Autor: Luis Eduardo Muñoz Guerrero

El arte de generar imágenes: Una introducción a Stable Diffusion

ISBN: 978-628-95805-4-9
Primera edición
Editado en Colombia ©
Septiembre 2023

Autor: Luis Eduardo Muñoz Guerrero

El Arte de Generar Imágenes: Una Introducción a Stable Diffusion

Autor:

Luis Eduardo Muñoz Guerrero

AGRADECIMIENTO:

"Dedico este libro a mi hermana Sonia y su preciada hija María José, quienes juntas son poderosa fuerza que me permiten dar vida a mis ilusiones"

PÁGINA LEGAL

Título : El arte de generar imágenes: una introducción a Stable Diffusion

ISBN obra independiente: 978-628-95805-4-9

Sello editorial: Corporación Centro Internacional de Marketing Territorial para la Educación y el Desarrollo (978-628-95805)

Tipo de Contenido: Ciencia y tecnología

Clasificación THEMA: Aplicaciones prácticas de tecnología de la información - Para enseñanza superior / educación universitaria

Público objetivo: Enseñanza universitaria o superior

Idioma: Español

Número de edición: 1

Ciudad de Edición: Medellín

Tipo de soporte: Libro digital descargable

Formato: Epub (.epub)

Tipo de contenido: Texto (legible a simple vista)

Tipos de acceso: Digital: descarga y online

Editorial: Corporación Centro Internacional de Marketing Territorial para la Educación y el Desarrollo

Número de identificación tributaria o de ciudadanía : 8110433950

Representante legal: Roger Loaiza Alvarez

Autor: Luis Eduardo Muñoz Guerrero

Correspondencia del autor: lemunozg@utp.edu.co

Tipo de soporte: Libro digital descargable

Formato: Epub (.epub)

Tipo de contenido: Texto (legible a simple vista)

Página de descarga:
www.editorialcimted.com

Editor: Corporación Centro Internacional de Marketing Territorial para la Educación y el Desarrollo. Corporación CIMTED Nit:811043395-0 editorialcimted@gmail.com Cuidado de la Edición: Juliana Escobar Gómez -Calle 41 No 80B 120 Medellín- Colombia www.memoriascimted.com - www.editorialcimted.com

ISBN: 978-628-95805-4-9

Primera Edición Septiembre 2023©

Depósito digital:

SOBRE EL AUTOR:

Luis Eduardo Muñoz Guerrero

Universidad Tecnológica de Pereira Colombia

Ingeniero de sistemas, Magister en Ingeniera de Sistemas por la Universidad Nacional de Colombia, PhD en ciencias de la educación RudeColombia Cade UTP. Su experiencia de trabajo ha girado, principalmente, alrededor del campo educativo, sus proyectos están asociados con áreas de evaluación educativa, educación basada en competencias, software educativo, enseñanza de la programación. Ha publicado artículos en revistas nacionales e internaciones. Autor de los libros; Programación Moderna con aplicaciones y Programación Funcional con Racket. Actualmente es profesor titular de tiempo completo programa de Ingeniería de Sistemas y Computación de la Universidad Tecnológica de Pereira y Pertenece al grupo de investigación informática.

Correspondencia: lemunozg@utp.edu.co

TABLA DE CONTENIDO

TABLA DE ILUSTRACIONES

INTRODUCCIÓN

Hoy día, existen numerosos modelos de inteligencia artificial (IA) cada vez más sofisticados. Estos modelos de IA se clasifican en diversas categorías, tales como los diseñados para automatizar tareas, generar contenido, revisar contenido, entre otros. Aunque pueda parecer lo contrario, el funcionamiento de estos modelos no es una novedad. De hecho, podemos afirmar que los modelos de IA forman parte intrínseca de nuestra vida cotidiana, a menudo sin que seamos conscientes de ello.

Antiguamente, la IA era considerada una fantasía de Hollywood o una tecnología exclusiva para empresas con presupuestos astronómicos. Sin embargo, hoy en día la situación es distinta: los modelos de IA son tan accesibles que cualquier persona puede utilizarlos y beneficiarse de sus diversas capacidades a través de un simple dispositivo móvil o una computadora personal.

Este cambio se debe a varios factores. En primer lugar, la continua evolución y desarrollo de los modelos de IA, fruto de avances e innovaciones significativas en el campo. En segundo lugar, el progreso en hardware capaz de manejar estos modelos de IA de manera eficiente.

Con estos avances, se han desarrollado muchos modelos gratuitos y de código abierto para el público, democratizando el acceso a estas nuevas tecnologías. Así, la IA ha dejado de ser un concepto limitado a películas de ciencia ficción o a compañías de alto poder económico, para convertirse en una herramienta al alcance de todos.

El propósito principal de este libro es introducir uno de los modelos de inteligencia artificial que ha causado gran revuelo a nivel mundial. Este interés no se debe únicamente a la capacidad que ofrece este modelo generativo de inteligencia artificial, sino también a la innovación significativa que representa en comparación con el estado actual del arte. Los avances más significativos en inteligencia artificial se centran en este modelo y sus variantes, así como en otros similares. A fin de cuentas, estamos hablando de una tecnología relativamente nueva, muy prometedora y, aunque todavía está en desarrollo, la realidad es que las expectativas sobre los resultados que este tipo de modelos pueden generar son muy altas. Esto es especialmente relevante si consideramos las capacidades actuales de estos modelos, las cuales son sorprendentes y atraen cada vez a más personas.

El modelo que se discute en este libro es Stable Diffusion, un modelo generativo de inteligencia artificial para imágenes desarrollado por una alianza de varios grupos de investigación, pero liderado principalmente por StabilityAI. Este modelo de inteligencia artificial permite generar imágenes de alta calidad y resultados

finales impresionantes, utilizando descripciones textuales o algunas herramientas visuales para crear imágenes completamente nuevas, creativas, de alta calidad y con detalles sorprendentes. Es como si contáramos con un artista profesional capaz de generar casi cualquier cosa que podamos imaginar en muy poco tiempo. Verdaderamente es increíble.

Hay mucho por aprender, y este libro no está necesariamente centrado en explicar desde un punto de vista demasiado profesional o especializado las diferentes capacidades de este modelo de inteligencia artificial. Más bien, el propósito general es servir como guía para

aquellos que quieran entender y aprender sobre este modelo. Además, no solo tendremos la oportunidad de depender de empresas que ofrezcan este modelo y conocer sus diferentes limitaciones, sino que también podremos usar una versión local y personalizada de este modelo utilizando Google Colab. Esto nos permitirá adaptarlo a nuestro gusto y programar lo que específicamente necesitemos.

¿PARA QUIÉN VA DIRIGIDO ESTE LIBRO?

Aunque el uso de un modelo de inteligencia artificial pueda parecer un tema intimidante, en realidad no es tan complejo como se podría pensar. Esto es especialmente cierto si se utilizan modelos que ya han sido optimizados a través de un sólido proceso de abstracción. Comprender cómo aplicar el Stable Diffusion para diversos propósitos tampoco resulta excesivamente complicado.

Este libro, aunque contiene información que podría ser considerada técnica y avanzada para algunos, ha sido diseñado para explicar los temas de la manera más clara posible. De esta forma, se asegura que un público amplio pueda entender su contenido. A diferencia de otros textos introductorios, este puede que no sea totalmente apropiado para lectores muy jóvenes, ya que gran parte de su contenido se centra en una perspectiva abstracta y práctica del uso de modelos de inteligencia artificial.

Hacia el final del libro, se presentan ejemplos y se discuten capacidades del modelo desde un punto de vista avanzado. Este análisis abarca tópicos no tan comunes, como el uso de repositorios, programación en Python, técnicas de machine learning y librerías de machine learning. Aunque esto pueda parecer un obstáculo para muchos, los temas son explicados de manera comprensible para resolver cualquier posible duda.

El propósito final del libro es mostrar las capacidades de este modelo de inteligencia artificial y las distintas tareas que se pueden realizar con él. No hay que preocuparse si algunos temas parecen confusos inicialmente, ya que estos modelos son fácilmente entendibles y cuentan con una documentación excelente. Aprender de este libro puede ser una experiencia gratificante, aunque posiblemente cansada para algunos. Sin embargo, cualquier persona, sin importar su formación académica, puede entender este libro.

Los temas se tratan a diferentes niveles de complejidad, empezando por usos básicos de sitios web, pasando por el funcionamiento de parámetros internos y explicaciones de técnicas de modelado, hasta la ejecución de estos tópicos en un entorno de programación. A pesar de esto, no se requiere un conocimiento profundo en inteligencia artificial para entender el libro. De hecho, una persona con conocimientos básicos de programación podría comprenderlo.

Dada la popularidad y el creciente número de personas con conocimientos de programación, el público general de este libro es bastante amplio. Algunos elementos pueden parecer triviales para aquellos con conocimientos previos sobre modelos de inteligencia artificial. No obstante, el propósito de este libro es introducir a los neófitos en este campo, proporcionándoles una base inicial para despertar su interés en aprender más.

Por lo tanto, el público objetivo de este libro no se limita a personas con especializaciones y un extenso conocimiento académico en inteligencia artificial o en el uso de modelos generativos de inteligencia artificial. Sin embargo, a pesar de la formación académica previa, este libro puede resultar bastante enriquecedor para muchos lectores.

CAPÍTULO 1: LA INTELIGENCIA ARTIFICIAL EN LA COTIDIANIDAD

Objetivos del Capítulo:

El propósito de este capítulo es ilustrar la importancia y la omnipresencia de la inteligencia artificial (IA) en nuestra vida cotidiana. La meta es exponer cómo la IA, a través de diferentes modelos y aplicaciones, ha transformado diversos sectores como el entretenimiento y la agricultura. Se busca también aclarar las diferencias entre conceptos similares como la IA, aprendizaje de máquina, y redes neuronales. A través de ejemplos detallados y análisis de caso, se pretende proporcionar una comprensión completa de cómo la inteligencia artificial se ha convertido en una herramienta esencial en la modernidad. Se introduce además el modelo Stable Diffusion, resaltando su relevancia en la generación de imágenes y su aplicación en múltiples campos.

El mundo ha cambiado drásticamente desde la aparición de la inteligencia artificial en nuestras vidas. S in darnos cuenta, modelos de inteligencia artificial son los principales intermediarios para realizar una amplia gama de tareas en muchos ámbitos y aspectos cotidianos. En contraste a la idea errónea de que la inteligencia artificial eventualmente dominará el mundo y controlará la vida de los seres humanos, hemos entendido que la inteligencia artificial tiene muchos más aspectos positivos de lo que creemos. Esto no es novedoso para muchos, pues como se mencionó anteriormente los modelos de inteligencia artificial han sido partícipes de demasiados aspectos de nuestra vida cotidiana sin que nos demos cuenta de que esto ocurre enfrente de nuestros ojos.

Grandes compañías y entidades usan tecnologías, algoritmos, herramientas y sistemas basados en inteligencia artificial, aprendizaje de máquina y redes neuronales; esto mismo debido a la facilidad y eficacia que estos ofrecen al automatizar tareas sin la necesidad de trabajo humano, el cual ocasionalmente puede llegar a cometer errores.

Desde las recomendaciones que recibimos en nuestra página principal de YouTube, hasta la lista de reproducción personalizada de Spotify y muchas otras aplicaciones y sitios web, son ejemplos de cómo se utiliza la inteligencia artificial para predecir y sugerir nuevo contenido en base a las diferentes interacciones y gustos personales de cada usuario.

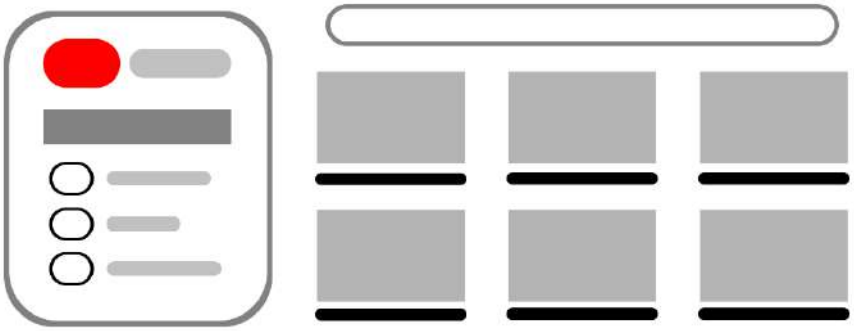


Ilustración 1 – Autores afirman que el algoritmo de YouTube está basado en Inteligencia Artificial. (Tufekci, 2018) (Bryant, 2020)

El uso de modelos de inteligencia artificial para automatizar tareas no es una capacidad exclusiva de las grandes empresas. Actualmente, cualquier persona que cuente con una computadora e internet tiene acceso a poderosos modelos de inteligencia artificial. En este libro nos centraremos en uno de estos modelos, recibiendo el nombre de Stable Diffusion.

Este modelo de inteligencia artificial ha ganado reconocimiento en el mundo académico, científico e informático. Su capacidad para generar imágenes y gráficos con resultados sorprendentes ha dejado sorprendido a muchos entusiastas del tema.

En esta primera sección exploraremos diversos ejemplos, situaciones y casos hipotéticos para comprender de manera explícita cómo la inteligencia artificial está presente en nuestra vida cotidiana, así como la importancia que tienen estos modelos en nuestro día a día.

La inteligencia artificial en el sector del entretenimiento

En la sección anterior hablamos de cómo la inteligencia artificial está presente en diversas aplicaciones y servicios que utilizamos día a día, incluyendo aquellas relacionadas con el entretenimiento.

Aunque a simple vista no es evidente cómo la IA está presente en estas plataformas, lo cierto es que su presencia es cada vez más común y relevante en este sector, tomemos como ejemplo a YouTube y Spotify.

En el caso de YouTube, la IA se utiliza para recomendar contenido relevante a los usuarios, en base a sus preferencias y patrones personales de la actividad en la plataforma.

Spotify, por otro lado, utiliza la IA para crear listas de reproducción personalizadas y sugerir canciones similares a las que el usuario ha escuchado previamente; estos dos son solo algunos ejemplos de cómo la IA está presente en el mundo del entretenimiento.

Considerando esto, ¿qué otro uso tiene la inteligencia artificial en este sector? Aunque no lo parezca, el uso de la IA es mucho más numeroso y útil que solo estos dos ejemplos.

Por ejemplo, la IA se utiliza como herramienta en la creación de videojuegos, generando entornos y personajes más auténticos, encargándose también de aspectos importantes como la optimización del rendimiento computacional y la experiencia de juego; sin mencionar que una lista extensa de herramientas basadas en inteligencia artificial se usan en la producción de películas y series televisivas, mejorando efectos visuales y creando mundos virtuales bastante detallados.

Un ejemplo de esto es el modelo de inteligencia NeRF (Neural Radiance Field), el cual consiste en un sistema de inteligencia artificial capaz de transformar un conjunto de fotografías 2D relacionadas en un espacio 3D sumamente realista y preciso en cuanto a pequeños detalles e iluminación (Stephens, 2022).

El uso de esta herramienta en la producción de recursos, tanto en la industria de videojuegos como en el renderizado de elementos en la industria del cine, es objeto de investigación y grandes expectativas, pues la aplicación de este tipo de herramientas puede aumentar significativamente la velocidad de producción y eficiencia en el trabajo.

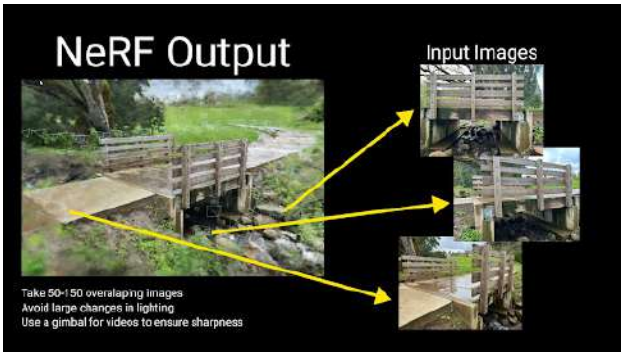


Ilustración 2 – Imagen representativa NeRF (Stephens, 2022)

En esta imagen, podemos observar a la derecha algunas de las fotografías que se introdujeron. A partir de estas imágenes, el modelo generó el espacio 3D que se muestra en el lado izquierdo de la ilustración, con tan solo unas pocas

imágenes es posible generar un espacio tridimensional hiperrealista usando esta herramienta.

En definitiva, la inteligencia artificial tiene un papel cada vez más importante en el sector del entretenimiento, además, se espera que el uso de estas herramientas siga en aumento los próximos años. Los modelos de IA utilizados en este sector, así como las tareas que desempeñan, son muy diversos y van desde algoritmos de recomendación de contenido, hasta sistemas de procesamiento de lenguaje natural y reconocimiento de elementos multimedia.

El impacto de estas herramientas tanto en la experiencia de usuario, así como la calidad del contenido que se produce es cada vez más evidente y no podemos dejarlo pasar de lado. Es más fácil entender estos conceptos analizando un ejemplo de cómo se aplica la inteligencia artificial en plataformas desarrolladas por grandes empresas, como es el caso de Netflix.

Caso ejemplar de estudio: *Netflix*

Ahora bien, es posible que la mayoría de nosotros ya conozcamos qué es Netflix. No obstante, vale la pena proporcionar una breve introducción sobre esta plataforma, asumiendo que algunos de los lectores quizás no estén familiarizados con ella. Netflix, según su sitio web, es uno de los servicios de transmisión en línea más destacados en la industria, ofreciendo una amplia variedad de películas, series y documentales; todo esto en prácticamente cualquier dispositivo con conexión a internet (Netflix, Inc, 2023). Esta plataforma mostró una ganancia de más de 30 mil millones de dólares en 2022, lo que convierte a su compañía en una de las más importantes y poderosas en servicios multimedia a nivel global (Forbes Media LLC, 2023).

Aunque no lo parezca, los modelos de inteligencia artificial tienen un papel fundamental en el funcionamiento y continuidad de esta empresa, pues los modelos que usan son los principales responsables de las sugerencias del nuevo contenido que se muestran a sus suscriptores. Netflix utiliza un algoritmo de recomendación de contenido conocido bajo el nombre de “Netflix Recommendation Engine” (NRE), o "motor de recomendaciones de Netflix" en español (Invisibly, 2023).

Este algoritmo de inteligencia artificial ha sido entrenado usando una enorme cantidad de datos basados en el comportamiento anónimo de los usuarios en la plataforma, creando así un sistema de recomendaciones bastante preciso, eficiente y flexible.

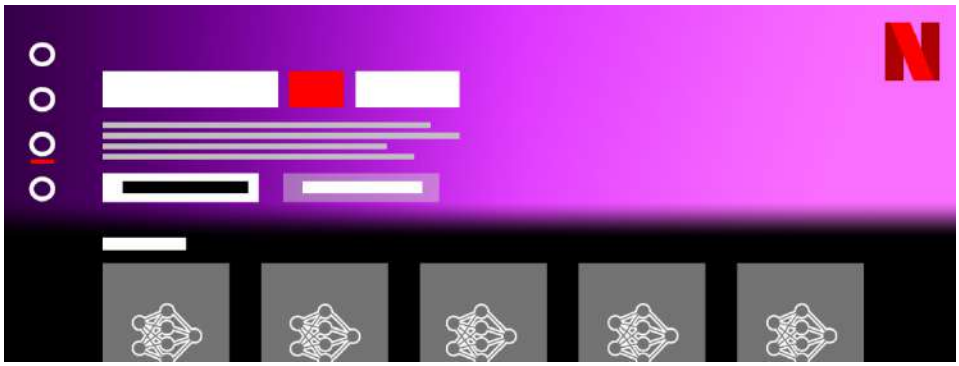


Ilustración 3 - Las recomendaciones de Netflix están basadas en la interacción del usuario con la plataforma.

Antes de que Netflix usara el algoritmo NRE como el principal sistema de recomendación de contenido, la plataforma hacía uso de un algoritmo basado en inteligencia artificial conocido como CineMatch, este algoritmo fue el predecesor del NRE, y a pesar de que haya sido reemplazado, CineMatch brindaba a las recomendaciones de Netflix una muy alta tasa de acierto (M, Srinivasulu, P, & B, 2020).

Un evento importante relacionado al algoritmo CineMatch de Netflix fue el evento que tuvo lugar en octubre de 2006 bajo el nombre de "The Netflix Prize".

En este evento, Netflix ofrecía una muy alta recompensa a cualquier grupo de personas que pudiera desarrollar un algoritmo (Usando una base de datos anónima de millones de usuarios y su interacción en la plataforma otorgada por Netflix) con rendimiento superior a su modelo actual CineMatch (Bennett & Lanning, 2007).

La inteligencia artificial en el sector agrícola

La inteligencia artificial se ha convertido en una herramienta importante en casi todas las áreas y profesiones laborales, esto también incluye el sector agrícola y la producción de alimentos.

Los algoritmos de aprendizaje y herramientas basadas en redes neuronales son de gran ayuda para automatizar tareas relacionadas con la producción de alimentos agrícolas, estas herramientas permiten reducir el consumo innecesario en recursos de producción y recolección, ofreciendo una eficiencia y velocidad que sería envidiable. ¿Por qué surge la necesidad de automatizar tareas relacionadas con la agricultura?

Hay varios factores que debemos considerar frente a esta necesidad, y no necesariamente están relacionados con la disminución del esfuerzo laboral de los agricultores. Uno de los aspectos principales que debemos tener en cuenta es el crecimiento poblacional a nivel global, ya que la tasa de producción de recursos

no es ilimitada y, eventualmente, será insuficiente para satisfacer las necesidades alimentarias básicas de gran parte del planeta.

En este sentido, el papel que desempeña la inteligencia artificial en la agricultura está directamente relacionado con la velocidad de producción, aliviando en cierta medida la insuficiencia de alimentos frente al crecimiento poblacional.

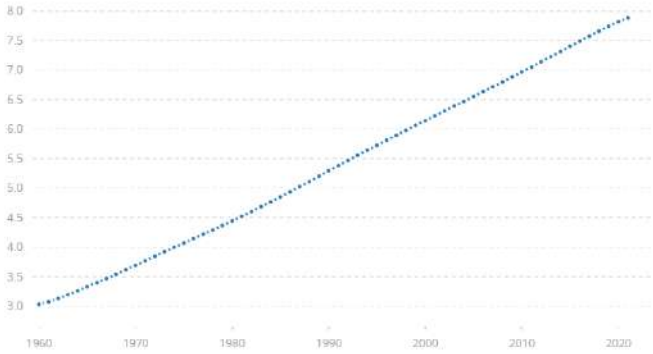


Ilustración 4 – Gráfico de relación entre el periodo de 1960 a 2021 y la población mundial (El eje X representa el tiempo en años, el eje Y representa unidades de miles de millones de habitantes). Fuente: (The World Bank Group, 2023)

La demanda de alimentos se espera que siga creciendo con el tiempo. Los métodos tradicionales utilizados por los agricultores no son suficientes para satisfacer esta demanda creciente, lo que lleva a los agricultores a optar por prácticas no tradicionales que puede afectar la fertilidad de la tierra, como el aumento de la cantidad de pesticidas utilizados en el proceso de producción. Gracias a la inteligencia artificial, es posible automatizar tareas que pueden facilitar todo este proceso (Jha, Doshi, Patel, & Shah, 2019).

En general, el uso de la inteligencia artificial en la agricultura y la industria alimentaria está en constante evolución y se espera que siga creciendo en el futuro. La automatización de tareas relacionadas con la producción de alimentos puede mejorar la eficiencia y la velocidad de producción, lo que es crucial para satisfacer la creciente demanda de alimentos en todo el mundo.

Caso ejemplar de estudio: Una granja “libre de manos” 1 en Australia

En 2021, se hizo popular la noticia de un proyecto que demostró el enorme potencial y aplicabilidad que puede tener la inteligencia artificial en el campo agrícola: una granja "libre de manos" en Australia. La Universidad de Charles Sturt, que proclama ser "la universidad #1 del país en empleabilidad de los graduados" (Charles Sturt University, 2023), es la entidad encargada de crear

[1] Se entiende un proceso “**libre de manos**” como aquel que **requiere de leve o nula interacción de trabajo humano**.

esta granja que hace uso de herramientas basadas en inteligencia artificial como tractores robóticos, drones, entre otros (Claughton & Condon, 2021).

Este tipo de casos no son únicos y existen esfuerzos similares en todo el mundo para automatizar tareas agrícolas usando la inteligencia artificial. Poder hacer esto ofrece la posibilidad de generar un impacto positivo en la calidad, velocidad y eficiencia de producción de los alimentos, al mismo tiempo que reduce la huella de carbono y su impacto en el calentamiento global (Pandey & Agrawal, 2014).

La inteligencia artificial en el sector automotriz

Desde un punto de vista estadístico, cada año aproximadamente 1.3 millones de personas pierden la vida o sufren consecuencias físicas graves debido a accidentes de tráfico, según indica un informe de la Organización Mundial de la Salud en junio de 2022 (World Health Organization, 2022). Los factores que causan estos accidentes varían bastante, según el gobierno de Jharkhand (India), algunos de los principales motivos causantes de los accidentes de tráfico son el exceso de velocidad, conducir en estado de embriaguez, pasar un semáforo en rojo, entre otros (Government of Jharkhand, 2023). Las estadísticas indican que entre 1993 y 2020, el porcentaje de accidentes vehiculares aumentó en un 831% (Injury Facts, 2023).

