Generación de Imágenes con Deep Convolutional Generative Adversarial Networks

Daniel Cabrera
Maestría en Ingeniería Informática
PUCP
Lima, Perú
dcabrerad@pucp.edu.pe

Joao Castro Pinto
Maestría en Ingeniería Informática
PUCP
Lima, Perú
a20226191@pucp.edu.pe

I. Introducción

En este trabajo de investigación vamos a enfocarnos en el Deep Convolutional Generative Adeversarial Network (DCGAN). Este es un modelo que trabaja con la generación de imágenes y con el aprendizaje de representación. Esto tiene una dificultad mediana ya que se debe identificar primero las arquitecturas que vamos a trabajar (discriminador y generador) para luego enfocarnos en el output de imágenes que son generadas por este modelo. Una estructura GAN, relacionada al trabajo de aprendizaje automático, se puede representar en la generación de imágenes contra imágenes reales que pasan por un discriminador de imágenes, el cual identifica si lo que se está pasando como entrada es real o falso. Con lo anterior explicado, tenemos como objetivo elaborar y entrenar una Generative Adversarial Network (GAN) para generar imágenes similares a las imágenes reales que son dadas como entrada al modelo discriminador. El presente informe va a explicar el estado del arte y la metología que va a ser usada para lograr dicho objetivo.

II. Estado del Arte

En [1] se centran en redes contradictorias, es decir, en las redes que se van a enfocar en la generación de imágenes y en las redes discriminatorias (D). Estas dos redes, se aplican cuando los modelos son multilayer perceptrons. Los autores definen un generator G que define una distrubución de probabiliadad que cumple con que de $G(z)$ se sabe que z es similar a Pz. Los autores usan el dataset de MNIST para la experimentación. Las redes generacionales usan activaciones lineales de rectificación y activaciones sigmoidales. Mientras que las redes discriminatorias usan activaciones maxout. Estos autores identifican como desventajas de usar estas estrategias que no se pueden identificar la representación de $Pg(x)$, además que D debe estar sincronizado con G durante el entrenamiento. Sin embargo, las ventajas que se presenta al usar esto, es que las redes contradictoras presentan una mayor ventaja en cuando a estadísticas sobre las redes de generación. Adicionalmente, las redes contradictoras tienen como ventaja el hecho de que representan cadenas muy fuertes que no pueden ser subjetivas al momento de generar análisis. Los autores concluyen, finalmente, que los modelos contradictorios pueden llegar a una dirección que puede ser útil.

En [2], los autores exploran y aplican los conceptos introducidos por el artículo [1]. Sin embargo, contribuyen con nuevas arquitecturas del modelo GAN, la DCGAN (Deep Convolutional GAN), mas con un aprendizaje más estable. Además, afirman no tener conocimiento de ningún otro estudio hasta la fecha de un modelo conseguir hacer aprendizaje de representaciones de forma no supervisada.

Los autores del documento [3] proponen evaluar diferentes formas de los GANs, en el cual se puede apreciar que usan el DGAN con modelos VGG, la estructura de red de un modelo DCGAN mejorado y el diseño de una estrategia de red de entrenamiento. LOs autores trabajan la experimentación de este modelo identificando en dataset, el entorno de experimentación y el entrenamiento del modelo, donde se puede apreciar y considerar que usaron 5000 épocas en esto último. No podemos olvidar que también se trabajó en la generación e identificación de la data. Se puede observar en la experimentación que el DCGAN mejorado que los autores proponen, con una estrcutira de VGG16 y REsNet50, se obtiene un accuracy de 86.07% y 90.34%. Por tanto, los autores concluyen que el modelo que proponen, el DCGAN mejorado para la generación de imágenes que consiste en una combinación de ResNet y VGG, abarcan muy bien el problema de la falta de data de imágenes, donde los autores identificaron que faltó imágenes con mayor resolución. Es por ello que se concluyen que se necesita mayor resolución en las imágenes para el uso de los modelos GAN.

Mientras tanto, en [4], se trabajó en un modelo que buscó generar ejemplos que se asemejen a la realidad, además se trabajó en un discriminador que pueda identificar los inputs de ejemplo en la data de entrenamiento. Para la experimentación, que se usó el dataset de CelebA Face Dataset, se introdujo las imágenes del dataset en dos modelos: DCGAN y WGAN. Sin embargo, también se probó en un tercer modelo que es parte del modelo de los autores. Los resultados de estos experimentos, mostraron mejores estadísticas en el modelo presentado por los autores para la generación de imágenes, el cuál presentó en dicho caso solo 19 caras sin detectar de las 1000 presentadas, frente a las 123 caras sin detectar de la WGAN y las 281 sin detectar de la DCGAN.

