Identificación de Imágenes Auténticas y Sintéticas : Abordando los Desafíos de las Imágenes Sintéticas en la Sociedad Actual

Deciré Jaimes - 2211835
Isidro Herrera - 2210088

JUSTIFICACIÓN

La generación de imágenes mediante IA plantea desafíos éticos y sociales, como la desinformación, difamación y acoso. Este proyecto busca desarrollar un sistema que diferencie entre imágenes reales y sintéticas, ayudando proteger la integridad de las personas en un entorno digital más seguro.

JUSTIFICACIÓN

OBJETIVOS y ALCANCE

Desarrollar modelo	Optimizar tasa de error	Evaluar Desempeño de Arquitecturas
Construir y validar un modelo de aprendizaje profundo que logre una precisión mínima del 75% en la clasificación entre imágenes reales y generadas por IA.	Reducir los falsos positivos y falsos negativos a una tasa combinada inferior al 10%, evaluada en un conjunto de prueba representativo.	Implementar y comparar el rendimiento de diferentes arquitecturas (como Xception y ResNet). Seleccionar la arquitectura que muestre al menos un 85% de precisión y recall.

IDENTIFICACIÓN Y DESCRIPCIÓN DE LOS DATASET

REALES

5.330

9.630

Human Faces Dataset

.10.919

SINTETICAS

5.589

IDENTIFICACIÓN Y DESCRIPCIÓN DE LOS DATASET

REALES

5.330

1.288

Fake-Vs-Real-Faces (Hard)

10.919

SINTETICAS

5.589

IDENTIFICACIÓN Y DESCRIPCIÓN DE LOS DATASET

IMAGENES REALES
5.330

EDAD

GENERO

POSICIONES

RAZA

IDENTIFICACIÓN Y DESCRIPCIÓN DE LOS DATASET

IMAGENES SINTÉTICAS
5.589

EDAD

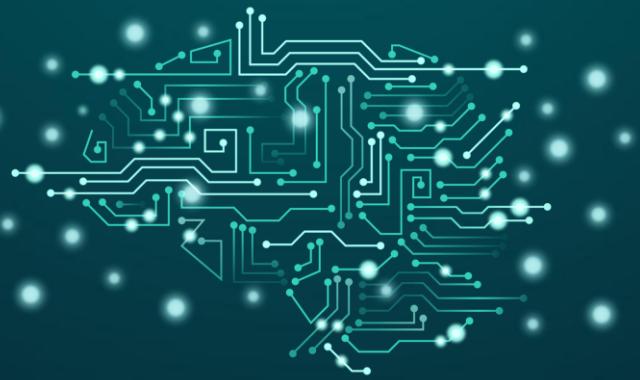
GENERO

POSICIONES

RAZA

REVISIÓN DEL ESTADO DEL ARTE

En los últimos años, las GANs han permitido la creación de imágenes extremadamente realistas, complicando la detección de falsificaciones. Investigaciones clave han empleado CNNs para analizar texturas y artefactos visuales, logrando excelentes resultados con arquitecturas como EfficientNet y Xception en la clasificación de imágenes sintéticas.



REVISIÓN DEL ESTADO DEL ARTE

Principales Avances

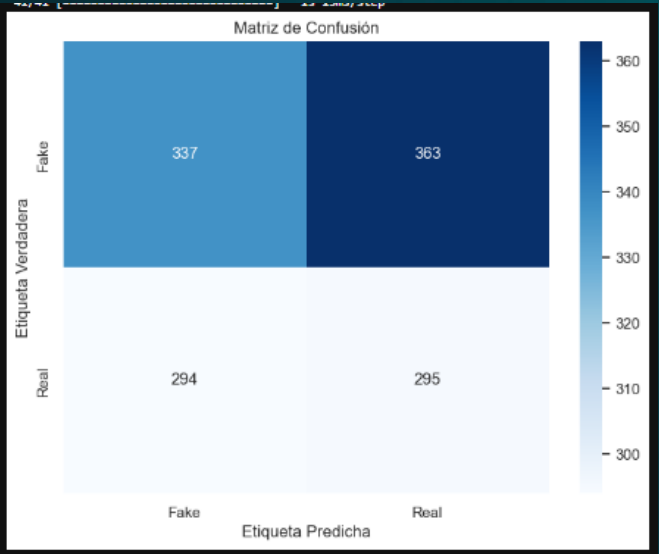
- Avances en Robustez y Generalización
- Eficiencia en el Entrenamiento y la Inferencia
- Aplicación de Redes Convolucionales Especializadas
- Mejora en la Detección de Imágenes Sintéticas

