**FACULTAD DE CIENCIAS Y TECNOLOGÍA**

INGENIERÍA EN CIENCIAS DE LA COMPUTACIÓN

**GENERACIÓN DE MÚSICA USANDO DCGAN**



Materia: Desarrollo de Aplicaciones Inteligentes (SIS330)

Estudiante: Oropeza Acosta Aylin Pola

C.U.: 111-310

Sucre – Bolivia

2021

**ÍNDICE**

1. ANTECEDENTES
2. PROBLAMÁTICA
3. PROPUESTA
4. OBJETIVO GENERAL
5. OBJETIVOS ESPECÍFICOS
6. MARCO TEÓRICO
7. METODOLOGÍA
8. MÉTODOS Y MATERIALES
9. BIBLIOGRAFÍA
10. **ANTECEDENTES**

Las matemáticas siempre estuvieron presentes en la música, dando explicación a los sistemas de afinación, ayudando a comprender el sonido. Con la aparición de las computadoras se comenzó a utilizarlas como instrumentos, brindando herramientas que revolucionaron la música, creando nuevos géneros como la música electrónica, el lo-fi y muchos más, estos géneros pasaron a tener una riqueza tímbrica a diferencia de la música acústica, cuyo fuerte es la armonía.

Los algoritmos estuvieron para facilitar el trabajo de composición de los músicos, llegando a ser utilizados también como instrumentos al poder crear inimaginables sonidos que pueden ser utilizados en la música. Pero ahora la Inteligencia Artificial es capaz de componer por su cuenta y no ser solamente una herramienta.

1. **PROBLEMÁTICA**

Se cree que composición está limitada únicamente a los humanos por su contenido emotivo, pero con composiciones de la IA que logren engañar a los melómanos, ésta se encargaría de la industrialización de la música, mientras los artistas pueden dedicarse más a experimentar y jugar con antiguos y nuevos elementos musicales.

1. **PROPUESTA**

Existen varios algoritmos capaces de componer música, se propone utilizar uno para obtener una pequeña obra musical.

1. **OBJETIVO GENERAL**

Utilizar la Inteligencia Artificial para generar audios con forma musical, con una precisión mayor al 90% como evaluación cuantitativa y aceptación de los oyentes como evaluación cualitativa.

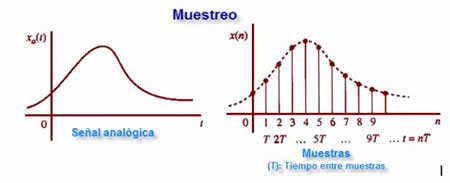
1. **OBJETIVOS ESPECÍFICOS**

* Identificar elementos matemáticos, digitales del estado del arte.
* Constituir un dataset de pistas de audio.
* Implementar y ajustar el modelo inteligente.
* Validar el modelo.

1. **MARCO TEÓRICO**

**EL SONIDO**

El sonido es una oscilación mecánica que se propaga en un medio y pueden ser percibidos por el oído. El sonido agradable tiene una percepción ordenada de la onda, ésta a su vez está compuesta de la amplitud, longitud de onda, periodo y frecuencia. Estas ondas son señales analógicas, que al ser captadas por un micrófono se convierte en señal digital tomando millones de muestras (una grabación de alta calidad consiste en 44 000 muestras por segundo), esta señal digital consiste en matrices de números.



**ARCHIVOS MIDI**

Las siglas MIDI corresponden a la abreviatura anglosajona de “Musical Instrument Digital Interface”, una breve explicación de este protocolo creado para la comunicación y mezcla de señales que provienen de varios instrumentos y dispositivos​ de creación musical. Estos podrán enviarse a una computadora digital, concretamente a un software que se encargará de la producción musical de nuestra obra y se obtiene de la grabación de interpretaciones en instrumentos netamente digitales o desde un software simulador, de esta forma los archivos midi contienen información como el nombre de las notas, tono, frecuencia, tempo, velocidad, instrumento, etc

