Detección de Imágenes Reales y Artificiales

Juliana Henao, Valentina Moreno, Geronimo Zuluaga y Santiago Cano

***Palabras clave*—Clasificación, Redes Neuronales Convolucionales (CNN), Deep Learning**

I. INTRODUCCIÓN

El aumento masivo de información es evidente para cualquier persona que tenga contacto con internet. Desde lo más formal o académico, como papers cientificos, hasta lo más trivial y cotidiano como un meme. Esto, como todo, tiene sus implicaciones beneficiosas y nocivas para la sociedad moderna. Un mundo conectado, información a la mano, comunicación global, etc. Son unos de los muchos beneficios que nos trae este crecimiento de la información, pero, tambien está su lado oscuro: Las fake news, la adicción a las redes sociales, el plagio, son también unos pocos ejemplos de los muchos existentes provocados por esta misma razón.

En específico, este proyecto está motivado por una tecnología insurgente causada por lo avanzada que está el área de inteligencia artificial, y es el Deepfake. El Deepfake es la manipulación de imagenes o videos por la cual se logra, o bien hacer mover una imagen inanimada, o colocar la cara de alguien en un video en el que no aparecía, donde, ha llegado a tal punto que puede ser indetectable para el ojo humano. No hay que forzar mucho a la intuición para imaginarse qué se podría hacer con esta tecnología, es por esto que es perentorio desarrollar en paralelo una tecnología capaz de contrarrestar los efectos nocivos que pudiera causar el Deepfake. Con este objetivo en mente, en el proyecto a realizar, se ha elegido la implementación de una herramienta, desarrollada con redes neuronales, para la identificación de imagenes reales o artificiales.

II. PLANTEAMIENTO DEL PROBLEMA

Videos, fotos y audios ya no son pruebas indiscutibles. En los últimos años tras el creciente auge de las herramientas tecnológicas se han empezado a desarrollar diferentes instrumentos que amenazan con la veracidad de una foto, los deepfakes, también conocidos como falsedades profundas son archivos de vídeo, imagen o voz manipulados mediante un software de inteligencia artificial de modo que parezcan originales, auténticos y reales.

Aunque los Deepfakes existen desde finales de 1990, cobraron interés en 2017, cuando un usuario de Reddit publicó material audiovisual falso con los rostros de varias actrices famosas.

La capacidad de parecer tan reales viene dada por la capacidad de modelado que tienen los programas informáticos dedicados a realizar los Deepfakes, que tratan de asemejarse lo más posible al funcionamiento de las redes neuronales y del cerebro humano.Los Deepfakes utilizan el aprendizaje automático de la inteligencia artificial. Esta tecnología se basa en sofisticados algoritmos que son capaces de analizar si un archivo es real o si está alterado y, de esta forma, la inteligencia artificial puede ir mejorando cada vez más en la labor de falsificar de manera más fidedigna. Los Deepfakes pueden ser generados directamente por softwares u ordenadores especializados en este aprendizaje automático, sin necesidad de intervención humana.

La falsedades profundas presentan un gran riesgo para la sociedad al ser capaces de inducir a error a las personas receptoras de los archivos, ya sea haciendo que un político diga algo en un vídeo que realmente nunca dijo para afectarle en una campaña política cerca de unas elecciones, o incluyendo la imagen de un famoso (o de cualquier persona) en un material explícito con el objetivo de perjudicarle o chantajearle.

III. MÉTODO

La identificación de objetos en imágenes es uno de los problemas clásicos de la Inteligencia Artificial. Al inicio de la tecnología de identificación de imágenes, el aprendizaje automático se basaba en redes neuronales y era más que suficiente para detectar elementos en imágenes pequeñas, sin embargo, esos mismos algoritmos se vuelven ineficientes cuando aumentamos el tamaño de dichas imágenes.

A mayor tamaño de la imagen mayor número de píxeles que producen un aumento exponencial de las variables de entrada (features) imposibles de manejar por una red neuronal o redes neuronales convolucionales de arquitectura tradicional.