Resultados - Sequential

Model: "sequential_1"

Layer (type)	Output Shape	Param #
rescaling_9 (Rescaling)	(None, 128, 128, 3)	0
conv2d_45 (Conv2D)	(None, 126, 126, 64)	1792
conv2d_46 (Conv2D)	(None, 124, 124, 64)	36928
max_pooling2d_27 (MaxPooling2D)	(None, 62, 62, 64)	0
batch_normalization_36 (BatchNormalization)	(None, 62, 62, 64)	256
conv2d_47 (Conv2D)	(None, 60, 60, 128)	73856
conv2d_48 (Conv2D)	(None, 58, 58, 128)	147584
max_pooling2d_28 (MaxPooling2D)	(None, 29, 29, 128)	0
batch_normalization_37 (BatchNormalization)	(None, 29, 29, 128)	512
dropout_27 (Dropout)	(None, 29, 29, 128)	0
conv2d_49 (Conv2D)	(None, 27, 27, 256)	295168
max_pooling2d_29 (MaxPooling2D)	(None, 13, 13, 256)	0
batch_normalization_38 (BatchNormalization)	(None, 13, 13, 256)	1024
dropout_28 (Dropout)	(None, 13, 13, 256)	0
flatten_9 (Flatten)	(None, 43264)	0
dense_10 (Dense)	(None, 128)	5537920
batch_normalization_39 (BatchNormalization)	(None, 128)	512
dropout_29 (Dropout)	(None, 128)	0
dense_11 (Dense)	(None, 2)	258

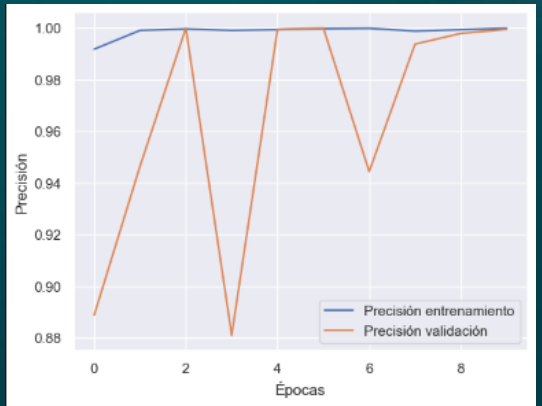
Total params: 6,095,810
Trainable params: 6,094,658
Non-trainable params: 1,152

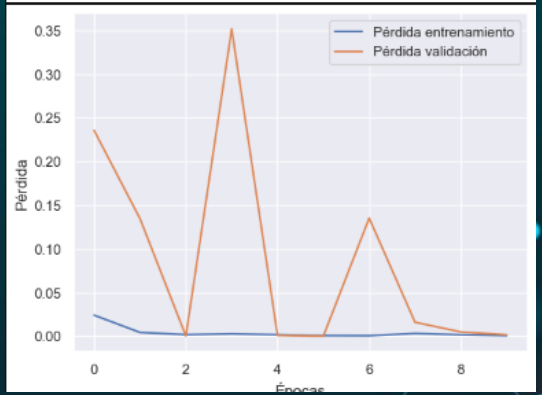


Reporte de Clasificación:

	precision	recall	f1-score	support
Fake	0.53	0.48	0.51	700
Real	0.45	0.50	0.47	589
accuracy			0.49	1289
macro avg	0.49	0.49	0.49	1289
weighted avg	0.49	0.49	0.49	1289

