

Proyecto - Avance 1

Semana: 07

Nombre del estudiante:

#61811583

Jasir Esaud Reyes Figueroa

#62211223

Jose Antonio Rodriguez García

#62111580

Marcio Moran Velasquez

Sede de estudio:

Sede Central

Docente:

Ing. Kevin Ivan Cruz Medina

Sección:

903

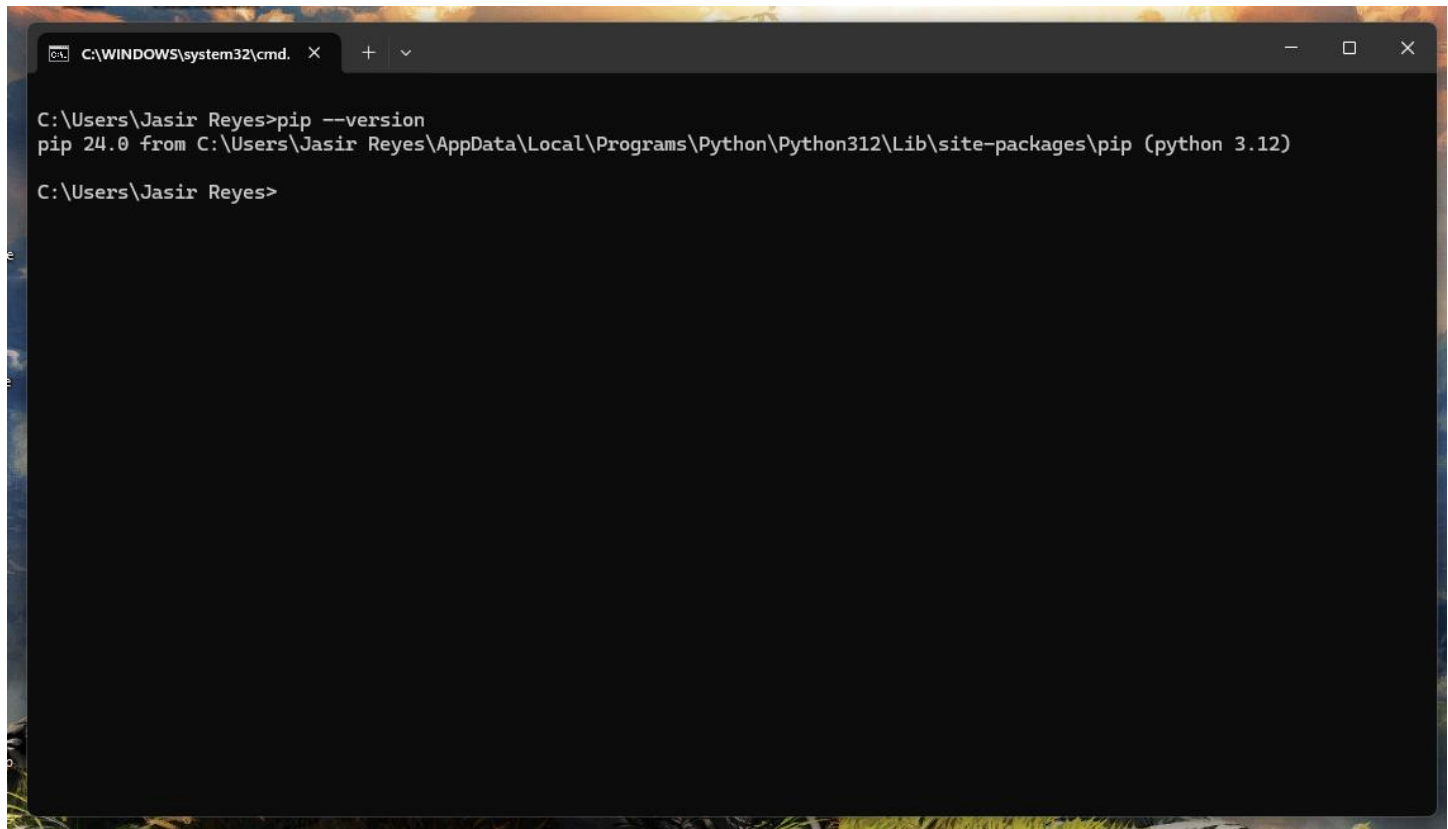
Fecha de entrega:

02/06/2024

¿Qué es pip en Python?