Las convolutional neural networks o redes neuronales convolucionales (CNN), solucionan este problema ya que asumen ciertas características espaciales de los inputs que permiten simplificar las arquitecturas de la red reduciendo, en gran medida, el número de variables de entrada. Por tanto, son especialmente útiles en problemas de visión por computador, y en particular, en el reconocimiento de objetos.

Como redes de clasificación, al principio se encuentra la fase de extracción de características, compuesta de neuronas convolucionales y de reducción de muestreo. Al final de la red se encuentran neuronas de perceptron sencillas para realizar la clasificación final sobre las características extraídas. La fase de extracción de características se asemeja al proceso estimulante en las células de la corteza visual. Esta fase se compone de capas alternas de neuronas convolucionales y neuronas de reducción de muestreo. Según progresan los

datos a lo largo de esta fase, se disminuye su dimensionalidad, siendo las neuronas en capas lejanas mucho menos sensibles a perturbaciones en los datos de entrada, pero al mismo tiempo siendo estas activadas por características cada vez más complejas.

Neuronas convolucionales: en la fase de extracción de características, las neuronas sencillas de un perceptron son reemplazadas por procesadores en matriz que realizan una operación sobre los datos de imagen 2D que pasan por ellas, en lugar de un único valor numérico. La salida de cada neurona convolucional se calcula como:

$$Y_j = g\left(b_j + \sum_i K_{ij} \otimes Y_i\right)$$

Donde la salida Y_j de una neurona j es una matriz que se calcula por medio de la combinación lineal de las salidas Y_i de las neuronas en la capa anterior cada una de ellas operadas con el núcleo de convolucional K_{ij} correspondiente a esa conexión. Esta cantidad es sumada a una influencia b_j y luego se pasa por una función de activación $g(\cdot)$ no-lineal.

El operador de convolución tiene el efecto de filtrar la imagen de entrada con un núcleo previamente entrenado. Esto transforma los datos de tal manera que ciertas características (determinadas por la forma del núcleo) se vuelven más dominantes en la imagen de salida al tener estas un valor numérico más alto asignados a los pixeles que las representan. Estos núcleos tienen habilidades de procesamiento de imágenes específicas, como por ejemplo la detección de bordes que se puede realizar con núcleos que resaltan la gradiente en una dirección en particular. Sin embargo, los núcleos que son entrenados por una red neuronal convolucional generalmente son más complejos para poder extraer otras características más abstractas y no triviales.

Neuronas de Reducción de Muestreo: Las redes neuronales cuentan con cierta tolerancia a pequeñas perturbaciones en los datos de entrada. Por ejemplo, si dos imágenes casi idénticas (diferenciadas únicamente por un traslado de algunos pixeles lateralmente) se analizan con una red neuronal, el resultado debería de ser esencialmente el mismo. Esto se obtiene, en parte, dado a la reducción de muestreo que ocurre dentro de una red neuronal convolucional. Al reducir la resolución, las mismas características corresponderán a un mayor campo de activación en la imagen de entrada.

La capa de pooling (POOL) es un tipo de capa que está presente en una gran cantidad de arquitecturas CNN. Su utilidad consiste en reducir las representaciones obtenidas de manera que estas se hagan más pequeñas y sean más manejables computacionalmente, reduciendo el número de parámetros necesarios. La operación de max-pooling encuentra el valor máximo entre una ventana de muestra y pasa este valor como resumen de características sobre esa área. Como resultado, el tamaño de los datos se reduce por un factor igual al tamaño de la ventana de muestra sobre la cual se opera.

Neuronas de Clasificación: después de una o más fases de extracción de características, los datos finalmente llegan a la fase de clasificación. Para entonces, los datos han sido depurados hasta una serie de características únicas para la imagen de entrada, y es ahora la labor de esta última fase el poder clasificar estas características hacia una etiqueta u otra, según los objetivos de entrenamiento.

$$y_j = g\left(b_j + \sum_i w_{ij} \cdot y_i\right)$$

Donde la salida y_j de una neurona j es un valor que se calcula por medio de la combinación lineal de las salidas y_i de las neuronas en la capa anterior cada una de ellas multiplicadas con un peso w_{ij} correspondiente a esa conexión. Esta cantidad es sumada a una influencia b_j y luego se pasa por una función de activación $g(\cdot)$ no-lineal.