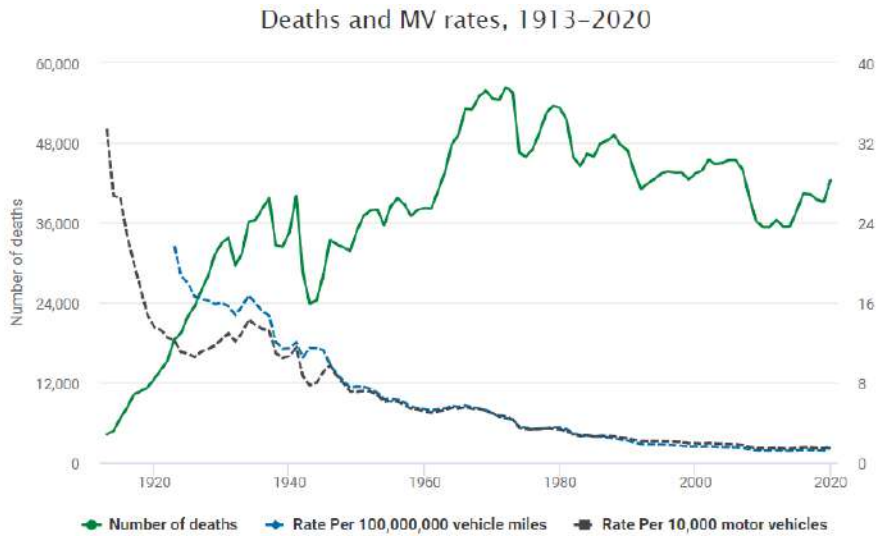


Ilustración 5 - Estadísticas de Injury Facts frente a las muertes causadas por accidentes de tráfico en el periodo 1913-2020

En cuanto a la relación entre la inteligencia artificial y los accidentes de tráfico, ¿es posible reducir las enormes cifras de accidentes utilizando la inteligencia artificial? Para responder a esta pregunta, es necesario analizar algunos ejemplos que ayuden a entender de manera más concreta el papel que pueden tomar estos modelos en este sector económico.

Caso ejemplar de estudio: *Tesla*

Tesla se define a sí misma como "una compañía que construye un mundo alimentado por energía solar, diseña sistemas sostenibles que puedan ser masivamente escalables e instala baterías para almacenar energía limpia" (Tesla, 2023). En 2008, Tesla presentó su primer automóvil completamente eléctrico, el Tesla Roadster, y desde entonces ha seguido fabricando vehículos eléctricos (Britannica, 2023).

Tesla se ha vuelto popular no solo por sus vehículos eléctricos, sino también por el uso de modelos de inteligencia artificial en sus automóviles. Esto ha generado bastante polémica para la empresa, especialmente a partir de 2015 cuando declaró que "los vehículos de Tesla serán capaces de conducirse solos" (KOROSEC, 2015). El sistema de inteligencia artificial en los vehículos de Tesla recibe el nombre de Tesla Autopilot y está diseñado para ofrecer "hardware capaz de proporcionar funciones de piloto automático y conducción autónoma" (Tesla, 2023).

Este sistema puede ayudar a reducir la tasa de accidentes de tráfico, ya que todas estas funciones y el ecosistema interior de los vehículos de Tesla ayudan a prevenir accidentes causados por errores humanos, y eventualmente pueden ser un gran paso hacia la conducción completamente autónoma y segura de los vehículos (Ingle & Phute, 2016). Los esfuerzos realizados por Tesla para disminuir significativamente los accidentes de tráfico se deben no solo a sus modelos de inteligencia artificial implementados en el sistema de sus vehículos, sino también al diseño general de estos, ya que proveen una gran resistencia ante diversos desastres. Las estadísticas son innegables y se ha demostrado que la capacidad de los vehículos de Tesla que usan el sistema Autopilot han reducido drásticamente las posibilidades y tasas de accidentes.

En el siguiente gráfico se pueden apreciar algunos ejemplos con más detalle: el color gris representa el promedio de millones de millas recorridas por un vehículo convencional en Estados Unidos antes de sufrir un accidente. Por otro lado, las líneas azules y celestes corresponden a los millones de millas recorridas por vehículos Tesla utilizando el sistema Autopilot y sin él, respectivamente. Se puede observar que la mejora es realmente significativa (Tesla, 2023).

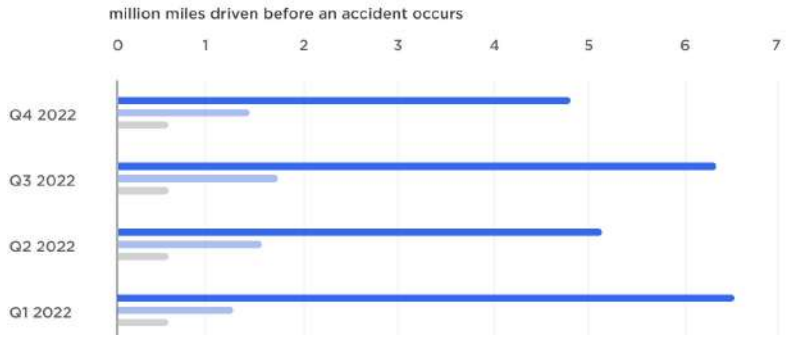


Ilustración 6 – Estadísticas de seguridad promedio de millones de millas recorridas antes de un accidente usando Tesla Autopilot (Tesla, 2023).

Podemos observar que los usos de modelos de inteligencia artificial en el sector automotriz realmente pueden generar un impacto positivo. Es importante destacar que, en el caso de Tesla, el desarrollo y mejora de este tipo de modelos ha sido un proceso que ha llevado varios años. Sin embargo, debido a las estadísticas positivas que presentan estos modelos de inteligencia artificial en vehículos particulares, numerosas compañías y productores de vehículos, también han comenzado a desarrollar sus propios modelos con filosofías similares a las de Tesla. Más allá de la competencia en el mercado, es fundamental reconocer que todo esto tiene un impacto sumamente positivo en la seguridad de las personas, incluyendo a conductores, peatones y al ambiente en general. Las posibilidades que ofrece la inteligencia artificial en este sector son prácticamente infinitas.

Conclusión del capítulo

Este capítulo ha proporcionado una visión exhaustiva de cómo la inteligencia artificial se ha entrelazado con muchos aspectos de nuestras vidas cotidianas. Se han desglosado ejemplos concretos y aplicaciones en el sector del entretenimiento y la agricultura, destacando cómo la IA puede mejorar la eficiencia, la personalización y la calidad en diversas áreas. A través de estudios de caso como Netflix y la exploración de algoritmos como NeRF, se ha demostrado la diversidad y la innovación en el campo de la IA. Además, se ha presentado el modelo Stable Diffusion, enfatizando su capacidad única en la generación de imágenes y su relevancia en la comunidad científica y académica. La discusión también ha arrojado luz sobre los desafíos y las oportunidades que enfrentamos en la era de la inteligencia artificial, haciendo hincapié en la importancia de comprender y utilizar estas herramientas de manera responsable y efectiva.

CAPÍTULO 2: FUNDAMENTOS DE INTELIGENCIA ARTIFICIAL

Objetivos del capítulo:

El presente capítulo tiene como propósito introducir al lector en los conceptos básicos y fundamentales de la inteligencia artificial (IA). Se explorará la definición y la historia de la IA, incluyendo los modelos de perceptrón y su influencia en el desarrollo de las redes neuronales. Se abordará el llamado "invierno de la IA", un período en la historia donde la investigación y financiación en IA se redujeron considerablemente, así como su impacto en el progreso de la disciplina. Finalmente, se discutirá el estado actual de la investigación en IA, con una reflexión sobre si estamos en una época dorada o en las puertas de un nuevo invierno de la inteligencia artificial. La intención de este capítulo es ofrecer una visión global y accesible de la IA, sentando las bases para una comprensión más profunda de Stable Diffusion en capítulos posteriores.

Antes de utilizar Stable Diffusion, una herramienta clasificada como modelo de inteligencia artificial, es importante tener una base teórica sobre qué es la inteligencia artificial, así como otros conceptos importantes para comprender lo que ocurre internamente en el modelo que estudiaremos.

Si bien es cierto que este libro no pretende ofrecer una explicación exhaustiva de los diferentes conceptos relacionados con la inteligencia artificial, ni describir cómo funcionan estos modelos desde una perspectiva técnica y profunda (a excepción de algunos casos directamente vinculados a Stable Diffusion), sí

brindaremos la oportunidad de entender ideas clave sobre la inteligencia artificial en general. Con esto en mente, se expondrá cada tema de la manera más abstracta y superficial posible, sin perder de vista que el proceso de aprendizaje debe ser ameno y fácil de comprender.

¿Qué es realmente la inteligencia artificial?

La tarea de describir qué es la inteligencia artificial no es tan complicada como parece. Existen numerosas definiciones que buscan explicar de manera sencilla el concepto, cada una de ellas propuesta por autores provenientes de diversos campos de estudio. A pesar de esto, podemos afirmar que el concepto de inteligencia artificial es universal, pues a pesar de que existen variaciones en las definiciones debido a los diferentes autores que abordan el tema, es evidente que todos comparten varios elementos en común en sus descripciones.

Con esto en mente, podemos afirmar que la inteligencia artificial se refiere a aquellos sistemas que emulan la inteligencia humana para llevar a cabo diversas tareas (Oracle, 2023). Lo que resulta fascinante de esto es su habilidad para procesar información y, eventualmente, simular el razonamiento y la toma de decisiones con un comportamiento semejante al de los seres humanos.

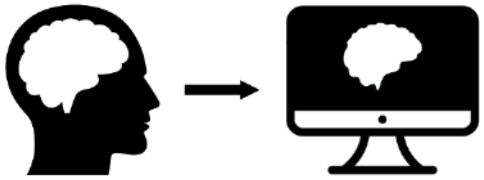


Ilustración 7 - "... la inteligencia artificial se refiere a aquellos sistemas que emulan la inteligencia humana ..."

Es importante destacar que la definición presentada es bastante general, pues engloba diversas ramas de campo de estudio que aplican modelos de inteligencia artificial en el desarrollo de sus profesiones. Aunque parezca que la inteligencia artificial solo hace parte del mundo informático, esta concepción realmente no es cierta.

La implementación de modelos y algoritmos de inteligencia artificial abarca una amplia lista de posibilidades, se extienden a numerosas profesiones, campos de estudio y situaciones específicas en la vida de las personas. Con esto en mente, podemos decir que esta definición y sus características mencionadas son comunes entre las otras varias definiciones que existen de este concepto, principalmente debido a que la inteligencia artificial es una herramienta tan versátil y flexible que puede adaptarse a una gran variedad de profesiones y contextos.

Aunque pueda parecer sorprendente, la inteligencia artificial como campo de estudio no es un fenómeno reciente. Los primeros eventos relacionados con este

campo se remontan a principios de la década de 1940, cuando Warren S. McCulloch y Walter Pitts propusieron un "modelo" que simulaba el funcionamiento de las neuronas. En la actualidad, estas neuronas artificiales se denominan perceptrones. Esta propuesta fue presentada en su artículo "A logical calculus of the ideas immanent in nervous activity" (McCulloch & Pitts, 1943).

No fue hasta finales de la década de 1950 cuando Frank Rosenblatt presentó oficialmente el perceptrón en su artículo "The perceptron: A probabilistic model for information storage and organization in the brain" (Rosenblatt, 1958). El perceptrón propuso una interpretación computacional que se asemeja fielmente al funcionamiento de una neurona biológica, convirtiéndose así en la estructura unitaria más simple de las redes neuronales. Profundizaremos en las redes neuronales en próximas secciones.

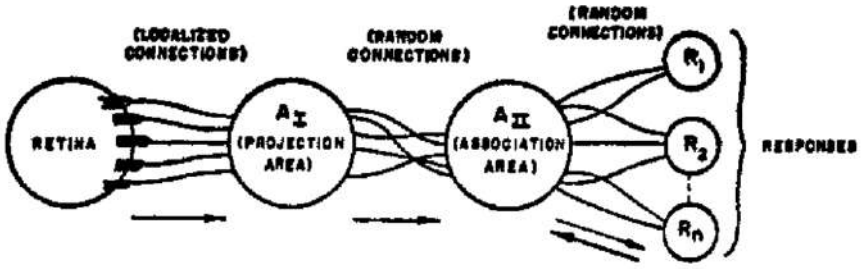


Ilustración 8 – Organización de un perceptrón (Rosenblatt, 1958), una de las primeras ilustraciones hechas

Desde entonces, la idea del perceptrón como una unidad en las estructuras de redes neuronales ha avanzado significativamente, facilitando la comprensión visual de un perceptrón.

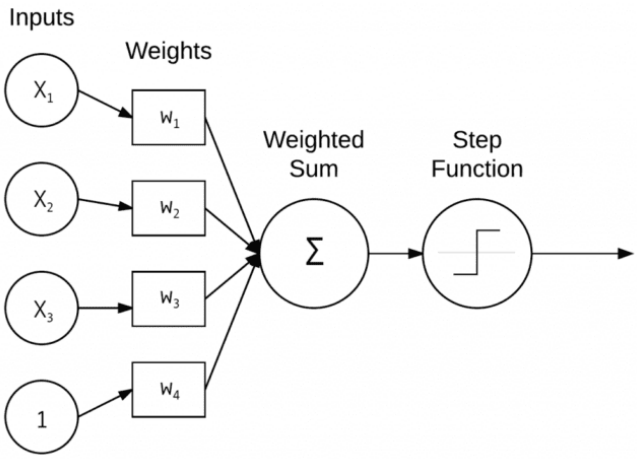


Ilustración 9 – Arquitectura de un perceptrón (Rosebrock, 2021)

Gracias a la actual comunidad científica y al poder computacional del hardware disponible, el progreso en la investigación de la inteligencia artificial avanza a un ritmo exponencial. Sin embargo, esto no siempre fue el caso.

El gran invierno de la IA

Hoy en día, el progreso en proyectos y diversos procesos de investigación relacionados con la inteligencia artificial ha cobrado gran relevancia. Distintos grupos de investigación y, en algunos casos, grandes empresas, están empezando a invertir sumas considerables de dinero en productos basados en inteligencia artificial (IA). Este avance, promovido por la comunidad científica, ha llevado a que la IA se convierta en un tema de interés para una amplia variedad de personas, como empleados, ciudadanos, educadores, emprendedores y creadores de contenido, entre otros.

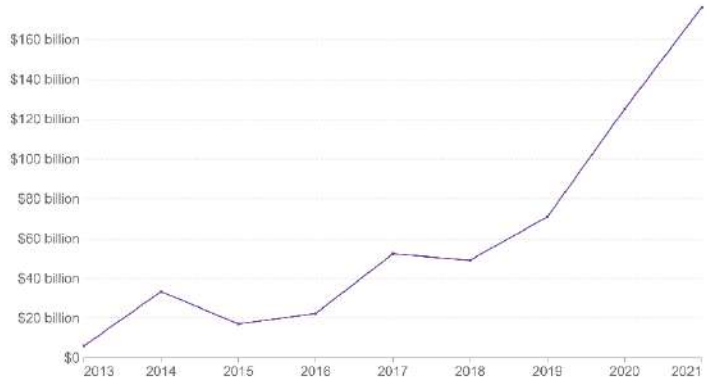


Ilustración 10 - Inversión anual global de corporaciones en inteligencia artificial (Our World In Data, 2021)

Es importante destacar que la investigación y la historia de la IA no son temas nuevos. De hecho, este fascinante campo ha sido objeto de estudio desde hace varias décadas. Sin embargo, a pesar del tiempo transcurrido, cabe preguntarse por qué los modelos de IA apenas están comenzando a entrar en su etapa dorada. Para comprender esto en detalle, debemos considerar varios factores que explican qué fue el "invierno de la IA" y por qué es tan importante hablar de este período de eventos. En términos simples, el "invierno de la IA" se refiere a un período en la historia en el que la investigación y financiación de proyectos de IA disminuían drásticamente.

La comunidad científica y el sector económico no mostraban mucho interés por financiar o realizar proyectos relacionados con la investigación de la IA. A fin de cuentas, era un tema de investigación en desarrollo y, lamentablemente, el hardware de la época no podía manejar modelos capaces de resolver tareas tan complejas como los actuales. Este proceso tuvo lugar entre las décadas de 1970 y 1980.

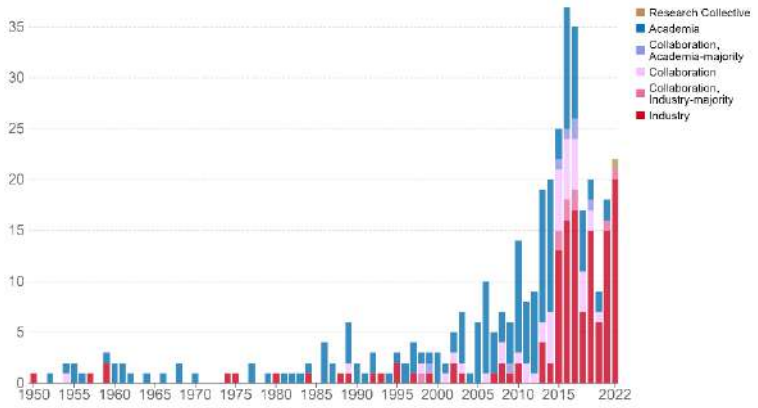


Ilustración 11 – Entidades de equipos de investigación construyendo sistemas notables de IA[2] a través del tiempo (Our World In Data, 2022).

Al principio, los investigadores en el campo de la IA tenían expectativas muy altas. Muchos autores de la época afirmaban que estos proyectos eran un paso hacia la creación de IA que simulara la inteligencia humana en solo unos años. Sin embargo, la realidad de aquel momento demostró que esta visión era demasiado optimista e irrealista, lo que llevó a una disminución en las expectativas y la decepción tanto de los investigadores como de los financiadores.

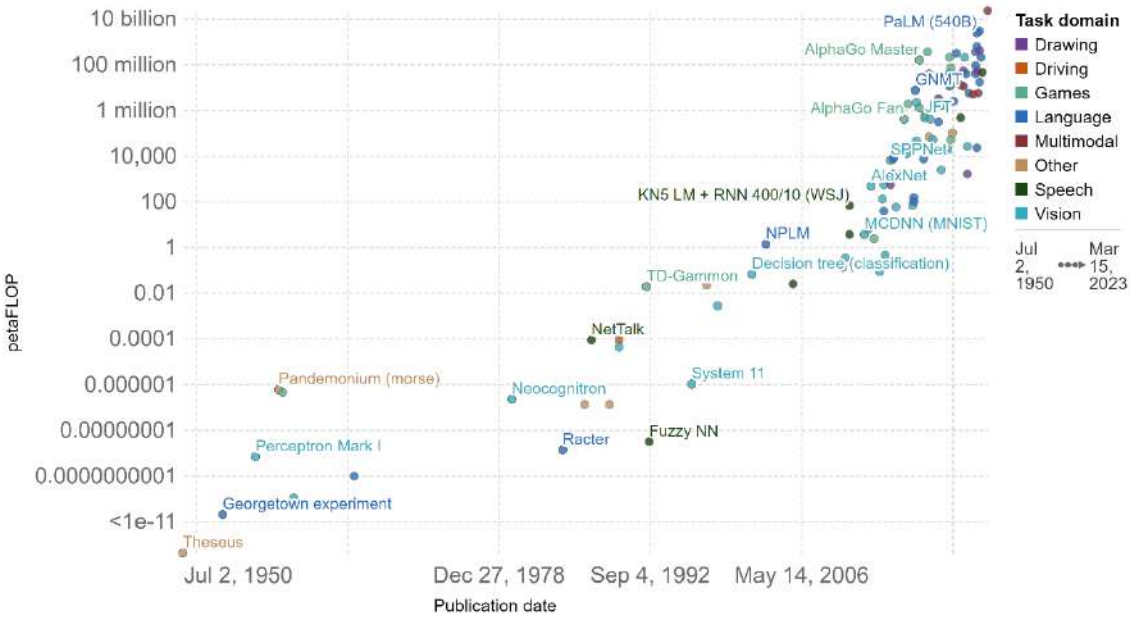


Ilustración 12 – Poder computacional usado para entrenar sistemas notables de IA a través del tiempo (Our World In Data, 2023).

[2] Los sistemas son definidos como "**notables**" en base a varios criterios, como su importancia histórica y avance del estado del arte.

¿Cómo interpretar este gráfico?

El objetivo de este gráfico es proporcionar una representación visual más clara de lo discutido en el texto, en el cual se mencionó que la investigación en inteligencia artificial durante el período del "invierno de la IA" estaba considerablemente limitada por el hardware de la época. En el gráfico, se muestra el poder computacional utilizado para entrenar diferentes modelos que se consideraron notables. Dicho poder computacional se presenta en petaFLOPs.

No fue hasta principios de la década de 2000 cuando el poder computacional para entrenar modelos empezó a aproximarse a 1 petaFLOP. Antes de eso, el poder computacional para entrenar modelos era considerablemente menor, incluso inferior a 0.01 petaFLOPs. Esto respalda la afirmación de que la idea prevaleciente durante el invierno de la IA, de que en pocos años existirían modelos de IA que replicarían la inteligencia

Esta situación provocó un retraso en el avance de los estudios de IA en general por varias décadas, lo que implica que los modelos de IA actuales podrían haber sido desarrollados con versiones mejores y el campo de la IA estaría mucho más avanzado en la actualidad.

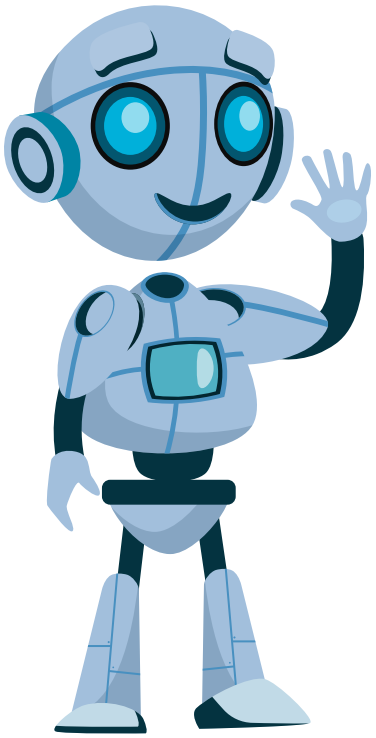
No obstante, en la actualidad disponemos de computadoras con una capacidad de procesamiento asombrosa en comparación con las de hace unas décadas. Esto permite que los modelos de inteligencia artificial y los diferentes grupos encargados de su desarrollo tengan muchas más posibilidades para entrenar modelos de gran envergadura, que cuentan con miles de millones o billones de parámetros. El progreso de la inteligencia artificial se ve cada vez más favorecido y potenciado gracias al desarrollo de nuevos dispositivos de hardware que ofrecen capacidades computacionales extraordinarias. Por lo tanto, podemos afirmar que todo esto contribuye de manera beneficiosa al avance de estos modelos.

Actualidad: ¿Época dorada o próximo invierno?

Ahora bien, es innegable que, como se ha mencionado en numerosas ocasiones a lo largo de este libro, el interés por investigar y desarrollar modelos de inteligencia artificial se ha vuelto cada vez más popular. Cada vez más empresas están invirtiendo grandes sumas de dinero en el desarrollo e investigación de modelos y algoritmos de inteligencia artificial, lo que sugiere un futuro prometedor para esta área, ¿verdad? La respuesta es bastante simple: sí. Todo esto contribuye significativamente al desarrollo de nuevos modelos y tecnologías cada vez más importantes y relevantes en este campo, no solo gracias a la inversión económica de las empresas y otras entidades que ven la investigación en inteligencia artificial como un área rentable, sino también debido a muchos otros factores que impactan positivamente en el desarrollo de estos modelos.

Sin embargo, aunque esto pueda parecer lógico, y a pesar de que no hay mucho que debamos considerar en este momento dado que solo forma parte de especulaciones futuras, algunas personas temen que la inteligencia artificial pueda enfrentar lo que se conoce como un nuevo "invierno de la inteligencia artificial". Entonces, ¿por qué ocurriría esto si acabamos de mencionar que el interés en el desarrollo de nuevos modelos e investigación en general en este campo sigue en pleno auge? La idea de un nuevo invierno de la IA puede generar diversas interpretaciones y opiniones, pero es importante tener en cuenta varios elementos al analizar esta posibilidad.

Conclusiones del capítulo

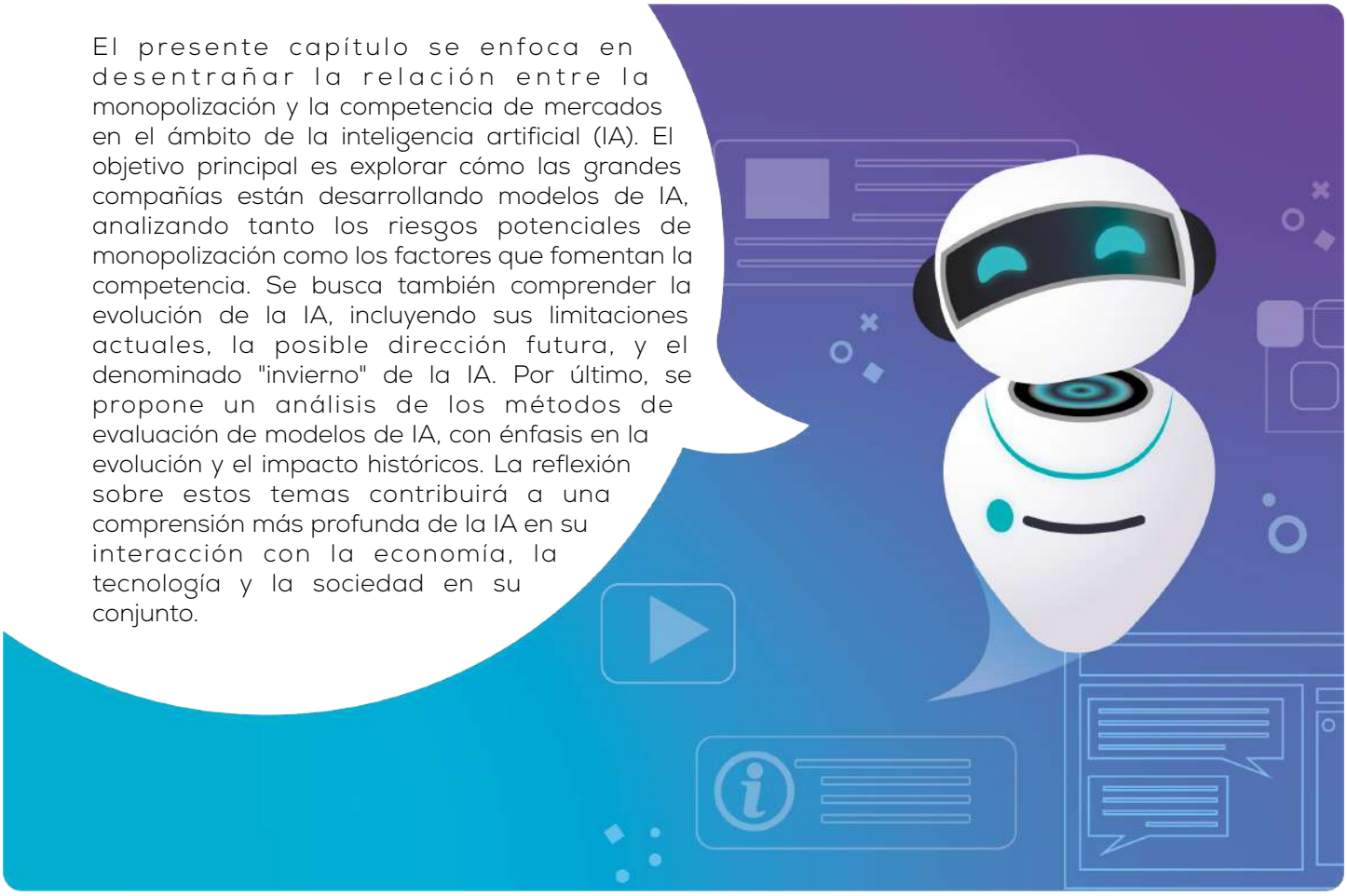


Este capítulo ha proporcionado una comprensión fundamental y completa de la inteligencia artificial, destacando su definición, historia, y aplicaciones diversas en la vida diaria y profesional. Se ha hecho énfasis en la evolución del campo, desde los primeros perceptrones hasta los desarrollos actuales, poniendo en perspectiva cómo la tecnología y el interés económico han influenciado este avance. La reflexión sobre un posible nuevo "invierno de la inteligencia artificial" nos invita a considerar el futuro de este campo con una perspectiva crítica y ponderada. La comprensión de estos temas es esencial para apreciar el papel de la IA en la sociedad moderna y en el contexto de la Stable Diffusion, que será el enfoque de los siguientes capítulos del libro.

