

RECONOCIMIENTO DE ROSTROS MEDIANTE INTELIGENCIA ARTIFICIAL HACIA LOS DISPOSITIVOS INTELIGENTES

FACE RECOGNITION THROUGH ARTIFICIAL INTELLIGENCE TOWARDS INTELLIGENT DEVICES

Pérez Escamilla Javier, Mendoza Guzmán Lorena, Pérez Bautista Mario

División de Ingeniería en Sistemas Computacionales, Instituto Tecnológico Superior del Occidente del Estado de Hidalgo. Paseo del Agrarismo 2000, Carr. Mixquiahuala - Tula, Km 2.5. Mixquiahuala de Juárez, Hidalgo, C.P.42700, [*javierperez@itsoeh.edu.mx](mailto:javierperez@itsoeh.edu.mx), mendozaq@itsoeh.edu.mx, mperez@itsoeh.edu.mx.

RESUMEN. La democratización de la inteligencia artificial, IA, ha llevado a la aumentación de capacidades humanas. La identificación de rostros en un dispositivo inteligente es eficaz gracias a la clasificación, haciéndolo más rápido y eficiente que un ser humano. En el caso de clasificación multi-clase, un científico de datos, puede aplicar métodos de la IA de aprendizaje profundo. Podrá desarrollar una prueba, por ejemplo, en la plataforma Kaggle.com. Al mismo tiempo, puede mejorar el modelo y revisar las métricas que le permitan obtener los resultados deseados. Tensorflow y Keras son frameworks para la ciencia de datos, donde se infiere el conocimiento al usar redes profundas. Aquí denotaremos el uso de Tensorflow y Keras para reconocimiento de rostros, tomando un set de imágenes provisto por Kaggle. Se realizó en una computadora personal, se introdujo un conjunto de imágenes de una nueva etiqueta, revisando el método y resultados, se concluye en las posibles aplicaciones que un investigador pudiera desarrollar. Se abona en conceptos de “alfabetización” de las tendencias de la IA, para que sirva de referencia a futuros investigadores. Los dispositivos y espacios inteligentes se hacen cada vez más presentes en la vida de los seres humanos, gracias a los avances técnicos en arquitecturas más potentes y al mismo tiempo más reducidas, como es el caso de microcomputadoras. Las microcomputadoras o tarjetas de desarrollo permiten ser un medio disruptivo en la creación de artefactos de utilidad. El uso de técnicas de la IA, en ellas, presenta retos como la transferencia de modelos y el tiempo de procesamiento. La democratización, trata de hacer disponibles los métodos de la IA, haciendo fácil el acceso y el uso, así un ciudadano es capaz de desarrollar su propio dispositivo inteligente. Un ejemplo de la democratización es el aprendizaje profundo, siendo este un avance técnico en la IA denotado como “IA mejorada”. Una aplicación para el reconocimiento de rostros puede usar internamente una técnica de IA, que permita a un dispositivo o computadora clasificar o etiquetar, basado en un conjunto de imágenes de entrenamiento, la salida del proceso es una etiqueta o una probabilidad de pertenencia.

Palabras clave: Redes profundas, alfabetización, aumentación humana.

ABSTRACT. The democratization of artificial intelligence, AI, has led to the augmentation of human capabilities. Face identification on a smart device is effective thanks to classification, making it faster and more efficient than a human. In the case of multi-class classification, a data scientist can apply deep learning AI methods. You can develop a test, for example, on the Kaggle.com platform. At the same time, you can improve the model and review the metrics that allow you to obtain the desired results. Tensorflow and Keras are frameworks for data science, where knowledge is inferred using deep networks. Here we will denote the use of Tensorflow and Keras for face recognition, taking a set of images provided by Kaggle. It was carried out on a personal computer, a set of images of a new label was introduced, reviewing the method and results, it is concluded in the possible applications that a researcher could develop. It is credited to “literacy” concepts of AI trends, to serve as a reference for future researchers. Smart devices and spaces are becoming more and more present in the lives of human beings, thanks to technical advances in more powerful and at the same time smaller architectures, such as microcomputers. Microcomputers or development cards allow to be a disruptive medium in the creation of useful artifacts. The use of AI techniques, in them, presents challenges such as model transfer and processing time. Democratization tries to make AI methods available, making it easy to access and use, so that a citizen is able to develop their own smart device. An example of democratization is deep learning, this being a technical advance in AI denoted as “enhanced AI”. An application for face recognition can internally use an AI technique, which allows a device or computer to classify or label, based on a set of training images, the output of the process is a label or a probability of belonging

Key words: Deep learning, artificial Intelligence literacy, human augmentation.

INTRODUCCIÓN

Los algoritmos son parte de la vida cotidiana, es de una forma tan natural que casi pasan desapercibidos para la mayoría de la gente, sin embargo en el ámbito computacional son parte fundamental del uso de la máquina, es el fundamento de un programa de cómputo¹.

