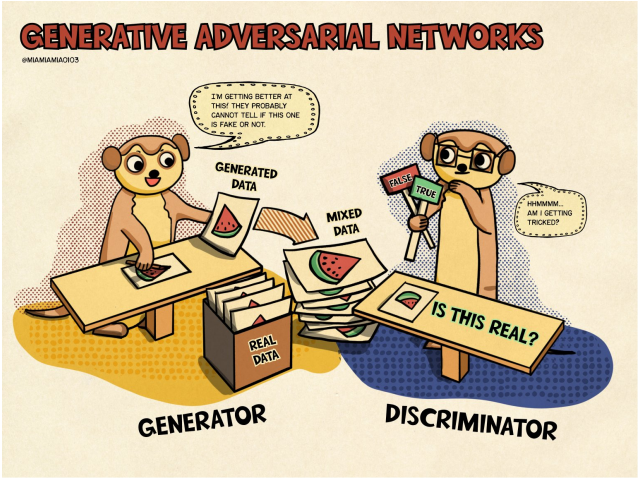
Modelos Generativos

Nahuel Costa

Grado en Ciencia e Ingeniería de Datos

Universidad de Oviedo
Universidá d'Uviéu
University of Oviedo

Redes GAN


GENERATIVE ADVERSARIAL NETWORKS
@MIAMIAMIAOIO3
I'M GETTING BETTER AT THIS! THEY PROBABLY CANNOT TELL IF THIS ONE IS FAKE OR NOT.
GENERATED DATA
MIXED DATA
REAL DATA
FALSE
TRUE
HHMMMM... AM I GETTING TRICKED?
IS THIS REAL?
GENERATOR
DISCRIMINATOR

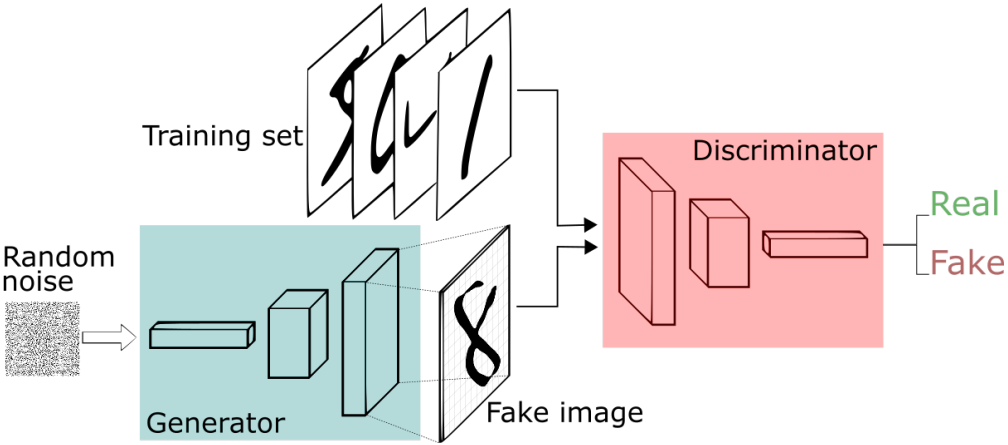
Redes GAN

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. Advances in neural information processing systems, 27.

Las redes generativas adversarias han demostrado grandes resultados en muchas tareas generativas para replicar contenido como imágenes, lenguaje humano y música. Están inspiradas en la teoría de juegos: dos modelos, un generador y un discriminador compiten y a la vez se mejoran entre sí.

"*The most interesting idea in the last 20 years in Machine Learning*" - Yann LeCun

Arquitectura



Fuente

Arquitectura

- Red generadora: recibe como entrada un vector de números aleatorio o ruido gaussiano (por eso a veces se le denomina z), a partir del cual se encargará de generar datos.
- Red discriminadora: Su labor es identificar si los datos que recibe son reales (parte del conjunto de entrenamiento) o falsos (generados por la red generadora).