Pip es un sistema de gestión de paquetes utilizado para instalar y administrar paquetes de software escritos en Python. Desde Python 3.4 viene incluido por defecto con los instaladores binarios de Python. Un entorno virtual es un entorno de Python parcialmente aislado que permite instalar paquetes para que los use una aplicación en particular, en lugar de instalarlos en todo el sistema.

Prueba de instalación de pip y versión en consola:

A screenshot of a Windows Command Prompt window. The title bar shows 'C:\WINDOWS\system32\cmd.' and standard window controls. The command prompt shows the command 'C:\Users\Jasir Reyes>pip --version' and the output 'pip 24.0 from C:\Users\Jasir Reyes\AppData\Local\Programs\Python\Python312\Lib\site-packages\pip (python 3.12)'. The prompt then returns to 'C:\Users\Jasir Reyes>'.

```
C:\Users\Jasir Reyes>pip --version
pip 24.0 from C:\Users\Jasir Reyes\AppData\Local\Programs\Python\Python312\Lib\site-packages\pip (python 3.12)
C:\Users\Jasir Reyes>
```

Librerías de Python

NumPy

NumPy es una biblioteca fundamental para el cálculo científico en Python. Ofrece soporte para arreglos multidimensionales y matrices, así como una colección de funciones matemáticas de alto nivel para operar con estos arreglos. NumPy se utiliza ampliamente en la computación científica y la inteligencia artificial por su eficiencia y facilidad de uso.

Proyectos y Contenido:

- Documentación y Tutoriales: NumPy proporciona una guía completa y tutoriales básicos y avanzados para aprender sobre el uso de la biblioteca. Esto incluye la creación de arrays, manipulación de matrices, y operaciones matemáticas avanzadas.
- Interoperabilidad: NumPy permite la interoperabilidad con otras bibliotecas como Pandas y TensorFlow. Proporciona métodos para convertir datos entre estas bibliotecas sin necesidad de copias adicionales, lo que es crucial para aplicaciones de aprendizaje profundo y procesamiento de datos.

Cursos y Certificados:

- Cursos en línea: Plataformas como Coursera y edX ofrecen cursos específicos de NumPy que forman parte de programas más amplios de ciencia de datos e inteligencia artificial.

Lineamientos y Laboratorios:

- Laboratorios Virtuales: NumPy se usa en numerosos laboratorios virtuales y entornos de notebooks Jupyter, lo cual facilita la experimentación y el aprendizaje interactivo.
- Mejores Prácticas: La documentación oficial de NumPy incluye lineamientos sobre el uso eficiente de arrays y funciones para mejorar el rendimiento de los cálculos.

Pandas

Pandas es una biblioteca de Python para la manipulación y el análisis de datos. Proporciona estructuras de datos y funciones diseñadas para trabajar con datos etiquetados o relacionales de manera intuitiva.

Proyectos y Contenido:

- Documentación Oficial: La documentación de Pandas es extensa y cubre desde conceptos básicos hasta técnicas avanzadas para el manejo de datos.
- Tutoriales y Ejemplos: Incluye ejemplos prácticos de cómo utilizar DataFrames, realizar operaciones de agrupamiento, y limpiar datos, lo cual es vital en proyectos de inteligencia artificial y ciencia de datos.

Cursos y Certificados:

- Capacitaciones en Línea: Pandas se enseña en muchos cursos de ciencia de datos y análisis de datos en plataformas como DataCamp, Coursera y edX, los cuales ofrecen certificados al completar los cursos.