MONOPOLIZACIÓN Y COMPETENCIA DE MERCADOS EN LA IA

Objetivos de esta sección:

El presente capítulo se enfoca en desentrañar la relación entre la monopolización y la competencia de mercados en el ámbito de la inteligencia artificial (IA). El objetivo principal es explorar cómo las grandes compañías están desarrollando modelos de IA, analizando tanto los riesgos potenciales de monopolización como los factores que fomentan la competencia. Se busca también comprender la evolución de la IA, incluyendo sus limitaciones actuales, la posible dirección futura, y el denominado "invierno" de la IA. Por último, se propone un análisis de los métodos de evaluación de modelos de IA, con énfasis en la evolución y el impacto históricos. La reflexión sobre estos temas contribuirá a una comprensión más profunda de la IA en su interacción con la economía, la tecnología y la sociedad en su conjunto.



Ahora, como se pudo observar en el gráfico anterior, el desarrollo de modelos de inteligencia artificial nuevos realmente no está siempre directamente relacionado con el mundo académico, sino que, al contrario, gran parte de los modelos de inteligencia artificial destacados actuales corresponden a modelos desarrollados por diferentes empresas privadas que tienen la capacidad de financiar grupos de investigación para el desarrollo de modelos muy grandes. Esto realmente, a pesar de tener muchos aspectos positivos, también puede presentar un aspecto negativo: el hecho de que grandes

compañías de desarrollo de modelos de inteligencia artificial puedan llegar al punto de lo que se considera como monopolización de los modelos.

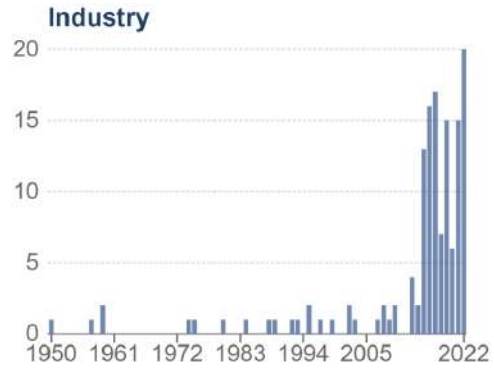


Ilustración 13 – Sistemas notables de IA a través del tiempo desarrollados por la industria (Our World In Data, 2022).

Es decir, si bien es cierto que el avance de la ciencia en general por parte de las empresas realmente no es un aspecto nuevo y se ha dado desde hace mucho tiempo, es un aspecto que también afecta a la inteligencia artificial. Empieza a hacer que el uso de este tipo de modelos se vuelva algo más cerrado, con fines lucrativos y bajo directivas de una sola empresa, haciendo que la inteligencia artificial quede únicamente en manos de unos pocos y se convierta en un elemento económico y rentable, además de un avance relevante para la ciencia.

Esta idea puede parecer realmente conspirativa, pero bastantes autores afirman esta preocupación de la monopolización de los modelos de inteligencia artificial (Nowak, Lukowicz, & Horodecki, 2018) (Gans, 2022). Sin embargo, afortunadamente no hemos llegado a presenciar un caso como estos. Hasta ahora, el desarrollo de modelos de inteligencia artificial y gran parte de estos sistemas notables han estado abiertos para la comunidad en general, no necesariamente como Open Source[3], sino también como modelos de uso libre para el resto de las personas interesadas en usar y aplicar estos tipos de modelos.

Además, es crucial destacar la competencia existente entre los diversos grupos de investigación, quienes se esfuerzan cada vez más por desarrollar un modelo superior al de sus competidores. Esta rivalidad evita que las empresas y entidades privadas monopolicen la inteligencia artificial. Otros factores que impiden esta monopolización incluyen el desarrollo de modelos de alto rendimiento por parte de grupos de investigación y comunidades comprometidas con la filosofía de código abierto (Shatnawi, Al-Bdour, Al-Qurran, & Al-Ayyoub, 2018). La creación de estos modelos de código abierto facilita el desarrollo de un

3 El software Open Source (Código abierto en español), es aquel cuyo código fuente original está disponible gratuitamente, además de poder modificarse y distribuirse.

número creciente de soluciones de inteligencia artificial accesibles para cualquier persona, contribuyendo así al avance más libre de esta tecnología.

Existen varios factores adicionales a considerar en relación con las limitaciones de la inteligencia artificial actual. Por ejemplo, aunque es cierto que se han desarrollado diversos modelos de inteligencia artificial con la intención de ser de tipo general, es decir, una inteligencia artificial capaz de desempeñar y adaptarse a distintas tareas, lo cierto es que probablemente estemos lejos de alcanzar un punto en el que la inteligencia artificial se convierta en completamente general y posea una inteligencia similar o incluso superior a la del ser humano.

Es indudable que, hasta ahora, la inteligencia artificial se ha demostrado un rendimiento asombroso en un conjunto específico de tareas. No obstante, toda esta emoción y esperanza en torno al progreso de los modelos de inteligencia artificial podría enfrentar una decepción en el futuro, no solo debido al paso del tiempo, sino también a limitaciones en hardware.

Además, es importante considerar otros factores, como el hecho de que la tecnología actual y el desarrollo de hardware avanzado, en cierta medida, eventualmente alcanzarán un límite. Después de todo la mayoría de los transistores han llegado a un punto en el que es casi imposible reducir aún más su tamaño. Esto no implica que el hardware dejará de avanzar, ni que el desarrollo de la inteligencia artificial se detendrá; simplemente significa que algunos expertos muestran preocupación debido a que las técnicas actuales empleadas en inteligencia artificial podrían llegar a un punto máximo, tal como ha ocurrido en el pasado.

No obstante, a pesar de todos estos factores, algo es indudable: la inteligencia artificial, al menos por ahora, muestra un crecimiento, interés y financiación en constante incremento. Esto implica que, al menos en el presente, seguiremos presenciando avances en este campo de estudio, lo que nos permitirá lograr avances más significativos en el mundo de la inteligencia artificial. Por el momento, solo podemos afirmar que, si en algún momento las preocupaciones de numerosos autores resultan ser ciertas, seremos testigos de un avance crucial y sumamente relevante en la inteligencia artificial. Además, es esencial aclarar que todas estas ideas solo pueden considerarse como especulaciones, dado que, después de todo, nadie puede predecir con certeza el futuro de la inteligencia artificial. Lo único seguro es que su investigación y desarrollo continúan experimentando un auge en constante crecimiento.

Por otro lado, desde una perspectiva menos pesimista, como la que sugiere que la inteligencia artificial alcanzará un punto en el que se considere otro "invierno" de la IA, algo es seguro: no debemos ignorar por completo el hecho de que los modelos de inteligencia artificial se vuelven cada vez más competentes. Es cierto que, hace apenas unos años, los modelos de IA realizaban tareas de manera que solo cumplían con algunos resultados limitados tanto en capacidad como en calidad, principalmente debido a las limitaciones técnicas de estos modelos. Sin

embargo, ahora contamos con modelos de inteligencia artificial capaces de realizar diversas tareas con tal eficacia que podemos afirmar que la calidad del material proporcionado por estos modelos es realmente sorprendente.

No es simplemente un fenómeno limitado a unos pocos modelos de hace algunos meses, sino un proceso constante de progreso y mejora que se ha estado llevando a cabo desde hace varios años. Por lo tanto, podemos afirmar que estamos viviendo lo que muchos autores denominan la época dorada de la inteligencia artificial, no solo otro invierno (Kaynak, 2021).

Después de todo, es importante considerar que nunca se habían registrado tantos avances y progresos significativos en el campo de la inteligencia artificial como hasta ahora. Ya no se trata solo de expectativas incumplidas, como ocurría hace décadas, sino que los modelos de inteligencia artificial actuales están ofreciendo resultados verdaderamente valiosos y positivos, en lugar de promesas vagas.

Teniendo en cuenta lo anterior, podemos afirmar que la denominación más adecuada para la actual etapa de avances en modelos de inteligencia artificial sería algo similar a una "época dorada de la inteligencia artificial" (Kaynak, 2021). No obstante, los modelos de inteligencia artificial todavía presentan numerosas limitaciones. Diversas empresas y grupos de investigación están desarrollando modelos de inteligencia artificial asombrosos, lo cual nos lleva a anticipar un crecimiento aún mayor en este campo de estudio. Después de todo, aunque es cierto que nos encontramos en lo que muchos autores denominan la época dorada de la inteligencia artificial, es posible que solamente estemos siendo testigos del inicio de una nueva y trascendental era en la ciencia.

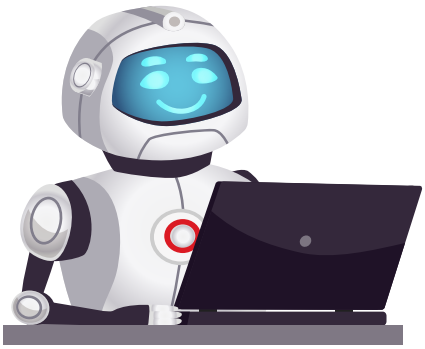
Métodos de evaluación

Naturalmente, los modelos de inteligencia artificial deben proporcionar resultados coherentes o, de alguna manera, ser capaces de resolver ciertos tipos de problemas. La forma de evaluar estos modelos puede variar según los enfoques escogidos. Aunque en la actualidad los modelos de aprendizaje automático (machine learning) han demostrado ser altamente eficientes para abordar diversos problemas, es importante recordar que no siempre ha sido así.

Teniendo esto en cuenta, podemos afirmar que la manera de evaluar a estos modelos ha experimentado una evolución y cambios significativos a lo largo del tiempo. Aunque pueda parecer trivial comprender esto, es crucial reconocer que la evaluación de los diferentes modelos desarrollados a lo largo de la historia constituye un paso esencial para profundizar en el conocimiento y la importancia de estos variados enfoques de evaluación. Esto nos permite apreciar la evolución y el impacto que la inteligencia artificial ha ejercido en la historia de la humanidad y el progreso científico, y también para comprender cómo el estándar de eficiencia de los modelos ha evolucionado a lo largo del tiempo.

En esta sección, examinaremos algunas de las formas en las que se ha evaluado el progreso de la inteligencia artificial en general a lo largo del tiempo, mediante pruebas generales que pueden indicar nuestro punto actual de avance en este campo. Sin embargo, estos elementos serán demasiado generales y no representarán métricas precisas, debido a la diversidad que presentan estos modelos (Handelman, y otros, 2019).

Conclusión de este capítulo:



Este capítulo ha ofrecido una perspicaz exploración de la monopolización y la competencia en el contexto de la inteligencia artificial. Se ha resaltado que, aunque existen preocupaciones sobre la monopolización de los modelos de IA por parte de grandes empresas, factores como la competencia entre grupos de investigación y el compromiso con el código abierto han prevenido tal monopolización hasta el momento. Se ha enfocado en la constante evolución y mejoramiento de la IA, destacando que estamos en una etapa que podría considerarse como la época dorada de esta tecnología. Además, se ha reconocido que, a pesar de los avances, aún hay desafíos y limitaciones significativas en el desarrollo de la IA. La reflexión sobre estos temas complejos no solo enriquece nuestra comprensión de la IA, sino que también arroja luz sobre las preocupaciones éticas y prácticas que la acompañan. La evaluación de estos factores resulta esencial para apreciar la influencia y el potencial de la IA en nuestra era contemporánea.

AJEDREZ, GO Y LA IA

Objetivos del capítulo

El presente capítulo tiene como objetivo principal explorar la relación entre la inteligencia artificial (IA) y los juegos de mesa, especialmente el ajedrez y el Go, y su influencia en la evolución de los modelos de IA. Se busca explicar cómo estos juegos han sido plataformas de pruebas fundamentales para desarrollar y evaluar modelos de IA, y de qué manera han contribuido a la comprensión y avance en este campo. Además, se aspira a ilustrar los hitos históricos alcanzados a través de estos juegos y cómo han ayudado a medir la progresión de la IA en términos de habilidad, eficiencia, y complejidad.

Curiosamente, una forma en la que se han evaluado los avances en inteligencia artificial, de modo que podamos obtener una idea precisa sobre el progreso de los modelos de IA a lo largo del tiempo, es a través del desarrollo de modelos de IA para jugar al ajedrez y otros juegos de mesa relacionados con la estrategia y el pensamiento lógico. Aunque no lo parezca, estos modelos de inteligencia artificial han sido de gran importancia para el avance general en el desarrollo de diferentes modelos de IA. Incluso, podemos afirmar que los modelos de inteligencia

artificial para jugar al ajedrez han desempeñado un papel crucial en la historia de este campo (Krishnamurthy, 2022).

Teniendo esto en cuenta, esta breve subsección tiene como objetivo explicar cómo se han empleado modelos de inteligencia artificial en el ajedrez y otros juegos de mesa, como el Go, a lo largo de la historia de la inteligencia artificial. Además, se busca detallar la importancia de estos avances.

El uso del ajedrez en la inteligencia artificial es una práctica bastante común. Desde la década de 1960, algunos pioneros en este campo emergente consideraban al ajedrez como la "Drosophila" de la inteligencia artificial. A simple vista, la relación entre un juego de mesa y la inteligencia artificial podría parecer poco evidente; sin embargo, el ajedrez fue percibido como una práctica común, pero útil para transferir información de elementos cada vez más complejos de manera experimental, accesible, familiar y sencilla.

Desde entonces, el ajedrez ha desempeñado un papel crucial en la investigación de la inteligencia artificial como método experimental para probar modelos. La decisión de centrarse en el ajedrez en esta área de estudio influyó de manera significativa en el programa de investigación de la inteligencia artificial en la década de 1970.

Aunque un juego de mesa relativamente simple como el ajedrez podría parecer que no guarda relación con los modelos de inteligencia artificial o su desarrollo, a lo largo de la historia se han utilizado modelos de inteligencia artificial y diversos tipos de algoritmos para crear programas y sistemas capaces de jugar ajedrez y otros juegos de mesa, incluso enfrentándose a jugadores humanos expertos. Esta era una forma experimental de evaluar la inteligencia artificial y sus capacidades (Ensmenger, 2012).

Lo que comenzó como un experimento, se ha transformado en una de las aplicaciones más populares de la inteligencia artificial. El desarrollo de modelos de inteligencia artificial que mejoran progresivamente su habilidad en el ajedrez es una realidad, y con el paso del tiempo, el profesionalismo y las capacidades de estos modelos continúan avanzando constantemente. A continuación, examinaremos más detenidamente algunas estadísticas e información histórica sobre esto.

Históricamente, una de las primeras máquinas computarizadas que empleaba inteligencia artificial para jugar ajedrez fue el programa Deep Blue. Este fue el primer sistema informático capaz de vencer a uno de los mejores jugadores de ajedrez de esa época, Garry Kaspárov (Newborn, 2012). Este acontecimiento ocurrió en la década de los años 90 y, en ese momento, representó un asombroso avance tanto para la comunidad ajedrecística como para la inteligencia artificial y la ciencia en general. Sin embargo, esto solo fue el punto de partida para el desarrollo de modelos cada vez más avanzados.

Desde entonces, han surgido diferentes máquinas y sistemas basados en modelos de inteligencia artificial, demostrando ser incluso más capaces que Deep Blue. Actualmente, los modelos de inteligencia artificial desarrollados para el ajedrez han alcanzado un nivel de perfección y ELO inalcanzable para cualquier ser humano. En otras palabras, hemos llegado a un punto donde hemos creado sistemas de inteligencia artificial que juegan ajedrez mucho mejor que todos los ajedrecistas de la historia e incluso mejor que cualquier ser humano vivo.

Aunque no lo parezca, esto representa uno de los avances más importantes de la inteligencia artificial en la historia. Esto no es lo único sorprendente, pues también lo es el hecho de que este tipo de modelos de inteligencia artificial tienen la capacidad de analizar cientos de millones de posibles resultados de una sola partida de ajedrez en un muy corto periodo de tiempo. Esto es bastante sorprendente desde un punto de vista computacional y también remarca la importancia del avance del hardware cada vez más capaz de manejar este tipo de modelos.

Para entender un poco más a qué nos referimos y tener una idea más visual de la importancia que ha tenido el ajedrez en el desarrollo de diferentes modelos de inteligencia artificial a lo largo del tiempo, podemos observar y analizar este gráfico. Aquí, se muestra cómo el ELO de las computadoras que juegan ajedrez basadas en inteligencia artificial ha aumentado drásticamente con el paso del tiempo. Del mismo modo, podemos ver las distintas clasificaciones que tienen este tipo de modelos frente a algunos de los jugadores más importantes en el mundo del ajedrez. De esta manera, podemos tener una mejor idea del avance de los modelos de inteligencia artificial que juegan ajedrez a lo largo del tiempo y por qué los modelos actuales son relativamente invencibles.

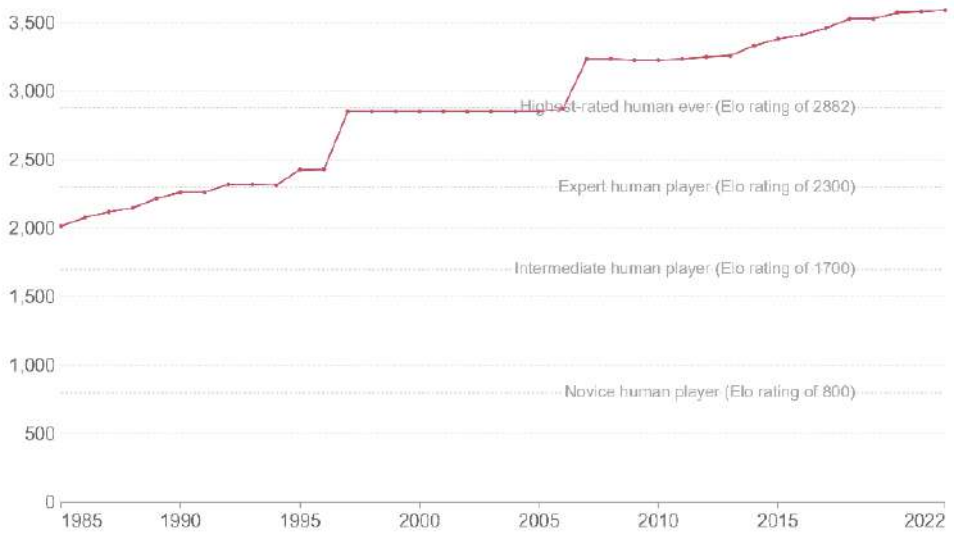


Ilustración 14 - Habilidad en el ajedrez de las mejores computadoras (Our World In Data, 2022).

En esta ilustración, podemos observar cómo han avanzado las computadoras y los diversos sistemas que juegan ajedrez, así como su habilidad en el juego. En el ajedrez, la habilidad de un jugador se mide mediante su ELO [4]. Actualmente, los modelos de inteligencia artificial han superado la barrera de 3591 puntos ELO, lo que representa un número significativamente mayor al puntaje ELO más alto alcanzado por un ser humano, el cual es de 2882 puntos. Esto nos permite comprender de manera más clara la diferencia abismal en el avance de la inteligencia artificial a lo largo del tiempo en el ajedrez, y por qué es común la relación hecha entre la inteligencia artificial y el ajedrez.

Ahora bien, el ajedrez no es el único juego de mesa de estrategia relacionado con la inteligencia artificial; de hecho, también existen otros juegos que han demostrado ser un desafío intelectual mayor para estos modelos, los cuales han logrado superar a seres humanos con décadas de experiencia. Por ejemplo, podemos mencionar un juego en particular que representa un avance significativo y reciente en el ámbito de la inteligencia artificial. Concretamente, nos referimos al juego de mesa estratégico Go, un juego bastante complejo de comprender al principio para aquellos que no están familiarizados con él. Lo interesante no es el juego en sí, sino un modelo de inteligencia artificial que fue creado con la intención de poder jugar este juego de mesa. Este modelo del cual estamos hablando es un modelo desarrollado por DeepMind, una de las empresas más importantes que llevan el liderazgo en el desarrollo de modelo de inteligencia artificial; con el paso del tiempo, esta empresa se ha convertido en una de las empresas más importantes en el desarrollo de modelos de inteligencia artificial.

4 El ELO es un sistema creado para calcular la habilidad relativa de los jugadores de ajedrez. Parte de la idea de que la habilidad de un jugador puede ser medida por su desempeño en partidas contra jugadores con diferentes niveles de habilidad.

El modelo al que nos referimos se llama AlphaGo y es el primer programa de inteligencia artificial que ha logrado derrotar a un jugador humano profesional de Go (DeepMind, 2022). No estamos hablando de un jugador profesional cualquiera, sino del campeón mundial de este juego de estrategia, Lee Sedol, lo que convierte a este modelo de inteligencia artificial en el "jugador" de Go más destacado de todos los tiempos. Este logro representa un avance significativo en el campo de la inteligencia artificial y un hito histórico en general.

Puede parecer trivial otorgar el título del jugador más fuerte de Go a una computadora. Sin embargo, es crucial destacar que la capacidad de una computadora para competir con el mejor jugador de este tipo de juego de estrategia es un logro relativamente reciente. Aunque no lo parezca, no podemos afirmar que la implementación de modelos de inteligencia artificial en computadoras para jugar al Go sea algo nuevo.

A pesar de las décadas de investigación en este campo, la existencia de una computadora capaz de jugar este juego de estrategia es un desarrollo reciente. Puede resultar incomprensible, ya que modelos de inteligencia artificial con la habilidad de jugar otros juegos de estrategia, que requieren analizar y evaluar millones de posibles resultados, no son una novedad. Por ejemplo, ya se han desarrollado modelos de inteligencia artificial capaces de derrotar a los mejores jugadores de ajedrez. Entonces, ¿por qué ha llevado tanto tiempo obtener resultados similares en el juego de Go? (Nielsen, 2016).

Es importante preguntarnos, si los modelos de inteligencia artificial fueron desarrollados durante tanto tiempo, ¿por qué solo recientemente se ha creado un modelo capaz de derrotar al mejor jugador humano de Go? Aunque no lo parezca hay muchos aspectos a considerar para responder a esta pregunta. Después de todo, aunque el ajedrez ya es un juego de mesa bastante complicado para algunas personas, el Go es un juego de mesa aún más complejo de lo que algunas personas podrían imaginar. Por lo general, las personas que tienen un buen desempeño en este juego no solo son muy inteligentes, sino que también han dedicado décadas a perfeccionar su estilo de juego.

Hay muchos ejemplos de diferentes modelos de inteligencia artificial que han tenido diversas aplicaciones y han permitido a las personas comprender realmente la capacidad que estos sistemas presentan.

Hoy en día, existen numerosas formas de evaluar el avance de la inteligencia artificial, desde la prueba de Turing hasta modelos de inteligencia artificial que resuelven exámenes matemáticos, que los seres humanos suelen realizar, y han logrado obtener excelentes resultados. En general, es cierto que el desarrollo de estos modelos y sistemas ha ido en aumento, aportando aspectos positivos al avance de la inteligencia artificial de forma general.

Estos modelos permiten a las personas comprender las numerosas aplicaciones y las diversas formas de evaluar su eficiencia. Con esto en mente, podemos

hacernos una idea de los distintos avances experimentados en el campo de la inteligencia artificial, así como la relación que guardan los juegos de mesa con ella. Esto nos permite entender, por ejemplo, el rendimiento de estos modelos en comparación con expertos en distintas áreas del conocimiento.

La utilización de la inteligencia artificial como herramienta resulta altamente positiva y representa un avance significativo para la humanidad. Después de todo, es como contar con un experto en casi cualquier tema, dependiendo del modelo específico que deseemos emplear, lo cual nos ayudará a comprender diversos asuntos que queramos abordar. Además, con la aparición de numerosos modelos de procesamiento del lenguaje natural, se ha observado que los modelos generativos pueden ser extremadamente útiles para obtener resultados esperados o similares a los mencionados, es decir, modelos de inteligencia artificial con la capacidad de comunicarse con nosotros de manera similar a un experto en un tema particular.

El ámbito de la inteligencia artificial es verdaderamente fascinante, y no existe mejor momento en la historia que el presente para comprender y estudiar en profundidad todos estos modelos y nuevas tecnologías.

Test de Turing: ¿Evaluación obsoleta?

