UNIVERSIDAD MAYOR REAL Y PONTIFICIA DE SAN FRANCICO XAVIER DE CHUQUISACA

GENERADOR DE PISTAS DE AUDIO MUSICALES APLICANDO REDES NEURONALES GENERATIVAS ADVERSARIAS PROGRESIVAS

ESTUDIANTE: YEISON MARIO FERNANDEZ ZARATE

C.U.: 111-161

CARRERA: ING. EN CIENCIAS DE LA COMPUTACIÓN

MATERIA: DESARROLLO DE APLICACIONES INTELIGENTES (SIS-330)

SUCRE - BOLIVIA

ÍNDICE

I. Descripción.-

Mediante el uso de Redes Neuronales Generativas Adversarias Progresivas, crear un sistema inteligente capaz de generar contenido musical (pistas de audio) a partir del entrenamiento con una gama de canciones que tengan características similares entre sí, tales como el género, autor, etc.

II. Antecedentes.-

La música y sus elementos han formado parte de la vida de los seres humanos desde tiempos inmemorables, y desde Pitágoras y su escuela, se tuvo una percepción de las matemáticas implícitas en las melodías.

Esta actividad, la creación y concepción de la música misma, ha sido asociada siempre como una capacidad propia de los humanos, como seres pensantes complejos capaces de producir ritmos y armonías que se traducían en la música que hoy conocemos.

Los mayores retos de la Inteligencia Artificial están ligados con la creación de modelos capaces de realizar esas tareas que son propias de los humanos, y en uno de los últimos avances de la misma, se han desarrollado las Redes Neuronales Generativas Adversarias y buscando eliminar la limitación de los tamaños de datos con los que estas funcionan, se evolucionó a las Redes Progresivas.

III. Problemática.-

La creación de recursos artísticos ya sean plásticos, visuales, musicales o cualquier forma de arte, son tareas que su creación está limitada a ser realizada por humanos.

IV. Objetivo General.-

Crear un sistema inteligente, capaz de generar pistas de audio con características de arte musical, con los elementos propios de la música: ritmo, melodía, armonía y matices.

V. Objetivos específicos.-

- Identificar investigaciones previas para verificar el estado del arte de las redes neuronales generativas adversarias progresivas.

- Formular una arquitectura de modelo generativo para generar información en formato de audio.
- Recopilar un banco de datos para entrenar el modelo diseñado y así obtener un generador funcional.
- Validar el modelo con datos experimentales para verificar la capacidad de generación de datos.

VI. Fundamento Teórico.-

La música cuenta con cuatro elementos esenciales que son: el ritmo, la melodía, la armonía y los matices. La forma en que se definen estos elementos varía de una cultura a otra.

En la actualidad, el proceso de "Composición musical", ha adquirido cierto grado de complejidad, el cual comúnmente es realizado por "Productoras musicales". Este proceso varía en sus elementos de acuerdo a muchas variables, tales como el género, los instrumentos musicales, las voces, el estilo musical, etc. Las etapas más generales de la producción son las siguientes: Pre-Producción, Grabación y Post-Producción, esta última dividida en la mezcla y la masterización.

La mezcla, es aquel proceso en el cual el productor, interactúa con los sonidos individuales, ya sean de instrumentos, voces o sonidos sintéticos.

A diferencia de la mezcla, el Ing. de masterización ya no tiene el acceso a hacer cambios individuales en los instrumentos o sonidos, sino que trabaja más bien con el resultado homogéneo de la mezcla.

El producto final es almacenado en un archivo de audio sin compresión, el cual tiene como característica un alto peso de almacenamiento. Para su distribución ya sea física o en plataformas digitales, este archivo es comprimido mediante algoritmos de compresión (mp3, wma, etc.) los cuales, mediante diversas técnicas, disminuyen el peso del archivo, eliminando sonidos imperceptibles para el oído humano.

Estos archivos de audio, cuentan con dos partes importantes: La señal y la frecuencia. La señal es almacenada en uno o varios canales, como una sucesión de números que representan la amplitud de la señal, cada unidad de información de esa secuencia es comúnmente denominada: sample. La frecuencia es la cantidad de samples reproducidas por unidad de tiempo, que en archivos de audio, es representada por un atributo llamado sample rate, que define la cantidad de samples reproducidos por segundo. En el caso de audios de dos canales, este conjunto de datos es representado en un array bidimensional.

