



REPORTE | MÓDULO 2

TAREA 4: GENERATIVE AI

Inteligencia artificial avanzada para la ciencia de
datos II (Gpo 503)

PROFESORA

NORA ELSA AGUIRRE SAMPAYO



Nombre del alumno:

Miguel Ángel Bermea Rodríguez

Matrícula:

A01411671

Carrera:

ITC



TECNOLÓGICO DE MONTERREY

FECHA DE ENTREGA: 24 DE NOVIEMBRE DE 2023

Contenido

Introducción.....	3
Análisis.....	3
Componentes de GANs	3
Diseño.....	3
Generador	3
Discriminador	3
Implementación	3
Carga de Datos	3
Entrenamiento	4
Pruebas.....	6
Antes del entrenamiento	6
Después del entrenamiento.....	7
Evaluación de Resultados.....	8
Problemas Encontrados.....	8
Ajustes Necesarios.....	9
Conclusiones	9

Introducción

Las Redes Generativas Adversarias (GANs) son modelos generativos que utilizan técnicas no supervisadas para generar nuevos datos. En este informe, se abordará la implementación de una GAN para la generación de imágenes de rostros humanos. Se explicarán los componentes clave de las GANs, su funcionamiento y aplicaciones prácticas.

Análisis

Componentes de GANs

Las GANs constan de dos componentes principales: el Generador, que crea datos similares a los de entrada, y el Discriminador, que distingue entre datos reales y generados. Estos dos componentes están en constante competencia, mejorando iterativamente sus habilidades.

Diseño

Generador

Existe un generador que toma un vector aleatorio de la distribución normal como entrada y pasa por capas densas y de convolución para generar imágenes de 128x128x3. Se utilizaron capas de batch normalization y activación LeakyReLU para mejorar la calidad de las imágenes generadas.