Los elementos llamados Tipos de Datos Abstractos (ADT, Abstract Data Type), son la estructura de

información mínima que tiene el algoritmo para guardar la información, esta es la base en la cual se realiza la implementación del algoritmo².

“En todos los casos que sea posible, el descubrimiento de algoritmos es la meta natural de las matemáticas”³

Académico A.N. Kolmogorov

Las operaciones de un algoritmo tienen una secuencia ordenada que tienen el fin de resolver un problema muy preciso, además este debe tener las siguientes propiedades⁴:

- Finitud.
- Ausencia de ambigüedad.
- Secuencia.
- Entrada y salida.
- Efectividad.
- Definición de alcance.

Inteligencia Artificial

La IA nace naturalmente de describir fenómenos, en el caso más simple, a los autómatas. El uso del estatuas como la de “Talos protector de la isla de Creta”⁵, el uso del reloj solar de sechat⁶ 1200 A.C y calendario maya, que data 3114 A.C⁷, son elementos que por su propia naturaleza reflejan aspectos de la evolución del pensamiento del ser humano y la automatización. Podría decirse que un autómata es un ente que tiene cuerpo y es capaz de realizar tareas humanas, entonces es una máquina inteligente, que tiene dos características: cuerpo y mente. Descartes, 1863, describió el tratado del dualismo cuerpo y mente, donde dice que un autómata es que realiza acciones humanas, entonces aplica tiene una inteligencia artificial⁸. No olvidar que la definición IA de encontrar la mejor solución a un problema, también es parte del concepto de la inteligencia artificial, e.d. Juego de Torres de Hanói⁹. Se podría decir que los primeros pasos en las matemáticas lo dieron Gerge Boole, con el álgebra booleana, Gottlob Frege con la lógica de primer orden y Alan Turing, que teorizaba una máquina de estados discretos digital¹⁰. Más avances se dan a lo largo del tiempo.

La cognición

Se toma primero la investigación en las emociones humanas, en el campo de la psicología, en esta se descubren muchos aspectos del proceso de las emociones humanas. En este caso tomamos la teoría lázaro, según Richard Lazarus, el pensamiento debe ocurrir antes que la experiencia de la emoción. Las aplicaciones de este tipo de teoría, centran claramente la atención en el espacio individual, pero son temas de investigación en diversas épocas¹¹.

Perceptrón y la Red Neuronal Artificial

En 1967 Marvin Minzky publica su obra “Perceptrones”, en que describe el comportamiento de una neurona, que se podría definir como un motor de inferencia que identifica las características de un

objeto, para sí poder clasificar su clase, Este elemento genera un modelo en un hiper-plano y para ello aplica un conjunto de algoritmos¹². Las redes neuronales artificiales responden a una imitación del ser humano de procesamiento paralelo, a diferencia de que las máquinas son secuenciales, estas redes adquieren conocimiento basado en la experiencia, son adaptables, tolerantes a fallas y tienen un comportamiento no lineal¹³.

Inteligencia artificial moderna

La tecnología actual, se centra en aumentar las capacidades humanas, así entonces se plantean dos categorías que agrupan las tendencias y aspectos de las IA: centrado en las personas y espacios inteligentes, ahí se denotan las siguientes categorías¹⁴:

1. Centrado en las personas

- Híper-automatización: Todo lo que se puede automatizar, se va a automatizar, ed. El uso de los Sistemas Expertos.
- Multi-Experiencia. El mundo es nuestra computadora, ed. Vehículos autónomos.
- Democratización: La apertura de los modelos y técnicas ayudan a mejorar y extender la IA, ed. Robo Advisors.
- Aumentación humana: Lo dispositivos superan y aumenta la capacidad humana, ed. Agricultura de presión mediante cámaras espectrales montadas en drones.
- Transparencia y trazabilidad, lo modelos de las IA, deben ser rastreables, debugables y éticos. Esta es una nueva tarea para los arquitectos de software¹⁵.

2. Centrado en los espacios.

- Empoderamiento de la frontera: Los dispositivos se dotan de IA para crear espacios inteligentes.
- Nubes de computo distribuidas: Sí se puede procesar en la frontera, microcomputadoras es más viable que generar más tráfico en la red, ed. El de Cómputo en la niebla es aquel procesamiento de la información que no hecho en la nube.
- Dispositivos autónomos. Los dispositivos son colaborativos entre ellos, ed. brazos robóticos colaborativos¹⁶.
- Cadenas de bloques prácticas: Las empresas integran los aspectos del negocio en un modelo computacional.
- Seguridad en IA: Los modelos y los set de entrenamiento se deben resguardar, se debe pensar en nuevas formas en una IA puede ser

vulnerada y ser capaz de detectar nuevos ataques.

La tecnología de la inteligencia artificial está mejorando otros sectores, como la aviación, la hospitalidad, la atención médica, los servicios financieros, el comercio y la industria manufacturera entre otros.