Adicional a la red neuronal convolucional, se presentan los métodos bagging que consisten en el uso de múltiples modelos en paralelo para luego producir una predicción en base a cada una de las predicciones realizadas por los modelos. El principal objetivo de los métodos bagging es aprovechar la independencia que tiene cada uno de los modelos, asi se puede combatir los problemas de overfitting.

Por tanto, se realizo un método bagging con múltiples modelos simples de la red neuronal utilizada y explicada posteriormente.

III-A. Datos

Los datos usados para el desarrollo de esta investigacion fueron tomados de [4], el cual cuenta con un total de 3993 fotos, de los cuales se tienen 1961 fotos reales y 2032 fotos generadas por una inteligencia artificial, por lo que se puede observar que se cuenta con un conjunto de datos balanceado.

Figura 1. Imágenes ejemplo de los datos con sus etiquetas

III-B. *Consideraciones de aprendizaje*

Se realizó una partición de los datos en 80 % y 20 % para entrenamiento y pruebas, respectivamente. En la figura 2 se puede detallar la proporción de datos que fueron designados, cada barra del histograma representa una de las dos clases, la barra al lado izquierdo corresponde a los datos generados por la inteligencia artificial y la barra al lado derecho a los datos reales. Con color rojo se puede identificar los datos de prueba y con color azul los datos de entrenamiento. Se usaran como etiquetas el valor 0 para las fotos generadas de manera artificial y el valor 1 las fotos reales.

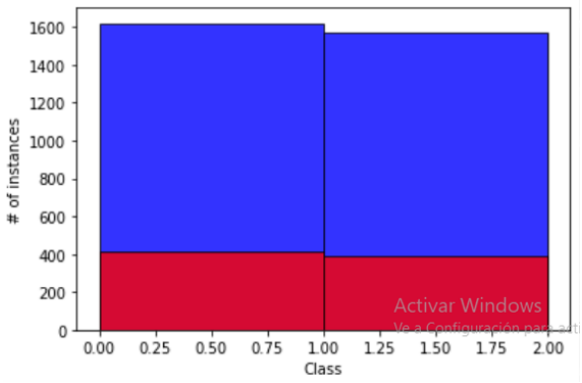

Figura 2. Frecuencia de las dos clases, para train y test

III-C. *Función de pérdida*

La función de pérdida trata de determinar el error entre el valor estimado y el valor real, con el fin de optimizar los parámetros de la red neuronal. En este caso se utilizó la función de entropía cruzada, o pérdida logarítmica:

$$H_p(q) = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot \log\left(p\left(y_i\right)\right) + \left(1-y_i\right) \cdot \log\left(1-p\left(y_i\right)\right) \quad (1)$$

Esta mide el rendimiento de un modelo de clasificación cuya salida es un valor de probabilidad entre 0 y 1. La pérdida de entropía cruzada aumenta a medida que la probabilidad predicha diverge de la etiqueta real.

III-D. *Estructura de la red*

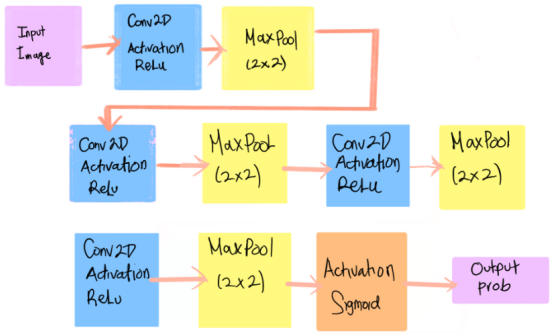

Figura 3. Arquitectura de la red neuronal

IV. RESULTADOS

En las siguientes secciones se pueden ver las tablas y las figuras donde se encuentra el rendimiento de las arquitecturas utilizadas.