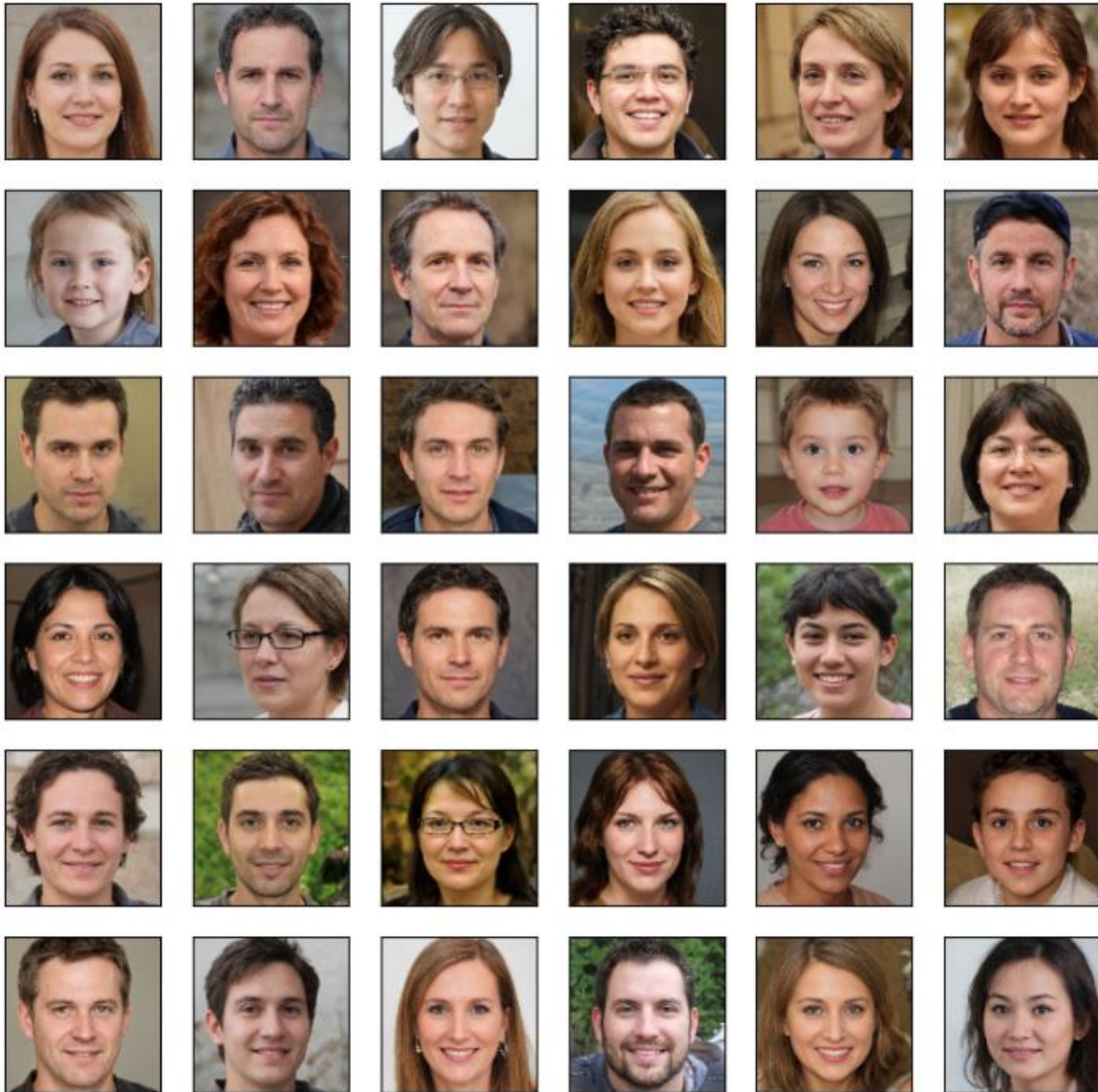
Discriminador

El discriminador es un modelo convolucional que toma imágenes de 128x128x3 como entrada y realiza capas de convolución para discernir entre imágenes reales y generadas. La salida utiliza la función de activación sigmoide para producir valores entre 0 y 1.

Implementación

Carga de Datos

Se utilizó el conjunto de datos "face-mask-lite-dataset" que se descargó directamente desde Kaggle. Se preprocesaron 9090 imágenes, convirtiéndolas a formato RGB y normalizándolas.



Entrenamiento

El modelo se entrenó durante 10 épocas utilizando el optimizador RMSprop con tasas de aprendizaje específicas. Se utilizaron funciones de pérdida de entropía cruzada binaria para el generador y el discriminador.

```
Epoch : 1
Training: 100%|██████████| 285/285 [09:03<00:00, 1.91s/it]
Time:543.0
Generator Loss: 0.7042951583862305 Discriminator Loss: 1.1644889116287231

Epoch : 2
Training: 100%|██████████| 285/285 [09:21<00:00, 1.97s/it]
Time:562.0
Generator Loss: 0.7087459564208984 Discriminator Loss: 1.1470086574554443

Epoch : 3
Training: 100%|██████████| 285/285 [09:21<00:00, 1.97s/it]
Time:562.0