Proceso de minería de datos

Para extraer información se aplica un modelo de proceso de siete tareas, que son¹⁷:

1. Entendimiento del dominio.
2. Selección y adición de diferentes fuentes de datos.
3. Pre-procesamiento y limpieza de los datos.
4. Transformación de los datos.
5. Minería de datos, aplicando una técnica de IA.
6. Evaluación e interpretación.
7. Visualización e integración.

Reconocimiento de Rostros

El reconocimiento facial (RF) se encuentra en un punto crítico de su desarrollo, la tecnología avanza y mejora rápidamente¹⁸. La adopción de herramientas de RF como un activo de aplicación de la ley está creciendo. Ahora es comúnmente accesible para las agencias de policía. Y más allá de sus de aplicaciones originales cae también en el ejército, la lucha contra el terrorismo y la aplicación de la ley¹⁹.

Las características de rostros las podemos calcular mediante la distancia euclidiana, es decir una medida de la referencia de distancia, así podemos orientarnos y saber dónde se encuentra la nariz, quiere decir, que hay es una representación que la computadora es capaz de entender, así entonces se puede reconstruir a partir de formas simples²⁰.

Red Neural Profunda

Las redes neuronales profundas son una técnica de reconocimiento de patrones, aplicadas en el aprendizaje supervisado y el aprendizaje por refuerzo. Aplica un conjunto de redes internas interconectadas, llamadas capas ocultas, así las entradas $E_1, E_2, E_n \dots$ son mapeadas a salidas a $S_1, S_2 \dots S_n$, así según las activaciones, pasarán a otras capas ocultas, aquí la importancia radica en los pesos del modelo, o mejor dicho que funciones de activación son más importantes para extraer las

características²¹. Existen varios modelos de Redes Neuronales Profundas:

- Red Neural Convolutiva o Convolutional Neural Network, CNN.
- Redes Generativas Adversas o Generative Adversarial Network, GAN.
- Autocodificadores o Redes Boltzman, RBN.

Arquitecturas de Redes Neuronales Profundas

La red de Incepción de Google se consideró una vez una arquitectura (o modelo) de aprendizaje profundo de última generación para resolver problemas de detección y reconocimiento de imágenes, sin embargo, presenta la desventaja de que requiere de gran cantidad de equipo para poder entrenar ese tipo de red neural, lo cual a menudo puede crear complicaciones:

Cuanto más grande es el modelo, más propenso es sobre-ajustarse²³. Esto es particularmente notable cuando los datos de entrenamiento son pequeños. Aumentar la cantidad de parámetros significa que necesita aumentar sus recursos computacionales existentes. Los nuevos avances, pueden indicar el uso de modelos de inferencia de Neuro-fuzzy para mejorar el desempeño²⁴.

Tensorflow y Keras

El uso de tensores en el cálculo matricial, hace eficiente el uso de las redes profundas. Tensorflow y Keras implementan un conjunto de herramientas matriciales y modelos ya entrenados para el rápido desarrollo en la ciencia de datos, algunos de esto modelos son: LeNet, para el reconocimiento de caracteres, AlexNet para el reconocimiento de objetos, ResNet que incremento la complejidad y profundidad de AlexNet²⁶.

Los elementos requeridos para trabajar en el contexto de un ejemplo de clasificación con Tensorflow y Keras son los siguientes:

- Conjunto de imágenes de entrenamiento: contienen la característica de lo que se desea clasificar.
- Conjunto de imágenes de test: son imágenes que sirven como referencia para probar la validez del modelo.
- Conjunto de prueba: son imágenes que son usadas para probar el modelo, más allá de del test.

Pre procesamiento: Es el proceso en el que se debe hacer que la entrada sea adecuada para el proceso, en este caso convertir a vectores²⁷.

Modelo: Es la representación de las capas de entrada, salida y ocultas en un red funcional. Existen dos tipos: secuencial que es un orden secuencial del modelo; funcional que permite interconectar varios modelos²⁸.

Capa convolucional: Realizan operaciones sobre la entrada mediante el cálculo de convoluciones. Ya describimos lo que es la convolución anteriormente²⁹.

$$y_{rcl} = \sum_{i=1}^r \sum_{j=1}^c F_{cyr+1-1c+j-1l-1} w_{ijl} + b_l$$

Donde y_{rcl} es el volumen de salida en $\{r,c\}$, F_r , F_c son el número de columnas y renglones el filtro de paso, w_{ijl} es el valor del filtro en la posición $\{i,j\}$, $y_{r+1-1c+j-1l-1}$ representa el valor de la entrada a esta capa en la posición $\{r+i-1,c+j-1\}$, b_l es valor de bias.

Ya que la capa neuronal representa el procesamiento de una entrada, provista por el filtro, estos serán entonces los parámetros de entrenamiento.

Capas densas: Representan las neuronas clásicas en los perceptrones. Su función principal es la regresión lineal o la clasificación. Aquí resaltaremos que la regresión lineal es una aproximación numérica de una serie de elementos en un hiper-plano y la clasificación es el grado de pertenencia a una categoría³⁰.