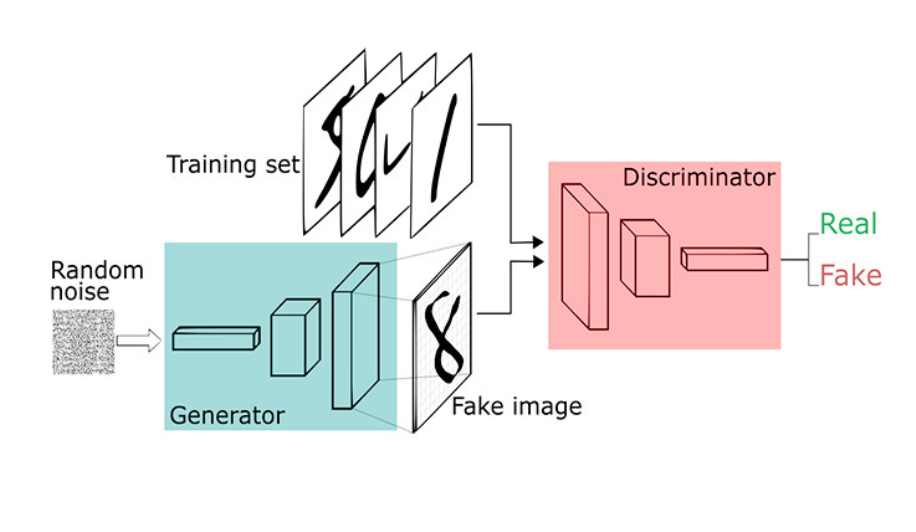
El resultado final de todo sonido que pasa por el midi es un archivo de gran calidad y de un tamaño no demasiado grande, que además permite modificaciones y ajustes.

**REDES NEURONALES GENERATIVAS ADVERSAS (GAN-DCGAN)**

Las GANs consisten en enfrentar dos redes neuronales que compiten en un constante juego de suma cero (si una gana, la otra pierde).

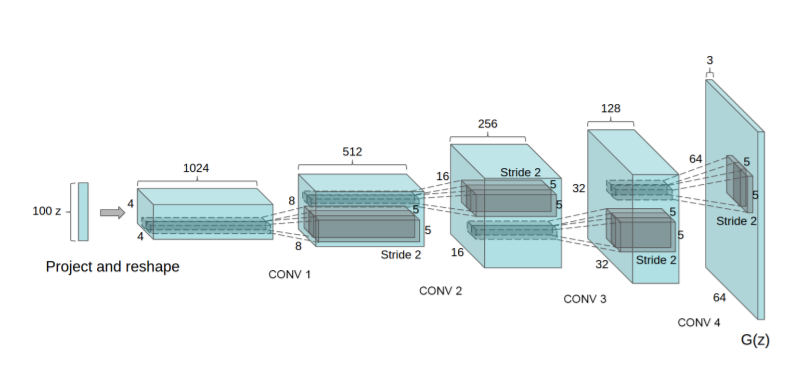
Una red es generativa, produce muestras de los que queremos crear (imágenes, textos, sonidos), que al comienzo generará ruido y luego irá mejorando en cada intento. Mientras la segunda red, la discriminadora entrenada en la identificación analizará el material producido por la red generativa y determina si se ajusta a lo que buscamos. Entonces, la red generativa trata de hacer pasar sus creaciones como reales, mientras la red discriminadora no deja pasar contenido que considera falso.

Así, puede haber cientos, miles o millones de intentos antes de que la red discriminadora acepte el resultado ofrecido por su rival. Mientras tanto, rechazo tras rechazo, la red generativa habrá ido 'aprendiendo' qué es lo que busca la discriminadora, quien a su vez le habrá ido guiando con la información que aportaba con sus porcentajes de acierto.



En este proyecto se utiliza la arquitectura DCGAN (Deep Convolutional Generative Adversarial Network), que consiste en una representación del aprendizaje no supervisado. Esta arquitectura tiene algunos cambios en las arquitecturas CNN:

1. La primera red del Generador es totalmente convolucional, reemplazando funciones de agrupación como el maxpooling con convoluciones escalonadas.
2. Se eliminan capas completamente conectadas, la GAN recibe el ruido y devuelve un tensor de 4 dimensiones.



