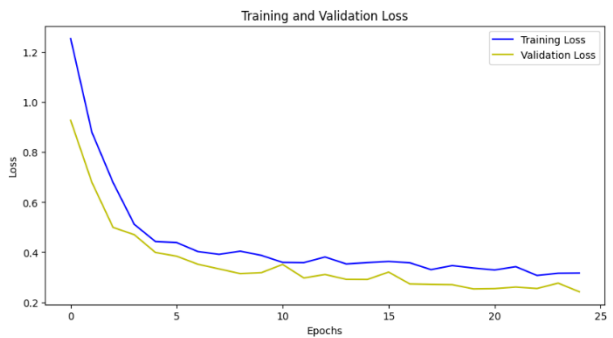


TABLA IV

RENDIMIENTO DE LA RED EN EL CONJUNTO DE ENTRENAMIENTO Y DE VALIDACIÓN UTILIZANDO UNA RED NEURONAL CONVOLUCIONAL Y UNA RED NEURONAL RECURRENTE

# of epochs	Train loss	Validation Loss	Train accuracy	Validation Accuracy
25	0.2997	0.2687	0.9036	0.9260

VISUALIZACIÓN DEL LOSS Y ACCURACY EN EL CONJUNTO DE ENTRENAMIENTO Y VALIDACIÓN POR ÉPOCAS UTILIZANDO SOLO UNA RED CONVOLUCIONAL



VI. ANÁLISIS

A. Análisis del rendimiento de una Red Neuronal Convolutiva en el conjunto de datos Fruit-360

Al utilizar el conjunto de datos Fruit-360 con un modelo de una red neuronal convolutiva para el reconocimiento, identificación y clasificación de frutas, se pudo observar en general un rendimiento sobresaliente en general del modelo en cuanto a la calidad de las predicciones y en el costo computacional de la red. De esta manera, en el conjunto de entrenamiento y validación, se observó un buen ajuste del modelo a los datos, lo que evidencia una gran adaptación del modelo, obteniendo (en 10 épocas) un *loss* (pérdida) de 0.0334 y un *accuracy* de 1, es decir, de un 100% en el conjunto de entrenamiento, y un *loss* de 0.0333 y un *accuracy* también de 100% en el conjunto de validación. Un rendimiento muy similar en el conjunto de entrenamiento y en el conjunto de validación, lo que nos indicó que el modelo aprendió patrones útiles durante el entrenamiento y es capaz de generalizar bien frente a datos no vistos anteriormente.

Como se observó en el gráfico con la curva de la evolución del *loss* y del *accuracy* para ambos conjuntos, se evidenció que el modelo se ajusta de buena manera a los datos utilizando 6 épocas, por lo que es posible reducir el número de épocas de 10 a 6, disminuyendo aún más el costo computacional y reduciendo/evitando la posibilidad de un sobreajuste del modelo.

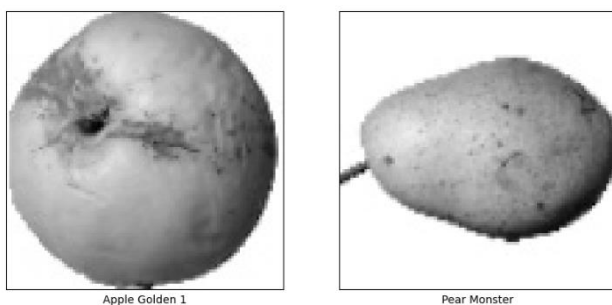
Por otra parte, en cuanto al conjunto de pruebas, se pudo observar un *accuracy* de 0.9748, esto quiere decir, que el modelo clasificó correctamente un 97.48% de las imágenes del conjunto de pruebas,

afirmando que la red neuronal convolucional utilizada tiene un buen rendimiento al momento de clasificar nuevos datos/imágenes que no había visto anteriormente.

Asimismo, en la matriz de confusión del conjunto de pruebas, fue posible notar fácilmente que el modelo en donde cometió más errores de clasificación fue al momento de clasificar/predecir incorrectamente las imágenes con etiqueta correspondiente al tipo “Pear Monster” como “Apple Golden 1”, clasificando correctamente el tipo “Apple Golden 1” en el 89% de los casos, y en un 11% de los casos clasificando incorrectamente dicho tipo, asignándole la etiqueta de “Pear Monster”.

Al analizar este caso más en detalle, se pudo identificar que hay imágenes con características similares entre el tipo “Apple Golden 1” y “Pear Monster” ubicando las frutas en un cierto ángulo cada una, tal como se ejemplifica en las imágenes adjuntadas posteriormente.

COMPARACIÓN TIPOS DE FRUTAS
IZQUIERDA: “APPLE GOLDEN 1”, DERECHA: “PEAR MONSTER”.



El ejemplo previo clarifica que, a diferencia de las demás frutas, fue posible observar que comparten ciertas características en común como “manchas” oscuras. Otra característica usual dentro de las muestras adjuntadas con anterioridad corresponde al mismo tono de grises y así sucesivamente con diversas similitudes encontradas. Debido a esto, se pudo inferir que es probable que suceda el error en clasificar incorrectamente la fruta “Pear Monster” como “Apple Golden 1”, ya que desde un distinto ángulo, por ejemplo, desde la parte de abajo de la fruta, la fruta “Pear Monster” tendrá una forma “redonda” parecida a la fruta “Apple Golden 1”, y combinando esto con las demás características

mencionadas anteriormente tales como manchas y/o color, es posible que los filtros correspondientes de la fruta “Apple Golden 1” tengan un peso más significativo al momento de clasificar la imagen.