Función de activación; Aportan la no linealidad de las funciones y de agregar activaciones de múltiples capas en la salida de la red. Las hay lineales, sigmoidales, función Softmax, Unidad Lineal Rectificada (RELU), tangente hiperbólica. Todas ellas son funciones matemáticas que no trataremos aquí³¹. Softmax y RELU son de interés.

- Softmax: $f_{ji} = \frac{e^{z_{ji}}}{\sum_k e^{z_{kj}}}$
- RELU: $f(x) = \max(0, x)$

Capa de Pooling: Capa para reducir el volumen de salida de las capas convolucionales. Así el perceptrón incrementa el campo de percepción de la red, así entonces toma características viables para el modelo³².

$$\text{MaxPooling: } h_{xyl} = \max_{i=0..s, j=0..shl-1x+i(y+j)}$$

Dropout: Es la capa que permite evitar el sobre entrenamiento. Elimina las contribuciones de algunas

neuronas junto a las conexiones de entrada y salida, basado en una probabilidad aleatoria³³.

Batch normalization: Aumenta la estabilidad del entrenamiento, normalizando las salidas de las capas de activación anterior. Aplica las operaciones restando la media provista y dividiendo por la desviación típica³⁴.

$$\text{Batch Normalization} = (z_i) = y_i z_i + y_i, z_i = z_i - E[z_i], \text{Var}[z_i]$$

Épocas: se puede decir que es un ciclo de entrenamiento, donde se realizan todas las operaciones del modelo, además estos se guardan en “pesos”, que son las funciones de activación que ofrecen mayor ganancia.

Batch Size: Refiere a un conjunto de imágenes, en las que una época de entrenamiento procesa la entrada, según el número provisto.

Matriz de confusión: en la clasificación, refiere a las instancias correctamente clasificadas y no correctamente clasificadas. TP o Verdaderos positivos, FP falsos positivos, TN o Verdaderos Negativos, FN o falsos negativos, las métricas asociadas son *Accuracy*, *Recall* y *Precision*³⁵. Asociados con la exactitud, el recuerdo y la precisión:

- $\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$, mide lo acertado del modelo.
- $\text{Recall} = \frac{TP}{TP+FN}$, mide la capacidad del modelo de discriminar entre las clases.
- $\text{Precision} = \frac{TP}{TP+FP}$, mide la predicción correcta de instancias.

Problema a tratar:

La democratización de la IA, en muchos casos, obtener el conjunto de entrenamiento y hacer fácil su uso. En este caso usaremos el conjunto de imágenes llamado “5 Celebrity Faces DataSet”³⁶.

En este dataset intercambiaremos una folder para agregar, una con una nueva etiqueta, aplicaremos una transformación y pondremos, así entonces generaremos el modelo y verificaremos resultados. Los argumentos del modelo se encuentran en la Tabla 2.

METODOLOGÍA

Aplicaremos la técnica de aprendizaje profundo y la clasificación supervisada, en una CNN con la siguiente arquitectura. Además refiero a la configuración de la máquina donde se realizó el proceso, se utilizó software de código abierto y es sólo para propósitos académicos. Los elementos son Python 3.7, en una pc con 16GB Ram, Ryzen 5 con 12 núcleos y un disco de estado sólido de 250GB, en una placa B450-M1, una tarjeta de vídeo gráfica 730, fuente de poder 800Wats. Monitor 27". Para ellos se aplicaron las etapas de la Tabla 1.

Pre procesamiento de la entrada

Del conjunto de imágenes de entrenamiento, se ha compuesto imágenes con extensiones JPG, PNG. Estos elementos son cargados al sistema convirtiendo cada uno de ellos a un numpy array, usando la librería Pillow y aplicado un modo RGB mediante Image.fromarray, en este caso, convirtiendo la imagen en un ítem de 3x8 bits de color verdadero sin mascara de transparencia. Una vez obtenido el producto, se redimensiona a un tamaño estandarizado de 150x150 pixeles.

Tabla 1. Etapas

- 1.- Entendimiento del problema
- 2.- Planteamiento del problema
- 3.- Adquisición de datos. (Imágenes)
- 4.- Implementación de un modelo
- 5.- Entrenamiento
- 6.- Validación

Etapas del proceso.

Clases

Al pre procesar los elementos del conjunto de imágenes de entrenamiento, se han obtenido las clases tomando la siguiente forma:

- Labels in SET : 144
 - 0 ben_afflek
 - 1 elton_john
 - 2 javier
 - 3 madonna
 - 4 mindy_kaling
- Images and Labels
 - <class 'list'> <class 'list'>
 - X type: (144, 150, 150, 3)
- Total of classes: 5
- Labels of the classes: [0 1 2 3 4]

Conjunto de entrenamiento y test

Utilizando la librería Sklearn, se ha construido un conjunto de entrenamiento y un conjunto de prueba, donde se ha utilizado el 20% del conjunto total para el test.

- Conjunto de entrenamiento: (115, 150, 150, 3) (92,)
- Conjunto de test : (29, 150, 150, 3) (29,)

(5)

Para el conjunto de validación, se separa un 20% de conjunto de entrenamiento quedando de la siguiente manera.

