

---

# UNIVERSIDAD AUTÓNOMA DE GUADALAJARA

---

CON RECONOCIMIENTO DE VALIDEZ OFICIAL DE ESTUDIOS DE LA SECRETARÍA DE  
EDUCACIÓN PÚBLICA SEGÚN ACUERDO No.158 DE FECHA 17 DE JULIO DE 1991

---

## POSTGRADO E INVESTIGACIÓN



Uso de Deep Learning para composición musical

---

## TRABAJO DE INVESTIGACIÓN

QUE PRESENTA

**EFRAIN ADRIAN LUNA NEVAREZ**

PARA OBTENER EL GRADO DE

**MAESTRÍA EN CIENCIAS COMPUTACIONALES**

DIRECTOR Y ASESOR:

MTRO. JUAN ANTONIO VEGA FERNANDEZ

GUADALAJARA, JALISCO. AGOSTO 2018

Número de registro en el sistema de investigación

---



# Tabla de Contenido

---

<b>Lista de Figuras</b>	<b>III</b>
<b>Lista de Tablas</b>	<b>v</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Descripción del Problema . . . . .	1
1.2. Objetivos . . . . .	1
1.2.1. Objetivo General . . . . .	1
1.2.2. Objetivos Específicos . . . . .	2
1.3. Justificación . . . . .	2
1.4. Delimitación . . . . .	2
1.5. Organización de la Tesis . . . . .	3
<b>2. Marco Teórico</b>	<b>5</b>
2.1. Bases teóricas . . . . .	5
2.1.1. Inteligencia Artificial . . . . .	5
2.1.2. Redes neuronales artificiales . . . . .	7
2.1.3. Deep Learning . . . . .	11
2.1.4. Redes neuronales recurrentes . . . . .	11
<b>3. Desarrollo</b>	<b>15</b>
<b>4. Conclusiones</b>	<b>17</b>
<b>Bibliografía</b>	<b>19</b>



# Lista de Figuras

---

2.1. Ramas de la inteligencia artificial . . . . .	5
2.2. Red neuronal biológica . . . . .	7
2.3. Estructura general de una neurona . . . . .	8
2.4. Neurona Artificial . . . . .	8



## Lista de Tablas

---





---

## Capítulo 1

# Introducción

---

La música es el idioma universal, no importa en qué lugar físico nos encontremos, todos alguna vez hemos escuchado alguna canción que se nos queda grabada en la cabeza por mucho tiempo. En ocasiones nos preguntamos cómo es que el autor compuso esa canción.

Si bien es cierto el componer una canción es un arte, las canciones se pueden interpretar en forma matemática, esto nos abre la posibilidad de usar algoritmos para la creación de nuevas canciones.

### 1.1. Descripción del Problema

El tiempo requerido para la creación de una nueva canción es muy variado, ya que depende del género, el número de instrumentos y por supuesto la imaginación del compositor. Sin embargo si pudiéramos usar un algoritmo de aprendizaje en una computadora, esto nos permitiría generar nuevas canciones de una manera más sencilla y tomando mucho menos tiempo.

Existen compositores que a lo largo del tiempo no les gusta explorar nuevas formas o ritmos musicales y eso ocasiona que sus composiciones sean muy similares entre si.

### 1.2. Objetivos

#### 1.2.1. Objetivo General

Creación de un programa que use algoritmos de Deep Learning para servir de apoyo en una composición musical.

### 1.2.2. Objetivos Específicos

- 1.- Implementar una red neuronal que sea capaz de crear melodías y armonías musicales en guitarra.
- 2.- Implementar una red neuronal que sea capaz de crear melodías musicales en bajo eléctrico.
- 3.- Implementar una red neuronal que sea capaz de crear ritmos básicos de batería.
- 4.- Conjuntar la salida de las redes neuronales para generar un solo archivo de audio.
- 5.- Verificar la cuadratura y tonalidad de la canción resultante.

## 1.3. Justificación

El uso de algoritmos de Deep Learning en la música no es algo nuevo, sin embargo, existen muchas áreas de oportunidad en cuestión de la implementación de estos algoritmos.

La mayoría de trabajos en esta área se basan en sonidos melódicos y no abarcan tan a fondo el campo armónico. Lo que se pretende en esta investigación es tener una manera fácil y rápida para la creación de nuevas composiciones musicales partiendo ya sea por melodías o armonías musicales de uno o varios instrumentos.

Un algoritmo eficaz para composición es aquel que es capaz de aprender a componer música partiendo de tres fundamentos musicales: la tonalidad, el ritmo y la cuadratura, el algoritmo que se pretende realizar en esta investigación evaluará estas variables y las tratará de generalizar para poder crear música a partir de una simple melodía o una armonía.