B. *Análisis del rendimiento de una Red Neuronal Convolucional combinada con una Red Neuronal Recurrente en el conjunto de datos AG News*

Al utilizar el modelo de solo una red neuronal recurrente en el conjunto de datos AG News para clasificar noticias, se pudo observar un *loss* de 0.3382 y un *accuracy* de un 89.49% en el conjunto de entrenamiento, y un *loss* de 0.2506 con un *accuracy* de un 91.61% en el conjunto de validación, utilizando 25 épocas para el entrenamiento y validación del modelo. En general, la red neuronal recurrente utilizada tuvo un buen rendimiento al clasificar las noticias, es decir, al identificar patrones en las cadenas de texto, en donde hubo un rendimiento similar entre el *accuracy* del conjunto de entrenamiento y del conjunto de validación. Si bien existió una leve diferencia de aproximadamente un 3%, esta no es lo suficientemente significativa para deducir que hay un error en el entrenamiento de la red o que hubo presencia de underfitting.

En la curva de evolución del *loss* y *accuracy* del conjunto de entrenamiento y validación, se pudo observar que hay épocas en las cuales el rendimiento es menor que el de la época anterior, lo que nos da a saber que, si utilizáramos más épocas, podríamos sufrir de overfitting en la red. Además, se observó que el *loss* y *accuracy* mejoran rápidamente entre la primera época y la décima, por lo que se podría disminuir el número de épocas y tener una disminución mínima en el rendimiento, pero se disminuiría el costo computacional y se evita el riesgo de sobreajustar el modelo o que las métricas de *loss* y *accuracy* comiencen a empeorar o no mejorar.

Al combinar una red neuronal convolucional con una red neuronal recurrente, utilizando la misma red neuronal convolucional anterior y agregando una capa LSTM bidireccional para prever la secuencia contenida en las características de salida de la red neuronal convolucional, se pudo observar una leve mejora en el *loss* y *accuracy* del conjunto de

entrenamiento y de validación, por lo que agregar una capa que ayude a comprender el orden temporal en el texto sí tiene un pequeño impacto positivo en el rendimiento, pero en términos más generales, es casi el mismo entre ambos modelos. Por otro lado, el fenómeno de las épocas es el mismo: es posible reducir el número de épocas y reducir aún más el costo computacional, que, de comienzo, es bajo.

de noticias completas (muchos párrafos), de distintos idiomas y semánticas.

VII. CONCLUSIONES GENERALES

La estructura o modelos de redes neuronales convolucionales implementadas tuvieron un buen rendimiento en general en los conjuntos de datos Fruit-360 y AG News para la clasificación de imágenes y de texto, respectivamente, siendo de aproximadamente un 97% en el primer modelo para el conjunto de pruebas, y de aproximadamente un 92% en los otros dos modelos utilizados para el conjunto de entrenamiento y de validación.

En el caso de la clasificación de imágenes, una pequeña diferencia en la imagen, como el ángulo de la fotografía, puede llevar al modelo a realizar clasificaciones/predicciones incorrectas, lo que puede tener un gran impacto en otras aplicaciones. Por ejemplo, si el problema que se enfrenta trata de identificar la existencia de contaminación en una fruta, en donde el ángulo de la fotografía podría llevar al modelo a identificar una fruta contaminada como una fruta saludable, pero de otro tipo, en donde comparten alguna característica como tener una “mancha”. Debido a esto es esencial construir una red adecuada y tener los suficientes datos de entrenamiento para garantizar la eficiencia del modelo.

El modelo implementado en el conjunto AG News para la clasificación de noticias deja en evidencia la gran capacidad que tienen las redes neuronales convolucionales de aprender representaciones jerárquicas para comprender el texto e identificar patrones para clasificar noticias nuevas. Agregar a este mismo modelo una capa LSTM bidireccional mantener el orden de los pasos temporales en el modelo, si bien en este caso no fue un aumento significativo en el rendimiento del modelo, sí proporciona una leve mejora en la clasificación de texto (noticias, en este caso), pudiendo tener un gran impacto en la clasificación

VIII. BIBLIOGRAFÍA

- [1] [Zhang, Xiang, and Yann LeCun. "Text understanding from scratch." arXiv preprint arXiv:1502.01710 \(2015\).](#)
- [2] Ignacio Bugueño, Alfonso Ehijo, Felipe Gómez, Camila Rapu, "Clasificación de frutas usando CNN", Código base Proyecto Semestral, Escuela de Ingeniería, Universidad de O'higgins.
https://colab.research.google.com/drive/1FqF23ASvssd9RP5s-h96D_CHKMjVEopu?usp=sharing
- [3] Ignacio Bugueño, Alfonso Ehijo, Felipe Gómez, Camila Rapu, "Clasificación de noticias textuales usando CNN y RNN", Código base Proyecto Semestral, Escuela de Ingeniería, Universidad de O'higgins.
<https://colab.research.google.com/drive/1RD7HovwEcIcp2icUDFxW3n7nH6hN89S?usp=sharing>
- [4] "Google Colab". Google Research. Accedido el 22 de diciembre de 2023. [En línea]. Disponible:
[https://research.google.com/colaboratory/intl/es/faq.html#:~:text=Conceptos%20básicos&text=Colaboratory.%20o%20Colab"%20para, análisis%20de%20datos%20y%20educación](https://research.google.com/colaboratory/intl/es/faq.html#:~:text=Conceptos%20básicos&text=Colaboratory.%20o%20Colab)