A continuación, abordaremos otra forma de evaluar el progreso de los modelos de inteligencia artificial en general. En este caso discutiremos una de las metodologías más populares para evaluar un modelo de inteligencia artificial: la prueba de Turing. Esta prueba es bastante conocida, debido a su aparición en varias películas de ciencia ficción relacionadas con la inteligencia artificial.

Es importante destacar varios aspectos a tener en cuenta. Por ejemplo, este tipo de pruebas no están necesariamente obsoletas. De hecho, en la actualidad debido al desarrollo de numerosos modelos de inteligencia artificial especializados en el procesamiento del lenguaje natural, se ha demostrado que son muy efectivos en este tipo de pruebas.

Para aquellos que no sepan de qué se trata la prueba de Turing, el concepto es bastante sencillo. Se trata de poner a prueba un modelo de inteligencia artificial al interactuar con un ser humano. La idea detrás de esta prueba es que el ser humano no tiene contacto físico con la máquina y, de hecho, no sabe que aquello con lo que está interactuando no es un ser humano. Si el ser humano es capaz de clasificar a la persona con la que estuvo interactuando como un modelo de inteligencia artificial, entonces se puede decir que el modelo habrá fracasado en la prueba. De lo contrario, si el ser humano realmente no pudo darse cuenta de que estaba interactuando con una inteligencia artificial y asume que estaba interactuando con otro ser humano, por lo que el modelo habrá pasado la prueba.

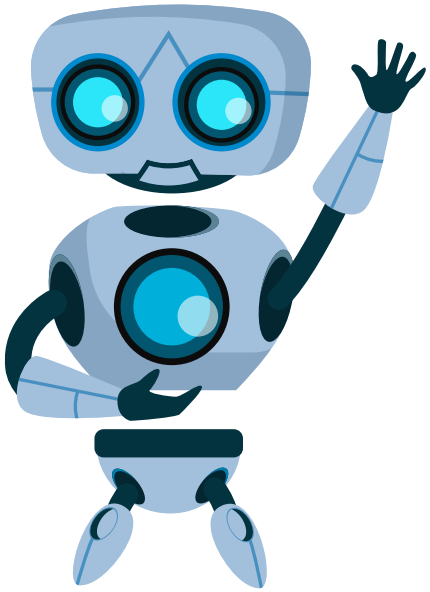
Puede parecer algo sacado de una película este tipo de pruebas para medir el rendimiento de un modelo de inteligencia artificial en diferentes ámbitos que involucre, por ejemplo, interacción directa con el ser humano, como es el caso de algunos modelos de inteligencia artificial conversacionales. Por otro lado, es cierto que modelos de inteligencia artificial que han pasado la prueba de Turing fueron desarrollados hace unos cuantos años, lo que demuestra que, si bien era sorprendente y difícil de superar hace unas décadas, en la actualidad, es una prueba que muchos de los modelos de inteligencia artificial generativos de conversaciones actuales pueden superar fácilmente.

El tema en cuestión, que mencionábamos anteriormente, es la obsolescencia de ciertas pruebas como la prueba de Turing. Actualmente, los modelos de inteligencia artificial ya tienen la capacidad de pasar dicha prueba, lo que indica que no está completamente alineada con el estado del arte. Por lo tanto, es necesario evaluar estos modelos de inteligencia artificial mediante otros tipos de pruebas, más allá de la prueba de Turing, para determinar si un modelo es eficiente o no en el contexto de la inteligencia artificial actual.

Existen varios factores a considerar al abordar la prueba de Turing. Por ejemplo, el objetivo principal de la prueba de Turing es determinar la inteligencia del modelo en cuestión. No obstante, algunos autores sostienen que la intención principal de esta prueba no es realmente medir la inteligencia del modelo, sino demostrar la inteligencia potencial que podría presentar un modelo que apruebe la prueba.

Es fundamental tener en cuenta que la inteligencia artificial no se trata de seres humanos conscientes, sino de modelos entrenados para proporcionar respuestas basadas en información previamente utilizada para su aprendizaje. Por lo tanto, es crucial comprender la diferencia entre la inteligencia humana y la inteligencia computacional, dos conceptos que muchos autores defienden como muy distintos (Trimble, 2023). La prueba de Turing no busca demostrar que los modelos de inteligencia artificial puedan entender, comprender y tener conciencia como los seres humanos, en realidad, es una forma algo inadecuada de evaluar la inteligencia computacional de los modelos actuales.

Conclusión del capítulo

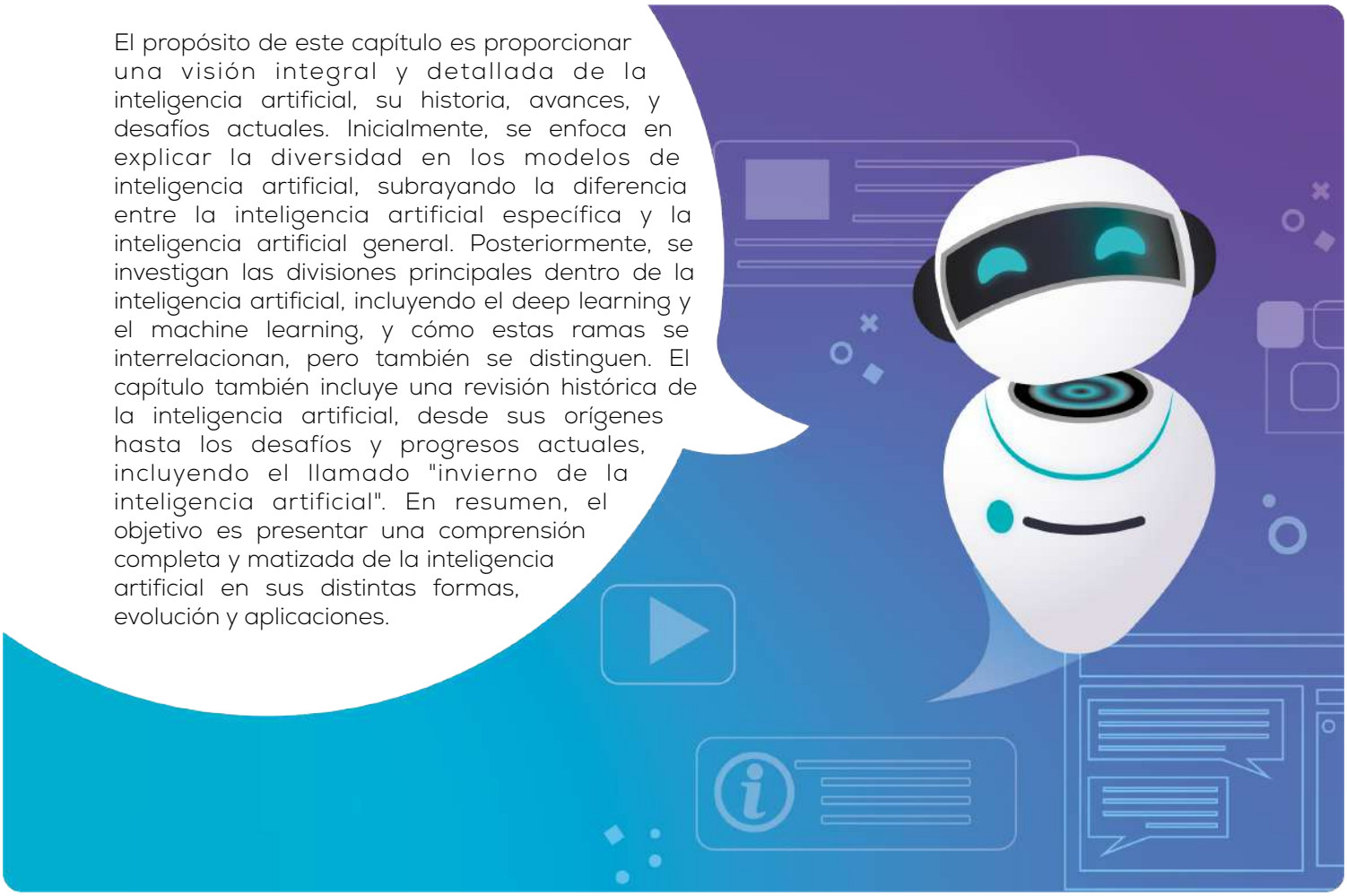


La conexión entre los juegos de mesa como el ajedrez y el Go y la inteligencia artificial ha sido una vía esencial en el progreso de los modelos de IA. La complejidad de estos juegos ha servido como un reto significativo para desarrollar y afinar algoritmos y técnicas, llevando a resultados sorprendentes como la victoria de la IA sobre campeones humanos. Esta sección ha expuesto cómo estos logros, aunque puedan parecer triviales a simple vista, representan avances fundamentales en la capacidad computacional y el entendimiento de la IA. La evolución constante de la IA en estos juegos simboliza no solo una mejora en habilidad sino también un reflejo del avance tecnológico en su totalidad. La relación entre los juegos de mesa y la IA es un testamento claro de cómo se puede utilizar un contexto lúdico para impulsar la innovación y la comprensión en un ámbito científico y tecnológico, haciendo de ellos una parte indispensable de la historia de la inteligencia artificial.

INTELIGENCIA ARTIFICIAL GENERAL

Objetivos del capítulo

El propósito de este capítulo es proporcionar una visión integral y detallada de la inteligencia artificial, su historia, avances, y desafíos actuales. Inicialmente, se enfoca en explicar la diversidad en los modelos de inteligencia artificial, subrayando la diferencia entre la inteligencia artificial específica y la inteligencia artificial general. Posteriormente, se investigan las divisiones principales dentro de la inteligencia artificial, incluyendo el deep learning y el machine learning, y cómo estas ramas se interrelacionan, pero también se distinguen. El capítulo también incluye una revisión histórica de la inteligencia artificial, desde sus orígenes hasta los desafíos y progresos actuales, incluyendo el llamado "invierno de la inteligencia artificial". En resumen, el objetivo es presentar una comprensión completa y matizada de la inteligencia artificial en sus distintas formas, evolución y aplicaciones.



En la sección anterior, abordamos brevemente las distintas maneras en que se puede evaluar el rendimiento de un modelo de inteligencia artificial en problemas de la vida real. No obstante, es crucial considerar diversos aspectos importantes de las comparaciones previamente mencionadas.

Por ejemplo, cada uno de los modelos de inteligencia artificial que analizamos tiene diferentes aplicaciones. Esto significa que no podemos esperar que un

modelo como AlphaGo pueda pasar la prueba de Turing, ya que ha sido entrenado exclusivamente para jugar al Go y no para realizar otras tareas.

De la misma forma, es probable que un modelo de inteligencia artificial generativo de conversaciones no pueda ser evaluado de la misma manera que AlphaGo. Esto nos lleva a reflexionar sobre la naturaleza de los modelos de inteligencia artificial actuales y cómo aún están lejos de convertirse en lo que se conoce como inteligencia artificial general[5], es decir, un modelo capaz de resolver prácticamente cualquier tarea basándose en el aprendizaje, con o sin entrenamiento previo.

La idea detrás de la inteligencia artificial general es desarrollar un modelo que pueda ofrecer un buen rendimiento en casi cualquier tarea. Aunque este tema ha sido objeto de investigación desde hace tiempo, como en el caso del *proyecto Gato* (DeepMind, 2022), reconocido como una de las primeras inteligencias artificiales generales, la realidad es que alcanzar un punto en el que los modelos de inteligencia artificial puedan realizar cualquier tarea puede parecer ciencia ficción o simplemente un objetivo aún no alcanzado en su totalidad.

Por ello, es importante destacar que las distintas formas de medir el rendimiento de los modelos de inteligencia artificial mencionadas anteriormente no son absolutas, sino que se centran en tareas específicas que cada modelo está diseñado para realizar. La evaluación que debemos llevar a cabo está altamente influenciada por el tipo de modelo que intentamos poner a prueba.

Inteligencia artificial, machine learning y deep learning

En el campo de estudio de la inteligencia artificial, es importante reconocer que existen diversas ramas principales que se agrupan bajo un mismo término. La inteligencia artificial se divide en diferentes capas, según el método o aspecto específico que se desee estudiar. Aunque hemos utilizado el término "inteligencia artificial" de manera general a lo largo del libro, la verdad es que abarca una amplia variedad de enfoques y técnicas.

Las principales ramas de la inteligencia artificial incluyen, por ejemplo, el deep learning y el machine learning. Estos términos, mencionados anteriormente, se refieren a las principales divisiones dentro de la inteligencia artificial. Al hablar de inteligencia artificial, en realidad estamos abarcando muchos más temas de los que imaginamos.

A pesar de que el deep learning y el machine learning forman parte de la inteligencia artificial, estos temas son bastante diferentes entre sí. Es decir,

5 La inteligencia artificial general es el nombre que recibe una máquina con habilidades cognitivas similares a los seres humanos, donde estas habilidades cognitivas son aplicables a cualquiera tarea.

aunque en nuestro campo de estudio de la inteligencia artificial existen varias ramas que pueden tener aspectos similares entre sí, no es del todo correcto tratar estos tres elementos como sinónimos. Con esto en mente, en la siguiente sección, analizaremos las principales diferencias entre estas ramas.

Diferencias entre inteligencia artificial, machine learning y deep learning

Como se mencionó anteriormente, el campo de estudio de la inteligencia artificial se ramifica en áreas más específicas. A pesar de que existen otras variaciones, las dos ramas principales de estudio en las que se divide la inteligencia artificial son el aprendizaje automático (machine learning) y el aprendizaje profundo (deep learning). Es frecuente confundir estos tres términos entre sí, pero tienen características principales que permiten distinguirlos claramente. Por lo tanto, es un error considerar estos conceptos como sinónimos.

La inteligencia artificial, definida en la sección anterior como "sistemas o máquinas que imitan la inteligencia humana para realizar tareas" (Oracle, 2023), engloba diversas ramas y variantes en su campo de estudio. Entre los ejemplos mencionados anteriormente, el aprendizaje automático y el aprendizaje profundo destacan como las dos principales ramas de la inteligencia artificial, siendo objeto de estudio e interés para la comunidad científica.

El aprendizaje automático es una rama de la inteligencia artificial que se centra en desarrollar métodos que permitan a los agentes artificiales aprender a partir de ejemplos y/o experiencias, según la definición de algunos autores (Pineda, s.f.). En resumen, se refiere al proceso a través del cual los modelos informáticos pueden "aprender" utilizando información de aprendizaje proporcionada.

Puede resultar confuso comprender cómo un modelo informático logra aprender de tal manera, pero este concepto será explorado en detalle a lo largo de los temas que se abordan en este libro. Por otro lado, el aprendizaje profundo es un tipo de aprendizaje automático que utiliza redes neuronales artificiales para permitir que los sistemas digitales aprendan y tomen decisiones basadas en datos no estructurados y sin etiquetar (Microsoft, 2023). Este campo de estudio es considerado tanto una rama de la inteligencia artificial general como del aprendizaje automático

Aunque puede parecer complicado entender la división de las ramas de la inteligencia artificial, en realidad, es más sencillo de lo que parece. Podemos observar de manera más visual la organización de las principales ramas de la inteligencia artificial en el siguiente gráfico, donde se muestra qué rama comprende a las demás y cuál es la relación entre ellas.

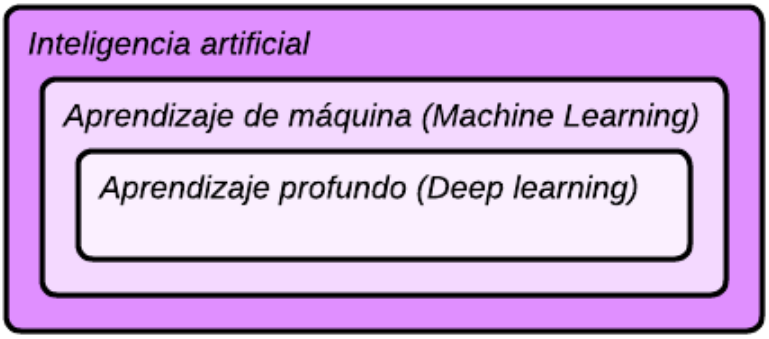


Ilustración 15 – Correlación y dependencia de las principales ramas de la inteligencia artificial

En este libro, exploraremos las distintas ramas de la inteligencia artificial, brindándonos la oportunidad de conocer sus fundamentos esenciales para obtener una comprensión general. Aunque no nos adentraremos en detalle en cada tema específico, pues no es el objetivo principal del libro, esta base nos permitirá comprender mejor las secciones posteriores relacionadas con las tres ramas principales de la IA.

Ideas principales de la inteligencia artificial, conceptos teóricos e historia

La Inteligencia Artificial se centra en la interpretación o imitación del proceso de toma de decisiones humanas mediante máquinas. Hay diversos enfoques para modelar este comportamiento, incluida la simulación de funciones cerebrales y su relación con la toma de decisiones humanas. Antes de analizar estos modelos, es esencial comprender la introducción y conceptos clave de la Inteligencia Artificial. Además, resulta valioso revisar la historia de la disciplina y su evolución a lo largo del tiempo. La Inteligencia Artificial engloba técnicas como el aprendizaje automático y las redes neuronales, las cuales se describirán con más detalle en secciones posteriores del libro.

Los orígenes de la inteligencia artificial se remontan a la década de 1950, cuando se acuñó el término (Rosenblatt, 1958). Al principio, la falta de comunicación efectiva entre la comunidad científica y la escasa popularidad del tema hicieron que la inteligencia artificial no despertara interés suficiente.

Antes de que la inteligencia artificial fuese conocida y estudiada como tal, la comunidad científica mostraba un interés particular en este tema, pero no lograba desarrollarse completamente. Aun así, realmente no era del todo sorprendente, a lo largo del tiempo se desarrolló lo que hoy se conoce como "sistemas expertos", los cuales son elementos fundamentales para la inteligencia artificial actual. Un sistema experto es un programa informático diseñado para imitar parcialmente el comportamiento de toma de decisiones humanas. En cierto modo, los sistemas expertos tienen una relación indirecta con la inteligencia

artificial y pueden considerarse precursores de esta, dado que ambos comparten el concepto de tomar decisiones mediante la simulación y esquematización del razonamiento humano (Liao, 2005).

Poco tiempo después de que se comprendió la importancia de los sistemas expertos para la comunidad científica en general, se observó un creciente interés en la investigación de la inteligencia artificial. La comunidad científica y académica comenzaron a explorar diversas aplicaciones de la IA, y algunos eventos clave contribuyeron a modificar la percepción pública acerca de la IA y su potencial impacto en la sociedad y la vida humana.

Ilustración 16 - Una máquina Symbolics 3640 Lisp: una de las primeras plataformas para sistemas expertos (Umbricht & Friend, 2010).

Es importante destacar que la historia de la inteligencia artificial no siempre ha sido un campo de estudio que ha generado interés y asombro en la comunidad científica. De hecho, la inteligencia artificial atravesó un periodo conocido como el "invierno de la inteligencia artificial". Durante este período, la comunidad científica descubrió que, debido a factores como la falta de capacidad computacional y el escaso desarrollo en este campo, la inteligencia artificial no producía los resultados esperados y no tenía la relevancia prevista en los proyectos de investigación donde se utilizaba (Shin, 2019).

Por un tiempo, el interés en el campo de la inteligencia artificial disminuyó drásticamente, lo que resultó en una pausa en su avance y una ausencia en el estudio científico y académico. Esta situación persistió durante varios años y retrasó significativamente el progreso en la investigación en inteligencia artificial. A pesar de las preocupaciones iniciales acerca de la inteligencia artificial, en la actualidad se han alcanzado importantes avances y se ha aumentado la financiación en proyectos relacionados con esta tecnología (Mondal, 2020).

Muchas entidades muestran un interés en financiar el desarrollo de la inteligencia artificial, gracias a los notables progresos impulsados por el surgimiento de internet y el fácil acceso a la información. Esto ha resultado en beneficios directos, como el crecimiento y avance exponencial en diversos aspectos de la tecnología y la ciencia en general.

Todo esto ha facilitado la creación de un entorno y capacidad de procesamiento adecuada para el desarrollo y ejecución de modelos de inteligencia artificial poderosos y relevantes para la comunidad científica. Incluso las personas que realmente no se involucran mucho en el mundo científico o académico, experimentan mayor confianza, seguridad y familiarización con modelos de inteligencia artificial actuales (Glikson & Woolley, 2020).

La popularidad de la inteligencia artificial comenzó a resurgir y ganar interés en la investigación a lo largo del tiempo, más precisamente, este renovado interés surgió durante la década de los 90 e inicios del 2000. Al mismo tiempo, a mediados de la década de 2010, aparecían cada vez más bases de datos recientes con enormes cantidades de información, nuevos algoritmos y diferentes métodos para medir la precisión de los algoritmos en los distintos modelos de inteligencia artificial. En ese entonces, los modelos de inteligencia artificial adoptaban diferentes enfoques de implementación, ya que principalmente se trataba de modelos diseñados para cumplir tareas específicas (Russell & Norvig, 2022). Con el paso del tiempo, diferentes ramas de estudio de la inteligencia artificial se volvieron cada vez más populares tanto en el mundo industrial como en el ámbito científico y de investigación. En este caso, podemos mencionar, por ejemplo, los modelos centrados en el procesamiento del lenguaje natural y la visión por computadora, los cuales representaron un enorme cambio en las aplicaciones que pueden ofrecer los modelos de inteligencia artificial. Esto, junto con el lanzamiento de diversos elementos del internet de las cosas, hizo que diferentes modelos de inteligencia artificial se integraran incluso en los hogares de muchas personas (Deng, Bradski, Coates, & Shah, 2017).

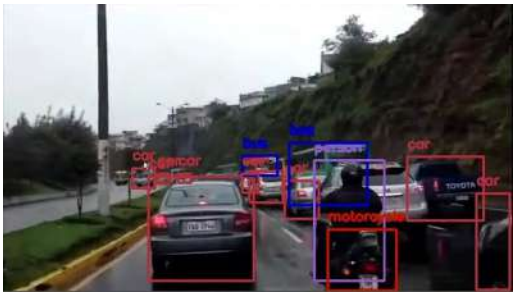
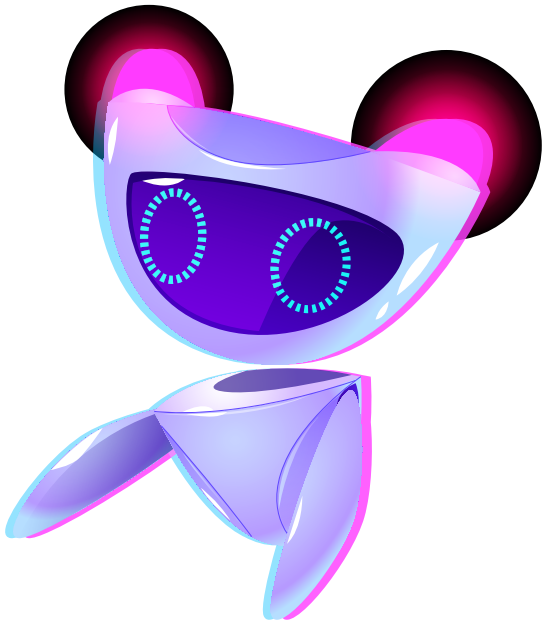


Ilustración 17 – Ejemplo visual de visión de computadora (Comunidad de Software Libre Hackem [Research Group], 2020).

El futuro de la inteligencia artificial es incierto; sin embargo, podemos estar seguros de que el desarrollo de muchos nuevos modelos y técnicas, así como la investigación en este campo, es un proceso que se espera continúe creciendo a lo largo del tiempo. Por ello, es una época increíble para presenciar el avance y desarrollo de estos modelos.

Conclusión del capítulo



Este capítulo ha ofrecido una profunda exploración en la complejidad de la inteligencia artificial, subrayando su naturaleza multifacética. La discusión sobre la inteligencia artificial general y específica ilustra la gran diversidad en los modelos y aplicaciones de la IA, enfatizando que aún hay desafíos en la creación de un modelo verdaderamente general. Se ha realizado un análisis detallado de las ramas principales como el deep learning y el machine learning, aclarando sus diferencias y cómo se interrelacionan bajo el paraguas de la inteligencia artificial. Además, se trazó la historia de la inteligencia artificial, desde su nacimiento hasta los obstáculos y logros recientes, proporcionando una visión integral de su desarrollo y estado actual. La revisión histórica resalta la transformación de la inteligencia artificial desde un campo poco comprendido hasta una tecnología omnipresente y esencial en la sociedad moderna. Con ello, el capítulo fortalece la base necesaria para comprender cómo la inteligencia artificial interactúa y contribuye a diversos campos, preparando el terreno para los temas específicos que se tratarán en las siguientes secciones del libro.

¿QUÉ ES EL APRENDIZAJE DE MÁQUINA?

Objetivos del capítulo

El objetivo principal de este capítulo es ofrecer una comprensión detallada y completa del aprendizaje de máquina, una de las ramas fundamentales de la inteligencia artificial. Se busca explorar sus conceptos clave, algoritmos y procesos, y destacar las diferencias entre el aprendizaje de máquina, la inteligencia artificial y el deep learning. El lector obtendrá una visión profunda de cómo funciona el aprendizaje de máquina, cómo se aplica y cómo simula el proceso de aprendizaje humano. También se presentarán algunos algoritmos esenciales en el aprendizaje automático, como la regresión lineal, y se explorarán diversas aplicaciones prácticas del aprendizaje automático en diferentes campos.

Hasta el momento, hemos tenido la oportunidad de estudiar y analizar de manera general qué es la inteligencia artificial. Sin embargo, es momento de empezar a explicar las diferentes ramas principales e importantes que componen la inteligencia artificial. Después de todo, es esencial comprender cada vez más las diferencias entre estos conceptos: IA, machine learning y deep learning.

Por ahora, podemos decir que el aprendizaje de máquina es la rama de inteligencia artificial que se lleva a cabo mediante algoritmos. Estos algoritmos tienen la capacidad de simular el razonamiento y la toma de decisiones, de una

forma un poco similar al razonamiento y la toma de decisiones por seres humanos. El objetivo final del aprendizaje de máquina es entender cómo razona nuestro cerebro y, luego, implementar ese proceso en un sistema computacional.