- Conjunto de entrenamiento: (92, 150, 150, 3) (92,)
- Conjunto de validación: (23, 150, 150, 3) (29,)

Donde en ejemplo 92 representa el número de instancias, 150, 150, representa el tamaño de la imagen en pixeles y 3, representa los canales de la imagen, en este caso RGB.

Diagrama de proceso

El cambio dimensional de los parámetros, cada etapa de convolución representa un cambio en la dimensión de los parámetros de entrada y salida de cada parte del proceso, por ellos es necesario visualizar estos cambios Figura 1. La capa "l" que representa una convolución podrá afectarse en función del filtro, padding y strides:

- Tamaño de filtro= f[l] -> entrada= nh[l-1]nw[l-1]nc[l-1].
- Padding= p[l] -> cada filtro= f[l]×f[l]×nc[l-1]×nc[l].
- Stride =s[l] -> pesos = f[l] ×f[l]×nc[l-1]×nc[l].
- Salida -> nh[l]×nw[l]×nc[l].
- Bias= 1×1×1×nc[l].

Entonces la dimensión final de una capa está dada por:

$$n [l]=n l-1+2p [l]-f [l] s [l] +1.$$

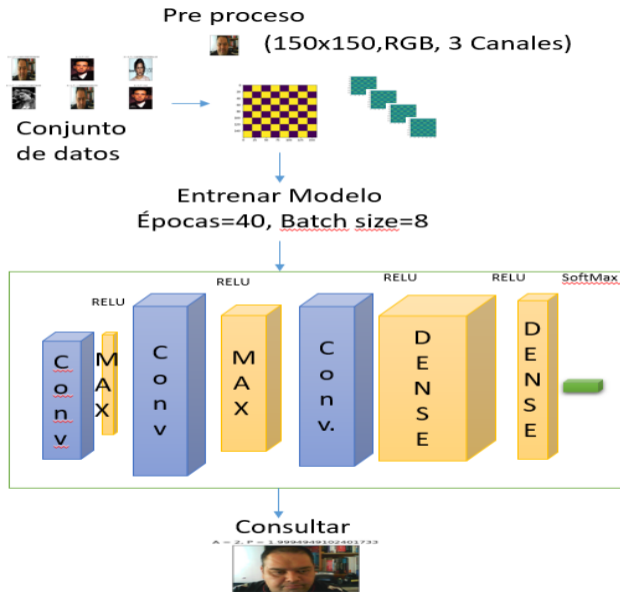


Figura 1.- Diagrama del proceso.

La CNN se entrenó con 40 épocas (iteraciones) y en cortes de 8 imágenes de las 115 totales. El set de datos y los resultados están disponibles en <https://github.com/javierperez83/art-2019-data-science>. El modelo se infiere acorde a la Tabla 2.

Tabla 2. Modelo secuencial.

Capa	Entrada/Forma Vector	Parámetros
Conv2D	(None, 150, 150, 64)	1792
MaxPooling2	(None, 75, 75, 64)	0
Conv2D	(None, 75, 75, 64)	36928
MaxPooling2	(None, 37, 37, 64)	0
Conv2D	(None, 37, 37, 64)	36928
Flatten	(None, 87616)	0
Dense	None, 256	22429952
Dense	(None, 5)	1285
Total de parámetros: 22,506,885		
Parámetros entrenables: 22,506,885		
Parámetros no entrenables: 0		

Modelo desarrollado con Tensorflow y Keras.

Las imágenes mostradas, son sólo con propósitos educativos, las clases quedan tendrán las siguientes etiquetas, notar que es el nombre del folder al que se hace referencia; Ben Afflek, clase 0; Elton John, clase 1; Javier, clase 2; Madonna, clase 3; Mindy Kaling clase 4.

El modelo seleccionado es arbitrario y se basa en una interpretación propia.

Se configuran los siguientes parámetros para el entrenamiento.

```
cnn.compile(loss=keras.losses.categorical_crossentropy, optimizer = 'adam', metrics=['categorical_accuracy'])
```

Función de pérdida “Categorical Cross Entropy”
La entropía cruzada categórica es una función de pérdida que se utiliza en tareas de clasificación multiclase. Aquí determinamos las características que permiten al modelo seleccionar una única clase de pertenencia, de entre muchas posibles, es decir, cuantifica la diferencia entre dos distribuciones de probabilidad.

$$CCE(p,t) = -c \sum_{i=1}^n p_i \log_{10}(p_i/c_i)$$

Optimizador Adam

Este algoritmo, permite optimizar el gradiente descendiente cuadrático, SGD, adaptando el promedio de aprendizaje en función de los momentos. Mantiene el foco el promedio de los valores anteriores del SGD.