Las redes neuronales generativas adversas, fueron creadas con el objetivo de generar imágenes con ciertas características, para lo cual se definió un nuevo método de entrenamiento, que consta de una red neuronal generadora, y una discriminadora, que para su entrenamiento, se produce una condición de competencia de suma cero. La red generadora, compite por crear cada vez imágenes sintéticas, más similares a imágenes reales, mientras que la discriminadora pretende determinar cuándo una imagen es real o sintética.

Una de las mayores limitantes de estas redes, es que no son capaces de generar imágenes de alta resolución, debido a la gran cantidad de datos que tienen las mismas, no es capaz de reconocer todos los patrones de los datos a gran escala y a pequeña escala a la vez. Como una solución a este problema, la corporación Nvidia, desarrolló un nuevo mecanismo, una adaptación de las tradicionales redes generativas adversas, acuñando el método progresivo en su investigación

denominada: *"Progressive Growing of GANs for Improved Quality, Stability, and Variation",* en la cual pudieron generar imágenes de hasta 1080x1080 px.

Esta investigación, implementa una variante de las redes neuronales adversarias denominada: "Wasserstein GANs + Gradient Penalty", la cual está basada en una investigación previa: "Wasserstein GANs". Ambas están basadas en la modificación de la función de costo de las originales redes adversas, que utilizan una función denominada minimax, la cual se encuentra basada en un concepto teórico ideal, el cual no siempre se da en casos reales, lo cual conduce a que las redes generadoras y discriminadoras ingresen en una competencia desfavorable, impidiendo la convergencia de las redes.

"Wasserstein GANs + Gradient Penalty", implementa como función de costo la distancia "Wasserstein", que puede ser definida como la cantidad de trabajo requerida para transformar una distribución probabilística en otra deseada. Además de esta distancia, es aplicado un "Gradient Penalty", la cual es una alternativa para forzar la condición de Lipschitz, estabilizando la fase de entrenamiento de estas redes, basada en la salida del discriminador o critic (denominada así en el paper original), respecto a una entrada interpolada entre datos reales y datos sintéticos o generados.

Tal como el documento *"Progressive Growing of GANs for Improved Quality, Stability, and Variation"* describe, se realizó con el objetivo de generar imágenes de alta resolución. Ya que los datos de pistas de audio son diferentes en formato a los de imágenes, se realizó algunas adaptaciones que se evidenciaron necesarias para mejorar la calidad de los datos generados, las cuales son descritas a continuación:

1. Función de costo.- Se realizó la sustitución de la función de costo "Wasserstein GANs + Gradient Penalty", que propone las siguientes fórmulas:

$$\boldsymbol{d_{loss} = \Big(D\big(G(z)\big) - D(x)\Big) + \lambda * GP}$$
$$\boldsymbol{g_{loss} = -D\big(G(z)\big)}$$

Las cuales proponen un escenario ideal, en el cual $\boldsymbol{D\big(G(z)\big)\ y\ D(x)}$ son valores positivos, y $\boldsymbol{D\big(G(z)\big) < D(x)}$, lo cual crearía una condición de competencia estable entre generador y discriminador, ya que la tarea del generador será maximizar el primer término de la función del discriminador y al contrario el discriminador tratará de minimizar toda la función en su conjunto. Tal escenario no siempre es posible de conseguir, ya que los valores de $\boldsymbol{D\big(G(z)\big)\ y\ D(x)}$ tienen una inicialización aleatoria definida por la inicialización de pesos y kernel de las capas ocultas. Además de esta condición, se evidencia una ambigüedad en la función del discriminador, ya que no se puede determinar si el discriminador, con el objetivo de minimizar su costo aumente o disminuya la diferencia entre $\boldsymbol{D\big(G(z)\big)\ y\ D(x)}$, lo cual en el caso de disminuir la diferencia, haría un trabajo contrario a su objetivo o interpretado de otra forma reducir su propia capacidad de diferenciar entre valores reales y falsos, lo cual da como resultado la minimización de su costo en ciertos escenarios, cuando $\boldsymbol{D\big(G(z)\big)\ y\ D(x)}$ presentan ciertos valores.