Generator Loss: 0.6789466738700867 Discriminator Loss: 1.1573047637939453

Epoch : 4
Training: 100%|██████████| 285/285 [09:02<00:00, 1.90s/it]
Time:543.0
Generator Loss: 0.7030712366104126 Discriminator Loss: 1.16166090965271

Epoch : 5
Training: 100%|██████████| 285/285 [09:02<00:00, 1.90s/it]
Time:542.0
Generator Loss: 0.6999533176422119 Discriminator Loss: 1.1664655208587646

Epoch : 6
Training: 100%|██████████| 285/285 [09:21<00:00, 1.97s/it]
Time:562.0
Generator Loss: 0.7053493857383728 Discriminator Loss: 1.1613579988479614

Epoch : 7
Training: 100%|██████████| 285/285 [09:21<00:00, 1.97s/it]
Time:562.0
Generator Loss: 0.6876885890960693 Discriminator Loss: 1.1684486865997314

Epoch : 8
Training: 100%|██████████| 285/285 [09:21<00:00, 1.97s/it]
Time:562.0
Generator Loss: 0.7333178520202637 Discriminator Loss: 1.1839215755462646

Epoch : 9
Training: 100%|██████████| 285/285 [09:21<00:00, 1.97s/it]
Time:562.0
Generator Loss: 0.7298811674118042 Discriminator Loss: 1.1333880424499512

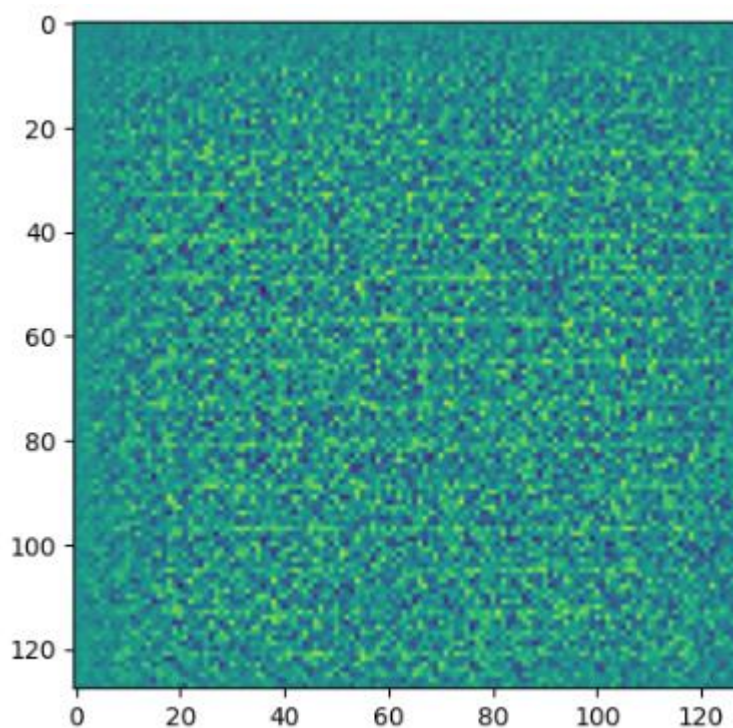
Epoch : 10
Training: 100%|██████████| 285/285 [09:02<00:00, 1.90s/it]
Time:543.0
Generator Loss: 0.6953165531158447 Discriminator Loss: 1.1961026191711426
```