Precisión Global: 0.490302560124127 2

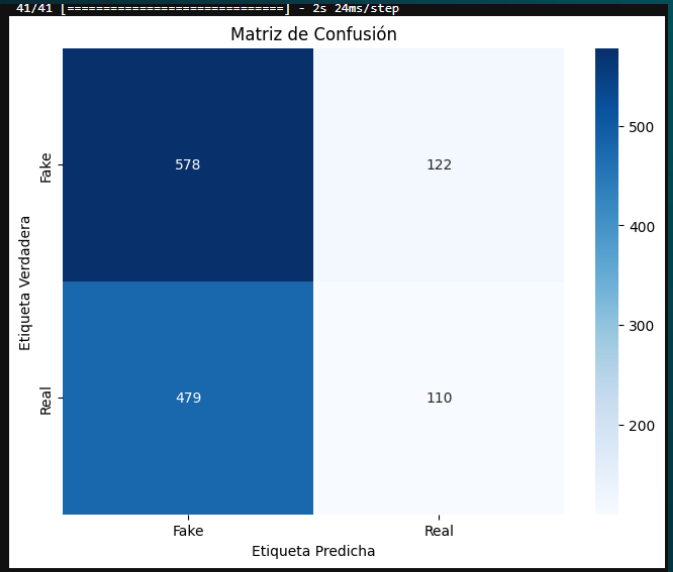




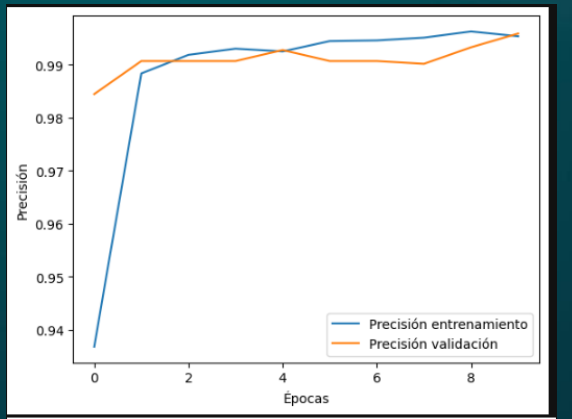
Resultados - MobileNetV2

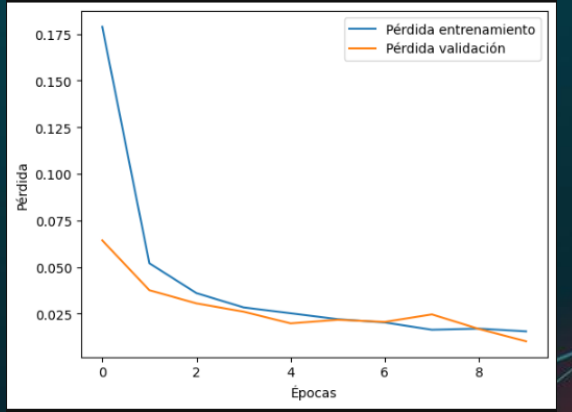
```
Model: "model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_2 (InputLayer)        [(None, 224, 224, 3)]     0

 sequential (Sequential)     (None, 224, 224, 3)       0

 tf.math.truediv (TFOpLambda  (None, 224, 224, 3)      0
 )

 tf.math.subtract (TFOpLambd  (None, 224, 224, 3)      0
 a)

 mobilenetv2_1.00_224 (Funct  (None, 7, 7, 1280)       2257984
 ional)

 global_average_pooling2d (G  (None, 1280)             0
 lobalAveragePooling2D)

 dropout (Dropout)           (None, 1280)              0

 dense (Dense)               (None, 1)                 1281

=================================================================
Total params: 2,259,265
Trainable params: 1,281
Non-trainable params: 2,257,984
_________________________________________________________________
```

```
41/41 [==============================] - 2s 24ms/step
```




```
Reporte de Clasificación:
              precision    recall  f1-score   support

        Fake       0.55      0.83      0.66       700
        Real       0.47      0.19      0.27       589

    accuracy                           0.53      1289
   macro avg       0.51      0.51      0.46      1289
weighted avg       0.51      0.53      0.48      1289

Precisión Global: 0.5337470907680373
```