Playground

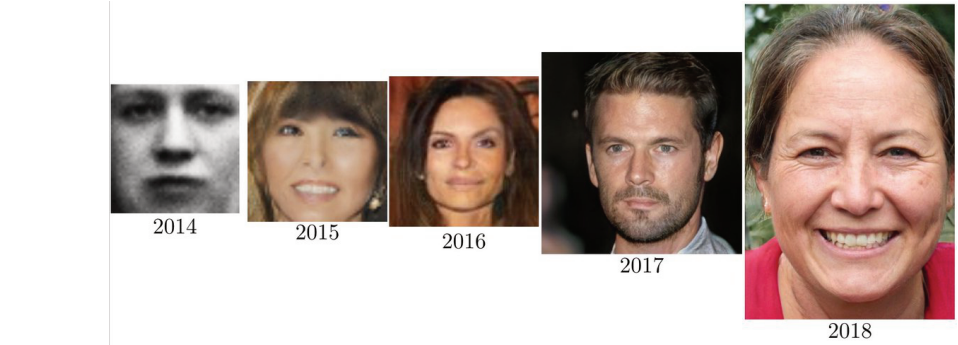
Explicación

Al principio del entrenamiento ninguna de las redes es buena en su tarea. La generadora producirá imágenes borrosas y sin coherencia y la discriminadora no será buena discerniendo entre datos reales o falsos. A medida que avanza el entrenamiento la generadora va mejorando en sus creaciones para intentar engañar a la discriminadora y la discriminadora, al ver más datos reales, va aprendiendo la distribución real de los datos, por lo que entran en un juego de suma 0 en el que al final del proceso ambas redes son muy buenas en las tareas que les han encomendado.

Normalmente, tras el entrenamiento la discriminadora se desecha, ya que el interés principal está en la generación de datos.

Aplicaciones

El área de aplicación más estudiada es la generación de imágenes.



De izquierda a derecha: GAN original (2014), DCGAN (2015), CoupledGAN (2016), ProgressiveGAN (2018), StyleGAN (2019). Fuente.

Aplicaciones

- **Conditional image generation**: por ejemplo, BigGAN [1] puede generar muestras de Imagenet condicionadas a una clase concreta. StyleGAN [2] es capaz de generar imágenes de caras en alta resolución condicionadas a características concretas. Aprende un embedding para luego interpolar entre diferentes características como peinado, gafas de sol, arrugas...
- **Paired image-to-image generation**: se pueden utilizar datos en la forma $(x_n; y_n)$ para construir modelos generativos condicionales $p(x|y)$. En algunos casos, la variable condicionante y tiene la misma dimensión que la variable de salida x. El modelo resultante $p(x|y)$ se puede utilizar entonces para realizar la traslación de imagen a imagen.

Aplicaciones

- **Unpaired image-to-image generation**: Una limitación de las GAN condicionales es la dificultad por recopilar datos emparejados. Es mucho más fácil recopilar datos no emparejados, por ejemplo, un conjunto de imágenes diurnas Dx, y un conjunto de imágenes nocturnas Dy. Suponemos que los conjuntos de datos Dx y Dy proceden de las distribuciones marginales $p(x)$ y $p(y)$ respectivamente. El objetivo es ajustar un modelo conjunto de la forma $p(x|y)$, de modo que podamos calcular los condicionales $p(x|y)$ y $p(y|x)$ y así traducir de un dominio a otro. Esto se denomina **unsupervised domain translation** y tiene aplicaciones como la transferencia de estilos.

Aplicaciones

- **Generación de vídeo**: La coherencia espacio-temporal se obtiene garantizando que el discriminador tenga acceso a los datos reales y a las secuencias generadas en orden, penalizando así al generador cuando genera fotogramas individuales realistas sin respetar el orden temporal.
- **Generación de audio**: Se han desarrollado muchas arquitecturas GAN diferentes para la generación de audio, incluyendo la generación de grabaciones de una sola nota de instrumentos por GANSynth, un modelo que trabaja con espectrogramas [3], la conversión de voz [4], y la generación directa de audios en WaveGAN [5].

