



# FACULTAD DE CIENCIAS INTELIGENCIA ARTIFICIAL

---

## Proyecto FINAL

---

Equipo: Skynet Scribes

Número de proyecto: 03

**Carlos Daniel Cortés Jiménez**  
420004846

**Sarah Sophía Olivares García**  
318360638

**Marco Silva Huerta**  
316205326

**Laura Itzel Tinoco Miguel**  
316020189

**Luis Enrique García Gómez**  
315063880

**Fernando Mendoza Eslava**  
319097690

Profesora: Cecilia Reyes Peña  
Ayudante: Tania Michelle Rubí Rojas

**Semestre 2024-2**

Fecha de entrega:  
**24 de Mayo del 2024**

# 1. Desarrollo del Proyecto

Nuestra propuesta se centra en el desarrollo de un sistema de inteligencia artificial capaz de generar poesía personalizada, reflejando los estilos, emociones y temas preferidos por cada usuario. Este sistema será una herramienta para aquellos que buscan explorar nuevas formas de arte literario y expandir su propia creatividad a través de la colaboración con la IA.

Queremos facilitar la creación de poesía mediante el uso de tecnología avanzada, ofreciendo a los usuarios la posibilidad de recibir poemas personalizados. El proyecto se enfocará en dos categorías principales de poesía, garantizando que el contenido generado sea relevante y de alta calidad.

- Crear un sistema de generación de poesía automática: Incorporar diferentes estilos literarios en la generación de poesía, asegurando variedad y riqueza en el contenido.
- Aplicar técnicas de procesamiento de lenguaje natural y aprendizaje automático: Modelar patrones poéticos y adaptar la generación de contenido a las preferencias del usuario, permitiendo una experiencia más personalizada.
- Comprender la importancia de un proceso correcto de datos para analizar todo el contenido en un poema y poder usarlo para que nuestro modelo aprenda y pueda generar el contenido nuevo.

## 1.1. Bases Teóricas

### Redes Neuronales

Una red neuronal es un modelo computacional inspirado en la estructura y funcionamiento del cerebro humano. Está compuesta por nodos interconectados llamados neuronas artificiales, organizados en capas que procesan y transmiten información. Cada neurona recibe entradas, las procesa mediante una función de activación y luego transmite la salida a las neuronas de la capa siguiente. Esto permite que las redes neuronales aprendan patrones complejos y realicen tareas como reconocimiento de patrones, clasificación de datos y predicción.

- Las redes neuronales utilizan una técnica de aprendizaje automático llamada aprendizaje automático supervisado para adaptarse y aprender nuevas tareas.
- Pueden representar información proporcionada durante la etapa de aprendizaje.
- Dado que las redes neuronales pueden calcular datos en paralelo, las máquinas con arquitectura de redes neuronales trabajan más rápido para proporcionar resultados
- Si una red neuronal está parcialmente dañada, puede provocar una caída en el rendimiento; sin embargo, la red neuronal puede conservar algunas de sus propiedades incluso cuando está dañada

### Ajuste de hiperparámetros y selección de modelo

Imaginemos que vamos a jugar un torneo de cartas, tenemos dos equipos para elegir jugar:

1. Equipo  $\alpha$  conformado por jugadores con experiencia en las cartas
2. Equipo  $\beta$  conformado por jugadores más novatos pero con habilidades en matemáticas

Con que equipo nos quedamos? Primero, vamos a probar ambos equipos en diferentes partidas y evaluaremos cómo se desempeñan en términos de estrategia y habilidades matemáticas. Tras varias partidas, notas que el equipo de  $\alpha$  tiene una ventaja clara en la toma de decisiones durante el juego, pero el equipo  $\beta$  tiene una mejor comprensión de las probabilidades y las estadísticas del juego.

Aquí es donde entra el ajuste de hiperparámetros. Vamos a ajustar la estrategia del equipo  $\alpha$  para que sea más cauteloso y evite riesgos innecesarios. Pero al equipo  $\beta$  le haremos un ajuste para equilibrar su enfoque analítico y hacer decisiones más intuitivas y rápidas.

Después de ajustar ambos equipos, y hacer pruebas, notamos como el equipo  $\alpha$  sigue siendo sólido, pero  $\beta$  ha mejorado su capacidad para adaptarse a situaciones cambiantes durante el juego.