III. Metodología

A. Proceso de entrenamiento de una GAN

El proceso de entrenamiento de una GAN es bastante diferente y más complejo que el proceso de entrenamiento de una red CNN convencional para la clasificación o regresión de datos. Para entrenar una GAN, además del modelo generativo, es necesario un modelo de discriminación. Al modelo de discriminación le pasamos como entradas imágenes reales o imágenes generadas por el modelo generativo. La figura 1 representa este proceso gráficamente.

El objetivo del modelo de generación es ser capaz de generar imágenes, las cuales el modelo de discriminación no consiga comprender si son imágenes generadas o reales. Por el contrario, el modelo de discriminación tiene como objetivo aprender cuales son imágenes generadas por el modelo de generación y cuales son imágenes sampleadas del conjunto de datos.

Las funciones de pérdida de cada red son definidas con base en los objetivos referidos: para el modelo generativo la perdida es definida con base en el número de imágenes generadas que clasifica como generadas; para el modelo discriminatorio la pérdida es definida en función del número total de errores de clasificación.

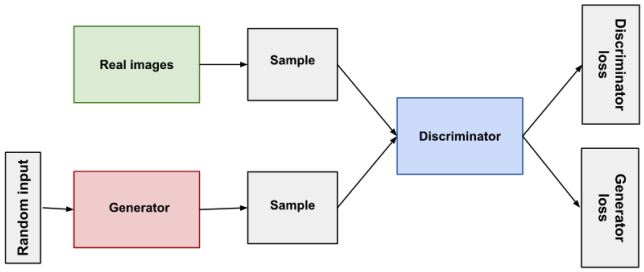

Fig. 1. Representación del proceso de entrenamiento de una GAN

B. Datos

Los datos necesarios para el entrenamientos del sistema que proponemos en este *paper* son imágenes y conjuntos de z números aleatorios, o sea ruido. El numero de valores de z propuesto en la bibliografía [2] es 100. Las imágenes son dadas como entrada de la red de discriminación y los conjuntos de ruido son entradas para la red de generación. La salida de la red de generación también es entrada de la red de discriminación.

La función de un modelo simple de generación de imágenes, tal como es el modelo propuesto en este documento, es altamente dependiente del conjunto de datos de entrenamiento. Si el aprendizaje del modelo fuer bien ejecutado, esté debería ser capaz de generar imágenes similares a las imágenes de entrenamiento.

Datasets muy específicos (p.ex.: MNIST) son muy útiles para comprender la capacidad de generaciones de imágenes del sistema, ya que lo que el modelo intenta representar es más comprensible. Por otro lado, al utilizar conjuntos de datos más generales, no se sabe lo que el modelo intenta representar. Por esa razón, se vuelve más complicado entender la calidad de las imágenes generadas por el modelo.

Por consiguiente, proponemos experimentaciones con 3 conjuntos de datos de niveles de especificidad diferentes: *MNIST* [7], *Dogs vs. Cats* [9] y *CIFAR-10* [8].

Si en el futuro se considerar necesario evaluar la capacidad de generalización de la red de discriminación, se deberá crear un conjunto de datos de prueba y otro de validación.

C. Evaluación de Modelos

Visto que hemos propuesto experimentar con 3 conjuntos de datos diferentes, nos gustaría señalar que planteamos hacer las evaluaciones que se siguen para el modelo entrenado en cada uno de los datasets referidos en la sección anterior.

Una forma trivial de evaluar la *performance* de un modelo de generación de imágenes podría ser pasar las imágenes generadas a un modelo preentrenado y confiar en el modelo la tarea de clasificar una imagen como plausible o no. Para una mejor ejecución de este método, sería necesario utilizar conjuntos de datos muy específicos y bien estudiados, tal como el *MNIST* por ejemplo. Esta técnica es utilizada en el *paper* [4].

Como fue mencionado en la sección II, los *papers* [5] y [6] proponen formas de evaluación para GANs. En el futuro utilizaremos algunas de estas técnicas para hacer una evaluación más profunda de cada modelo.

Debido a que el aprendizaje del modelo de discriminación es supervisado, para evaluar su capacidad de generalización, se podrá utilizar las técnicas de evaluación de aprendizaje supervisado convencionales, con un conjunto de datos de validación y otro de prueba.

D. Arquitectura

Como fue descrito en la sección III-A, el entrenamiento de una GAN necesita de dos redes neuronales. En esta subsección abordaremos la arquitectura propuesta el articulo introductorio de la DCGAN [2].

Con la intención de facilitar el aprendizaje de las redes, y producir mejores resultados, se realizaron las siguientes cinco recomendaciones arquitectónicas:

- Reemplazar capas de *pooling* por *strided convolutions* en la red de discriminación y *fractional-strided convolutions* en la red de generación.
- Utilizar *Batch Normalization* en todas las capas.
- Remover capas *fully connected* en arquitecturas profundas.
- En la red de generación, utilizar la activación *ReLU* en todas las capas, excepto la capa de output que debe utilizar *Tanh*.
- En la red de discriminación, utilizar la activación *LeakyReLU* en todas las capas.