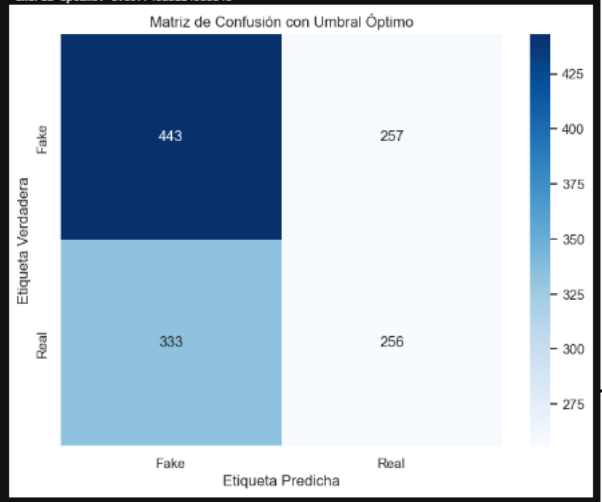




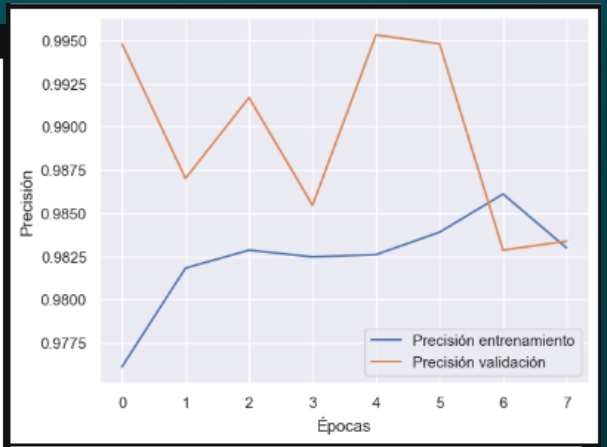
Resultados - EfficientNetB0

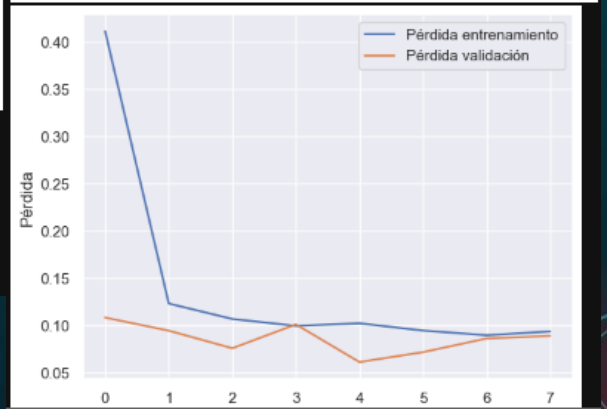
```
Model: "model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_2 (InputLayer)        [(None, 224, 224, 3)]     0

 sequential (Sequential)     (None, 224, 224, 3)       0

 efficientnetb0 (Functional)  (None, 7, 7, 1280)       4049571

 global_average_pooling2d (G  (None, 1280)             0
 lobalAveragePooling2D)

 dropout (Dropout)           (None, 1280)              0

 dense_1 (Dense)             (None, 128)               163968

 dense (Dense)               (None, 1)                 129

=================================================================
Total params: 4,213,668
Trainable params: 164,097
Non-trainable params: 4,049,571
_________________________________________________________________
```

```
41/41 [==============================] - 3s 36ms/step
Umbral Óptimo: 0.09774088114500046
```