Aplicaciones

- **Generación de texto**: existen varias tareas para datos de texto para las que se han desarrollado enfoques basados en GAN como la generación de texto condicional y la transferencia de estilo de texto. Los datos de texto suelen representarse como valores discretos (a nivel de caracteres o de palabras), que indican la pertenencia a un conjunto de un determinado tamaño de vocabulario (tamaño del alfabeto o número de palabras).
- **Domain adaptation**: Una tarea importante en Machine Learning es corregir los cambios en la distribución de los datos. Los enfoques para la generación de imágenes mencionados se centran en adaptaciones a nivel de píxel, como pix2pix [6]. Las extensiones de estos enfoques para el problema general de domain adaptation buscan hacer esto no sólo en el espacio de datos observado, sino también a nivel de características.

Función de error

Por un lado, queremos asegurarnos de que las decisiones del discriminador D sobre datos reales sean precisas. Maximizando $\mathbb{E}_{x\sim p_r}[logD(x)]$ forzamos a que $D(x)$ sea cercano a 1, lo que corresponde a una confianza alta en que x es realmente real. Dada una muestra falsa $G(z), z \sim p_z(z)$ se espera que el discriminador prediga una probabilidad, $D(G(z))$, cercana a cero maximizando $\mathbb{E}_{z\sim p_z}[log(1 - D(G(z)))]$.

Función de error

Por otra parte, el generador se entrena para aumentar las posibilidades de producir una alta probabilidad para un ejemplo falso, y así minimizar $\mathbb{E}_{z\sim p_z}[log(1 - D(G(z)))]$.

Al combinar ambos aspectos, estamos jugando un juego minimax en el que debemos optimizar la siguiente función de pérdida:

$$min_G max_D L(D, G) = \mathbb{E}_{x\sim p_r(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))] = \mathbb{E}_{x\sim p_r(x)}[\log D(x)] + \mathbb{E}_{x\sim p_g(x)}[\log(1 - D(x))]$$

Limitaciones

Aunque las GAN han demostrado un gran éxito en la generación de imágenes realistas, el entrenamiento no es fácil si no que es un proceso lento e inestable.

Cada modelo actualiza su coste de forma independiente sin tener en cuenta al otro "jugador". La actualización simultánea del gradiente de ambos modelos no garantiza la convergencia. Esto en ocasiones puede desembocar en el **mode collapse**. Durante el entrenamiento, el generador puede colapsar a una configuración en la que siempre produce las mismas salidas. Aunque el generador sea capaz de engañar al discriminador correspondiente, no consigue aprender a representar la compleja distribución de datos del mundo real y se queda atascado en un espacio pequeño con una variedad extremadamente baja.

Limitaciones

El entrenamiento de una GAN se enfrenta a un dilema:

- Si el discriminador no es bueno, el feedback que recibe el generador tampoco es bueno y en consecuencia la función de pérdida no puede representar la realidad.
- Si el discriminador hace un gran trabajo, el gradiente de la función de pérdida cae hasta cerca de cero y el aprendizaje se vuelve lento o incluso se atasca (**vanishing gradient**).

Este dilema puede hacer que el entrenamiento de GAN sea muy difícil. Evidentemente existen métodos para hacer este proceso más simple [7].

[1] A. Brock, J. Donahue, and K. Simonyan, "Large scale gan training for high fidelity natural image synthesis," *arXiv preprint arXiv:1809.11096*, 2018.

[2] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4401–4410, 2019.

[3] J. Engel, K. K. Agrawal, S. Chen, I. Gulrajani, C. Donahue, and A. Roberts, "Gansynth: Adversarial neural audio synthesis," *arXiv preprint arXiv:1902.08710*, 2019.

[4] T. Kaneko, H. Kameoka, K. Tanaka, and N. Hojo, "Cyclegan-vc3: Examining and improving cyclegan-vcs for mel-spectrogram conversion," *arXiv preprint arXiv:2010.11672*, 2020.

[5] C. Donahue, J. McAuley, and M. Puckette, "Adversarial audio synthesis," *arXiv preprint arXiv:1802.04208*, 2018.

[6] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1125–1134, 2017.

[7] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," *Advances in neural information processing systems*, vol. 29, 2016.