El proceso que suele llevarse a cabo para aplicar un modelo de machine learning consta de varias etapas. Para visualizar mejor esto, el modelo parte de lo que se conoce como datos de entrenamiento. Estos datos corresponden a la información en cuestión, de la cual el modelo intentará buscar las mejores inferencias para comprender cuáles son las reglas y similitudes que comparte este tipo de datos.

Luego, la aplicación del modelo. Las personas pueden aplicar muchos algoritmos y modelos de machine learning a diferentes tipos de casos.

Dependiendo del algoritmo o modelo en cuestión que se haya escogido, los pasos y la forma de identificar características principales entre los datos de entrenamiento pueden llegar a variar. Por eso es importante elegir correctamente el algoritmo adecuado para la tarea o problema específico que se intenta resolver.

Finalmente, en base a los datos de entrenamiento y el algoritmo seleccionado, se evalúa el modelo con el objetivo de determinar su precisión frente a nuevos datos y la predicción de resultados, este resultado varía en función del proceso de entrenamiento que haya experimentado. Existen muchos más elementos a considerar en el ciclo de vida de un modelo de aprendizaje automático, por ejemplo, la elección del marco de trabajo que se utilizará, el monitoreo constante del rendimiento del modelo, la planificación previa al desarrollo y muchas otras etapas (Hovsepyan, 2022).

Podemos afirmar que las mencionadas anteriormente son las etapas principales en el desarrollo de un modelo de aprendizaje automático en su ciclo de vida. Por ahora puede parecer algo complejo, pero podemos ilustrar esto con el siguiente gráfico.

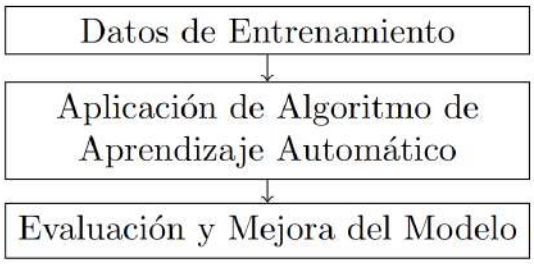


Ilustración 18 - Proceso de Aprendizaje Automático.

Aunque al principio pueda no parecer relacionado, realmente podemos decir que el proceso realizado dentro de los modelos de machine learning tiene mucha similitud con el proceso que nosotros, los seres humanos, llevamos a cabo para

aprender algo. Por eso, es importante para el desarrollo de este tipo de modelos tener una idea previa de cómo funciona el aprendizaje, para poder entender y aplicar directamente estos sistemas. Entre las características distintivas y cruciales del machine learning se encuentra la capacidad de imitar el proceso de aprendizaje durante la ejecución del modelo a través de algoritmos, técnicas y esquemas matemáticos.

El proceso de aprendizaje a través de machine learning puede parecer complejo para algunas personas. Los algoritmos empleados, así como las diversas técnicas y prácticas que se implementan para lograr un aprendizaje efectivo en los modelos, no implican necesariamente una simulación directa de cómo funcionan las neuronas del cerebro. Por ejemplo, en el caso del deep learning, sí se busca imitar directamente el funcionamiento de las conexiones neuronales de nuestro cerebro.

Sin embargo, en el machine learning, en lugar de simular el funcionamiento de nuestras conexiones cerebrales o el proceso de las diferentes inferencias que nuestro cerebro puede realizar biológicamente, se emplean distintos modelos matemáticos y algoritmos que se acercan al proceso de aprendizaje que llevamos a cabo los seres humanos. Es decir, en lugar de esquematizar directamente el funcionamiento del cerebro, el machine learning se basa en algoritmos y técnicas diversas que permiten obtener resultados mediante el aprendizaje a partir de varios datos iniciales de entrenamiento (Reese, 2017) (Jakhar & Kaur, 2020).

Los principales algoritmos utilizados en el aprendizaje automático están estrechamente vinculados a conceptos matemáticos. Algunas de las técnicas más empleadas incluyen la regresión lineal (Ksenija, Šverko, Salarić, Martinović, & Lucijanić, 2021), un tipo de modelo de aprendizaje de máquina que busca encontrar la mejor función matemática que se ajuste a la distribución de los diferentes tipos de datos presentes en el proceso de entrenamiento, para así realizar una clasificación más adecuada de los posibles nuevos y desconocidos datos. La regresión lineal es solo una herramienta para encontrar relaciones lineales simples entre dos o más variables; explicaremos su funcionamiento con mayor detalle en secciones posteriores del libro.

Otros algoritmos utilizados en el aprendizaje automático incluyen la regresión logística, los árboles de decisión y muchos otros modelos y algoritmos que intentan resolver algún tipo de problema específico. Por ahora, puede parecer mucha información para aprender desde el inicio, sin embargo, estudiaremos más a detalle y profundizaremos en los diferentes tipos de algoritmos populares empleados en el machine learning, para poder tener una idea de cómo funcionan y por qué son tan populares y utilizados (Mahesh, 2018).

Es comprensible que algunas personas encuentren confuso el concepto de la capacidad de los modelos matemáticos para reproducir o simular el aprendizaje humano, uno de los procesos más complejos y enigmáticos del cerebro. Para abordar mejor este tema, es fundamental comprender claramente qué es el

aprendizaje de seres humanos y cómo se lleva a cabo en cada individuo. Aunque todos los seres humanos tienen la capacidad de aprender, el mecanismo completo de cómo ocurre sigue siendo un enigma sin resolver.

Con esto en mente, surge la pregunta de cómo es posible replicar este proceso mediante algoritmos simples. A continuación, se presenta una breve explicación de cómo se puede lograr esto. Podemos afirmar que un algoritmo como regresión lineal simula el aprendizaje utilizando una cantidad definida de datos. Es decir, se le proporcionan una cantidad de datos de entrenamiento y, en base a ellos, intenta trazar una línea recta que mejor se ajuste a los mismos.

No obstante, es importante entender que este algoritmo no emula completamente el aprendizaje humano. De hecho, la regresión lineal es un algoritmo bastante sencillo y simple en comparación con la complejidad del aprendizaje humano.

Machine Learning: Algoritmo de regresión lineal

El modelo de regresión lineal es un algoritmo fundamental en el aprendizaje de máquina. Aunque se haya presentado previamente una introducción básica de su funcionamiento puede ser difícil de comprender para aquellos sin conocimiento previo sobre algoritmos y los elementos que se utilizan en la ejecución y desarrollo de estos modelos.

En esta sección, se examinará en detalle el funcionamiento de la regresión lineal. Se profundizará en su concepto y en los diferentes pasos que sigue para hacer predicciones. Además, se explorarán las diferentes formas en que se puede mejorar su relevancia en el campo de la inteligencia artificial y los modelos de regresión lineal en general.

Una vez que se tenga una comprensión más clara del concepto superficial de un modelo de regresión lineal, se puede examinar de manera más directa su funcionamiento y cómo se puede aplicar en un ejemplo para comprender su utilidad en el campo de la inteligencia artificial.

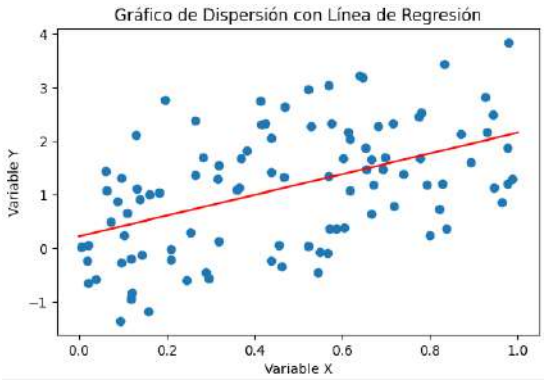


Ilustración 19 - Relación entre las variables X e Y con la línea de regresión

La regresión lineal es una técnica empleada en aprendizaje automático que busca hallar la línea que mejor representa la relación entre variables. Aunque el concepto pueda parecer complicado al principio, veremos superficialmente este tema para que quede claro, aclarando cualquier duda sobre su funcionamiento.

Podemos analizar los fundamentos de este algoritmo de regresión lineal y entender cómo funciona un modelo de este tipo. Un modelo de regresión lineal busca establecer una representación matemática que relacione una variable predictiva con un conjunto de variables (datos) que influyen en ella. Imaginemos que queremos analizar cómo distintos factores ambientales afectan la tasa de reproducción de un grupo de ardillas en un bosque. Para ilustrar de manera más clara y gráfica cómo funciona el modelo, consideremos el siguiente ejemplo.

Supongamos que tenemos información sobre el tamaño del bosque (x_1), la cantidad de árboles (x_2) y el tipo de árboles (x_3) en el hábitat de las ardillas. La tasa de reproducción de las ardillas (y) es lo que queremos predecir. El modelo de regresión lineal busca encontrar una ecuación que relacione estas variables y que tenga la forma:

$$y = a + b_1 * x_1 + b_2 * x_2 + b_3 * x_3$$

Donde "a" es una constante y "b_1", "b_2" y "b_3" son los coeficientes que determinan cómo cada variable (x_1, x_2, x_3) influye a la tasa de reproducción (y). El objetivo es encontrar los valores de "a", "b_1", "b_2" y "b_3" que mejor ajusten la ecuación a los datos disponibles. Una vez que se ha encontrado la ecuación que mejor ajusta los datos, podemos utilizarla para predecir la tasa de reproducción de las ardillas en diferentes situaciones, cambiando los valores de x_1, x_2 y x_3 según las condiciones del bosque.

Aplicaciones del Machine Learning

En esta sección, exploraremos diversos ejemplos de aplicaciones del aprendizaje automático, examinando sus impactos y resultados en distintas tareas. El aprendizaje automático posee un amplio espectro de aplicaciones en campos relevantes, desde la gestión de correo electrónico hasta procesos más complejos, como la predicción de precios y la medicina. Esto implica que el aprendizaje automático puede emplearse en numerosas áreas de estudio y empresas para aumentar la eficiencia y automatización de tareas.

Algunas personas podrían subestimar la capacidad de los algoritmos de aprendizaje automático para resolver problemas cotidianos. Un ejemplo de ello es el uso de modelos de aprendizaje automático para predecir la duración de las llamadas telefónicas. Aunque algunos pueden considerar esta aplicación trivial o poco importante, en realidad, utilizar modelos de machine learning para predecir la duración de las llamadas telefónicas ofrece múltiples ventajas, ya que el uso de estos modelos puede mejorar la eficiencia de la red telefónica, la experiencia del usuario y la capacidad de análisis de datos.

Otra de las aplicaciones que pueden tener los modelos y algoritmos de machine learning en ejemplos de la vida cotidiana como utilizar modelos de machine learning para predecir el tiempo de uso de dispositivos inteligentes en función de un contexto específico (Sarker, Kayes, & Watters, 2019).

El aprendizaje automático es una tecnología en constante evolución, y su aplicación se expande cada vez más a diferentes industrias y áreas de estudio. Una de sus mayores ventajas es su habilidad para analizar grandes cantidades de datos y aprender de ellos en tiempo real. Además de la aplicación mencionada para predecir la duración de las llamadas telefónicas, el aprendizaje automático también se utiliza en la detección de spam en correos electrónicos. Los filtros de spam emplean algoritmos de aprendizaje automático para identificar correos no deseados y marcarlos como spam, lo que ayuda a los usuarios a mantener sus bandejas de entrada organizadas y evitar la avalancha de correos no deseados.

En medicina, el aprendizaje automático se utiliza para mejorar la precisión en la identificación de enfermedades y facilitar la toma de decisiones clínicas. Por ejemplo, los modelos de aprendizaje automático pueden analizar imágenes médicas, como radiografías y tomografías, y ayudar a los médicos a identificar lesiones y enfermedades con mayor precisión.

El aprendizaje automático también tiene una influencia significativa en la industria financiera, especialmente en la prevención de fraude en tarjetas de crédito. Los modelos de aprendizaje automático pueden analizar patrones en datos de transacciones y detectar comportamientos sospechosos que puedan indicar fraude, lo que permite a las instituciones financieras proteger a sus clientes y reducir pérdidas debido al fraude. Las aplicaciones del aprendizaje automático

son bastantes, abarcando desde la gestión de correo electrónico hasta la medicina y la industria financiera. La capacidad de estos modelos al analizar grandes cantidades de datos y aprender de ellos en tiempo real permite mejorar la eficiencia y la precisión en una variedad de tareas, lo que conduce a mejores resultados y soluciones más efectivas (Shinde & Shah, 2018).

Con esta información, ahora podemos comprender mejor el aprendizaje automático y los distintos modelos y algoritmos que se utilizan para llevar a cabo diversas tareas que involucran o requieren el uso de este tipo de modelos. Sin embargo, como hemos mencionado previamente, en realidad estos modelos abordan una amplia gama de temas importantes que merecen ser estudiados.

No obstante, es imposible analizar todas las técnicas de aprendizaje automático existentes y estudiarlas en profundidad en este libro sin apartarnos de nuestro objetivo principal. A pesar de ello, esto nos permite tener una idea más clara, aunque superficial, del funcionamiento de estos algoritmos y modelos de aprendizaje automático. A pesar de que los algoritmos de este tipo pueden ser bastante fascinantes, en realidad aún no hemos explorado a profundidad el funcionamiento de estos modelos ni su aplicación en el mundo real. Esto no implica que estos algoritmos sean incapaces de lograrlo, y como mencionamos previamente, esto representa solo una pequeña fracción de las múltiples e innumerables posibilidades que este campo de estudio puede llegar a ofrecer.

Hasta este punto, hemos tenido la oportunidad de analizar los temas más elementales, sin embargo, podemos afirmar que existen muchos más temas por investigar y numerosos conceptos esenciales, potentes e intrigantes por descubrir para abordar este tipo de tareas en futuras secciones.

Por el momento, podemos afirmar que poseemos las bases fundamentales para comprender de manera simple qué es el aprendizaje automático y por qué esta área es tan popular en el campo de la inteligencia artificial. No obstante, todavía hay muchos más temas que debemos tener en cuenta y comenzar a explorar. No debemos preocuparnos excesivamente si al principio resulta algo confuso, pues en este libro, antes de llevar a la práctica lo que deseamos alcanzar al final (aprender cómo utilizar Stable Diffusion para generar imágenes empleando inteligencia artificial), es crucial que adquiramos un conocimiento más profundo y claro sobre las distintas ramas y el funcionamiento de la inteligencia artificial.

Conclusión del capítulo

En este capítulo, hemos logrado desentrañar los aspectos fundamentales del aprendizaje de máquina, explorando su relación con la inteligencia artificial y su similitud con el aprendizaje humano. Se han descrito varios modelos y algoritmos clave, proporcionando una visión detallada de cómo funcionan y por qué son relevantes en la actualidad. Además, se han ilustrado aplicaciones reales y ejemplos de cómo el aprendizaje de máquina está transformando diferentes industrias y áreas de estudio. La sección ha servido para demystificar algunos de los conceptos complejos y enigmáticos que rodean al aprendizaje de máquina, y ha presentado una introducción sólida y accesible para aquellos interesados en profundizar en este campo crucial de la tecnología contemporánea.

¿QUÉ ES EL APRENDIZAJE PROFUNDO?

Objetivos del capítulo

El objetivo principal de este capítulo es proporcionar una visión detallada y comprensible de lo que es el aprendizaje profundo, con énfasis en conceptos clave como las redes neuronales, perceptrones, y cómo estos elementos imitan la función cerebral humana. Se pretende ilustrar el funcionamiento de las conexiones neuronales artificiales y explicar cómo estos conceptos se han aplicado en modelos como el GPT-3 y Stable Diffusion. Además, se busca esclarecer los aspectos fundamentales de la construcción y funcionamiento de perceptrones y redes neuronales, y presentar de forma accesible cómo estos elementos interactúan para resolver tareas complejas. Finalmente, se incluyen gráficos e ilustraciones para facilitar una mejor comprensión de estos conceptos para el lector.

La inteligencia artificial ha ganado gran popularidad en la actualidad, particularmente en el campo del aprendizaje profundo o "deep learning". Este auge ha sido posible gracias a la aparición de diversos modelos, como el GPT-3 y el Stable Diffusion, siendo este último el enfoque que abordaremos en este libro. Es importante destacar que estos dos modelos son solo algunos de los más destacados en la actualidad; no obstante, la lista de modelos basados en deep learning y sus diversas aplicaciones es bastante extensa.

La popularidad de los modelos de inteligencia artificial basados en deep learning no es casualidad, se justifica debido a que se justifica debido a las complejas tareas que estos modelos pueden llevar a cabo, particularmente en el ámbito del procesamiento del lenguaje natural, donde se puede afirmar que dichos modelos destacan. Por lo tanto, es comprensible que estos modelos de inteligencia artificial resulten tan fascinantes y sorprendentes para muchos.

Esta sección presenta una visión general de las causas tras la popularidad de los modelos de inteligencia artificial basados en deep learning, así como una introducción breve a su funcionamiento, conceptos clave e ideas relevantes. Se espera proporcionar una comprensión más clara y precisa de estos nuevos conceptos.

Antes de profundizar en el funcionamiento de los modelos de inteligencia artificial basados en deep learning, es fundamental comprender algunos conceptos básicos, como las redes neuronales y los perceptrones, entre otros. Estos conceptos son esenciales para una comprensión clara del funcionamiento de estos modelos.

El deep learning es una rama de la inteligencia artificial que se enfoca en la creación de sistemas o algoritmos que puedan resolver tareas complejas. Los modelos de deep learning se diferencian de otras técnicas y modelos de inteligencia artificial en la solución de problemas sin la necesidad de una programación detallada previa del sistema. En lugar de ser programados específicamente para hacerlo, los modelos de deep learning aprenden por sí mismos cómo solucionar las tareas.

El funcionamiento de los modelos de inteligencia artificial basados en aprendizaje profundo está estrechamente relacionado con el funcionamiento del cerebro humano. De hecho, se puede decir que los modelos de deep learning tienen un comportamiento parcialmente similar al proceso de aprendizaje y razonamiento cerebral humano. Por lo tanto, es importante comprender cómo funciona el cerebro humano para comprender por qué existe esta similitud. No significa que debamos ser expertos en anatomía, pero comprender el funcionamiento del cerebro puede ayudarnos a entender mejor esto.

El funcionamiento del cerebro humano se lleva a través de conexiones de neuronas. Este proceso incluye el funcionamiento electroquímico del cerebro, permitiendo procesos como la memoria y el desarrollo de la interpretación cerebral abstracta.

La comunicación entre las neuronas se realiza a través de un proceso electroquímico en el que se transmiten señales eléctricas y procesos químicos para compartir información. Todo esto sucede en una fracción de segundo y se repite millones de veces en el proceso cerebral humano.

Deep Learning: Conceptos clave e ideas principales

En esta sección, exploraremos a fondo el funcionamiento de los modelos de inteligencia artificial basados en aprendizaje profundo. Cabe recordar que un modelo de inteligencia artificial que usa Deep Learning modela el funcionamiento de las conexiones cerebrales humanas y la transmisión de señales del sistema cerebral. Esto se logra mediante estructuras denominadas redes neuronales, que representan un esquema de las conexiones de las neuronas en el cerebro humano.

La unidad básica de un sistema de conexión de redes neuronales se conoce como perceptrón, y consiste en un solo nodo que forma parte de una red neuronal. Un nodo es un elemento de procesamiento de información básico que puede recibir, transformar y enviar señales a otros nodos o capas en la red. En este contexto, un perceptrón puede considerarse una representación simplificada de una sola célula neuronal en el cerebro y, por lo tanto, unidad dentro de un modelo más amplio y complejo.

Un perceptrón en una red neuronal intenta imitar el funcionamiento de las neuronas, ya que el proceso de recibir, transformar y enviar señales a otros nodos de parte de un perceptrón es bastante similar al proceso que realiza una sola neurona en nuestro cerebro. Los componentes clave de un perceptrón incluyen la recepción de información, la transformación mediante funciones y la salida de datos.

A continuación, se presenta un gráfico que ilustra la estructura más simple del perceptrón, en el cual se pueden apreciar claramente estos componentes.

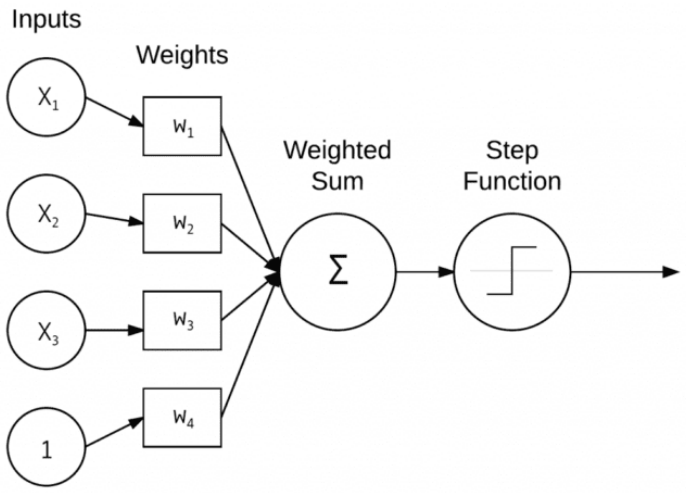


Ilustración 20 – Representación gráfica de un perceptrón (Rosebrock, 2021).

Por ahora, puede ser algo confuso entender la relación que tienen estos perceptrones con las conexiones cerebrales. Sin embargo, tal como mencionamos anteriormente, podemos interpretar la manera en la que se organiza este perceptrón con la estructura misma de una sola neurona.

Es decir, podemos buscar la relación que tiene este tipo de gráfico con las formas y las diferentes partes que componen una célula neuronal, observando así por qué se entiende que las partes de un perceptrón son una especie de esquematización del funcionamiento de las neuronas reales de nuestro cerebro. Podemos comprender mejor lo que estamos mencionando con el siguiente gráfico.

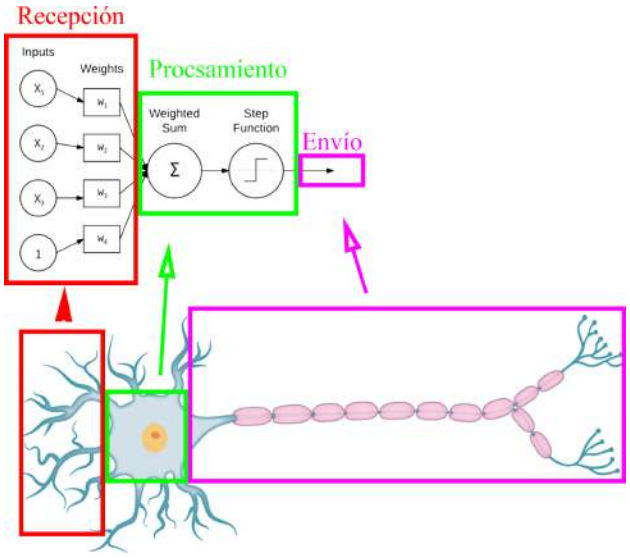


Ilustración 21 – Relación entre el perceptrón y una neurona

Con esto, ahora podemos tener una idea más clara acerca de la relación que pueden presentar los perceptrones con las neuronas de nuestro sistema nervioso. Así, observamos las diferentes partes que componen a una neurona real y, de la misma manera, cómo estas mismas partes son representadas dentro del perceptrón y sus diversas funciones.

A pesar de que este gráfico muestra las diferentes partes que componen un perceptrón, algunos elementos pueden resultar confusos para aquellos sin experiencia previa en el funcionamiento de las redes neuronales. A continuación, se proporcionan breves explicaciones de conceptos como la sumatoria de pesos y la función de activación.

Durante el proceso de entrada de datos, el perceptrón recibe un conjunto de datos al que llamaremos "S" por ahora, los cuales son generados por otros perceptrones. Cada entrada individual en el perceptrón tiene un valor numérico conocido como "peso", que indica qué parte de la entrada de una neurona es más relevante y cuánta información debe considerarse. Un peso con un valor numérico

de 1 representa la totalidad de la información entrante al perceptrón, mientras que un peso de 0 indica que no hay entrada de información en esa neurona. Cada perceptrón puede recibir múltiples conjuntos de datos como entrada, dependiendo del número de neuronas que lo conformen. Por ahora, puede parecer confuso, pero explicaremos detalladamente el proceso paso a paso para resolver cualquier duda que pueda surgir.

Una vez que el perceptrón ha recibido los datos y ha realizado la multiplicación por los pesos correspondientes de cada entrada, se procede a la etapa de procesamiento de datos.

Primero, se suman los valores de entrada multiplicados por sus pesos correspondientes para obtener un valor único. Luego, se aplica una función de activación a este resultado. Este proceso se puede describir mediante la siguiente fórmula:

$$y = f\left(\sum_{i=1}^{n} w_i x_i \right)$$

Aunque el perceptrón es la unidad más básica en el aprendizaje profundo como rama del aprendizaje de máquina, por sí solo no puede realizar tareas muy importantes. Al igual que una sola neurona en el cerebro humano, un perceptrón necesita estar conectado e interrelacionado con otros perceptrones para realizar operaciones y comenzar a ejecutar tareas más complejas relacionadas con la inteligencia artificial.

En este contexto, resulta esencial comprender cómo emplear estos perceptrones para lograr una interconexión efectiva. De este modo, podemos utilizar redes neuronales para realizar una lista más extensa de tareas, y refleja el potencial y la función principal de estas estructuras en la inteligencia artificial. Con esto en mente, podemos apreciar aún más el motivo por el cual estos modelos reciben tanta atención y popularidad en la comunidad de investigación de la IA.

Es fundamental entender cómo implementar lo mencionado previamente. Las estructuras de perceptrones se basan, fundamentalmente en tres componentes: la entrada de datos, el procesamiento mediante funciones y la salida de datos. Imaginemos una estructura simple que contenga una sola capa de conexiones entre perceptrones; esto resultaría en lo que se denomina una red neuronal.

El siguiente gráfico ilustra lo que acabamos de mencionar, mostrando dos capas de redes neuronales: la primera capa cuenta con tres nodos, mientras que la segunda contiene cuatro. Todos los nodos están interconectados. En este caso, cada uno de los "nodos", representados como círculos en la imagen, hace alusión a un perceptrón, lo que indica que estos perceptrones están interconectados entre sí.