Aquí es donde entra la validación de exclusión y la validación cruzada. Guardamos algunas partidas para evaluar nuevamente a ambos equipos después de los ajustes. Al hacer esto, podemos asegurar de que los cambios realizados si mejoran el rendimiento de los equipos y no son temporales o un golpe de suerte en una única prueba.

Finalmente la elección del equipo deberá estar basada en quien ha demostrado una mejora constante y sólida en todas las evaluaciones, solo así podremos estar seguros de jugar el torneo.

## Mecanismo de las Redes Neuronales Recurrentes (RNNs)

Las redes neuronales recurrentes (RNNs) son un tipo especial de red neuronal que se utiliza principalmente para datos secuenciales, como texto o series temporales. Lo que las hace especiales es su capacidad para mantener información a lo largo del tiempo, lo que les permite aprender patrones en secuencias de datos.

### Componentes de una RNN

**Memoria y Bucles:** Las RNNs tienen una memoria interna que les permite recordar la información anterior en la secuencia. Esto se logra mediante la conexión en bucle dentro de las neuronas, lo que permite que la información persista.

**Capa Recurrente:** Esta es la capa central de una RNN. A diferencia de las redes neuronales tradicionales, cada neurona en esta capa no solo recibe la entrada actual, sino que también recibe información de la entrada anterior. Esto se logra mediante una conexión recurrente que permite a la red recordar información a lo largo del tiempo.

Como ejemplo imaginemos que vamos a leer una historia. Cada palabra que lees se añade a tu comprensión de la trama. De manera similar, una RNN procesa una palabra a la vez, y cada nueva palabra se añade a su *memoria* de las palabras anteriores.

### Proceso de una RNN

1. **Entrada Secuencial:** La RNN toma una secuencia de datos como entrada. Por ejemplo, una oración dividida en palabras.
2. **Propagación de la Información:** A medida que cada palabra se procesa, la RNN actualiza su estado interno (su "memoria") basado en la palabra actual y el estado anterior.
3. **Salida Secuencial:** La RNN puede generar una salida en cada paso, por ejemplo, predecir la siguiente palabra en una oración.

## 1.2. Estrategia del proyecto

1. Recopilación de Poemas: Juntamos una serie de poemas en dos categorías: amor y tristeza. Son dos porque nos permite enfocarnos en como debe ser el procesamiento y extraer las mejores características para entrenar el modelo. Estos datos han sido almacenados en un archivo .json.
2. Pre-procesamiento de Datos: Debemos hacer una limpieza de los datos
3. Desarrollo de Modelos
  - El primer modelo lo queremos entrenar para que pueda aprender y clasificar nuestros dos categorías de poemas utilizando técnicas de procesamiento de lenguaje natural y aprendizaje automático
  - El segundo modelo es para decirle al algoritmo como se conforma un poema y como esperamos que nos devuelva el resultado
  - Finalmente el tercer modelo será entrenando para que pueda tomar lo aprendido de los dos modelos de arriba y pueda construir un poema nuevo. Hasta el momento no hemos considerado más variables que pueda influir en la creación de modelo pero es muy posible que se integren.
4. Personalización: Implementación de funcionalidades que permitan a los usuarios influir en el estilo y contenido de la poesía generada, adaptándose a sus preferencias y emociones.
5. Evaluación y Mejora: Desarrollo de métricas para evaluar la calidad poética y realizar ajustes basados en la retroalimentación de los usuarios.

## 2. Limpieza de datos

## Referencias

- [1] A. Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media, 2022, cap. 16, ISBN: 9781098122461. dirección: <https://books.google.com.mx/books?id=X5ySEAAAQBAJ>.
- [2] L. Tunstall, L. von Werra y T. Wolf, *Natural Language Processing with Transformers*. O'Reilly Media, 2022, ISBN: 9781098103194. dirección: <https://books.google.com.mx/books?id=nTxbEAAAQBAJ>.
- [3] A. Müller y S. Guido, *Introduction to Machine Learning with Python: A Guide for Data Scientists*. O'Reilly Media, 2016, cap. 7, ISBN: 9781449369903. dirección: <https://books.google.com.mx/books?id=1-4lDQAAQBAJ>.
- [4] S. Bird, E. Klein y E. Loper, *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly Media, 2009, ISBN: 9780596555719. dirección: <https://books.google.com.mx/books?id=KGibfiiPli4C>.