Buscando resolver estos conflictos, se propone la implementación de las siguientes funciones de costo:

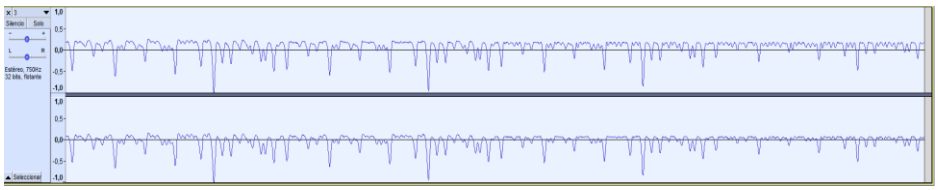
$$d_{loss} = \left(\frac{\left(D(G(z)) - D(x)\right)}{\left|D(G(z)) - D(x)\right|} + 0.0001 \right) * D(x) + \left(\frac{\left(D(G(z)) - D(x)\right)}{\left|D(G(z)) - D(x)\right|} + 0.000999 \right) * D(G(z))$$

$$g_{loss} = \left(\frac{\left(D(G(z)) - D(x)\right)}{\left|D(G(z)) - D(x)\right|} + 0.0001 \right) * D(G(z)) + \left|D(G(z)) - D(x)\right|$$

Dado que $|D(G(z)) - D(x)|$, representa la diferencia absoluta entre los valores de salida del discriminador para datos reales vs. datos falsos, las funciones descritas en la parte superior, para el caso del discriminador, al minimizarse mediante backpropagation, maximiza esta diferencia absoluta, y el discriminador al minimizar su función de costo minimiza esta diferencia, sean cuales sean los valores de $D(G(z))$ y $D(x)$, y sea cual sea su relación de magnitud, sea $D(G(z)) > D(x)$ o $D(G(z)) < D(x)$.

Los valores de las constantes 0.0001 y 0.000999, son aplicados para evitar el valor de cero, ya que se puede evidenciar que al ser cero la función del discriminador, los gradientes obtenidos se convierten en cero, anulando las variables entrenables de toda la red, es decir es un punto muerto a partir del cual la red deja de optimizarse.

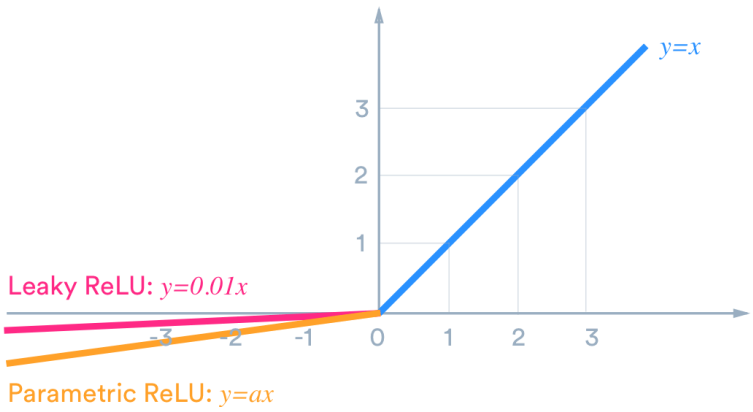
2. Función de activación intermedia.- En la generación de imágenes, se utilizó como función de activación entre los bloques del discriminador y el generador el rectificador LeakyRELu, el cual tiene por objetivo la eliminación de valores negativos, lo cual puede ser útil y lógico para trabajos relacionados con imágenes, ya que los valores de los datos sólo pueden ser positivos: de 0 a 255 o normalizados de 0 a 1. En el caso de las pistas de audio, los valores de sus datos incluyen valores negativos, es decir que sis valores van de -1 a +1.
   Mantener LeakyRELu como función de activación, provoca que al avanzar el entrenamiento, el generador se ve dificultado a crear datos en los dos sentidos: positivos y negativos, obteniendo como salida una imagen de las señales similar a la siguiente:



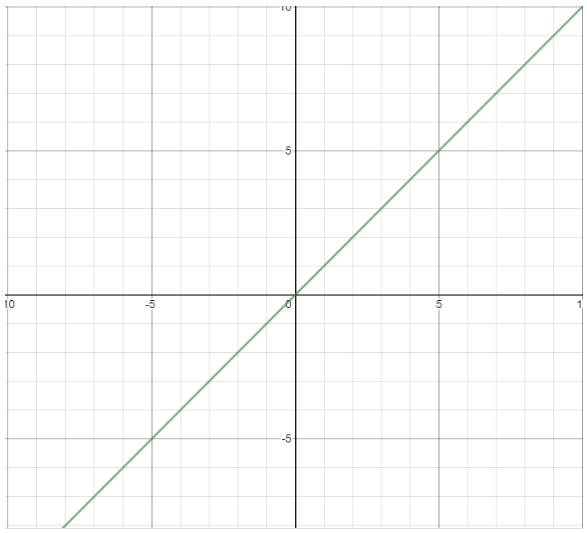
La primera elección tras evidenciar este problema provocado por la función de activación, fue eliminar el rectificador, dejando las capas intermedias con una función de activación lineal, lo cual condujo a un segundo problema, ya que al avanzar el entrenamiento, los valores de entrada de cada capa, crecían desproporcionadamente, provocando que pasadas ciertas iteraciones los valores fueran demasiado grandes, que hacían que las operaciones sean computacionalmente imposibles de realizar, dando como resultado un valor Nan.

De esta forma, se optó por el modelado de una función de activación propia, la cual tiene como principio el escalado de los datos de entrada de manera proporcional a su tamaño.

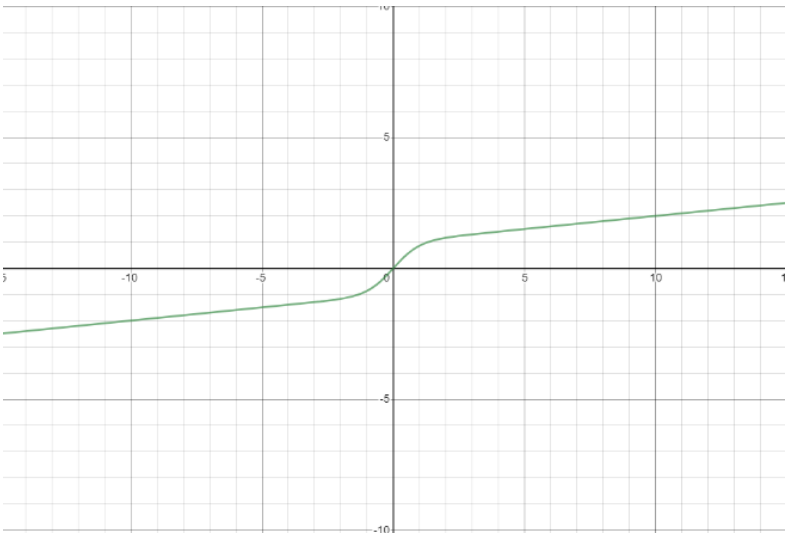
Las 3 diferentes funciones, representadas gráficamente, se ven de la siguiente manera:



LeakyRELu



Función de activación lineal



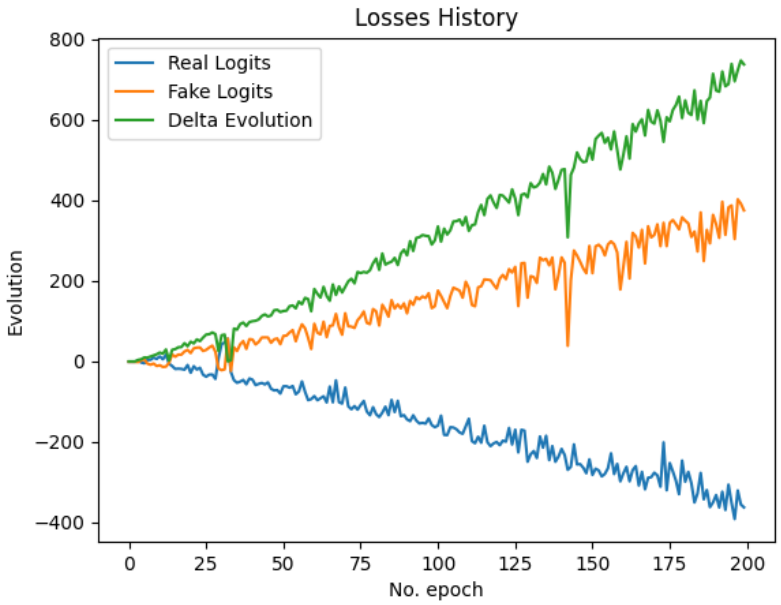
Función de activación propia

La fórmula de la función de activación propuesta para escalar los valores con relación a su tamaño esta dada de la siguiente forma:

$$f(x) = tanh(x) + \frac{x}{alpha}$$

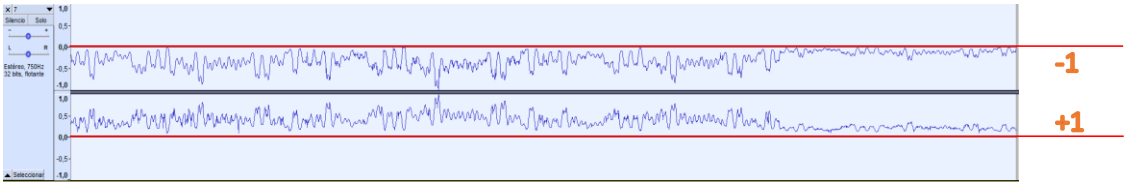
Donde alpha es un parámetro que define la pendiente de la curva para valores más grandes de x, de forma que mientras alpha sea más grande, menor es la pendiente, haciendo que valores de x más grandes, se escalen a un valor más pequeño.
Para este caso, se utilizó un valor diferente de alpha, para cada escala de las progresiones de la red, es decir que para las redes de dimensiones más grandes, un valor mayor de alpha, ya que al ser de mayores dimensiones, los valores a escalar son cada vez más grandes.

3. Función de activación de salida del discriminador.- En el caso de la función de activación de salida del discriminador, se sustituyó la función lineal del trabajo con imágenes, por la misma función definida en el punto anterior, pero con un valor de alpha mayor al de las intermedias.
   También se realizó la prueba con la función de la tangente hiperbólica, ya que también podría adecuarse para que la función de costo sea efectiva, ya que esta nos da como salida valores entre -1 y +1, pero esta condición estática a partir de ciertos valores, es decir los límites de -1 y +1, quitan la posibilidad al discriminador de seguir optimizándose al llegar a cierto punto.
   Es por esta razón que con una pequeña modificación, nuestra función de activación, permite continuar su optimización al discriminador, aunque haya llegado a cierto punto, limitando el crecimiento para valores mayores, evitando así que a diferencia de la función lineal, los valores se salgan de un rango calculable computacionalmente.
4. Resultados obtenidos de las optimizaciones.- Como resultados más resaltantes de la optimización de los puntos mencionados, tenemos la posibilidad de verificar durante el entrenamiento si las redes tienden a mejorar o no, verificando el valor de la diferencia absoluta de los valores del discriminador respecto de datos reales y falsos: $|D(G(z)) - D(x)|$
   Tal como se puede observar en la siguiente gráfica de la evolución:

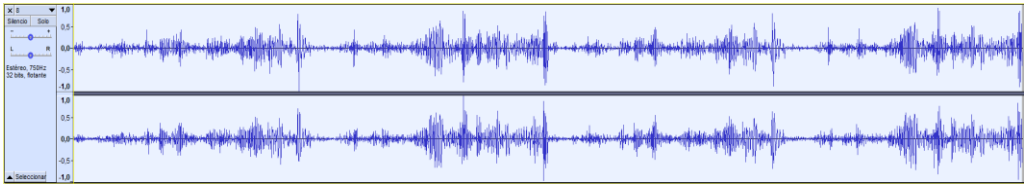


Donde Delta representa esta diferencia absoluta con línea verde, y las líneas naranja y azul representan $D(G(z))$ y $D(x)$ respectivamente.
El resultado esperado después de un número considerable de iteraciones, es ver una tendencia de Delta hacia el descenso, lo cual representará que el generador ha sido capaz de ajustarse para generar datos de características similares a los originales, de forma que el discriminador tiene menor capacidad de diferenciarlos.