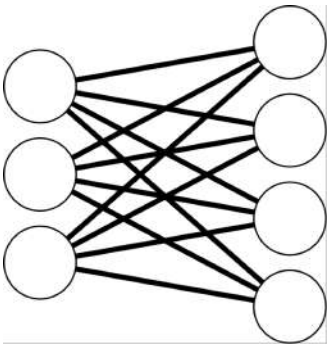


Ilustración 22 – Redes neuronales de dos capas interconectadas

Uno de los aspectos fundamentales en el funcionamiento de las redes neuronales son las capas. Cada modelo basado en redes neuronales se organiza en capas, de manera similar a lo descrito previamente. La particularidad de este enfoque radica en que cada capa genera una salida, la cual se emplea como entrada para la siguiente capa, y así sucesivamente. No obstante, las neuronas no establecen conexiones con capas diferentes ni con otras neuronas pertenecientes a la misma capa.

En otras palabras, mantienen únicamente una relación directa con la capa anterior, que les provee de entradas, y con la capa siguiente, a la cual suministran entradas a través de sus salidas, pero no se relacionan directamente con otras capas.

Este tipo de estructuras de redes neuronales se conoce como redes neuronales convencionales, caracterizadas por una interconexión más lineal entre las neuronas. La finalidad de este libro no consiste en proporcionar información exhaustiva y compleja sobre el proceso que permite transformar estas estructuras de redes neuronales en modelos de inteligencia artificial capaces de realizar tareas específicas. A continuación, se presenta un resumen conciso y general sobre el funcionamiento interno de las redes neuronales en el contexto de la inteligencia artificial, así como su capacidad para llevar a cabo tareas relacionadas en este campo.

Conclusión del capítulo

En este capítulo, hemos explorado en profundidad la naturaleza del aprendizaje profundo, sus componentes esenciales y su relación con el cerebro humano. Hemos desglosado la estructura y funcionamiento de perceptrones y redes neuronales,

utilizando gráficos e ilustraciones para ayudar en la comprensión. Además, hemos resaltado la importancia y popularidad de los modelos de aprendizaje profundo en la actualidad, especialmente en el procesamiento del lenguaje natural. Este capítulo ha servido como una introducción detallada y didáctica, preparando al lector para una comprensión más avanzada de cómo el aprendizaje profundo se aplica en diversos campos, incluyendo la generación de imágenes y traducciones de imagen a imagen, como se examina en otros capítulos del libro. La información presentada en este capítulo pone las bases para un entendimiento más profundo de las técnicas y aplicaciones específicas que se discutirán en los capítulos subsiguientes.

TIPOS DE REDES NEURONALES

Objetivos del capítulo

Este capítulo tiene como objetivo principal explicar los diferentes tipos de redes neuronales, con especial énfasis en la arquitectura de red neuronal "feedforward". La sección introducirá las distintas partes que componen estas estructuras, incluyendo las capas de entrada, ocultas y de salida, así como las neuronas artificiales, los pesos y la función de activación. Se describirá la relevancia y aplicabilidad de cada uno de estos elementos en el proceso de construcción y entrenamiento de una red neuronal. Además, se proporcionará una visión clara de las representaciones gráficas comunes de las redes neuronales y su interpretación.

Ahora, veremos dentro de esta sección los diferentes tipos de redes neuronales que podemos encontrar. Sin embargo, esta sección podría llegar a parecer algo trivial, después de todo, ¿cuál es la importancia de poder ver otros tipos de redes neuronales cuando realmente el concepto de la conexión cerebral entre las neuronas de nuestro cerebro es solo una? Es decir, solamente tendríamos que preocuparnos por entender y esquematizar una sola idea detrás del funcionamiento de las redes neuronales, ¿verdad? Esta idea puede parecer algo cierta, pero la realidad es que en el ámbito de la

computación es importante tener diferentes tipos y formas de conexión para poder realizar distintos tipos de redes neuronales.

Después de todo, el funcionamiento de los diferentes elementos y fenómenos que ocurren dentro de nuestro cerebro sigue siendo un misterio para muchas personas. Realmente, no podemos afirmar que las esquematizaciones que hemos mencionado hasta ahora, utilizadas en una red neuronal simple, emulen completamente las funciones y diversas capacidades que las conexiones neuronales pueden presentar en un cuerpo biológico. Por lo tanto, se vuelve crucial generar distintos tipos de redes neuronales con variaciones en sus conexiones.

Dependiendo del tipo de red neuronal que se haya elegido, los resultados y las capacidades que ofrecen estos tipos de redes para realizar tareas pueden llegar a variar, ofreciendo diferentes rendimientos y resultados más rápidos o eficientes dependiendo del tipo de red neuronal y la tarea en cuestión. Estos "tipos de redes neuronales" también se conocen como arquitecturas de redes neuronales, lo cual es una terminología más adecuada. En términos simples, la arquitectura de una red neuronal se refiere al tipo de conexión presente dentro del sistema. Veremos ejemplos de las distintas estructuras que pueden adoptar las redes neuronales a lo largo de este libro.

Redes neuronales feedforward

Primero, es importante comenzar con el tipo de red neuronal que ya hemos introducido, es decir, la estructura de red neuronal convencional mencionada anteriormente. Lo relevante e interesante de este tipo de redes neuronales es que están organizadas en diferentes capas que pueden representar distintos niveles de abstracción de información durante la ejecución del modelo. Esta estructura también es conocida en la comunidad de inteligencia artificial como una estructura de red neuronal "feedforward", que en español se traduciría como una estructura de red neuronal de avance directo.

Estas estructuras de redes neuronales constan de varias capas compuestas por múltiples neuronas, como se explicó previamente. Podemos dividir y describir la función de las diferentes capas en tres partes esenciales: capas de entrada, capas ocultas y capas de salida.

En términos simples, la capa de entrada del modelo consiste en múltiples neuronas encargadas de recibir la información de entrada del modelo. Por ejemplo, si el modelo debe procesar una frase, la capa de entrada es la parte donde el modelo recibe la información para ser analizada durante la ejecución del resto del modelo.

La sección de capas ocultas representa la parte más importante del modelo, generalmente compuesta por un amplio conjunto de capas que contienen

múltiples perceptrones (denominadas neuronas artificiales a partir de ahora para facilitar la comprensión del concepto).

La función de estas capas ocultas es llevar a cabo todo el proceso interno de la estructura, es decir, es la parte esencial del modelo donde se recibe la información de la capa de entrada y luego se procesa, compartiendo la información entre las capas y generando una interconexión entre cada una de ellas. Las capas ocultas en un modelo de inteligencia artificial basado en redes neuronales feedforward suelen estar compuestas por muchas subcapas, que pueden llegar a ser miles o incluso millones.

Finalmente, la capa de salida de este tipo de estructuras consta de otras neuronas artificiales encargadas de recibir la información generada en las capas ocultas y presentarla como una salida del modelo.

Podemos visualizar de manera más clara las diferentes partes de una red neuronal feedforward en el siguiente gráfico, que muestra varias capas de redes neuronales interconectadas, además, resalta de manera comprensible cuáles son las capas de entrada, las capas ocultas y las capas de salida.

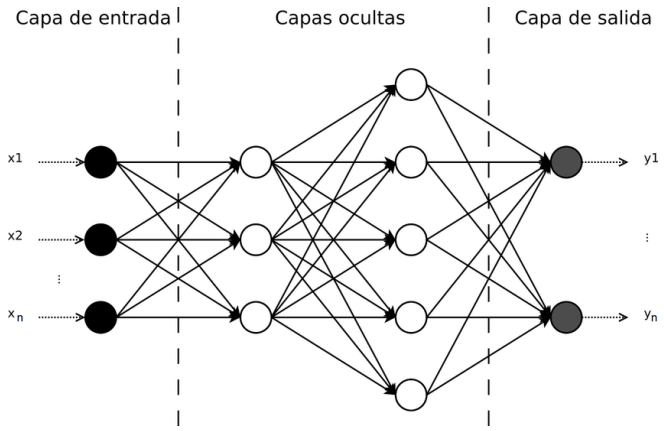


Ilustración 23 – Red neuronal artificial con sus respectivas capas (Alvarado & Meneses-Bautista, 2017).

Esta es una de las múltiples maneras en que podemos representar las diferentes capas de las redes neuronales en un gráfico. Aunque es común que los gráficos de las redes neuronales se muestren en forma horizontal, esto no responde a ninguna regla específica. Lo realmente relevante no es el orden o la manera en que se grafican las redes neuronales, sino entender los distintos tipos de capas que pueden aparecer en estas estructuras y, a su vez, comprender los roles que desempeñan dichas capas en las redes que estamos analizando.

Es posible que todos estos conceptos nuevos sean confusos para muchas personas al principio, pero no hay que preocuparse demasiado ni esforzarse en entender algo que puede resultar complicado en este momento. Basta con comprender los distintos roles e importancia de estas capas en relación a las

estructuras de redes neuronales. Cabe destacar que los gráficos para representar las conexiones de redes neuronales también pueden realizarse de forma vertical, como veremos ahora.

En la parte inferior de la imagen, se ubican las capas de entrada, identificadas con la notación "I{n}", donde "I" hace referencia a "Input" (entrada). De manera similar, en la capa intermedia, la letra "H" representa "Hidden" (oculta), mientras que, en la capa superior y final, la letra "O" simboliza "Output" (salida).

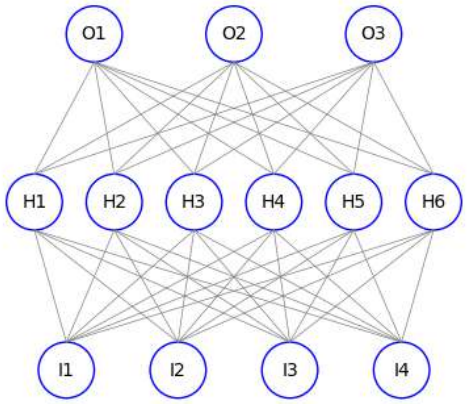


Ilustración 24 - Red Neuronal Feedforward con Capas de Entrada, Oculta y Salida

Dentro de las neuronas feedforward, cada entrada independiente en las neuronas artificiales posee un valor numérico, usualmente en el rango de 0 a 1, conocido como peso. Este peso indica la relevancia de los datos y cuán significativo debe ser el escalado de la información que ingresa a la neurona. Al crear el modelo, los pesos iniciales no son críticos, dado que se modificarán posteriormente mediante un proceso llamado entrenamiento. El ajuste de los valores de los pesos de las entradas en cada capa se denomina "ajuste de pesos".

En general, el proceso para encontrar los pesos adecuados en el modelo puede ser bastante subjetivo. El proceso consiste en modificar y seguir ajustando continuamente los pesos de cada entrada de las neuronas hasta encontrar los valores óptimos. Sin embargo, esto no implica que los pesos de las neuronas sean modificados de manera aleatoria, sino que estos son ajustados mediante una serie de reglas conocidas como "algoritmos de optimización" u "optimizadores". Estos algoritmos permiten modificar los pesos en función del error. Se explicará con más a detalle en secciones futuras.

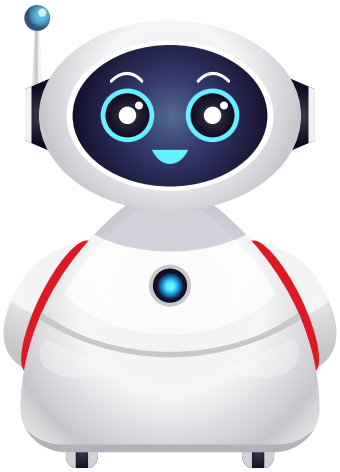
Además del proceso de optimización del algoritmo mencionado anteriormente, cuya función principal es especificar los pasos necesarios para modificar los pesos de las entradas en las neuronas, existen otros elementos que influyen en el rendimiento y comportamiento de la neurona. Por ejemplo, podemos mencionar la función que se ejecuta dentro de los modelos de redes neuronales para

modificar el resultado que se generará como salida, es decir, la función de activación. Tal como mencionamos previamente al explicar las diferentes partes de un perceptrón, existe una parte esencial dentro de la neurona encargada de modificar el resultado en base a la entrada del modelo, conocida como función de activación.

El proceso interno que ocurre en la neurona para transformar los datos ingresados se realiza mediante una función llamada "función de activación", como ya mencionamos en secciones anteriores de este libro. A lo largo de los estudios en este campo, se han propuesto varias funciones de activación, como la "función sigmoide" o "RELU", entre otras. La principal tarea de estas funciones de activación es convertir la suma ponderada de las entradas en un valor de salida adecuado para la neurona.

Finalmente, se lleva a cabo un proceso conocido como "entrenamiento del modelo", en el cual el modelo se entrena utilizando "datos de entrenamiento" y "datos de validación". Estos términos hacen referencia a los datos que instruirán a nuestro modelo sobre lo que debe aprender. Aunque el proceso de entrenamiento es complejo y extenso, puede simplificarse como un proceso en el que nuestro modelo se evalúa con base en el conocimiento que ha adquirido de los datos de entrenamiento. En este proceso, ajusta los pesos de sus neuronas para alcanzar un equilibrio en el que la salida sea eficiente y se alinee con lo esperado. Generalmente, el entrenamiento del modelo requiere de gran poder computacional y tiempo, no obstante, esto realmente puede variar dependiendo de varios factores, por ejemplo, el tipo y tamaño del modelo en sí, o incluso los datos de entrenamiento empleados en el modelo.

Conclusión del capítulo



En este capítulo, se ha proporcionado una comprensión detallada de la arquitectura de las redes neuronales feedforward y sus componentes esenciales. La importancia de tener diferentes tipos y formas de conexión en las redes neuronales se ha subrayado, mostrando su relevancia en el rendimiento y la eficiencia de los modelos. Se han explicado los distintos roles y la importancia de las capas y neuronas, así como los procesos de optimización y entrenamiento que permiten el ajuste y funcionamiento eficaz de la red. La ilustración y explicación de estas estructuras han permitido visualizar cómo se interconectan y operan. Al entender estas bases, los lectores están mejor preparados para explorar otras arquitecturas de redes neuronales y aplicaciones específicas en campos como la inteligencia artificial y el aprendizaje profundo.

REDES NEURONALES RECURRENTE

Objetivos del capítulo

El objetivo de este capítulo es ofrecer una visión completa y comprensiva de diferentes arquitecturas de redes neuronales, incluyendo redes neuronales recurrentes, de atención, LSTM y Transformers. A través de explicaciones detalladas y ejemplos hipotéticos, el lector se familiarizará con las características clave de cada una de estas estructuras, sus diferencias y similitudes, y su relevancia en el ámbito de la inteligencia artificial. A pesar de que el enfoque principal del libro está en el modelo de Stable Diffusion, este capítulo pretende establecer una sólida base conceptual sobre arquitecturas neuronales que facilitará la comprensión de conceptos más avanzados.

A continuación, estudiaremos otro tipo de arquitectura de red neuronal, en este caso, la arquitectura conocida como "redes neuronales recurrentes". Cabe destacar que estas arquitecturas tienen una estrecha relación con las redes neuronales feedforward que analizamos previamente. En cierto modo, las redes neuronales recurrentes pueden verse como una "variante" de las redes neuronales feedforward. A pesar de compartir la idea conceptual de las redes neuronales feedforward, difieren en un aspecto clave: la salida de una capa de neuronas se convierte en su propia entrada un paso después en el tiempo.

La idea puede parecer confusa al principio; sin embargo, como hemos observado, muchas de estas ideas complejas no son realmente difíciles de entender, sino que simplemente requieren práctica para comprenderlas por completo. Podemos

ilustrarlo con un ejemplo hipotético de cómo funcionan las redes neuronales recurrentes.

Consideremos la creación de una red neuronal recurrente como un modelo de inteligencia artificial. Los componentes de esta estructura son muy similares a los de las redes neuronales feedforward estudiadas previamente. Cada entrada y salida de datos en una neurona ocurre en una serie de pasos S, donde S1 es el primer paso, S2 el segundo y así sucesivamente. La principal diferencia entre una red neuronal recurrente y una feedforward radica en la forma en que se manejan las salidas. Si se produce una salida en un paso Sn dentro de una capa, dicha salida también funcionará como entrada dentro de la misma capa en el paso S(n+1).

Por ejemplo, si un conjunto de datos C se genera como salida en una capa de una red neuronal recurrente en el paso S7, este conjunto C no solo servirá de entrada para la siguiente capa, sino que también actuará como entrada en el paso S8 de su propia capa.

Sin embargo, ¿a qué se debe esto? ¿Existe algún cambio significativo en este proceso que hacen que estas estructuras sean tan distintas de las redes neuronales feedforward? La característica principal de la red neuronal recurrente, en contraste con la red neuronal feedforward, es su habilidad para tomar en cuenta información de eventos previos. Es decir, el modelo "recuerda" lo que ha sucedido a lo largo de su ejecución. Esta capacidad de retener información sobre eventos anteriores es lo que la diferencia de las redes neuronales feedforward.

El objetivo principal de este libro no es enseñar ni proporcionar ejemplos concretos de este tipo de conexiones en redes neuronales, sino entrenar y familiarizarse con el uso del modelo de Stable Diffusion. No obstante, resulta útil tener un conocimiento básico sobre el funcionamiento de estos modelos. Aunque no abordaremos un ejemplo completamente directo, esto nos ayudará a comprender mejor la estructura de dichas redes.

Redes neuronales de atención

Existen diversas estructuras de redes neuronales, y en este caso, examinaremos una última estructura que difiere de las feedforward y recurrentes mencionadas previamente. Para mantenernos enfocados en el propósito principal de este libro, que es aprender a utilizar Stable Diffusion en la generación de imágenes mediante inteligencia artificial, la tercera y última estructura de redes neuronales que abordaremos en esta sección son las redes neuronales de atención.

Estas redes son complejas, por lo que intentaremos explicar su funcionamiento y características principales de manera abstracta y sencilla. Los investigadores han desarrollado estas redes con el fin de mejorar la capacidad de las redes para

identificar y enfocarse en elementos cruciales dentro de un conjunto de datos, ignorando los elementos menos relevantes. Esto las hace especialmente útiles en tareas que requieren enfoques selectivos y adaptativos en lugar de analizar cada parte de los datos de entrada de manera uniforme.

El mecanismo de atención en estas redes opera asignando "pesos de atención" a los elementos de las secuencias y entradas de datos, resultado de un proceso de aprendizaje en el que la red identifica qué partes son más relevantes para la tarea en cuestión.

Así, la red puede dirigir sus recursos computacionales de manera más eficiente, enfocándose en áreas de mayor relevancia. Las redes neuronales de atención pueden adaptarse dinámicamente a diferentes contextos y situaciones, siendo útiles, por ejemplo, en el procesamiento del lenguaje natural.

Es crucial no confundir las redes neuronales de atención con otras estructuras, como las LSTM o los Transformers. Estas redes son eficientes al retener información a lo largo del tiempo, lo que las hace útiles en diversas aplicaciones, como en los modelos generativos de conversación, incluyendo ejemplos como GPT. No obstante, esto no implica que la base de estos modelos se limite exclusivamente a este tipo de estructuras, sino que representa una de las posibles aplicaciones que podrían tener.

Otros aspectos para considerar son el hecho de que estas conexiones son realmente útiles para mantener modelos actualizados a lo largo del tiempo. Aunque las redes neuronales recurrentes tienen la capacidad de recordar eventos a lo largo del tiempo, las conexiones de atención resultan aún más eficientes para preservar información en el transcurso del tiempo.

Las redes neuronales recurrentes son buenas para retener información temporalmente, sin embargo, las conexiones de atención son superiores en este sentido.

A pesar de esto, podemos afirmar que el principal enfoque de este tipo de neuronas es prestar diferentes niveles de atención a las partes más relevantes de una entrada del modelo. Si bien es cierto que estos modelos pueden almacenar información durante un poco más de tiempo, esto realmente no significa que se trate del enfoque principal o una de las capacidades más importantes del modelo.

Redes neuronales LSTM

Ahora, exploraremos un tipo específico de arquitectura de redes neuronales recurrentes, conocida como "redes neuronales LSTM" (Long Short-Term Memory). Estas redes están estrechamente vinculadas a las redes neuronales recurrentes que mencionamos anteriormente y, de hecho, pueden considerarse una

"evolución" de ellas. A pesar de estar basadas en la idea fundamental de las redes neuronales recurrentes, las LSTM introducen un mecanismo adicional: la habilidad para recordar y olvidar información a lo largo del tiempo de manera selectiva.

La noción podría parecer un poco abstrusa al comienzo; no obstante, como vimos anteriormente, muchas de estas ideas intrincadas no son en realidad difíciles de asimilar, sino que simplemente requieren dedicación para dominarlas por completo. Para ilustrar esto, imaginemos un caso hipotético de cómo funcionan las redes neuronales LSTM.

Al construir una red neuronal LSTM como modelo de inteligencia artificial, los elementos de esta arquitectura se asemejan en gran medida a las redes neuronales recurrentes discutidas previamente. En una LSTM, cada neurona cuenta con una estructura interna adicional llamada "unidad de memoria", que permite almacenar información a lo largo de los pasos S, donde S1 es el primer paso, S2 el segundo y así sucesivamente. La principal innovación de una red neuronal LSTM respecto a una recurrente radica en cómo se gestionan las conexiones y el flujo de información. Si se produce una salida en un paso Sn dentro de una capa, dicha salida puede ser almacenada, olvidada o actualizada en función de las necesidades del modelo.

Por ejemplo, supongamos que un conjunto de datos D se genera como salida en una capa de una red neuronal LSTM en el paso S5. Este conjunto D no solo servirá de entrada para la siguiente capa, sino que también podría ser almacenado en la unidad de memoria de su propia capa para ser utilizado en pasos futuros como S6, S7 o incluso más adelante.

Entonces, ¿por qué es esto relevante? ¿Existe algún beneficio considerable en este enfoque que haga que estas estructuras sean tan diferentes de las redes neuronales recurrentes? La característica esencial de las redes neuronales LSTM, en contraposición a las recurrentes, radica en su capacidad para manejar dependencias temporales de largo alcance. Es decir, el modelo "aprende" a retener información crucial y a desechar lo irrelevante, permitiendo un mejor manejo de secuencias extensas. Esta habilidad de gestionar información de manera selectiva es lo que distingue a las redes neuronales LSTM de las recurrentes.

Todo este proceso, en el que las conexiones de la red neuronal tienen la capacidad de recordar información y almacenarla, ocurre por medio de celdas de memoria presentes en cada una de las neuronas de este tipo de redes que utilizan esta arquitectura. Generalmente, el proceso que se lleva a cabo para entender esto es que la neurona se encarga de guardar en estas celdas la información que considera relevante.

Debido a que este tipo de redes neuronales procesan grandes cantidades de datos en periodos de tiempo muy cortos, es importante tener en cuenta que las

conexiones y las celdas de memoria de estas redes neuronales son relativamente pequeñas y se actualizan constantemente durante la ejecución del modelo.

Es decir, la información guardada en estas celdas no es permanente, sino que se trata únicamente de la información que el modelo clasifica como relevante en ese momento de la ejecución. Sin embargo, la información relevante almacenada en las celdas de memoria podría ser reemplazada en un momento posterior.

Todo este proceso, en el que el modelo aprende a guardar información, borrarla o incluso enviarla a otras neuronas, ocurre por medio de compuertas que se encargan de controlar estas acciones. Estos pequeños elementos funcionan como la entrada principal de las celdas de memoria y, generalmente, se encargan de decidir el manejo de la información en la celda de memoria. Existen tres tipos de compuertas: la compuerta de entrada, la salida y la de olvido.

El propósito central de este libro no es enseñar ni ofrecer ejemplos detallados de estos tipos de conexiones en redes neuronales, es provechoso adquirir un conocimiento básico sobre el funcionamiento de estos modelos. Aunque no nos adentraremos en un ejemplo totalmente explícito, esta comprensión nos permitirá apreciar mejor la estructura de dichas redes y su relación con el modelo de Stable Diffusion.

Redes neuronales Transformers

Ahora, es momento de estudiar una de las estructuras de redes neuronales más recientes que han surgido y que han revolucionado el mundo de la investigación en inteligencia artificial. Esto se debe principalmente a las increíbles capacidades que ofrecen este tipo de arquitecturas de redes neuronales. Después de todo, son estas conexiones las que han permitido que grandes modelos de inteligencia artificial actuales existan y tengan la capacidad de realizar tareas extremadamente complejas. Un ejemplo claro y directo de esto es los diferentes modelos de GPT, los cuales fueron desarrollados basados en neuronas que siguen este tipo de estructura.

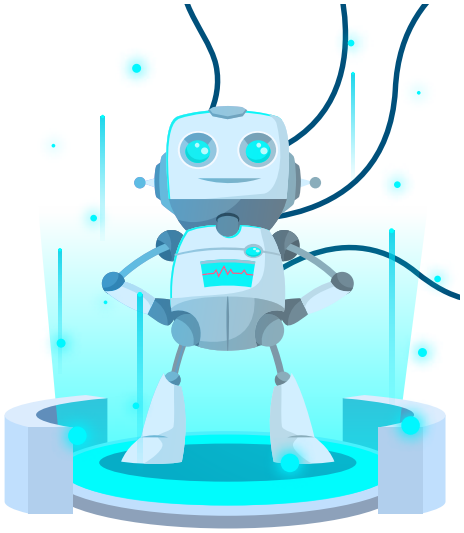
No se trata solamente de este modelo, sino que realmente existen muchos más ejemplos de las diferentes aplicaciones que han tenido este tipo de neuronas en el desarrollo de modelos de inteligencia artificial que realizan tareas bastante complejas. Por ahora, puede parecer algo confuso entender por qué este tipo de conexiones de redes neuronales son tan populares y se han vuelto tan importantes y relevantes en los modelos actuales de inteligencia artificial. Sin embargo, para poder entender un poco más en detalle los elementos que componen este tipo de estructuras y al mismo tiempo los diferentes aspectos que las hacen tan especiales, debemos entender primero su funcionamiento y las capacidades técnicas que pueden ofrecer.