## 1.4. Delimitación

Este proyecto se centrará en crear melodías musicales de guitarra y bajo eléctrico así como también ritmos básicos de batería, no se analizarán otros instrumentos.

Las nuevas canciones creadas por estos algoritmos no buscan ser ideas finales de composición, la única intención es crear una serie de ideas para nuevas canciones.

La base de datos usada para entrenar a la red neuronal consta de canciones de género rock y pop únicamente, no se entrenara la red para reconocer y generar canciones de otros géneros musicales.

## 1.5. Organización de la Tesis



---

## Capítulo 2

# Marco Teórico

---

## 2.1. Bases teóricas

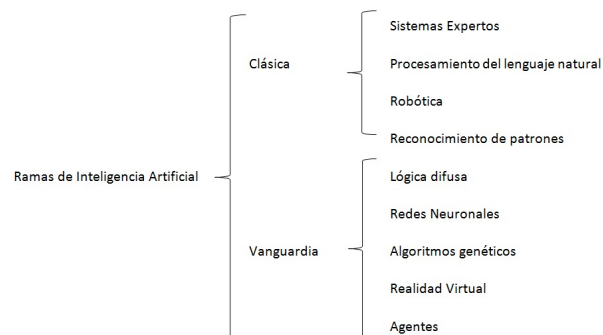
### 2.1.1. Inteligencia Artificial

En computación el término Inteligencia Artificial se define como la facultad de razonamiento de un programa informático, este agente racional tiene la capacidad de percibir su entorno y lleva a cabo acciones para maximizar el éxito en una tarea asignada.

Cuando una maquina es capaz de imitar las funciones cognitivas del ser humano para aprender y resolver problemas se puede decir que la maquina posee inteligencia artificial.

Dentro de la inteligencia artificial tenemos varias ramas que se enfocan en la resolución de problemas aplicando un principio muy específico de la inteligencia.

Las ramas de la inteligencia artificial se pueden dividir en áreas clásicas y áreas de vanguardia de acuerdo con la época cuando surgieron:



**Figura 2.1:** Ramas de la inteligencia artificial

**Sistemas expertos.-** son conocidos también como sistemas de conocimientos y estos programas informáticos aplican el proceso de razonamiento de un humano experto en la materia en la solución de problemas específicos. El modo de procesamiento de estos sistemas es en base a una gran base de datos y utilizando una heurística avanzada para la determinación de las posibles soluciones a un problema.

**Procesamiento del lenguaje natural.-** estos son sistemas capaces de reconocer, procesar y en cierto punto emular la comunicación humana, estos sistemas buscan dejar el uso de lenguajes de programación o conjunto de comandos, para procesar el lenguaje humano natural. Para procesar el lenguaje es necesario dividirlo, primero se obtiene la comprensión del lenguaje natural, el cual investiga los métodos para que una computadora sea capaz de entender las instrucciones de este lenguaje, la segunda etapa consiste en la generación de lenguaje natural, aquí es donde la maquina intenta comunicarse en el lenguaje humano.

**Robótica.-** un robot es un dispositivo programado para realizar una tarea en específico. Se dice que un robot está adquiriendo inteligencia artificial si es capaz de responder a cambios en su entorno en lugar de seguir instrucciones programadas con anterioridad.

**Reconocimiento de patrones.-** esta es la parte de la inteligencia artificial encargada del procesamiento visual de un entorno, las imágenes son captadas por cámaras y posteriormente procesadas para el reconocimiento de patrones del entorno.

**Lógica difusa.-** es una forma matemática de representar el lenguaje natural, y el principio es generalizar la lógica clásica haciendo que las variables tomen valores lingüísticos de verdad.

**Redes neuronales.-** estas redes tratan de emular el comportamiento de las redes neuronales biológicas que poseen los humanos en su cerebro, partiendo de una neurona artificial y conectándola con otras para crear sinapsis entre ellas.

**Algoritmos genéticos.-** estos algoritmos son capaces de mutar para producir mejores respuestas a un entorno, estos sistemas tratan de imitar el proceso de selección natural en el cual al ir mutando algunos genes se van obteniendo sistemas más capaces para un entorno. Su función es seleccionar de una población de soluciones candidatas e intentar producir nuevas generaciones de soluciones las cuales se buscan que sean mejores que las anteriores.