1. Normalización por lotes, se utiliza batch normalization a todas las capas, menos a la capa de salida del Generador y la capa de entrada del Discriminador para mantener la estabilidad de la arquitectura. El Generador la función de activación ReLU excepto en la capa de salida, la cual utiliza la función Tanh y el Discriminador utiliza la función de activación LeakyReLU en todas las capas.

**FRÉCHET INCEPTION DISTANCE (FID)**

Fréchet Inception Score es una de las mejores métricas aplicable a las redes GAN, que nació para suplir las carencias de su antecesora Inception Score al comparar las estadísticas de las muestras reales y las muestras generadas, en lugar de sólo evaluar las muestras generadas. Desde la propuesta de Heusel, Ramsauer, Unterthiner, Nessler y Hochreiter el año 2017 se usa la distancia Fréchet entre dos matrices gaussianas multivariables.

FID=||μr−μg||2+Tr(Σr+Σg−2(ΣrΣg)1/2)

Donde y son las activaciones de la capa pool3 del modelo pre entrenado Inception-v3 para muestras reales y generadas, respectivamente.

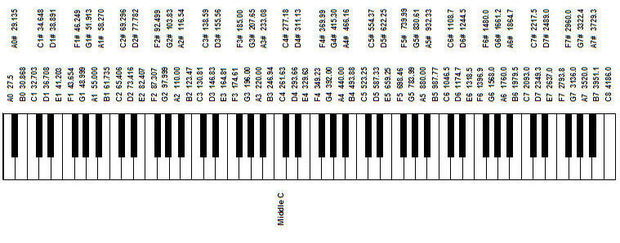
**FUNCIÓN DE PÉRDIDA**

El discriminador es entrenado para clasificar correctamente muestras reales y generadas. Esto se logra maximizando el logaritmo de la probabilidad predicha de las muestras reales y el logaritmo de la probabilidad invertida de las muestras generadas, promediado sobre cada mini lote de muestras.

Entonces podemos entender esta función de pérdida buscando probabilidades cercanas a 1.0 para muestras reales y probabilidades cercanas a 0.0 para muestras generadas. La suma de estos valores significa que los valores más pequeños de probabilidad de esta función de pérdida resultan en un mejor rendimiento para el discriminador.

1. **METODOLOGÍA**

Se tienen 331 obras para piano de compositores clásicos como Bach, Mozart y Chopin, cuidando que no sean obras demasiado densas musicalmente y evitando las disonancias (notas que no pertenecen a la misma familia, consiguiendo un sonido menos agradable). Con la librería music21 se procesará los archivos midi del dataset y se tendrá la información de cada nota por su nombre, por ejemplo C#5, lo que corresponde a la notación americana de la nota do, el “#” indica que es sostenida (un semitono más arriba) o “-“ para bemol (un semitono más abajo) y el 5 es la octava en la que está ubicada la nota. Una descripción gráfica se encuentra en la figura:

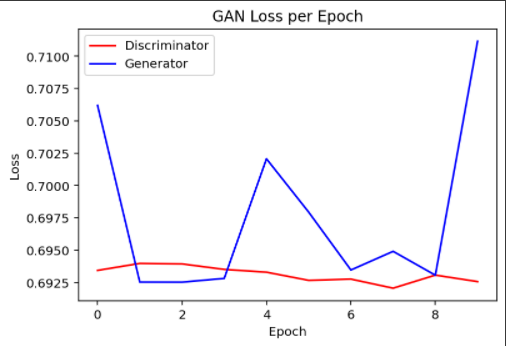


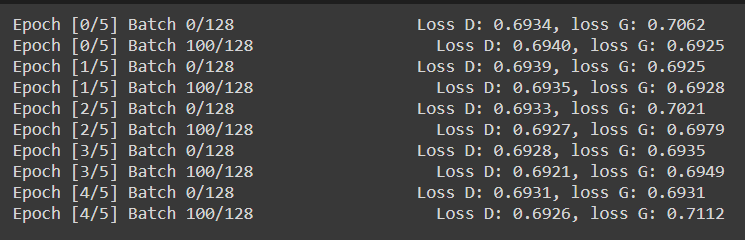
Los acordes son la ejecución de dos o más notas simultáneamente, se accederá a éstos con la concatenación de las notas que lo componen.