Algo es seguro, y es que este tipo de conexiones ha demostrado ser muy prometedoras frente a otros modelos que buscan realizar tareas complejas que sorprenden a muchas personas con sus resultados. Además del ejemplo que vimos anteriormente, veremos algunos otros ejemplos de esto más adelante. No solo eso, sino que realmente podemos decir que este tipo de conexiones de redes neuronales también tienen la capacidad de simular de forma bastante fiel algunos de los elementos y fenómenos que se llevan a cabo dentro de nuestro cerebro (Ornes, 2022).

Otra característica importante de este tipo de arquitectura es su utilidad y eficiencia para tareas relacionadas con el procesamiento del lenguaje natural, la cual es una de las innovaciones más importantes en inteligencia artificial actualmente. Hay varios aspectos relacionados con este tipo de conexiones, por ejemplo, al igual que las arquitecturas LSTM, estas neuronas también tienen la capacidad de almacenar información a lo largo del tiempo, es decir, pueden recordar eventos pasados.

Los Transformers funcionan mediante un mecanismo llamado autoatención, en el que cada entrada (una palabra, un píxel, un número en una secuencia o cualquier otro elemento que forme parte de la estructura con la que se está trabajando) siempre está conectada a todas las demás entradas. Como dijimos, puede parecer algo confuso para muchas personas, sin embargo, entender más a fondo el funcionamiento de este tipo de estructuras solo nos alejaría de nuestro objetivo, que es aprender a usar Stable Diffusion.

Por ahora, es más que suficiente entender que este tipo de estructuras forma parte de los diferentes tipos de conexiones y variantes que existen en los modelos de redes neuronales, y que ofrecen un rendimiento significativo positivo frente a tareas como el procesamiento del lenguaje natural. El mundo de las diferentes arquitecturas de conexiones de redes neuronales es un campo amplio e interesante de estudiar.



Conclusión del capítulo

En este capítulo, hemos explorado en detalle diversas arquitecturas de redes neuronales, proporcionando una base sólida en conceptos fundamentales. Desde las redes neuronales recurrentes y su habilidad para retener información temporal, hasta las complejas redes de atención y su capacidad para enfocarse en elementos cruciales, hemos abordado cómo estas estructuras funcionan y en qué se diferencian entre sí. Las redes LSTM y Transformers también fueron discutidas, destacando su relevancia en la manipulación de información y adaptabilidad en distintas situaciones. Aunque estos temas son complejos, una comprensión de ellos es crucial para apreciar el contexto en el cual el modelo de Stable Diffusion opera, y cómo este se relaciona con estas arquitecturas en la generación de imágenes y otros aplicativos de inteligencia artificial.

APLICACIONES Y PRÁCTICAS DEL DEEP LEARNING

Objetivos del capítulo

El objetivo de este capítulo es proporcionar una comprensión detallada de las aplicaciones y prácticas en el campo del deep learning, con un enfoque en el procesamiento del lenguaje natural y los asistentes de voz. Se busca explicar cómo los modelos de inteligencia artificial basados en aprendizaje profundo están siendo utilizados en la vida cotidiana, detallando ejemplos como la traducción automática, generación de texto, reconocimiento de imágenes, entre otros. Además, se tiene la intención de analizar en profundidad el caso particular de Siri, uno de los asistentes de voz más populares, ofreciendo una visión técnica de su funcionamiento y relevancia en la industria.

Naturalmente, la simple existencia de estos modelos de inteligencia artificial no es suficiente para comprender completamente lo que se puede lograr con ellos. La pregunta crucial es: ¿para qué podemos utilizar estos modelos de inteligencia artificial basados en deep learning y cuál es su impacto? Aunque esta pregunta pueda parecer trivial para algunos, sigue siendo esencial destacar las aplicaciones de estos modelos de inteligencia artificial para entender plenamente sus beneficios y tener una idea más clara de las distintas tareas que pueden realizar en nuestra vida cotidiana. Por ello, en esta sección mencionaremos y ofreceremos ejemplos de algunas de las principales aplicaciones de los modelos de inteligencia artificial basados en aprendizaje profundo.

Uno de los principales y más populares usos de los modelos de inteligencia artificial basados en estructuras de redes neuronales es el procesamiento del lenguaje natural, conocido comúnmente por sus siglas en inglés, NLP (Natural

Language Processing). Este concepto podría parecer un poco confuso al principio, ya que surge la pregunta: ¿a qué se refiere exactamente este término "lenguaje natural" y cuál es la relevancia de los modelos de aprendizaje profundo en la realización de este tipo de tareas? En pocas palabras, el lenguaje natural abarca todas aquellas señales de comunicación que utilizamos los seres humanos para interactuar entre nosotros, como el habla, las expresiones corporales, entre otros.

Los modelos basados en redes neuronales son muy eficientes en la interpretación de esta información que solemos transmitir al comunicarnos. De hecho, han demostrado un rendimiento impresionante en la interpretación de este tipo de información. El rendimiento de estas redes neuronales en el procesamiento del lenguaje natural no se limita solo a esto.

Estos modelos no sólo han demostrado ser capaces de procesar lenguaje natural como entrada, sino que también pueden generar texto, demostrando su habilidad en la generación de lenguaje natural.

En cuanto a aplicaciones más específicas de estos modelos de inteligencia artificial, se han utilizado para realizar tareas como traducción automática, generación de texto, respuestas automáticas a preguntas, reconocimiento de imágenes, visión por computadora, generación de arte y música, entre otras. Podemos decir que las aplicaciones de los modelos de inteligencia artificial son un campo muy amplio. Después de todo, el estudio de este campo no es nada nuevo y desde hace bastante tiempo se viene investigando. Desde entonces, se han hecho esfuerzos significativos para lograr progresos en la ciencia frente al avance de la inteligencia artificial.

Estas estructuras también han demostrado ser capaces de realizar muchas tareas, no solo automatizando procesos que generalmente podrían tomar mucho más tiempo de lo habitual, sino también potenciando la eficiencia y la innovación en diversos ámbitos. Es evidente que el uso de herramientas basadas en inteligencia artificial para realizar diferentes tipos de tareas ha representado un gran avance para la humanidad, reduciendo considerablemente el tiempo requerido.

Sin embargo, puede que no resulte del todo claro o que algunas personas no entiendan completamente la capacidad que estas herramientas pueden ofrecer, y simplemente su comprensión de los casos de aplicación de estas herramientas se vea limitado. Con esto en mente, la intención de las diferentes secciones que se abarcan es mostrar y hablar de algunas de las más importantes y relevantes aplicaciones que han tenido los diferentes modelos de inteligencia artificial. Esto no necesariamente partiendo de las aplicaciones más importantes o según el paso del tiempo, sino que serán aplicaciones y avances que son usados en la actualidad, es decir, modelos de inteligencia artificial cuyo uso sigue siendo relevante actualmente. Eventualmente, después de estudiar las diferentes aplicaciones y usos actuales de los modelos de inteligencia artificial, llegaremos a mencionar finalmente el modelo de inteligencia artificial que de hecho queremos

estudiar dentro de este libro, es decir, el modelo de inteligencia artificial de Stable Diffusion.

Asistentes de voz

Ahora, mencionaremos uno de los usos más populares de los modelos de inteligencia artificial, especialmente en el ámbito del consumo por parte de personas no técnicas. No nos referimos a personas con mucho conocimiento técnico en el tema, sino a un grupo de consumidores finales que utilizan el producto. Hablamos de modelos de inteligencia artificial de procesamiento del lenguaje natural que funcionan como asistentes de voz. Algunos ejemplos son Siri, Google Assistant y Alexa, los cuales son los más populares y utilizados actualmente.

El concepto detrás de los asistentes de voz es bastante sencillo. En términos simples, estos modelos de inteligencia artificial interpretan las órdenes del usuario. Por lo general, estos modelos no se basan en un solo elemento de inteligencia artificial para determinar los pasos a seguir, sino en un conjunto de procesos internos en el sistema que se está utilizando. Por ejemplo, hay una subdivisión del modelo de inteligencia artificial encargada de tomar en tiempo real las señales de voz recibidas y convertirlas en texto legible por el modelo.

Estos modelos de inteligencia artificial suelen conectarse a diversos ecosistemas, dependiendo de la función específica que estén intentando realizar. Por ejemplo, si Alexa es solicitada para apagar las luces mediante sus conexiones con dispositivos del Internet de las cosas en una casa, significa que Alexa y todo su sistema deben tener acceso al ecosistema de los objetos bajo su control. En el caso de Google Assistant, muchas de las respuestas del sistema provienen de búsquedas en internet o del uso de diferentes aplicaciones a las que tiene acceso el asistente dentro de su ecosistema.

La gran importancia de estos asistentes y la novedad que aportó a la industria de la inteligencia artificial hace algunos años radica en la capacidad de estas inteligencias artificiales para ser realmente coherentes ante las solicitudes de los usuarios. Además, la interacción con estos asistentes se basa simplemente en hablar en voz alta, dándoles la instrucción que queremos que realicen. Otras innovaciones importantes incluyen su capacidad para identificar voces de prácticamente cualquier persona con cualquier tono de voz y acento, además de funcionar en varios idiomas.

En esta sección, tendremos la oportunidad de analizar uno de estos modelos de inteligencia artificial de los asistentes de voz más populares en uso actualmente: Siri. Examinaremos algunas de las novedades de este asistente de voz, y principalmente, por qué es tan importante en la industria. Veremos cuál es la importancia de este modelo en el uso de las tareas diarias y cómo funciona este modelo de inteligencia artificial desde un punto de vista técnico. Tendremos la

oportunidad de observar el funcionamiento interno de este modelo de inteligencia artificial y cómo puede resultarnos útil. Podemos ver a lo que nos referimos en la siguiente sección.

Caso ejemplar: Siri

Ahora, tal y como mencionamos en la sección anterior, la principal idea es poder entender, de forma superficial, cómo funcionan estos modelos de inteligencia artificial para asistentes de voz. De esta manera, es importante comprender en detalle el funcionamiento y los diferentes procesos internos que tienen lugar en estos modelos de asistentes de voz. Puede parecer algo trivial, pero realmente es importante entender cómo funcionan estos asistentes de voz desde un punto de vista más profundo.

Sin embargo, antes de abordar el aspecto técnico del funcionamiento interno de este asistente de voz, es posible que no todas las personas comprendan a qué se refiere con "asistente de voz de Siri" y por qué este asistente es tan relevante en la industria. En términos sencillos, Siri es el nombre que se le da al asistente virtual de voz desarrollado por Apple Inc.

La popularidad de este asistente se atribuye principalmente a su integración en la mayoría de los dispositivos Apple actuales. El asistente de voz ofrece una amplia gama de utilidades para ejecutar diferentes tipos de tareas en nuestras cuentas o dispositivos personales. Además, este modelo de inteligencia artificial permite realizar funciones específicas que involucran el ecosistema de Apple y realizar consultas generales, entre otras cosas.

Otra de las importantes novedades que este modelo de inteligencia artificial tiene frente a otros competidores es el hecho de que con Siri no es necesario utilizar un lenguaje formal, ya que no se comporta como una simple computadora sin muchas capacidades, sino que Siri también tiene la capacidad de poder entender el lenguaje coloquial y una conversación más natural, esto hace que el modelo sea más ameno y amigable para usuarios sin mucha experiencia en computadoras.

Hay muchos aspectos dentro de este asistente que no son públicos, después de todo, estamos hablando de una compañía como Apple que generalmente no desea que la información del funcionamiento interno y profundo de sus productos sea explorada a fondo. Sin embargo, podemos mencionar algunos aspectos que contribuyen al funcionamiento general de este asistente de voz.

Al hablar con Siri, la voz del usuario es procesada por algoritmos de reconocimiento automático de voz (ASR), que transforman la señal de voz en texto utilizando redes neuronales profundas. Luego, Siri aplica procesamiento de lenguaje natural (NLP) para entender la intención del usuario, identificar

entidades, extraer información relevante y comprender el contexto, permitiendo así respuestas y conversaciones más naturales.

Siri toma decisiones según la información del usuario, contexto y datos en el dispositivo, accediendo a datos como el calendario y realizando búsquedas web. Utiliza síntesis de voz (TTS) para comunicarse y mejora constantemente mediante aprendizaje automático, apoyándose en interacciones de usuario y avances en investigación. Apple emplea aprendizaje profundo para optimizar la eficacia de Siri.

Apple ha hecho énfasis en la importancia de la privacidad en sus dispositivos. La mayoría de las interacciones con Siri se procesan en el dispositivo del usuario, lo que significa que no se almacenan en servidores externos. Sin embargo, algunas solicitudes, como las búsquedas en la web, pueden requerir una conexión a Internet y el procesamiento en la nube. Podemos ver un poco más a detalle los pasos que realiza este modelo en el siguiente gráfico:

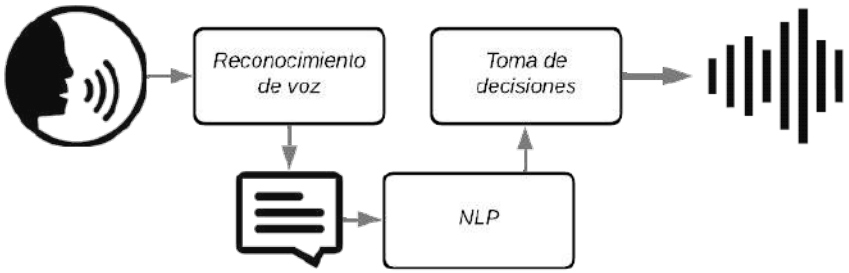
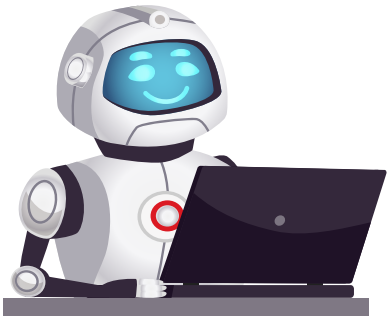


Ilustración 25 – Representación visual del funcionamiento de Siri

Ahora podemos comprender de manera más detallada el funcionamiento de este modelo de inteligencia artificial. Es importante subrayar que, en general, estos modelos de inteligencia artificial operan de manera muy similar. El proceso que acabamos de describir puede ser incluso aplicado a otros modelos que también desempeñan la función de asistentes de voz.

Conclusión del capítulo



En este capítulo, hemos explorado las diversas aplicaciones de los modelos de inteligencia artificial basados en deep learning, mostrando su versatilidad y eficacia en diversas tareas. A través del análisis detallado de las funciones de los asistentes de voz y el estudio en profundidad de Siri, hemos resaltado cómo estas tecnologías se han convertido en herramientas fundamentales en nuestra vida cotidiana. El capítulo enfatiza la capacidad de los modelos de inteligencia artificial para no solo automatizar procesos y mejorar la eficiencia en diversos campos, sino también enriquecer la interacción humana con las tecnologías a través de un lenguaje más natural y accesible. La discusión también abordó temas como la privacidad y la seguridad en el uso de estos asistentes, resaltando la complejidad y la importancia de estos factores en el diseño y la aplicación de los modelos de inteligencia artificial. Finalmente, el capítulo sirvió como una introducción a los modelos de aprendizaje profundo y estableció una base sólida para el estudio posterior del modelo de inteligencia artificial de Stable Diffusion, que se discutirá en el libro.

CAPÍTULO 3: ¿QUÉ ES STABLE DIFFUSION?

Objetivos del capítulo

En este capítulo, se busca proporcionar al lector una visión completa y detallada del modelo de inteligencia artificial Stable Diffusion, enfocándose en su funcionamiento, historia y aplicaciones. Se explicará cómo este modelo genera imágenes a partir de descripciones de texto, haciendo hincapié en la técnica de "Diffusion". Se contrastará Stable Diffusion con otros modelos similares y se ofrecerán ejemplos de su potencial en la generación de imágenes. Además, se abordará su origen, su desarrollo a lo largo del tiempo y su impacto en la industria y en la vida cotidiana. Por último, se pretende analizar las preocupaciones y debates actuales en torno a la posibilidad de que la inteligencia artificial reemplace el trabajo humano en el campo del arte digital.

Ahora, ha llegado el momento de abordar el tema principal de este libro, la generación de imágenes utilizando inteligencia artificial, específicamente Stable Diffusion.

Stable Diffusion es un modelo de inteligencia artificial diseñado para generar imágenes y tiene la capacidad de crearlas a partir de una entrada de texto que describe lo que se desea generar. Es decir, al proporcionar al modelo una descripción de lo que quieres obtener, este generará una imagen que representa lo solicitado.

Este proceso de generación de imágenes se lleva a cabo mediante una técnica llamada "Diffusion" (Difusión), de ahí el nombre del modelo. El concepto de Diffusion y su funcionamiento se explicarán con más detalle en secciones posteriores.

El desempeño de este modelo de inteligencia artificial en la generación de imágenes ha sorprendido a quienes han tenido la oportunidad de utilizarlo.

Incluso, su rendimiento ha superado el de otros modelos de inteligencia artificial desarrollados por grandes compañías, como, por ejemplo, DALL-E 2. Este modelo, similar a Stable Diffusion y creado por OpenAI, también es capaz de generar imágenes de alta calidad a partir de descripciones en texto.

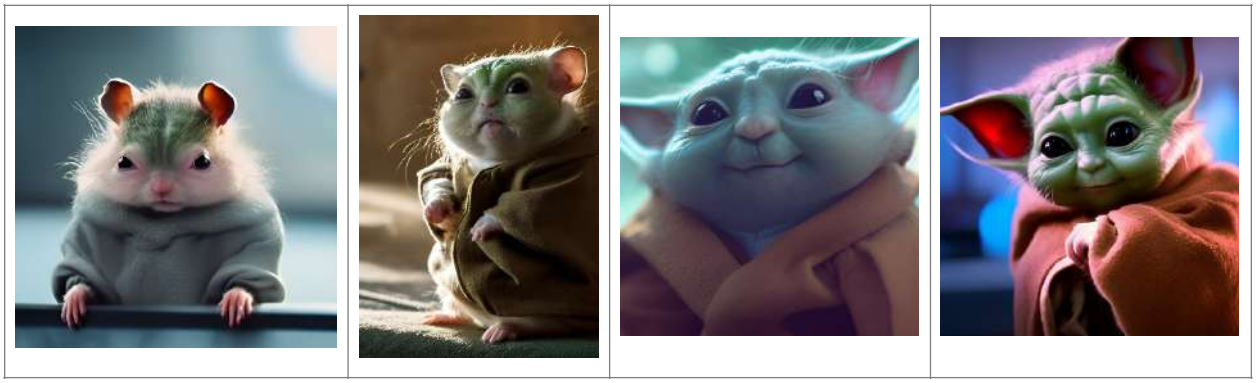
El concepto de Stable Diffusion es bastante sencillo de entender, y es importante destacar que las funciones que ofrece este modelo de inteligencia artificial son amplias y permiten generar y controlar imágenes de diversas maneras. Aunque su función principal y más popular es la generación de imágenes a partir de descripciones en texto, también exploraremos otras capacidades de Stable Diffusion y comprenderemos cómo funcionan en profundidad.

A continuación, se presentan algunas imágenes basadas en descripciones textuales. Muchas de estas imágenes han sido creadas por la comunidad de Stable Diffusion, pero también tendremos la oportunidad de utilizar este modelo de inteligencia artificial en secciones posteriores.

Es importante mencionar que estas imágenes han sido generadas exclusivamente por este modelo de inteligencia artificial.

Stable Diffusion ofrece la posibilidad de generar diversas imágenes a partir de una misma descripción en texto. Esto es un ejemplo de lo que mencionamos anteriormente.

Generación Stable Diffusion 1 - A close-up shot of a lion, low exposure, national geographic (Una toma cercana de un león, baja exposición, National Geographic)



Generación Stable Diffusion 2 - hamster yoda, movie still, star wars, cinematic, sharp focus, cinematic grain, cinematic lighting, 8 k, photo, realistic, high quality, professionally --cartoon, 3d (Hamster Yoda, fotograma de película, Star Wars, cinemático, enfoque nítido, grano cinema)

Generación Stable Diffusion 3 - Variación de una fotografía utilizando ControlNet Depth Map, un modelo de modificación de imágenes que tiene en cuenta la profundidad de los elementos, basado en Stable Diffusion.

Generación Stable Diffusion 4 - floating (silhouette of face:1.2), mutation, deformed, distorted face, (wrapped in millions of ribbons, threads, fractal:1.2), grayscale, black and white, blurry, (symmetric, symmetry, symmetric composition:1.2) dreamlike, floating, light, high key, white, white light (pencil drawing, intricate, very detailed:1.2)

– (No hay una traducción completamente adecuada, la traducción literal es: flotando (silueta de cara: 1.2), mutación, deformada, cara distorsionada, (envuelta en millones de cintas, hilos, fractal: 1.2), escala de grises, blanco y negro, borroso, (simétrico, simetría, composición simétrica: 1.2) onírico, flotante, ligero, clave alta, blanco, luz blanca (dibujo a lápiz, intrincado, muy detallado: 1.2))

Así pues, ahora tenemos una visión general de las enormes posibilidades que nos brinda Stable Diffusion en la generación de imágenes. Como podemos observar, tiene sentido que estos modelos de inteligencia artificial hayan cobrado tanta popularidad y sean objeto de debate en diversos temas actuales. Después de todo, Stable Diffusion ha revolucionado la forma en la que se crean imágenes digitales.

Es probable que aún haya aspectos difíciles de comprender, como la manera en que estructuras de redes neuronales y otras tecnologías nos permiten generar arte digital de alta calidad y con gran fidelidad a nuestras indicaciones. Resulta desconcertante para muchos entender cómo simples líneas de texto pueden dar lugar a imágenes tan asombrosas como las que hemos visto.

Es normal tener este tipo de confusión, sin embargo, no debemos preocuparnos por ello. Tendremos la oportunidad de estudiar más a fondo estos modelos de inteligencia artificial y comenzar a generar nuestras propias imágenes, experimentar con ellas, explorar qué tipos de imágenes podemos crear y conocer sus funcionalidades, utilidades y limitaciones.

Pero, más allá de la gran abstracción que ofrece este modelo de inteligencia artificial, en el cual basta con escribir una breve descripción en texto de lo que queremos generar, ¿qué sucede tras bambalinas? ¿Cómo es posible que esto ocurra y cuál es el proceso interno de estos modelos para generar dichas imágenes?

Responder a estas preguntas no es tarea sencilla. Stable Diffusion es el resultado de mucha investigación, programación y contribuciones comunitarias. Si realmente queremos entender cómo funciona este modelo de inteligencia artificial, tendremos mucho que procesar y estudiar. En este libro, se intentará explicar el proceso interno que tiene lugar en Stable Diffusion de tal manera que podamos comprender todo lo relacionado a este modelo de inteligencia artificial.

Dado que esto está directamente relacionado con el funcionamiento de Stable Diffusion, se hará todo lo posible para que la explicación de estos fenómenos sea lo suficientemente clara y detallada, de modo que no queden dudas.

Stable Diffusion: Un poco de historia

Antes de abordar en detalle el funcionamiento de Stable Diffusion, es útil comprender el origen de este poderoso modelo de inteligencia artificial, dónde se

ha desarrollado y cómo ha avanzado a lo largo del tiempo. Por ello, en esta sección examinaremos información relevante sobre este tema, donde tendremos la oportunidad de conocer las diferentes versiones de Stable Diffusion, el comienzo de su desarrollo, algunos de sus predecesores y diversos estudios relacionados con la generación de imágenes.

Stable Diffusion es un modelo de inteligencia artificial de código abierto, desarrollado y publicado por CompVis (Computer Vision and Learning LMU Múnich), un grupo de investigación en Visión por Computadora y Aprendizaje Automático. Esto se llevó a cabo en colaboración con Stability AI, una compañía centrada en modelos generativos de inteligencia artificial de código abierto; Runway, una plataforma que permite experimentar y trabajar con modelos de inteligencia artificial y deep learning sin necesidad de tener un amplio conocimiento en estos temas; y LAION, una organización sin ánimo de lucro que busca apoyar y hacer modelos de machine learning, bases de datos y código relacionado a inteligencia artificial completamente accesible de forma pública, bases de datos y código relacionado a inteligencia artificial completamente accesible de forma pública. Este modelo está disponible de forma gratuita para su uso en la página oficial de CompVis en GitHub.

Stable Diffusion no es un modelo que parte desde cero, de hecho, en base a otro proyecto previamente desarrollado por el grupo de investigación CompVis, llamado High-Resolution Image Synthesis with Latent Diffusion Models (Síntesis de imágenes de alta resolución con modelos de difusión latentes).

Este proyecto buscaba ofrecer un rendimiento similar a otros modelos como la primera versión de DALL-E y VQGAN. El modelo de inteligencia artificial fue publicado originalmente para uso libre en agosto de 2022. Con el tiempo y la ayuda de la comunidad, ha evolucionado, pasando por varias versiones del mismo modelo. Al momento de escribir este libro, la versión más reciente de este modelo de inteligencia artificial se denomina Stable Diffusion XL, también conocido como Stable Diffusion 2.2. Sorprendentemente, a pesar de la novedad que este modelo de inteligencia artificial presenta, se ha demostrado que tiene la capacidad de generar imágenes simplemente increíbles, además de superar las limitaciones de los modelos generativos de imágenes anteriores, por ejemplo, incluir texto en las ilustraciones.

Aunque la capacidad de generar texto dentro de las imágenes puede parecer no del todo importante para muchos, es impresionante ver cómo estos modelos de inteligencia artificial cada vez tienen mejores habilidades para superar y resolver sus limitaciones anteriores. Por ahora es una función que apenas está en desarrollo y solo presenta una versión beta, sin embargo, los avances y resultados que obtenemos actualmente siguen siendo de muy buena calidad.

Aplicaciones y casos de uso de Stable Diffusion

Queda evidente que Stable Diffusion es un potente modelo de inteligencia artificial capaz de generar imágenes de alta calidad a partir de simples instrucciones, lo que agiliza y optimiza considerablemente el proceso de creación de arte digital. Sin embargo, no se trata simplemente de un modelo de inteligencia artificial para entretenerse generando imágenes atractivas mediante unas pocas líneas de texto. De hecho, incluso antes de que Stable Diffusion alcanzara popularidad y surgieran modelos similares, ya se discutía y se percibía el gran impacto de estos modelos en la industria y en diversos aspectos de la vida cotidiana. Es posible que esto no sorprenda a muchos, ya que forma parte de las preocupaciones asociadas al surgimiento de este tipo de modelos de inteligencia artificial. La inquietud principal es que la inteligencia artificial pudiera reemplazar el trabajo humano, que para muchas familias representa el sustento económico y garantiza la alimentación.