Lineamientos y Laboratorios:

- Entornos Interactivos: Pandas se utiliza en combinación con Jupyter Notebooks para crear entornos de aprendizaje interactivos que permiten experimentar con conjuntos de datos en tiempo real.
- Best Practices: La documentación incluye guías sobre cómo manejar grandes volúmenes de datos y optimizar el rendimiento de las operaciones de manipulación de datos.

TensorFlow

TensorFlow es una biblioteca de código abierto desarrollada por Google para el aprendizaje automático. Permite a los desarrolladores construir y entrenar modelos de aprendizaje profundo.

Proyectos y Contenido:

- Modelos Pre-entrenados: TensorFlow Hub ofrece una colección de modelos pre-entrenados que se pueden usar y adaptar para diferentes tareas de inteligencia artificial.
- Documentación y Tutoriales: TensorFlow proporciona una extensa documentación y tutoriales que cubren desde la construcción de modelos básicos hasta técnicas avanzadas de entrenamiento y despliegue de modelos.

Cursos y Certificados:

- TensorFlow Developer Certificate: Google ofrece un certificado oficial de desarrollador de TensorFlow, que demuestra habilidades en el uso de esta biblioteca.
- Cursos en línea: Existen numerosos cursos en plataformas como Coursera, donde se puede aprender desde los conceptos básicos hasta el desarrollo de aplicaciones avanzadas de IA usando TensorFlow.

Lineamientos y Laboratorios:

- TensorFlow Playground: Un laboratorio interactivo que permite a los usuarios experimentar con redes neuronales sin necesidad de escribir código.
- Guías de Mejores Prácticas: TensorFlow ofrece guías detalladas sobre cómo optimizar el rendimiento de los modelos, gestionar el hardware, y asegurar la reproducibilidad de los experimentos.

- [NumPy](<https://numpy.org/>)
- [Pandas](<https://pandas.pydata.org/>)
- [TensorFlow](<https://www.tensorflow.org/>)

Scikit-Learn

Descripción: Scikit-Learn es una biblioteca de Python para aprendizaje automático que proporciona herramientas simples y eficientes para la minería de datos y el análisis de datos. Construida sobre Numpy, Scipy y Matplotlib, ofrece una amplia gama de algoritmos de clasificación, regresión, clustering y reducción de dimensionalidad.

Aplicaciones: Ideal para tareas de machine learning como clasificación de textos, predicción de datos numéricos, y clustering. Es ampliamente utilizada para la creación rápida de modelos de machine learning.

Documentación oficial: [Scikit-Learn Documentation](#)

Keras

Descripción: Keras es una biblioteca de alto nivel para la construcción y entrenamiento de modelos de redes neuronales profundas. Se integra perfectamente con TensorFlow, y proporciona una API simple y consistente para construir modelos complejos de deep learning.

Aplicaciones: Es utilizada para construir, compilar y entrenar redes neuronales para tareas como clasificación de imágenes, procesamiento de texto, y análisis de series temporales.

Documentación oficial: [Keras Documentation](#)

Matplotlib

Descripción: Matplotlib es una biblioteca completa para la creación de visualizaciones estáticas, animadas e interactivas en Python. Es altamente customizable y se puede utilizar para crear una amplia variedad de gráficos, desde simples gráficos de líneas hasta complejas visualizaciones tridimensionales.

Aplicaciones: Es ideal para crear gráficos de datos, visualizaciones científicas y visualizaciones interactivas que permiten la exploración detallada de los datos.