- $mt = 1mt_{-1} + 1 - 1gt$
- $vt = 2vt_{-1} + 1 - 2gt^2$
- mt es el primer momento media del gradientes
- vt es el segundo momento (variancia descentrada) del gradiente
- α y β_2 son valores de decaimiento
- mt y vt son inicializado como 0-vectores, así entonces, el vías tiende a 0.
- Computar la corrección del primer bias y calcular el segundo momento:
 - $mt = mt_{-1} - 1t$
 - $vt = vt_{-1} - 2t$
- Actualización de la regla de Adam
 - $\theta_{t+1} = \theta_t - \eta vt + \epsilon mt$

Ajuste del modelo,

```
fitter = cnn.fit(xt, yt, epochs=40, verbose=1, batch_size=8, validation_data=(valid_X, valid_label))
```

RESULTADOS Y DISCUSIÓN

Aún que no podemos interpretar los datos, nos podemos dar una idea de lo que ocurre con la gráfica de la Figura 2. La línea roja representa la precisión del modelo al predecir nuevas clases con un conjunto de imágenes de prueba, en esta vemos que el modelo empieza a mejorar, pero después ya no mejora significativamente, la línea verde representa la precisión de todas las clases, pero hay que tomar en cuenta que se agregó una nueva clase, la línea azul representa la pérdida del modelo, es decir como se ha equivocado el modelo al predecir, se busca que tienda a cero. Se debe tomar en cuenta que se cambió una clase por una instancia nueva.

A continuación, se presenta la matriz que permite validar los resultados.

Clase 0 24 0 1 4

Clase 1 24 2 2 1

Clase 2 16 0 1 12

Clase 3 24 2 1 2

Clase 4 20 0 3 6

El total de instancias es 29, que es la suma de todos los elementos de la matriz, la diagonal inversa, es son los elementos falsamente clasificados, la suma del primer renglón representa las instancias que no corresponden a la clase y el segundo, las clases que estamos revisando.

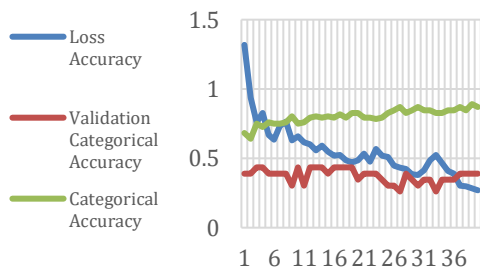


Figura 2. Categorical Accuracy, Validation Categorical Accuracy, Loss Accuracy Vs Epochs (40)

Sólo por observación, se denota que la clase 1 y 3 tienen errores de clasificación, por el contrario la clase 1, 2 y 4, tienen mejores predicciones.

Precisión total del modelo 87%. Esto indica que hay elementos que no son clasificados correctamente. Claramente tiene que ver con que los datos no están balanceados. Este desbalance, provoca errores en la clasificación.

En la evidencia de la clasificación mostrada en la Figura 3 y Figura 4, se observa que sí identifica correctamente instancias del conjunto de imágenes, o en su caso se aproxima bastante. Donde “A” representa la clase a la que pertenece y “P” la probabilidad calculada.

A = 0, P = 0.0



Figura 3. Instancia clasificada correctamente.

A = 2, P = 1.9994949102401733



Figura 4. Instancia clasificada correctamente, pues se acerca más a la clase 2 que a la clase 1.

Sin embargo en las instancias no clasificadas correctamente, es dónde se denota que características hay involucradas, mostradas en la Figura 5. Donde la clase a la que pertenece es 2 y la probabilidad es más cercana a la clase 0.

A = 2, P = 0.11264414340257645



Figura 5. Instancia clasificada correctamente, pues se acerca más a la clase 2 que a la clase 1.

Tomando como referencia una entrada de la clase 1, Figura 6, entonces, podemos denotar que aspectos son similares en la extracción de características.



Figura 6. Instancia de clase 1.

Ya aquí podemos decir, que los perceptores se activan aspectos como los ojos y sus elementos como la boca.

Aquí es donde radica la importancia de la democratización de la IA, la revisión de los pares académicos abona al mejoramiento, dando transparencia, legibilidad y trazabilidad en los cambios, aunque en ocasiones no sea claro el aporte. Incluir los aspectos de la configuración de la máquina y el software ocupado, ofrece la posibilidad de la replicación y así mismo la validación del mismo. La verificación, va más allá del contexto del prototipo planteado y una nueva habilidad para los científicos de datos. Estos tendrán nuevas responsabilidades en la construcción de modelos de IA, que se asocian a un rol de arquitectura de software. Aún más, empoderar las microcomputadoras, para hacerlos dispositivos inteligentes es una nueva ciencia que surge.

Quizá se esté preguntado ¿qué nuevos dispositivos se pueden crear desde este punto?,

¿Cómo puedo transferir el modelo a un dispositivo de la frontera?

La respuesta a la primera pregunta, no tiene límites y va referenciada a las tendencias de IA. La segunda pregunta, tiene una respuesta más técnica.