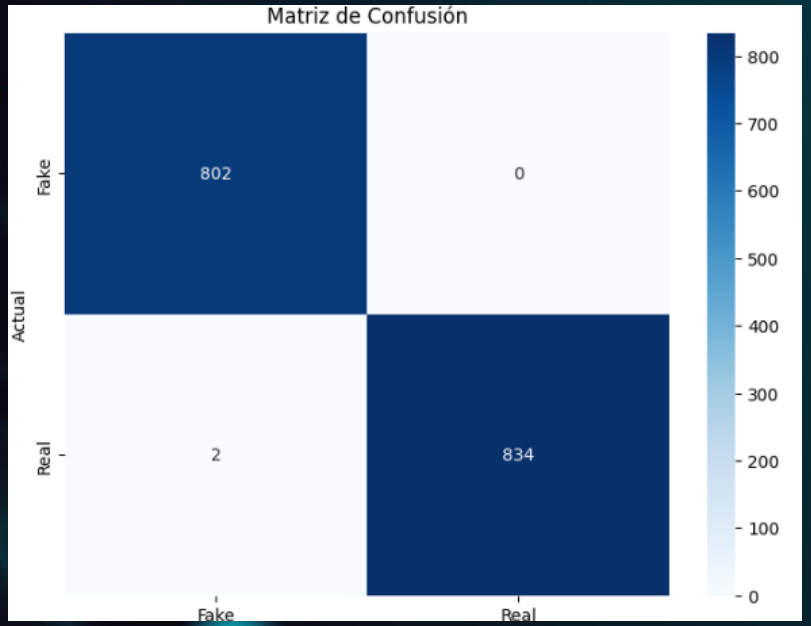


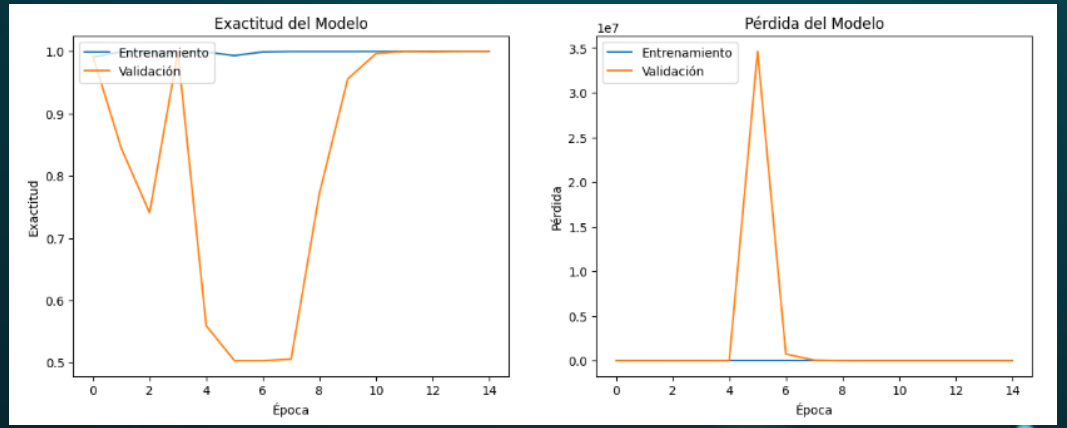

```
Reporte de Clasificación con Umbral Óptimo:
              precision    recall  f1-score   support

        Fake       0.57      0.63      0.60       700
        Real       0.50      0.43      0.46       589

    accuracy                           0.54      1289
   macro avg       0.53      0.53      0.53      1289
weighted avg       0.54      0.54      0.54      1289

Precisión Global con Umbral Óptimo: 0.5422808378588053
```