Pruebas

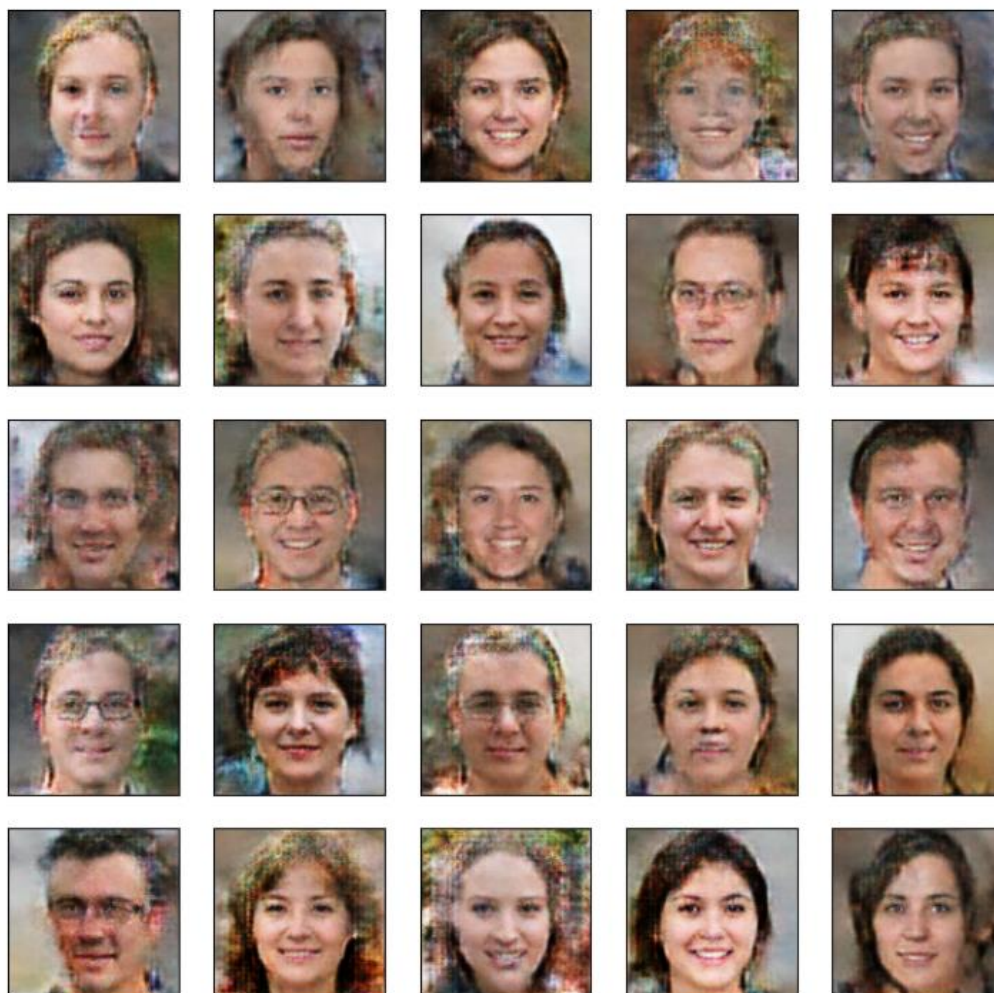
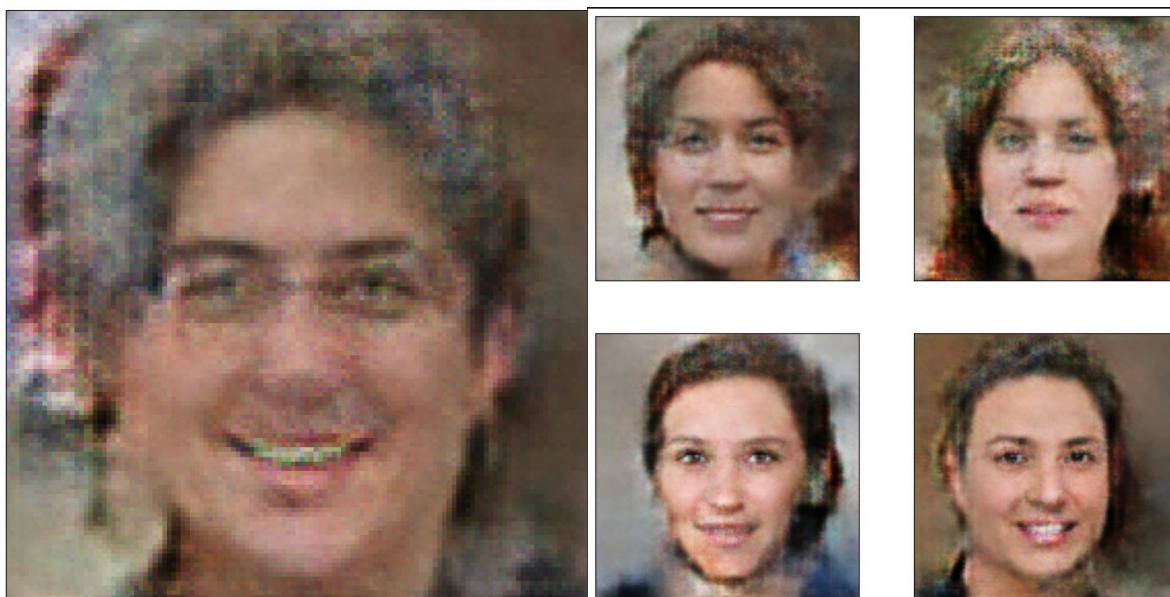
Se realizaron pruebas generando imágenes utilizando el generador antes y después del entrenamiento. Se visualizaron algunas imágenes generadas para evaluar la calidad y la mejora después del entrenamiento.