Además se observa en los datos generados, tras cierto número de iteraciones, la eliminación de la tendencia hacia cierto valor: -1 o +1. Esta mejora es posible apreciarla de mejor manera en los siguientes gráficos:



Antes



Ahora

Tal como se puede evidenciar en las gráficas de las señales, se experimentó una notable mejora en la calidad de los datos generados, en comparación con los ejemplos de datos reales, de los cuales las señales graficadas son similares a las siguientes:

Datos reales de entrenamiento

VII. Métodos y Herramientas.-

Se utilizará como plataforma hasta la fase de entrenamiento y evaluación del modelo, la herramienta Colaboraty Pro de Google.

Una vez entrenado y almacenado el modelo, este se utilizó para implementar una plataforma web, con el framework Django, que realiza predicciones con los modelos cargados.

Se utilizaró la librería Keras con Tensorflow como backend para la implementación de la red neuronal, además de diversas librerías desarrolladas en Python para el procesamiento de los datos.

VIII. Evaluación del modelo.-

Para la etapa de evaluación se aplicó el Inception Score, una medida propuesta inicialmente con el clasificador de imágenes Inception de Google. Esta medida puede aplicarse con la ayuda de cualquier clasificador, y evalúa principalmente 2 aspectos de los datos: la calidad y la diversidad de los datos generados.

Para este objetivo se entrenó un total de 7 clasificadores, uno para cada dimensión de la progresión, capaces de clasificar pistas de audio por autor, con el mismo conjunto de datos de entrenamiento que posteriormente será utilizado en la red generativa.

La publicación de generación de imágenes de Nvidia, en la sección de resultados obtenidos, presentan los Inception Score obtenidos con diferentes conjuntos de datos, obteniendo en su mejor resultado una puntuación de 8.80 ± 0.05 y un promedio de 8.56 ± 0.06.

En el caso de este trabajo, previo a realizar las mejoras mencionadas, se obtuvo valores entre 1 y 2,5 de Inception Score, y posterior a las modificaciones, se alcanzó máximas de 7,01.

El proceso de evaluación aún queda pendiente, ya que los clasificadores podrían basar la clasificación en características no perceptibles por el oído humano, causando que el generador obtenga un puntaje mayor pero al no ser características perceptibles por el oído, no resultan agradables para el oyente final. Para realizar una evaluación más seria de los resultados, se deberán considerar otras formas de evaluación y se requerirá de mayores recursos para completar los entrenamientos de las generativas en términos de mayores iteraciones y posibilidad de completar todas las escalas de las progresiones definidas.

IX. Cronograma.-

Nº	ACTIVIDAD	INICIO	FIN
1	RECOLECCIÓN DE DATOS	20-11-2020	22-11-2020
2	PREPROCESAMIENTO DE DATOS	23-11-2020	27-11-2020
3	DISEÑO DE ARQUITECTURA DEL MODELO	28-11-2020	30-11-2020
4	IMPLEMENTACIÓN DEL MODELO	01-12-2020	15-12-2020
5	ENTRENAMIENTO DEL MODELO	16-12-2020	20-01-2021
6	VALIDACIÓN DEL MODELO	20-01-2021	10-02-2021
7	IMPLEMENTACIÓN DE INTERFAZ GRÁFICA	10-02-2021	15-02-2021

X. Bibliografía.-

- *"Progressive Growing of GANs for Improved Quality, Stability, and Variation"* - https://research.nvidia.com/publication/2017-10_Progressive-Growing-of
- *"Los elementos de la música"* - http://colegiodemaria.com.ar/materiales/3/musica/Apuntes-de-MUSICA-3AyB.pdf
- *"Etapas De La Producción Musical"* - https://www.audioproduccion.com/etapas-de-la-produccion-musical/
- *"UNDERSTANDING AUDIO FILE FORMATS: FLAC, WMA, MP3"* - https://www.retromanufacturing.com/blogs/news/understanding-audio-file-formats-flac-wma-mp3
- *"WGAN + GP"* - https://arxiv.org/pdf/1704.00028v3.pdf