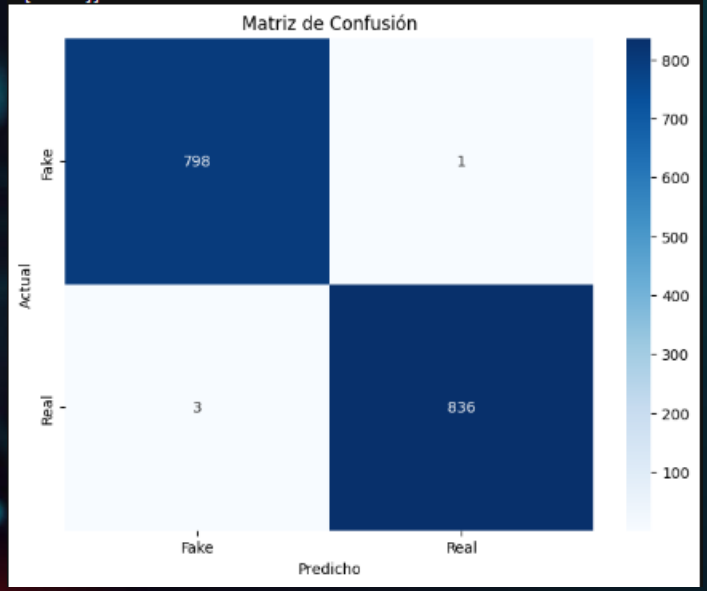
Resultados - 1era versión MobileNetV3





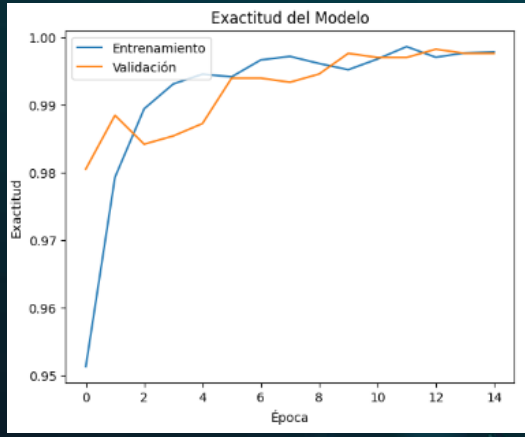
Resultados - MobileNetV3

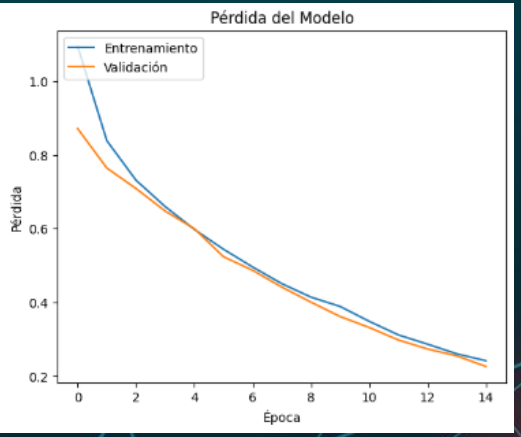
```
Pérdida en Prueba: 0.22447088360786438
Exactitud en Prueba: 0.9975579977035522
52/52 [==============================] - 1s 18ms/step
Reporte de Clasificación:
              precision    recall  f1-score   support

        Fake       1.00      1.00      1.00       799
        Real       1.00      1.00      1.00       839

    accuracy                           1.00      1638
   macro avg       1.00      1.00      1.00      1638
weighted avg       1.00      1.00      1.00      1638

Matriz de Confusión:
[[798   1]
 [  3 836]]
```



multiply_19 (Multiply)	(None, 7, 7, 960)	0	['Conv_1/BatchNorm[0][0]', 'tf.math.multiply_27[0][0]']
flatten (Flatten)	(None, 47040)	0	['multiply_19[0][0]']
dense (Dense)	(None, 512)	24084992	['flatten[0][0]']
dropout (Dropout)	(None, 512)	0	['dense[0][0]']
dense_1 (Dense)	(None, 2)	1026	['dropout[0][0]']

```
Total params: 27,082,370
Trainable params: 24,857,458
Non-trainable params: 2,224,912
```





MobileNetV3

```
tf.keras.backend.clear_session()

# Cargar el modelo MobileNetV3 Large
mobilenetv3 = MobileNetV3Large(weights='imagenet', include_top=False, input_shape=(height, width, 3))

# Congelar las primeras capas del modelo base
for layer in mobilenetv3.layers[:-20]:
    layer.trainable = False

# Aprendizaje por transferencia: Personalizar agregando nuevas capas sobre el modelo MobileNetV3 Large
x = Flatten()(mobilenetv3.output)
x = Dense(512, activation='relu', kernel_regularizer=l2(0.001))(x)
x = Dropout(0.45)(x)
output = Dense(len(labels), activation='softmax', kernel_regularizer=l2(0.001))(x)

# Crear el nuevo modelo
model_mobilenetv3 = Model(mobilenetv3.input, output)

# Compilar el modelo con una tasa de aprendizaje reducida
optimizer = Adam(learning_rate=1e-4)
model_mobilenetv3.compile(loss='categorical_crossentropy', optimizer=optimizer, metrics=['accuracy'])
model_mobilenetv3.summary()
```

¿Por qué es mejor?

MobileNetV3 funciona bien porque equilibra:

- Captura de patrones relevantes: Identifica artefactos generados por IA.
- Eficiencia: Se ajusta a datasets de tamaño moderado sin sobreentrenarse.
- Rapidez: Es más ligero y rápido, lo que ayuda en procesos iterativos como la diferenciación de caras.

Comparativa de resultados

#Intento	Modelo	Resultado
1	Sequential	Sin compilar
4	Sequential	0.4903025601241272
6	MobileNetV2	0.5337470907680373
8	MobileNetV2	0.4569433669511249
11	EfficientNetB0	0.5422808378588053
15	MobileNetV2	0.5057962172056132
16	MobileNetV2	0.5076266015863331
21	MobileNetV3	0.9975579977035522

Conclusiones

Desempeño del modelo:
El modelo alcanzó una exactitud del 99.76% y métricas casi perfectas,.

Impacto del Modelo:
El modelo supera los objetivos planteados.

Análisis de Errores:
Con solo 4 errores en 1638 predicciones y una tasa combinada de errores del 0.24%.

Recomendaciones Futuras:
- ❑ Validar con conjuntos más diversos.
- ❑ Explorar ensamblajes de modelos para mayor robustez.
- ❑ Implementar monitoreo en producción para mantener su desempeño a largo plazo.

REFERENCIAS

Wang, S., Wang, Y., Xu, C., & Sun, X. (2020). CNN-based detection of GAN-generated images. https://arxiv.org/abs/1911.12020

Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. https://arxiv.org/abs/1905.11946

Westerlund, M. (2021). The Emergence of Deepfake Technology: A Review. https://www.frontiersin.org/journals/communication/articles/10.3389/fcomm.2021.632317/full

F. Xavier Gaya-Morey, Silvia Ramis-Guarinos, Cristina Manresa-Yee, Jose M. Buades-Rubio. (2024). Unveiling the Human-like Similarities of Automatic Facial Expression Recognition: An Empirical Exploration through Explainable AI https://arxiv.org/pdf/2401.11835v3

GRACIAS