**Realidad virtual.-** es la recreación de un mundo artificial en tiempo real que puede ser captado por distintos canales sensoriales del espectador. Agentes.- estos son

pequeños programas que actúan como espías observando las acciones comúnmente realizadas por el usuario, estas acciones son almacenadas y registradas, si en dado caso se llega a dar una anomalía el programa lanzara una alerta y dará una serie de soluciones.

### 2.1.2. Redes neuronales artificiales

Las redes neuronales artificiales son un conjunto de neuronas creadas artificialmente para el desarrollo de inteligencia artificial en una computadora.

Estas redes están basadas en las redes biológicas del cerebro humano, modelando todos los factores biológicos de las neuronas.



*“Jiménez, J., Pensamientos habituales y desarrollo personal. [Red neuronal]. Recuperado de <http://poderpersonalmexico.com/tag/redes-neuronales>”*

**Figura 2.2:** Red neuronal biológica

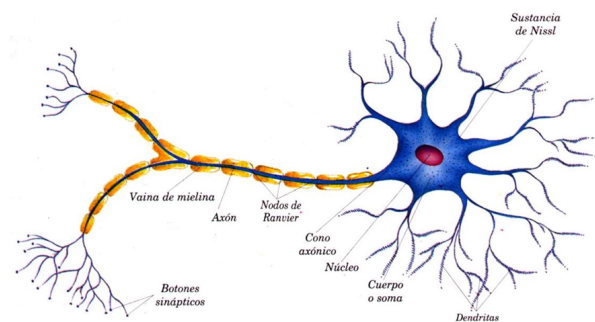
Debido a su diseño las redes son capaces de aprender de la experiencia, de generalizar de casos anteriores a nuevos casos, de abstraer características esenciales a partir de entradas que representan en ocasiones información irreverente.

La capacidad de aprendizaje adaptativo es una característica fundamental de las redes neuronales y les permiten llevar a cabo ciertas tareas mediante un entrenamiento previo, pueden aprender a diferenciar patrones y generalizar a partir de estos. Son considerados sistemas dinámicos ya que son capaces de adaptarse a nuevas condiciones de entrada.

Tienen una alta tolerancia a fallos ya que son capaces de detectar patrones aun cuando estos patrones posean ruido, distorsión o simplemente están incompletos. Estos programas son capaces de seguir funcionando incluso si parte de la red presente fallas.

La información se almacena de forma distribuida en las conexiones de las neuronas, provocando redundancia de información, es decir se guardara sus valores en base a la función de activación que posee cada neurona, de esta manera si una neurona es destruida o presenta fallas, las otras neuronas podrán aprender la información de la neurona que fallo.

Las neuronas humanas poseen diferentes secciones:



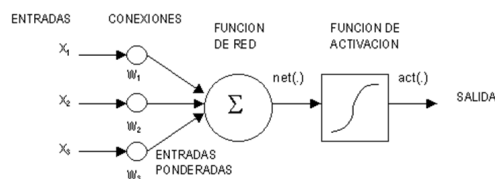
"Autor anónimo, Cuadros sinópticos y mapas conceptuales sobre las neuronas [Estructura general de una neurona]. Recuperado de <http://cuadrocomparativo.org/cuadros-sinopticos-sobre-las-neuronas>"

**Figura 2.3:** Estructura general de una neurona

En una neurona artificial se busca la emulación de las principales secciones de una neurona las cuales son:

- **Cuerpo.-** Se encarga de producir un impulso eléctrico en base a las entradas de la neurona.
- **Dendritas.-** son filamentos capaces de crear conexiones con otras neuronas.
- **Axón.-** Es el encargado de transmitir el impulso eléctrico generado por el cuerpo.

A continuación se muestra una imagen de como luciría una neurona artificial:



"Autor anónimo, Introducción a las redes neuronales artificiales [Neuronal artificial]. Recuperado de <http://www.rna.50webs.com/>"

**Figura 2.4:** Neurona Artificial

Esta neurona posee las siguientes secciones:



- **X1, X2, ..., Xn**.- Son las entradas de la neurona.
- **W1, W2, ..., Wn**.- Pesos específicos que tendrá cada entrada, esto hace que las entradas no valgan lo mismo ponderadamente.
- **Función de red**.- Esta es una función de sumatoria de las entradas.
- **Función de activación**.- Si la suma de las entradas es mayor o igual que el umbral definido por esta función se tendrá una señal a la salida.