Documentación oficial: [Matplotlib Documentation](#)

Propuesta de proyecto

IA que predice resultados de un partido (usando la librería de TensorFlow)

Usaremos la librería de TensorFlow puesto que trabaja con redes neuronales lo cual nos permite hacer uso del machine learning y así crear un proyecto capaz de predecir partidos con la información que se le ha brindado. Ya se ha demostrado en otros proyectos que el uso de redes neuronales nos permite a clasificar información que se ha brindado cómo, por ejemplo: clasificar correos SPAM, determinar el valor de una casa, clasificar imágenes, cálculo de tiempos de x acción, predecir acciones en el mercado, mostrar anuncios, etc. Por lo que un proyecto basado en la predicción del resultado de un partido puede ser muy viable ya que disponemos de mucha información como por ejemplo: resultado de partidos en años anteriores, estadísticas de cada jugador que ayudan a determinar la estadística del equipo, faltas, tarjetas, fuera de juego, penales, tiros de esquina, tiros libres, tiros al arco, asistencias, goles, etc. Algunas ventajas de usar TensorFlow:

1. Capacidad para Manejar Datos Complejos y Voluminosos

TensorFlow es capaz de manejar grandes volúmenes de datos y procesar información compleja. En el contexto del fútbol, esto incluye estadísticas de jugadores, historial de partidos, condiciones del partido y muchas otras variables. TensorFlow facilita la creación de modelos que pueden procesar y aprender de estos datos de manera eficiente.

2. Modelos de Aprendizaje Automático y Redes Neuronales

TensorFlow es especialmente adecuado para construir y entrenar redes neuronales, que son esenciales para problemas de clasificación y predicción como este. Utilizando redes neuronales profundas, el modelo puede capturar patrones complejos en los datos que serían difíciles de detectar con métodos más simples. Esto aumenta la precisión y la fiabilidad de las predicciones.

3. Flexibilidad y Escalabilidad

TensorFlow es una biblioteca muy flexible que permite a los desarrolladores personalizar modelos según las necesidades específicas del proyecto. Además, está diseñada para ser escalable, lo que significa que los modelos pueden ser entrenados y ejecutados en diferentes plataformas, desde dispositivos móviles hasta clusters de servidores en la nube.

4. Soporte para Implementación y Producción

TensorFlow tiene un fuerte soporte para la implementación de modelos en producción. Con TensorFlow Serving, se pueden desplegar modelos entrenados de manera eficiente y escalable, permitiendo que el sistema de predicción de resultados de partidos de fútbol se integre fácilmente en aplicaciones web o móviles.

5. Ecosistema y Comunidad Amplia

TensorFlow cuenta con una extensa comunidad de usuarios y desarrolladores, así como una abundancia de recursos, tutoriales y ejemplos. Esto facilita la resolución de problemas y la mejora continua del modelo. Además, el ecosistema de TensorFlow incluye herramientas complementarias como TensorBoard para la visualización del entrenamiento y TFX para la implementación en producción.

6. Optimización y Rendimiento

TensorFlow está optimizado para rendimiento, utilizando operaciones de bajo nivel altamente eficientes y soporte para aceleración de hardware como GPUs y TPUs. Esto es crucial para entrenar modelos de aprendizaje profundo de manera rápida y eficiente, especialmente cuando se trabaja con grandes conjuntos de datos como los utilizados en el análisis de partidos de fútbol.

Bibliografía

https://www.udemy.com/course/tensorflow-quickstart-for-beginners/?utm_source=adwords&utm_medium=udemyads&utm_campaign=DSA_Catchall_la.EN_c.c.ROW&campaigntype=Search&portfolio=ROW-English&language=EN&product=Course&test=&audience=DSA&topic=&priority=&utm_content=deal4584&utm_term=.ag_88010211481.ad_535397282064.kw.de.c.dm.pl.ti.dsa-841699839303.li_9069970.pd.&matchtype=&gad_source=1&gclid=Cj0KCQjwsPCyBhD4ARIsAPaaRf3wP1MFW2VN37pHoz8OoCoj3Rnn3LZBjPDNJmYj-Y11S2G7WDt0Um0aAmYIEALw_wcB&couponCode=2021PM25

<https://www.tensorflow.org/tfx?hl=es-419>

https://www.tensorflow.org/api_docs

<https://blog.tensorflow.org/2024/02/graph-neural-networks-in-tensorflow.html?hl=es-419>