IV-A. *Red Neuronal Convolocional*

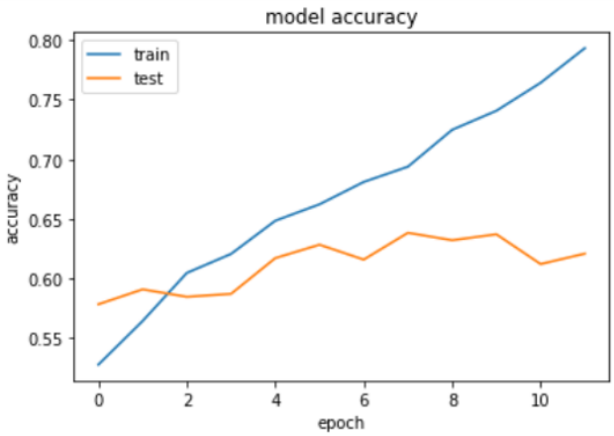

Figura 4. Accuracy de la red neuronal

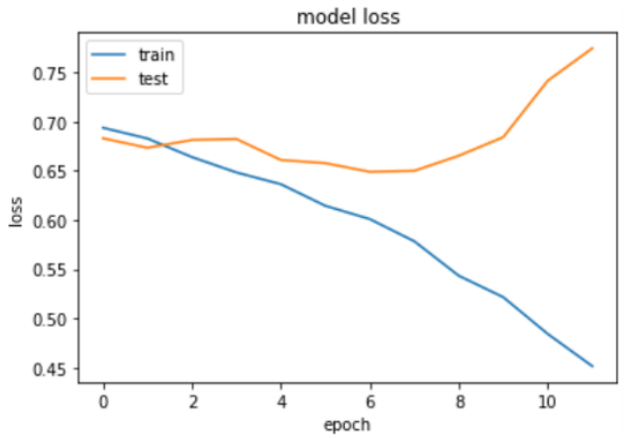

Figura 5. Función de pérdida de la red neuronal

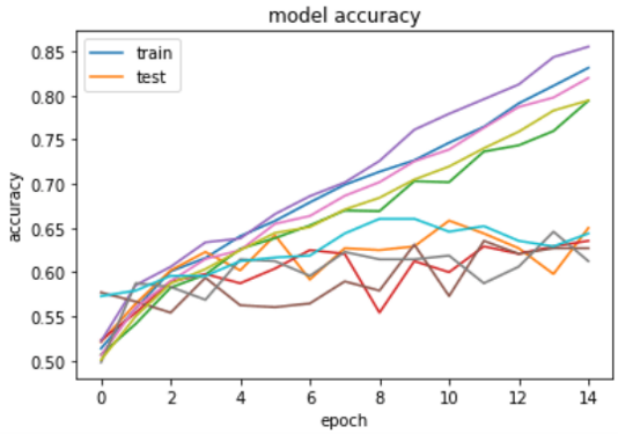

Figura 6. Accuracy de la red neuronal con bagging

Modelo	Accuracy test	Accuracy train	Loss	Error clase 0	Error clase 1
Red Neuronal	0.6333	0.8240	0.8735	0.5430	0.581

Cuadro I
RESULTADOS EN VALIDACIÓN PARA MODELO BÁSICO

Se evidencia, por la diferencia que hay entre los resultados de train y test, que existe un alto overfitting en el modelo. En la figura 4 se observa que en las primeras épocas los resultados en train y test no difieren sustancialmente, pero, a medida que avanza en las épocas, el modelo parece estar arrojando mejores resultados en train pero en el conjunto de datos de test empieza a decaer el accuracy.

Esta diferencia sugiere que el modelo se está aprendiendo los datos, generando así overfitting. Este comportamiento también se puede evindenciar en la función de pérdida, pues, aproximadamente en la séptima época, la función en test empieza a crecer y crear una brecha considerable con la de train, análogo a lo que sucedió con el accuracy.