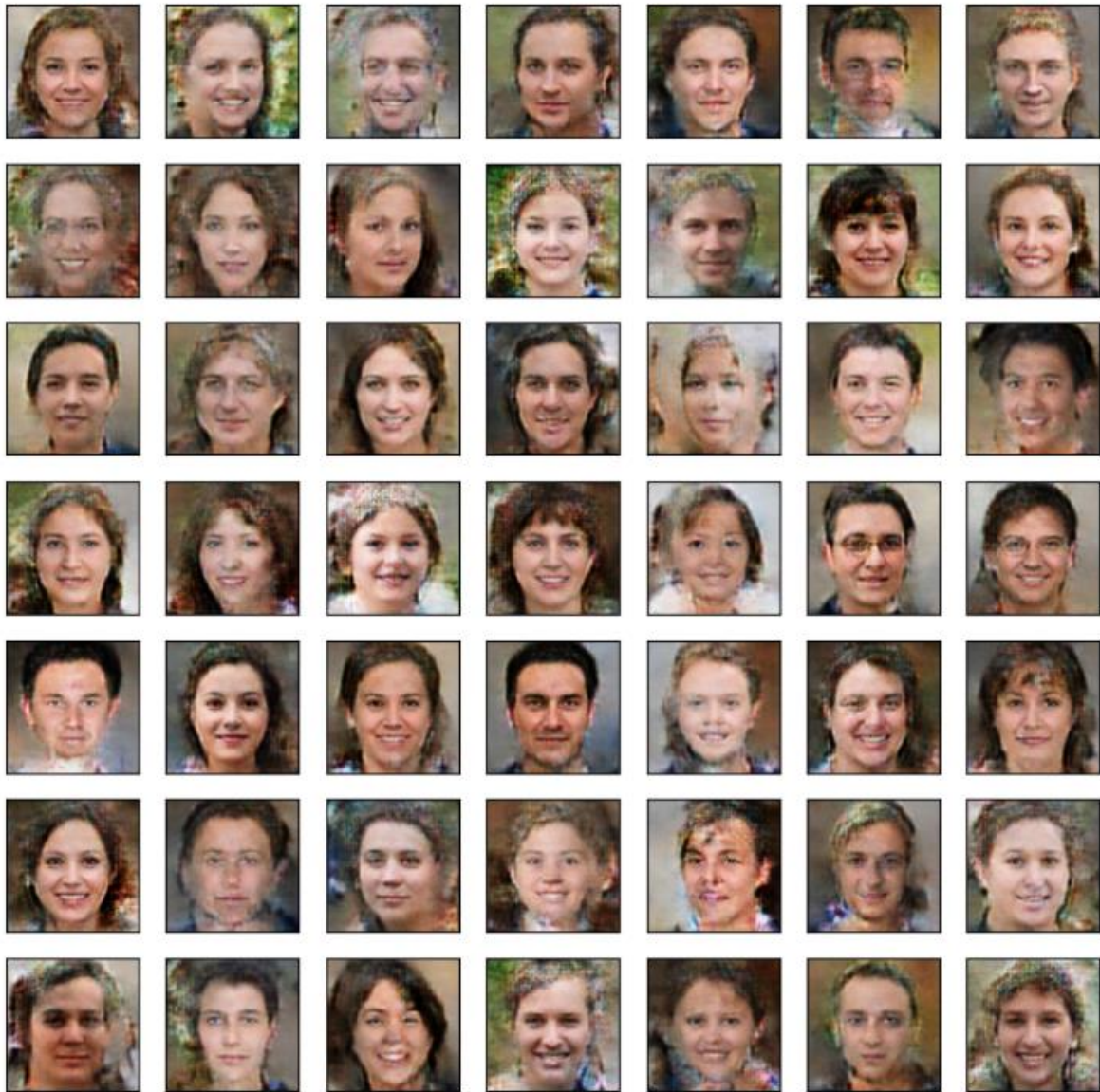
Antes del entrenamiento

Se utilizó el generador previo al entrenamiento, en este caso lo que se muestra no guarda ninguna relación con los datos (imágenes del dataset) y vendría a ser un “ruido aleatorio”. Es por esto por lo que se recurre al entrenamiento para que el propio generador aprenda a producir imágenes más realistas y relevantes para el objetivo del problema a solucionar (generación de imágenes).



Después del entrenamiento





Evaluación de Resultados

Durante el entrenamiento, se monitorearon las pérdidas del generador y del discriminador. Se observó que las pérdidas disminuyeron, lo que indica que el modelo estaba mejorando en la generación de imágenes realistas.

Problemas Encontrados

Durante la implementación, se enfrentó a un problema en relación con el hiperparámetro 'decay' del optimizador, por lo que se tuvo que señalar que era una característica *legacy* del optimizador para poder hacer uso de este.

Ajustes Necesarios

Se realizaron ajustes en la tasa de aprendizaje del optimizador para mejorar la convergencia del modelo. Además, se puede explorar la posibilidad de ajustar otros hiperparámetros como el tamaño del lote y la arquitectura de la red.

Asimismo, como ya se mencionó en la sección anterior se hizo un ajuste en el optimizador para poder contar con el hiperparámetro legacy denominado 'decay'.

Conclusiones

La implementación de la GAN logró generar imágenes realistas de rostros humanos. El modelo podría beneficiarse de ajustes adicionales y una mayor cantidad de épocas de entrenamiento. Las GANs ofrecen un enfoque prometedor para la generación de datos sintéticos en diversas aplicaciones.

En resumen, este reporte proporciona una visión general del diseño, implementación y evaluación de una GAN para la generación de imágenes de rostros humanos, destacando los desafíos enfrentados y las lecciones aprendidas durante el proceso.