Dado que la complejidad de lo que la IA se propone resolver la solución entonces, tendrá conjuntos de entrenamiento con más instancias, requieren entonces un tiempo de entrenamiento más largo. La solución radica en el paralelismo, su propósito es reducir el tiempo de entrenamiento.

Transferir el conocimiento a un dispositivo requiere transferir el modelo y los pesos de activación de la CNN. En nuestro caso 386Kb de modelo y 263.8MB de las activaciones, de un set de entrenamiento de 5.77MB con 51 imágenes de aproximadamente 150x150 píxeles. Mejorar el modelo y la precisión es labor de un científico de datos.

En la Tabla 3 se muestran la comparativa de las arquitecturas usadas en el conjunto original. Se debe destacar, que sólo aplican el dataset original y no introducen nuevas clases e instancias.

Tabla 3. Comparativa de clases.

URL	Arquitectura	Accuracia
https://www.kaggle.com/yhuan95/face-recognition-with-facenet	Face Net	100%
https://www.kaggle.com/iljoong/celebrity-face-classification-using-keras	VGG16	100%
NA	Modelo propuesto	87%

Comparativa de métodos, el método propuesto es un conjunto no balanceado.

CONCLUSIONES

La verificación, surge como una habilidad de todo científico de datos, apegado a sus normas éticas y su juicio ingenieril. Las habilidades de un ingeniero de datos, científico de datos o afín van desde los conceptos de algoritmos hasta el paralelismo, sin olvidar la representación gráfica. Las técnicas de IA deberán ser revisadas y mejoradas. Así mismo empoderar la frontera de los dispositivos, eventualmente, fijará nuevas arquitecturas y capacidades, pero también la habilidad de interactuar entre ellos y con los elementos existentes.

La democratización de la IA, se ha visto beneficiada con el apoyo de los aficionados a las tarjetas de desarrollo, donde se exponen nuevas habilidades y formas de resolver problemas. Así entonces un científico de datos debe ser un innovador en implementación, para así poner en práctica los conocimientos y buscar que la IA realmente abone a hacer super-humanos, o mejor dicho a humanos con habilidades aumentadas para el beneficio de la humanidad y del mundo en el que vivimos.

La frontera es difusa, pero la meta es clara. Un nuevo nivel de seguridad se podrá afrontar con el desarrollo de aplicaciones embebidas en tarjetas de desarrollo como Raspberry PI u otros,

Un dispositivo que integre elementos del ecosistema de Raspberry, podrá colocar nuevos retos en la personalización de este dispositivo, extendiendo su uso y forzando a un mejoramiento del mismo. Así entonces, nacen dispositivos inteligentes de seguridad.

AGRADECIMIENTOS Y/O RECONOCIMIENTOS

Agradezco al Instituto Tecnológicos Superior del Occidente del Estado de Hidalgo, ITSOEH, por brindarme la oportunidad de realizar Investigación y el apoyo para publicar el presente artículo en el congreso, así como a mis compañeros colaboradores por su apoyo y compromiso con este proyecto.

REFERENCIAS

1. Algoritmo. (2020, 22 de septiembre). Wikipedia, La enciclopedia libre. Fecha de consulta: 23:10, septiembre 27, 2020 desde <https://es.wikipedia.org/wiki/Algoritmo>.
2. Weiss, M.A.: Data Structures and Algorithm Analysis in C++, 4th Edition, Pearson/Addison Wesley, 2014.
3. Landa, L. N. (1974). Algorithmization in learning and instruction. Educational Technology. Pg.16.
4. Turing, A. M. (1950). Computer machinery and intelligence. Mind, 59(236), 433-460.
5. Talos. (2020, 8 de agosto). Wikipedia, La enciclopedia libre. Fecha de consulta: 02:17, septiembre 28, 2020 desde <https://es.wikipedia.org/w/index.php?title=Talos&oldid=128334029>.
6. Sechat. (2019, 24 de julio). Wikipedia, La enciclopedia libre. Fecha de consulta: 02:34, septiembre 28, 2020 desde <https://es.wikipedia.org/w/index.php?title=Sechat&oldid=117668631>.
7. Smithsonian Museo Nacional del Indígena Americano, El sistema de calendario, consultado el septiembre 27-2020 desde <https://maya.nmai.si.edu/es/calendario/el-sistema-calendario>
8. Torres de Hanói. (2020, 23 de junio). Wikipedia, La enciclopedia libre. Fecha de consulta: 01:50, septiembre 28, 2020 desde https://es.wikipedia.org/w/index.php?title=Torres_de_Han%C3%B3i&oldid=127173119.
9. Problema mente-cuerpo. (2020, 21 de septiembre). Wikipedia, La enciclopedia libre. Fecha de consulta: 01:45,