Por último en la tabla II se encuentran los valores exactos de lo mencionado anteriormente y, claramente, hay una diferencia evidente entre los resultados de train y test. Debido a este comportamiento, se procedió a realizar un método de ensamble (bagging) con el objetivo de reducir el overfitting, como se puede ver en la siguiente sección.

IV-B. Redes Neuronales Ensambladas con Bagging

Modelo	Accuracy test	Accuracy train	Loss	Error clase 0	Error clase 1
Red con bagging	0.6778	0.8046	0.7524	0.5156	0.484

Cuadro II
RESULTADOS EN VALIDACIÓN PARA MODELO CON BAGGING

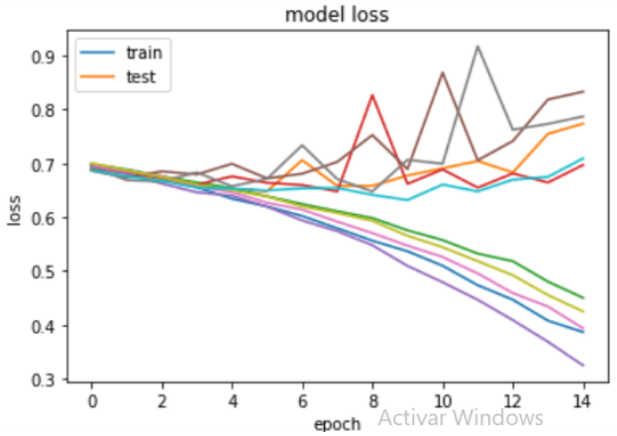

Figura 7. Función de pérdida de la red neuronal con bagging

A pesar de que hubo una mejora del en los resultados, puesto que se logró reducir un poco la diferencia entre los resultados de train y de test y, como se puede evidenciar en la gráfica 6, a el modelo le tomaba más épocas para empezar a presentar overfitting.

V. PLAN DETALLADO

El plan que se siguió para la realización de este proyecto se encuentra en el Cuadro III, donde se pueden ver los objetivos semanales que se deben cumplir cronológicamente hasta la última semana del semestre.

Cuadro III
PLAN DE TRABAJO

2°Actividad	*Semanas*											
	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*
Reunión grupal	■	■	■							■	■	■
Revisión literatura y planteamiento del problema	■	■	■									
Obtención, análisis y limpieza de datos			■	■					■			
Creación y preparación del modelo									■	■	■	■
Entrenamiento y evaluación del modelo									■	■	■	■
Presentación												■

Durante el desarrollo del plan se evidenció que la etapa de obtención, análisis y limpieza de datos es una de las más

importantes. También en muchos casos es la que requiere más tiempo. Por esto, en un trabajo futuro se puede priorizar el tiempo en estas etapas, aunque es bien sabido que este tipo de datos como lo son las imágenes o videos. También que las etapas de modelación, entrenamiento y evaluación, son cíclicas.

VI. Implicaciones éticas

El procesamiento de imágenes ha cobrado mucha fuerza y ha sido aplicado de diversas maneras en ambitos como la medicina, el arte, la vigilancia y muchos otros más, donde claramente ha tenido un impacto positivo para la sociedad. Sin embargo, el crecimiento y desarrollo acelerado de estas técnicas ha llevado a su mal uso. El Deepfake es un una técnica de manipulación de imágenes y videos, que permite intercambiar identidades [6]. Su finalidad va desde parodiar como entretenimiento, hasta hacer ataques dirigidos contra individuos o instituciones [7].

Hoy en día, se vive una crisis global con respecto a la posverdad y a las fake news, debido a la propagación exponencial de información en internet. Este uso mal intencionado del Deepfake presenta una amenaza para la seguridad, tanto de las personas como de los gobiernos, empresas e instituciones. Con estas técnicas de suplantación de identidad se puede llegar a manipular las elecciones de gobierno de un país, falsificando discursos de los candidatos. También es posible afectar la economía, emitiendo falsos mensajes sobre una empresa o sobre el estado de un activo en la bolsa.