Una vez obtenidas las notas y acordes de las pistas se las pasa a un diccionario asignándoles un número.

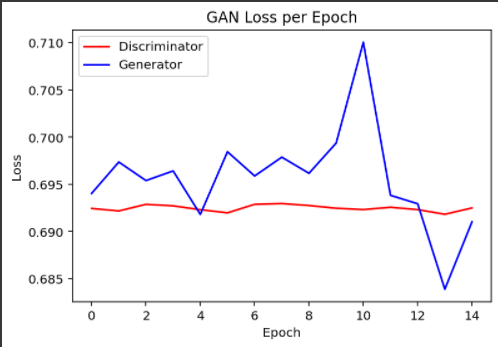
Dada la estructura del modelo descrito anteriormente se pasará esta lista de datos a un tensor que distribuya todos los valores en 4 dimensiones. Las notas pasarán de un modelo unidimensional a una matriz 64x64 repartidos en la mayor cantidad de lotes que permita el dataset, entonces la entrada al modelo sería un tensor de 128x1x64x64. Al contener un rango mayor de valores que una imagen, los datos del tensor se normalizarán entre -1 y 1.

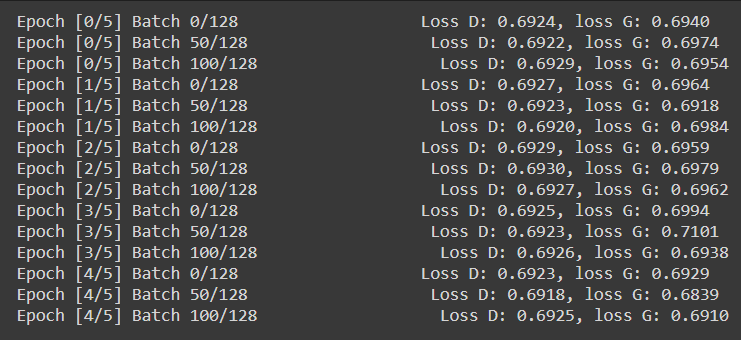
Al momento de hacer el entrenamiento, con un learning rate de 0.0002 para el generador, el discriminador y sin dropout en el modelo se obtuvo los siguientes valores de la función de pérdida:





Con dropout para el generador, un learning rate de 0.0001 en el generador y 0.0004 en el discriminador se obtuvieron los siguientes valores:





1. **MÉTODOS Y MATERIALES**

Se utilizará Google Colab como herramienta para la creación, entrenamiento y pruebas del modelo siguiendo la arquitectura DCGAN, para el preprocesamiento de archivos midi se utilizará la librería music21 y el modelo pre entrenado Inception-v3.

1. **RESULTADOS Y CONCLUSIONES**

El proyecto va tomando forma, pero no se puede hacer una evaluación cualitativa al no contar con la transformación de datos numéricos a notas. En cuanto a la función cualitativa se tiene las probabilidades de las funciones de pérdida, pero las métricas son necesarias para una evaluación más objetiva.

1. **BIBLIOGRAFÍA**

Dataset:

* <https://www.midiworld.com/bach.htm>
* <http://www.jsbach.es/bbdd/index01_20.htm>
* <http://www.piano-midi.de/midi_files.htm>

DCGAN paper:

* <https://arxiv.org/pdf/1511.06434.pdf>

Fréchet Inception Distance:

* <https://nealjean.com/ml/frechet-inception-distance/>

DCGAN in PyTorch with FID:

* <https://towardsdatascience.com/generating-pokemon-inspired-music-from-neural-networks-bc240014132>