- septiembre 28, 2020 desde https://es.wikipedia.org/w/index.php?title=Problema_mente-cuerpo&oldid=129450127.
10. Russell, S. J., & Norvig, P. (2004). Inteligencia Artificial: un enfoque moderno (No. 04; Q335, R8y 2004.). Pg. 9-10
11. Lazarus, R. S., Kanner, A. D., & Folkman, S. (1980). Emotions: A cognitive-phenomenological analysis. In Theories of emotion (pp. 189-217). Academic Press.
12. Stephen, I. (1990). Perceptron-based learning algorithms. IEEE Transactions on neural networks, 50(2), 179.
13. Izaurieta, F., & Saavedra, C. (2000). Redes neuronales artificiales. Departamento de Física, Universidad de Concepción Chile.
14. Paneta K. (2020, Octubre 21). Gartner Top 10 Strategic Technology Trends for 2020, reference <https://www.gartner.com/smarterwithgartner/gartner-top-10-strategic-technology-trends-for-2019/>
15. Galvan P. (2018, Mayo). Impacto de la Inteligencia Artificial en el Desarrollo de Software, Software Gurú, recuperado de <https://sg.com.mx/revista/56/inteligencia-artificial-desarrollo-software>
16. EDS Robotisc (2020, Junio 3). Brazo robótico vs colaborativo, recuperado de <https://www.edsrobotics.com/blog/brazo-robotico/>
17. Descubrimiento del Conocimiento (KDD): "El Proceso de minería", recuperado de <http://mineriadatos1.blogspot.com/2013/06/descubrimiento-del-conocimiento-kdd-el.html>
18. Andersen B. (2015, noviembre 18), Microsoft Publishes API for Emotion Recognition, recuperado de <https://codingexplained.com/news/microsoft-publishes-api-emotion-recognition>
19. Cohen, A., Shen, X., Torrellas, J., Tuck, J., Zhou, Y., Adve, S.,... & Beck, M. (2018). Interdisciplinary research challenges in computer systems for the 2020s. USA, Tech. Rep.
20. Feng, R., & Prabhakaran, B. (2013, October). Facilitating fashion camouflage art. In Proceedings of the 21st ACM international conference on Multimedia (pp. 793-802).
21. Código Fuente (2019, Abril 25), Redes neuronales profundas – Tipos y Características. <https://www.codigofuente.org/redes-neuronales-profundas-tipos-caracteristicas/>
22. Faizan Shaik (Octubre 2018), Deep Learning in the Trenches: Understanding Inception Network from Scratch, recover from <https://www.analyticsvidhya.com/blog/2018/10/understanding-inception-network-from-scratch/>.
23. Pal, S. K., & Mitra, S. (1999). Neuro-fuzzy pattern recognition: methods in soft computing. John Wiley & Sons, Inc
24. Ramsundar, B., & Zadeh, R. B. (2018). TensorFlow for deep learning: from linear regression to reinforcement learning. O'Reilly Media, Inc. Pg. 6-7.
25. Brownlee J. (December 2019) How to prepare your data for machine learning in python with Scikit-Learn. *Machine Learning Mastery*, recover from <https://machinelearningmastery.com/prepare-data-machine-learning-python-scikit-learn/>.
26. Keras API reference / Models API (2020), recuperado de <https://keras.io/api/models/>
27. Redes neuronales convolucionales. (2020, 25 de agosto). Wikipedia, La enciclopedia libre. Fecha de consulta: 00:55, octubre 18, 2020 desde https://es.wikipedia.org/w/index.php?title=Redes_neuronales_convolucionales&oldid=128749601.
28. Gupta, Anindya & Harrison, Philip & Wieslander, Håkan & Pielawski, Nicolas & Kartasalo, Kimmo & Partel, Gabriele &

- Solorzano, Leslie & Suveer, A. & Klemm, Anna & Spjuth, Ola & Sintorn, Ida-Maria & Wählby, Carolina. (2018). Deep Learning in Image Cytometry: A Review. *Cytometry Part A*. 95. 10.1002/cyto.a.23701.
29. 7 Types of Neural Network Activation Functions: How to Choose?. Recuperado de <https://missinglink.ai/guides/neural-network-concepts/7-types-neural-network-activation-functions-right/>.
 30. Brownlee J. (December 2019) A Gentle Introduction to Pooling Layers for Convolutional Neural Networks, *Machine Learning Mastery*, recover from <https://machinelearningmastery.com/pooling-layers-for-convolutional-neural-networks/>.
 31. Jansma H. (2018) Don't Use Dropout in Convolutional Networks, recover from <https://www.kdnuggets.com/2018/09/dropout-convolutional-networks.html>.
 32. Wikipedia contributors. (2020, September 4). Batch normalization. In *Wikipedia, The Free Encyclopedia*. Retrieved 01:22, October 18, 2020, from https://en.wikipedia.org/w/index.php?title=Batch_normalization&oldid=976700850.
 33. Matriz de confusión. (2020, 12 de julio). Wikipedia, La enciclopedia libre. Fecha de consulta: 01:24, octubre 18, 2020 desde https://es.wikipedia.org/w/index.php?title=Matriz_de_confusi%C3%B3n&oldid=127660930.
 34. DabB (2017). 5 celebrity faces dataset, recuperado de <https://www.kaggle.com/dansbecker/5-celebrity-faces-dataset>.