Aunque los deepfakes podrían utilizarse de forma positiva, como en el arte y los negocios, existe la posibilidad de que sean usados por grupos insurgentes y organizaciones terroristas, para representar a sus adversarios pronunciando discursos agresivos [8]. Por esto, es urgente desarrollar tecnicas de detección de imagenes falsas o creadas por una inteligencia artificial. Estos modelos para predecir la veracidad de una imagen o un video, tiene más implicaciones éticas además de su utilidad para protejer la integridad de una persona. Es importante que, como cualquier modelo de aprendizaje, no tenga sesgos raciales, étnicos o de género. Es decir, que los grupos mayoritarios y dominantes no sean los únicos beneficiados por mayores índices de precisión [9].

Garantizar modelos que detecten la falsificación de imágenes faciales, sin sesgos discriminatorios, es una necesidad latente frente al panorama de excesiva exposición a la información, a través de los medios audiovisuales.

VII. Aspectos legales y comerciales

Debido al desafío que existe hoy en día con respecto a los Deepfakes, existen grandes oportunidades para este proyecto. Empresas como Meta y Kaggle han creado competencias y retos para el desarrollo de técnicas para la detección de Deepfakes en archivos de video [10]. Debido a que la falsificación de imágenes faciales afecta a individios e instituciones, existe la posibilidad de comercializar el uso de las técnicas de detección, para personas potenciales a ser afectadas. Sin dejar de lado los aspectos jurídicos y políticos que pueden llegar a tener estas detecciones, con respecto, tanto a los afectados como a los emisores.

VIII. CONCLUSIONES

Basado en los resultados y en la bibliografía revisada, concluimos que, para reducir el overfitting y mejorar el acuraccy de las predicciones, se hace necesario aumentar el tamaño de la muestra. Esto demuestra que no es trivial encontrar un balance entre el sesgo y la varianza.

Sin embargo, también se pudo ver que aumentar la complejidad de la red neuronal no implica necesariamente una mejora en los resultados. De hecho esto puede implicar una desmejora, por esto métodos de regularización como el Dropout pueden ser útiles, o de ensamble como el bagging.

Finalmente, se puede ver que las redes neuronales convolucionales son una buena herramienta para realizar clasificacion de imagenes y representar datos no estructurados y en altas dimensiones.

Referencias

[1] Huaxiao Mo, Bolin Chen, and Weiqi Luo. 2018. Fake Faces Identification via Convolutional Neural Network. In IHMMSec '18: 6th ACM Workshop on Information Hiding and Multimedia Security, June 20–22, 2018, Innsbruck, Austria. ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3206004. 3206009

[2] Diana Vanessa Gomez Trejos and Alejandra Guerrero Guzman. Estudio y analisis de técnicas para procesamiento digital de im agenes. 2016.

[3] Keiron O'Shea and Ryan Nash. An introduction to convolutional neural networks. arXiv preprint arXiv:1511.08458, 2015.

[4] Yonsei University Department of Computer Science. Real and fake face detection. https://www.kaggle.com/datasets/ciplab/real-and-fake-face-detection, 2018.

[5] Alexander Reben. 1 million fake faces using a nvidia model. https://archive.org/ details/1mFakeFaces, Apr 2019.

[6] Brian Dolhanskya, Joanna Bittona, Ben Pflaum, Jikuo Lu, Russ Howes, Menglin Wang, and Cristian Canton Ferrer. The deepfake detection challenge dataset, 2020.

[7] Luciano Floridi. Artificial intelligence, deepfakes and a future of ectypes. Philosophy Technology, 31(3):317–321, 2018.

[8] Ashish Jaiman. Debating the ethics of deepfakes. 2020.

[9] Naroa Martinez and Helena Matute. Discriminacion racial en la inteligencia artificial. 2020.

[10] Meta AI. The deepfake detection challenge (dfdc) dataset. 2020.