La salida de la neurona viene dada por esta ecuación:

$$y_j = f\left(\sum_{i=1}^n (W_{ij}x_i + \theta_j)\right) \quad (2.1.1)$$

Una red neuronal no es más que la interconexión de varias neuronas artificiales, en la cual podemos identificar al menos tres secciones:

- **Capa de entrada**.- En esta capa se procesan todas las entradas, y si estas entradas son capaces de excitar las neuronas de esta capa se producirá una señal de salida.
- **Capa oculta**.- En esta capa se encuentran las neuronas encargadas del aprendizaje de la red.
- **Capa de salida**.- Esta neurona o neuronas de salida tendrán la salida del sistema.

La forma en que las redes aprenden es mediante la modificación de los pesos de las entradas.

Las redes neuronales artificiales han ido evolucionando con el paso del tiempo, hoy en día existen muchos modelos de redes neuronales, todas ellas con ventajas y desventajas si son comparadas entre ellas.

## Aprendizaje de las redes neuronales

El procedimiento utilizado para llevar a cabo el proceso de aprendizaje en una red neuronal se denomina entrenamiento.

El problema de aprendizaje en las redes neuronales se formula en términos de la minimización de la función de error (o pérdida) asociada.

Normalmente, esta función está compuesta por dos términos, uno que evalúa cómo se ajusta la salida de la red neuronal al conjunto de datos de que disponemos, y que se denomina término de error, y otro que se denomina término de regularización, y que se utiliza para evitar el sobreaprendizaje por medio del control de la complejidad efectiva de la red neuronal.

Por supuesto, el valor de la función de error depende por completo de los parámetros de la red neuronal: los pesos sinápticos entre neuronas, y los bias asociados a ellas, que, como suele ser ya habitual, se pueden agrupar adecuadamente en un único vector de peso de la dimensión adecuada, que denotaremos por  $w$ . En este sentido, podemos escribir  $f(w)$  para indicar que el valor del error que comete la red neuronal depende de los pesos asociados a la misma. Con esta formalización, nuestro objetivo es encontrar el valor  $w^*$  para el que se obtiene un mínimo global de la función  $f$ , convirtiendo el problema de aprendizaje en un problema de optimización.

En general, la función de error es una función no lineal, por lo que no disponemos de algoritmos sencillos y exactos para encontrar sus mínimos. En consecuencia, tendremos que hacer uso de una búsqueda a través del espacio de parámetros que, idealmente, se aproxime de forma iterada a un (error) mínimo de la red para los parámetros adecuados.

De esta forma, se comienza con una red neuronal con algún vector inicial de parámetros (a menudo elegido al azar), a continuación se genera un nuevo vector de parámetros, esperando que con ellos la función de error se reduzca (aunque dependiendo del método elegido, no es obligatorio, y temporalmente se puede admitir un empeoramiento del error siempre y cuando conduzca a una disminución posterior más acusada). Este proceso se repite, normalmente, hasta haber reducido el error bajo un umbral tolerable, o cuando se satisfaga una condición específica de parada.

El Descenso del Gradiente es el algoritmo de entrenamiento más simple y también el más extendido y conocido. Solo hace uso del vector gradiente, y por ello se dice que es un método de primer orden.

Este método para construir el punto  $w_{i+1}$  a partir de  $w_i$  se traslada este punto en la dirección de entrenamiento  $-g_i$ . Es decir:

$$w_{i+1} = w_i - g_i v_i \quad (2.1.2)$$

Donde el parámetro  $\eta$  se denomina tasa de entrenamiento, que puede fijarse a priori o calcularse mediante un proceso de optimización unidimensional a lo largo de la dirección de entrenamiento para cada uno de los pasos (aunque esta última opción es preferible, a menudo se usa un valor fijo,  $\eta_i = \eta$  con el fin de simplificar el proceso).

Aunque es muy sencillo, este algoritmo tiene el gran inconveniente de que, para funciones de error con estructuras con valles largos y estrechos, requiere muchas iteraciones. Se debe a que, aunque la dirección elegida es en la que la función de error disminuye más rápidamente, esto no significa que necesariamente produzca la convergencia más rápida.

Por ello, es el algoritmo recomendado cuando tenemos redes neuronales muy grandes, con muchos miles de parámetros, ya que sólo almacena el vector gradiente (de tamaño  $n$ ).

### 2.1.3. Deep Learning

### 2.1.4. Redes neuronales recurrentes

La idea detrás de las RNN es hacer uso de la información secuencial. En una red neuronal tradicional suponemos que todas las entradas (y salidas) son independientes entre sí. Pero para muchas tareas eso es una muy mala idea. Si quieres predecir la siguiente palabra en una oración, es mejor que conozcas qué palabras vienen antes. Las RNN se llaman recurrentes porque realizan la misma tarea para cada elemento de una secuencia, y la salida depende de los cálculos previos. Otra forma de pensar acerca de las RNN es que tienen una "memoria" que captura información sobre lo que se ha calculado hasta ahora. En teoría, los RNN pueden hacer uso de la información en secuencias arbitrariamente largas, pero en la práctica se limitan a mirar hacia atrás solo unos pocos pasos.