1) Red de generación: La figura 2 es una representación gráfica de la arquitectura propuesta para la red. Son utilizadas 5 capas, de las cuales una densa para hacer la projección inicial y cuatro convolucionales. Se nota que excluyendo la primera

capa convolucional se utiliza siempre un *paso* de 2 en las convoluciones. El número de filtros de los tensores resultantes de cada capa son: 1024, 512, 256, 128, 3. El último tensor tiene 3 filtros que representan cada uno los canales de colores de una imagen (RGB). De ese modo, la salida de esta red neuronal es una imagen generada.

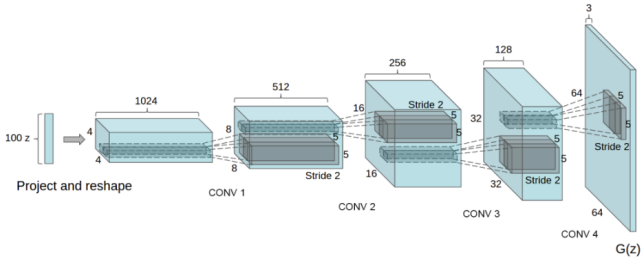

Fig. 2. Representación del proceso de entrenamiento de una GAN

2) Red de discriminación: En el articulo referido [2], no fue propuesta una arquitectura para la red de discriminación. Sin embargo, visto que el aprendizaje de esta red es del tipo supervisado, se puede tratar como una red convolucional de clasificación convencional.

Por lo tanto, utilizamos la propuesta de arquitectura [10] descrita por el siguiente código de *Python* con la *framework "Tensorflow"* y *"Keras"*.

```
def make_discriminator_model():
    model = tf.keras.Sequential()

    model.add(layers.Conv2D(64, (5, 5),
        strides=(2, 2), padding='same',
        input_shape=[28, 28, 1]))
    model.add(layers.BatchNormalization())
    model.add(layers.LeakyReLU())
    model.add(layers.Dropout(0.3))

    model.add(layers.Conv2D(128, (5, 5),
        strides=(2, 2), padding='same'))
    model.add(layers.BatchNormalization())
    model.add(layers.LeakyReLU())
    model.add(layers.Dropout(0.3))

    model.add(layers.Flatten())
    model.add(layers.Dense(1))

    return model
```

REFERENCES

[1] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y. (2014, June 10). Generative Adversarial Networks. arXiv.org. Retrieved November 1, 2022, from https://arxiv.org/abs/1406.2661v1.

[2] Radford, A., Metz, L., Chintala, S. (2016, January 7). Unsupervised representation learning with deep convolutional generative Adversarial Networks. arXiv.org. Retrieved November 1, 2022, from https://arxiv.org/abs/1511.06434v2.

[3] Min, F., Xiong, W. (2021). Butterfly Image Generation and recognition based on improved generative adversarial networks. 2021 4th International Conference on Robotics, Control[3] and Automation Engineering (RCAE). https://doi.org/10.1109/rcae53607.2021.9638915.

[4] Zhang, Ting, Tian, Wen-Hong, Zheng, Ting-Ying, Li, Zu-Ning, Du, Xue-Mei, Li, Fan (2019). Realistic face image generation based on generative Adversarial Network. 2019 16th International Computer Conference on Wavelet Active Media Technology and Information Processing. https://doi.org/10.1109/iccwamtip47768.2019.9067742.

[5] Lucic, M., Kurach, K., Michalski, M., Bousquet, O., Gelly, S. (2018). Are GANs Created Equal? A Large-Scale Study. arXiv:1711.10337v4

[6] Shmelkov, K., Alahari, K., amp; Schmid, C. (n.d.). How good is my gan? Retrieved from https://arxiv.org/pdf/1807.09499.pdf

[7] Deng, L. (2012). The mnist database of handwritten digit images for machine learning research. IEEE Signal Processing Magazine, 29(6), 141–142.

[8] Krizhevsky, A. (2009). CIFAR-10 and CIFAR-100 datasets. Retrieved November 6, 2022, from https://www.cs.toronto.edu/ kriz/cifar.html

[9] Kaggle. (2016). Dogs vs. Cats Redux: Kernels Edition. Retrieved November 6, 2022, from https://www.kaggle.com/competitions/dogs-vs-cats-redux-kernels-edition/data.

[10] Tensorflow. (2019). Deep convolutional generative Adversarial Network Tensorflow Core. Retrieved November 6, 2022, from https://www.tensorflow.org/tutorials/generative/dcgan