La decisión de una red recurrente alcanzada en el paso de tiempo  $t-1$  afecta la decisión que alcanzará un momento más tarde en el paso de tiempo  $t$ . Entonces, las redes recurrentes tienen dos fuentes de entrada, el presente y el pasado reciente, que se combinan para determinar cómo responden a los datos nuevos, de forma similar a como lo hacemos en la vida.

Esa información secuencial se conserva en el estado oculto de la red recurrente, que logra abarcar muchos pasos de tiempo a medida que avanza para afectar el procesamiento de cada nuevo ejemplo. Está encontrando correlaciones entre eventos

separados por muchos momentos, y estas correlaciones se llaman "dependencias a largo plazo", porque un evento en el tiempo depende de, y es una función de, uno o más eventos que vinieron antes. Una forma de pensar acerca de las RNN es esta: son una forma de compartir pesos a lo largo del tiempo.

Así como la memoria humana circula invisiblemente dentro de un cuerpo, afectando nuestro comportamiento sin revelar su forma completa, la información circula en los estados ocultos de las redes recurrentes.

Describiremos el proceso de llevar la memoria hacia adelante matemáticamente:

$$h_t = \phi(Wx_t + Uh_{t-1}) \quad (2.1.3)$$

El estado oculto en el paso de tiempo  $t$  es  $h_t$ . Es una función de la entrada al mismo tiempo paso  $x_t$ , modificada por una matriz de ponderación  $W$  (como la que usamos para las redes feedforward) agregada al estado oculto del paso de tiempo anterior  $h_{t-1}$  multiplicado por su propio estado oculto matriz  $U$  de estado oculto, también conocida como matriz de transición y similar a una cadena de Markov. Las matrices de peso son filtros que determinan la importancia de acuerdo tanto con la entrada actual como con el estado oculto pasado. El error que generan volverá a través de la propagación inversa y se usará para ajustar sus ponderaciones hasta que el error no pueda bajar más.

Debido a que este ciclo de retroalimentación ocurre en cada paso de la serie, cada estado oculto contiene rastros no solo del estado oculto anterior, sino también de todos los que precedieron a  $h_{t-1}$  mientras la memoria pueda persistir.

## LSTM

A mediados de los años 90, los investigadores alemanes Sepp Hochreiter y Juergen Schmidhuber propusieron una variación de la red recurrente con las denominadas unidades de memoria a largo plazo, o LSTM, como una solución al problema del gradiente de fuga.

Los LSTM ayudan a preservar el error que se puede volver a propagar a través del tiempo y las capas. Al mantener un error más constante, permiten que las redes recurrentes continúen aprendiendo durante muchos pasos de tiempo (más de 1000), abriendo así un canal para vincular causas y efectos de forma remota. Este es uno de los desafíos centrales para el aprendizaje automático y la IA, ya que los algoritmos se

enfrentan con frecuencia a entornos en los que las señales de recompensa son dispersas y diferidas, como la vida misma.

Los LSTM contienen información fuera del flujo normal de la red recurrente en una celda cerrada. La información puede almacenarse, escribirse o leerse desde una celda, al igual que los datos en la memoria de una computadora. La célula toma decisiones sobre qué almacenar y cuándo permitir las lecturas, escrituras y borraduras, a través de puertas que se abren y cierran. Sin embargo, a diferencia del almacenamiento digital en computadoras, estas puertas son análogas, implementadas con la multiplicación de elementos por sigmoides, que están todas en el rango de 0-1. Siendo Analógica tiene la ventaja sobre digital de ser diferenciable y, por lo tanto, adecuado para la propagación inversa.

Esas puertas actúan sobre las señales que reciben, y de forma similar a los nodos de la red neuronal, bloquean o transmiten información en función de su fuerza e importación, que filtran con sus propios conjuntos de ponderaciones. Esos pesos, como los pesos que modulan los estados de entrada y ocultos, se ajustan a través del proceso de aprendizaje de redes recurrentes. Es decir, las células aprenden cuándo permiten que los datos entren, salgan o se eliminen a través del proceso iterativo de hacer conjeturas, volver a propagar el error y ajustar los pesos mediante el descenso del gradiente.



---

Capítulo 3

# Desarrollo

---





---

Capítulo 4

# Conclusiones

---



## Bibliografía

---