La situación se plantea así: ¿por qué las grandes empresas contratarían a trabajadores con salarios elevados y tiempos de producción más largos, cuando podrían simplemente implementar un modelo de inteligencia artificial que realice el trabajo artístico por ellos? Este tema ha generado amplios debates en diversas partes del mundo, especialmente entre los

profesionales relacionados con la creación de arte, dibujos, animación, música, entre otros. Todo se centra en una idea: el temor a que la inteligencia artificial reemplace el trabajo humano.

A pesar de ello, se puede afirmar con cierta seguridad que esta afirmación no es del todo precisa. Para entenderlo, es necesario tener en cuenta varios factores. En primer lugar, es esencial destacar que la inteligencia artificial no es igual a la inteligencia humana. La inteligencia artificial es una herramienta, no un ser consciente. Esta ocurre mediante conexiones y estructuras digitales, mientras que la inteligencia humana se basa en complejas y avanzadas conexiones neuronales. Por lo tanto, no se debe considerar a la inteligencia artificial como un medio para reemplazar a los seres humanos. No solo debido al factor de error, ya que estas inteligencias artificiales no son perfectas y pueden cometer errores si no cuentan con supervisión humana, sino que se debe ver como una herramienta para potenciar la eficiencia de un trabajador al emplearla en sus tareas.

De hecho, la idea de utilizar la inteligencia artificial como una herramienta para mejorar el rendimiento de los trabajadores ha llevado a que Stable Diffusion tenga un impacto significativo, principalmente positivo, en varias industrias clave. No solo se relaciona con la creación de arte digital, sino que también se ha implementado en casi cualquier campo que requiera gráficos, diseño, ilustración, animación, desarrollo de videojuegos, modelado 3D, entre otros. Muchas compañías han tratado de disipar el temor de que sus empleados sean reemplazados por inteligencia artificial y, como una estrategia empresarial

positiva e interesante, han decidido capacitar a sus empleados en el uso de estas herramientas como una alternativa para aumentar su velocidad y eficiencia en la producción. Esto puede ser útil para hacer bocetos, generar nuevas ideas o agilizar procesos de pintura o diseño.

Stable Diffusion ha tenido un impacto significativo en el ámbito laboral, al igual que otros modelos destacados en el campo de la inteligencia artificial. Empresas de gran envergadura vinculadas a diversos sectores industriales, como diseño, arte, producción musical e incluso desarrollo de videojuegos, han adoptado este avanzado modelo de inteligencia artificial para generar ideas innovadoras que permitan materializar sus conceptos creativos. Al mismo tiempo, proporciona a los artistas profesionales la oportunidad de trabajar en las creaciones generadas por la inteligencia artificial.

En esencia, uno de los principales usos de Stable Diffusion en la industria es facilitar el proceso de diseño artístico al transformar ideas plasmadas en texto en bocetos generados por el modelo de inteligencia artificial. Dado que las creaciones producidas por este sistema no son perfectas, el rol principal de los profesionales en este contexto es supervisar el contenido generado, trabajar en él y desarrollar nuevas ideas, entre otras funciones.

Conclusión del capítulo

Este capítulo ha ofrecido una introducción exhaustiva a Stable Diffusion, un modelo de inteligencia artificial revolucionario para la generación de imágenes. Se han explorado los mecanismos técnicos subyacentes, sus múltiples aplicaciones y las implicancias en el mundo del arte y la industria. A través de una comprensión detallada de su historia, sus versiones y sus capacidades, el lector debería estar ahora preparado para apreciar su importancia y el papel que juega en la configuración de la era moderna del arte digital. El análisis de los debates en curso sobre la posible sustitución del trabajo humano por la inteligencia artificial también ha proporcionado una perspectiva importante sobre las preocupaciones éticas y prácticas en este campo. En resumen, este capítulo ha servido como una base sólida para comprender y apreciar Stable Diffusion, sentando las bases para una exploración más profunda en los siguientes capítulos.

CAPÍTULO 4: “UNA IMAGEN VALE MÁS QUE MIL PALABRAS”: PRIMEROS PASOS EN STABLE DIFFUSION

Objetivos del capítulo

El objetivo de este capítulo es presentar una introducción detallada y accesible a Stable Diffusion, destacando su impacto y versatilidad en la generación de imágenes. Se enfoca en guiar a los lectores a través del proceso práctico de utilizar la difusión estable, sin necesidad de conocimientos profundos en programación, y en explorar la conexión con Hugging Face, una empresa tecnológica clave en el acceso a este modelo. Además, el capítulo busca mostrar cómo generar imágenes utilizando Stable Diffusion mediante ejemplos prácticos, ofreciendo una comprensión clara y aplicable de sus capacidades y potencialidades.

Al analizar Stable Diffusion, es innegable el impacto positivo que ha generado en diversas industrias y aplicaciones. Su versatilidad y potencial en distintos campos la convierten en una herramienta valiosa y prometedora. Sin embargo, en lugar de limitarnos a discutir teóricamente sus méritos, resulta útil explorar su uso práctico. Después de todo, es esencial

experimentar con la difusión estable para determinar si realmente cumple con las expectativas planteadas.

En esta sección, se presentará cómo utilizar la difusión estable de manera gratuita y sencilla, sin necesidad de conocimientos previos en programación o enfrentarse a códigos y terminologías complejas. ¿Cómo se logra esto? Aunque quizás no sea sorprendente para algunos, la accesibilidad de la difusión estable como modelo de inteligencia artificial de código abierto ha permitido que numerosos desarrolladores lo integren en sitios web y aplicaciones. Esto significa que los usuarios pueden aprovechar este modelo sin la necesidad de instalar paquetes que consuman espacio en su almacenamiento ni de ajustar manualmente los diferentes parámetros que estos modelos requieren.

En resumen, es posible utilizar diferentes versiones de la difusión estable en múltiples sitios web a través de internet. En esta sección, veremos de manera sencilla diferentes imágenes generadas usando la difusión estable, lo que nos ayudará a obtener una idea más clara y precisa de las diversas capacidades que tenemos para usar este modelo. Ahora bien, aunque en esta sección tendremos la oportunidad de ver estas generaciones de imágenes mencionadas, lo cierto es que, tal y como se ha señalado, este modelo sigue en constante actualización. Además, diferentes aplicaciones y técnicas de generación con estas imágenes surgen con el paso del tiempo, haciendo que las tecnologías que utilizan la difusión estable sean cada vez más frecuentes y numerosas.

Por lo tanto, realmente no significa que lo que vamos a ver en esta sección o dentro de este libro sea absoluto, sino que corresponde solamente a un proceso mucho más grande de evolución de este modelo. Por ello, también es importante mantenerse actualizado y siempre atento a los nuevos lanzamientos de este tipo de modelos.

Stable Diffusion y Hugging Face: La comunidad de IA construyendo el futuro

Como se ha mencionado previamente, Stable Diffusion es un modelo de inteligencia artificial de código abierto. Esto implica que cualquier persona puede acceder a los archivos y elementos utilizados en su creación y entrenamiento. De esta manera, se brinda la oportunidad de incorporar este modelo de inteligencia artificial en proyectos propios, fomentando así la innovación y el desarrollo en diversos ámbitos.

Esta accesibilidad posibilita que Stable Diffusion se implemente en diversas plataformas, incluidos sitios web, lo cual representa una de las aplicaciones más comunes de este modelo de inteligencia artificial. Diferentes compañías y particulares han creado sitios web donde los usuarios pueden acceder y utilizar

este modelo, facilitando aún más su adopción y exploración en distintos contextos.

No obstante, surge la pregunta: ¿qué es Hugging Face y cuál es su relación con Stable Diffusion? A pesar de que no parezca evidente, estos dos conceptos están estrechamente relacionados y, de cierta manera, Hugging Face será el elemento clave que permitirá utilizar Stable Diffusion de forma sencilla y rápida. En este contexto, se hace referencia a la posibilidad mencionada anteriormente de emplear Stable Diffusion a través de un sitio web.

En términos simples, Hugging Face es una empresa tecnológica especializada en el ámbito de la inteligencia artificial, con un enfoque particular en el aprendizaje profundo (deep learning) y el procesamiento del lenguaje natural. Hugging Face ha ganado popularidad gracias a su capacidad para ofrecer acceso a modelos de inteligencia artificial potentes y reconocidos, como GPT y el modelo central de este libro, Stable Diffusion.

Para utilizar Hugging Face y Stable Diffusion, en primer lugar, es necesario visitar la página oficial de Hugging Face. Esta se puede encontrar fácilmente realizando una búsqueda en cualquier motor de búsqueda, como Google, por ejemplo.

Una vez localizada y accedido a la página oficial, se debe emplear el sistema de búsqueda que el sitio web proporciona para encontrar el modelo de Stable Diffusion.

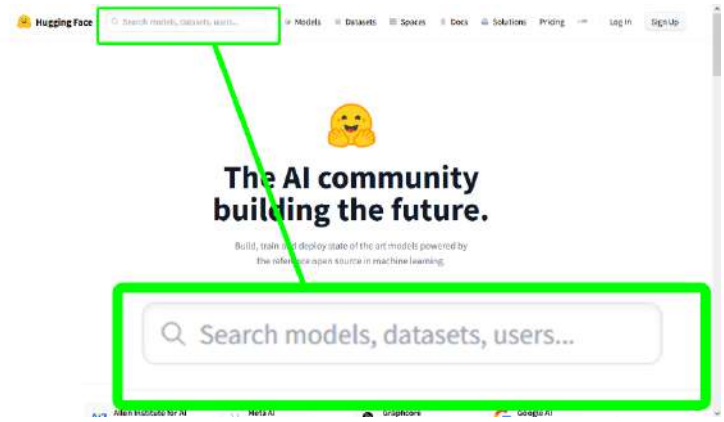


Ilustración 26 – Sistema de búsqueda en la página principal de Hugging Face

La sección del cuadro de búsqueda es la que resulta de mayor interés en este caso. Aquí, se pueden introducir palabras clave para encontrar diversos elementos disponibles en Hugging Face. Además, al buscar el término "Stable Diffusion", se podrá localizar el sitio web "oficial" que facilita la experimentación y prueba de Stable Diffusion de manera sencilla.

Al introducir el término "Stable Diffusion" en el cuadro de búsqueda, se observará que Hugging Face proporciona numerosos resultados. Es importante destacar

que Hugging Face es una plataforma que ofrece diversas opciones y resultados relevantes relacionados con la comunidad de inteligencia artificial. Esto ha permitido que Hugging Face se convierta en una de las principales plataformas preferidas para trabajar y encontrar modelos, bases de datos y recursos vinculados a la inteligencia artificial, con un enfoque especial en el aprendizaje profundo (deep learning).

En la búsqueda que se realice, se podrán encontrar diversos resultados importantes, incluidos los repositorios de este potente modelo de inteligencia artificial, junto con sus respectivas versiones, bases de datos y otros recursos relacionados.

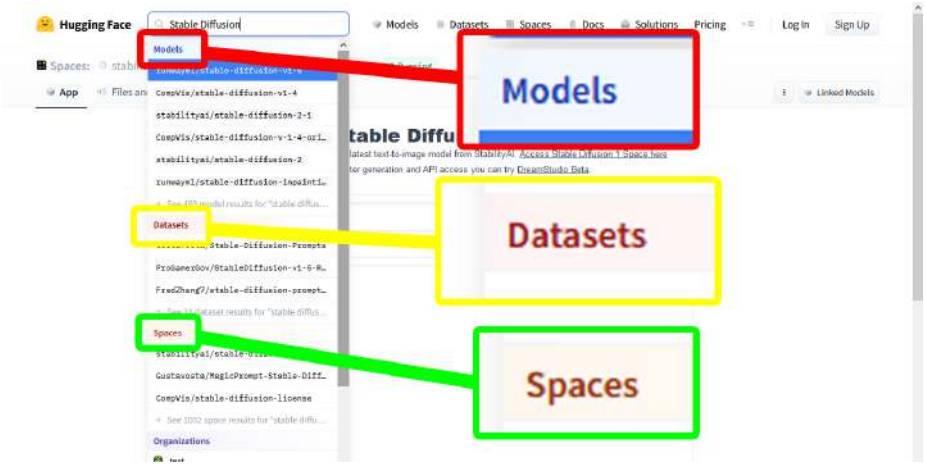


Ilustración 27 – Diferentes espacios y resultados que ofrece Hugging Face en su plataforma

A pesar de que es excelente contar con la variedad de recursos que Hugging Face ofrece al buscar el término "Stable Diffusion" en su plataforma, el área de mayor interés en este momento es la sección de "Spaces", la cual se encuentra resaltada en verde en la ilustración proporcionada.

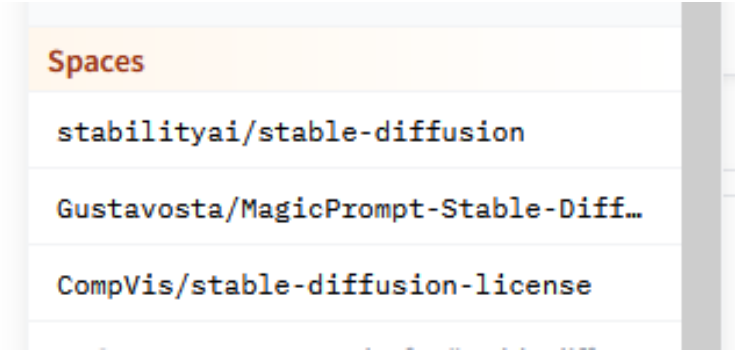


Ilustración 28 – Sección "Spaces" de término de búsqueda "Stable Diffusion" en Hugging Face

El resultado de interés en este caso es aquel denominado "stabilityai/stable-diffusion", que corresponde a un espacio en el sitio de Hugging Face publicado

por StabilityAI. Este espacio permite a los usuarios probar fácilmente la versión más reciente del modelo Stable Diffusion de una manera sencilla y sin salir del sitio web.

Una vez ingresado, se accederá al sitio web de demostración del modelo de inteligencia artificial Stable Diffusion. Esta sección, publicada por StabilityAI, permite utilizar este potente modelo de inteligencia artificial en el navegador web de manera sencilla, sin la necesidad de escribir código ni acceder a archivos importantes del sistema o comprender a fondo el funcionamiento interno del modelo de inteligencia artificial.

En este sitio, se brinda la oportunidad de disfrutar de este asombroso modelo simplemente introduciendo las indicaciones de lo que se desea generar con él.

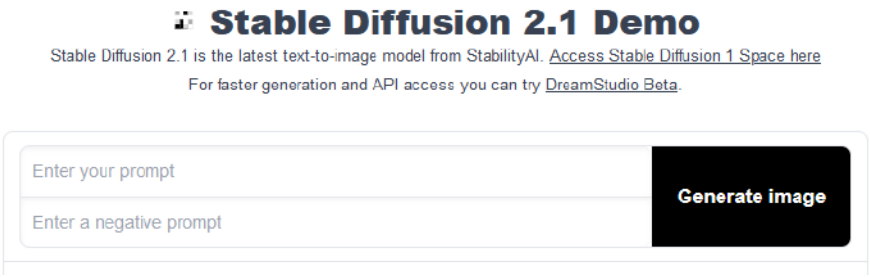


Ilustración 29 – Demo de Stable Diffusion 2.1 en el sitio web de Hugging Face

¡Y eso es todo! Ahora se cuenta con la posibilidad de utilizar Stable Diffusion dentro del navegador web, lo cual representa una de las formas más sencillas y prácticas para aprovechar fácilmente este modelo de inteligencia artificial, generar resultados y guardarlos en la computadora. Simplemente basta con escribir la descripción de lo que se desea generar y, en poco tiempo, se obtendrán resultados generados por este modelo de inteligencia artificial directamente en la pantalla.

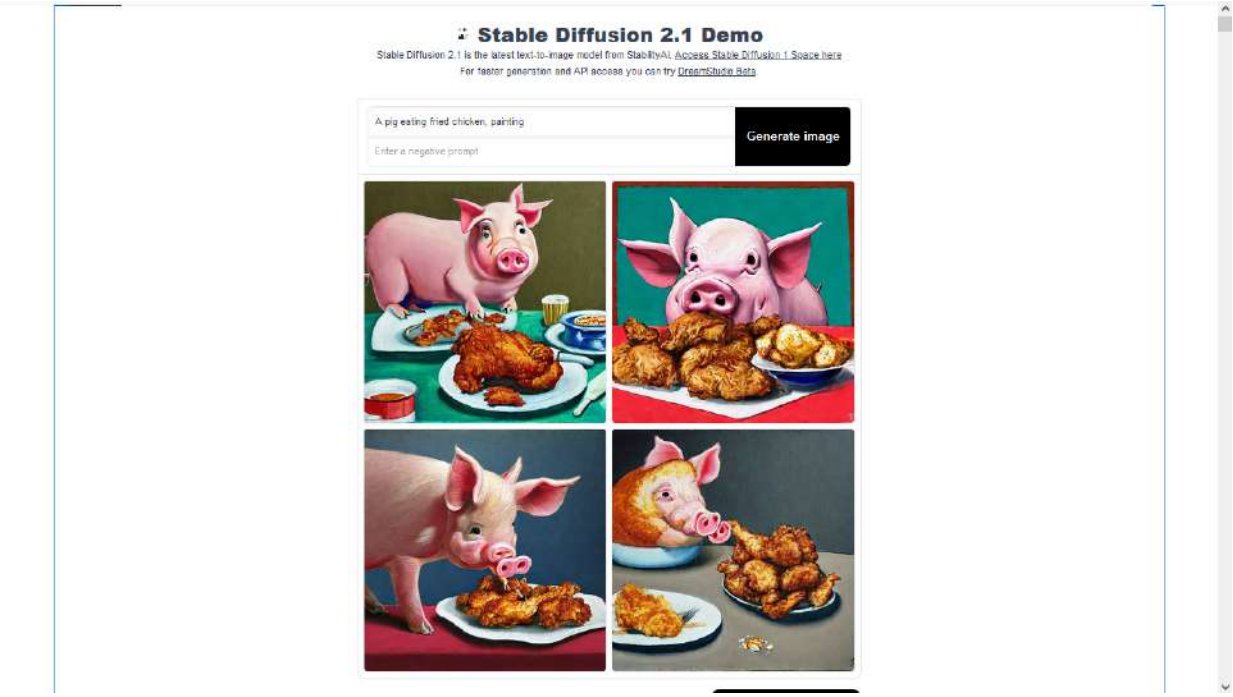


Ilustración 30 – Captura de pantalla de generación de imágenes de Stable Diffusion

En este espacio, se puede experimentar proporcionando al modelo la descripción de la imagen que se desea generar. Simplemente hay que escribir el texto en inglés de lo que se quiere en la caja de texto que indica "Enter your prompt". Por ahora, se debe ignorar la caja de texto que dice "Enter a negative prompt"; este aspecto se explicará con mayor detalle en la siguiente sección.

A continuación, se muestran ejemplos de imágenes generadas por este modelo utilizando el espacio en el sitio web de Hugging Face, acompañadas de sus respectivas descripciones e imágenes generadas:

Generación Stable Diffusion 5 – Modern house made of Legos (Casa moderna hecha de Legos)

Generación Stable Diffusion 6 - A futuristic village in the space (Un pueblo futurista en el espacio)

Conclusión del capítulo

El capítulo ha logrado proporcionar una visión integral y accesible de Stable Diffusion y su aplicación en la generación de imágenes. A través de una explicación detallada y guiada, se ha demostrado cómo cualquier persona puede utilizar este potente modelo de inteligencia artificial, incluso sin experiencia en programación. La relación entre Stable Diffusion y Hugging Face se ha ilustrado claramente, permitiendo a los lectores comprender cómo aprovechar estas tecnologías en su navegador web. Además, se ha destacado la naturaleza en constante evolución de este modelo, lo que subraya la importancia de mantenerse actualizado con los desarrollos futuros. En resumen, este capítulo ha servido como una puerta de entrada a la exploración práctica y creativa de Stable Diffusion, posibilitando a los usuarios a generar imágenes detalladas a partir de descripciones de texto.

¿QUÉ SON EL “PROMPT” Y “NEGATIVE PROMPT”?

Objetivos del capítulo

El objetivo principal de este capítulo es explorar y analizar en profundidad los conceptos de "prompt" y "negative prompt" en el contexto de la generación de imágenes utilizando el modelo de Stable Diffusion. Esto incluirá la definición de estos términos, la importancia de la ingeniería de prompts en la generación de imágenes, y la exploración de cómo el "prompt" y el "negative prompt" pueden ser utilizados para lograr resultados específicos y precisos. Se proporcionarán ejemplos prácticos para ilustrar estos conceptos y se examinará cómo interactúan con el modelo de Stable Diffusion en la plataforma de Hugging Face.

Como se analizó previamente, al utilizar el modelo de Stable Diffusion en el entorno proporcionado por Hugging Face, nos encontramos con la solicitud de ingresar un "prompt". Este espacio de texto permite introducir la descripción que la inteligencia artificial debe utilizar para generar la imagen. No obstante, ¿qué implica realmente este término? Aunque para muchos no resulte sorprendente, el término "prompt" posee cierta relevancia e importancia en el ámbito de la inteligencia artificial.

La forma más sencilla de definir un "prompt" en este campo hace referencia a una cadena de texto de entrada para un modelo de procesamiento de lenguaje natural, de manera que contenga instrucciones o sugerencias para que el modelo genere una respuesta basada en su interpretación.

A pesar de que redactar prompts puede parecer una tarea simple, al momento de utilizar un modelo de inteligencia artificial de procesamiento de lenguaje natural para realizar alguna tarea, el término "prompt" sigue siendo esencial para determinar el resultado obtenido por la IA. Es probable que, en tareas básicas, la "calidad" del prompt no tenga tanta relevancia; no obstante, a medida que se requieran resultados más precisos y adaptados a nuestras necesidades para llevar a cabo tareas más complejas, la importancia de los prompts introducidos en el modelo aumenta significativamente.

A raíz de este fenómeno, surgió un concepto bastante reciente en la comunidad de inteligencia artificial denominado "ingeniería de prompts" (prompt engineering). En términos simples, se trata del proceso o conocimiento mediante el cual se intenta establecer los pasos a seguir para proporcionar el prompt adecuado a un modelo de inteligencia artificial. De esta forma, el modelo se ajusta y genera resultados acordes a las expectativas. Cuanta más información relevante se suministre al modelo con respecto a la respuesta deseada y las tareas específicas que debe realizar, más preciso será el resultado que se busca obtener.

Este fenómeno puede observarse directamente al interactuar con Stable Diffusion, ya que, en efecto, indicamos al modelo las instrucciones para obtener lo que deseamos. Cuanto más específicos seamos en nuestras indicaciones, mejores resultados generarán el modelo.

A continuación, se presenta un ejemplo que ilustra esta idea, comparando dos entradas similares, pero con distintos niveles de detalle:

Generación Stable Diffusion 7 - Pumpkin ballerina in garden (Bailarina de calabaza en el jardín)

Generación Stable Diffusion 8 - (beautiful pumpkinpunk ballerina) in the (bottom of the garden), pumpkins, intricate, highly detailed, digital painting, artstation, concept art, sharp focus, cinematic lighting, illustration, art by artgerm and greg rutkowski, alphonse mucha, cgsociety ((hermosa bailarina de calabaza punk) en (el fondo del jardín), calabazas, intrincado, muy detallado, pintura digital, artstation, arte conceptual, enfoque nítido, iluminación cinematográfica, ilustración, arte de artgerm y greg rutkowski, alphonse mucha, cgsociety)

Es importante destacar que las capacidades que ofrece Stable Diffusion son amplias e importantes; no obstante, la razón de la "apariencia extraña" de las imágenes generadas por el modelo se debe a que se está empleando una versión de Stable Diffusion con menor cantidad de parámetros. Esto se realiza con el objetivo de optimizar el rendimiento de tal manera que pueda ejecutarse rápidamente dentro del sitio web. Sin embargo, también tendremos la oportunidad de utilizar el modelo con un mayor poder computacional para obtener resultados de mejor calidad.

Como podemos observar, ambos resultados fueron ingresados con prompts relativamente similares, aunque el segundo contiene mucha más información sobre lo que se desea obtener. Esto permite obtener de manera precisa el resultado esperado por parte de la inteligencia artificial. Así, podemos apreciar de forma más directa la aplicación de lo que se conoce como "ingeniería de prompts". A continuación, se presenta una comparación del mejor resultado entregado por ambos prompts, analizando la calidad y el resultado obtenido entre ambos casos.

En este ejemplo, el de la izquierda corresponde al prompt ingresado con pocas descripciones, mientras que el segundo es donde se aplica la "ingeniería de prompts".

	
"Pumpkin ballerina in garden"	"(beautiful pumpkinpunk ballerina) in the (bottom of the garden), pumpkins, intricate, highly detailed, digital painting, artstation, concept art, sharp focus, cinematic lighting, illustration, art by artgerm and greg rutkowski,

Como podemos observar, existe una gran diferencia entre los resultados obtenidos, lo que nos permite comprender que la ingeniería de prompts y la forma en que ingresamos el prompt en nuestro modelo influyen significativamente en los resultados generados.

Por esta razón, es crucial adquirir conocimientos y experiencia sobre el funcionamiento de estos modelos de inteligencia artificial, así como entender cómo aprovechar al máximo su potencial. De este modo, podremos obtener resultados óptimos a través de Stable Diffusion, basándonos en nuestras metas y en los objetivos que deseamos alcanzar.

En las siguientes secciones, presentamos varios ejemplos adicionales de este concepto, en los cuales se pueden apreciar diversos casos de resultados sorprendentes obtenidos al aplicar la ingeniería de prompts en Stable Diffusion, todos ellos utilizando la plataforma de Hugging Face:

Generación Stable Diffusion 9 - (cool black neon-lights ninja) in the (futuristic neon-lights dark foggy cyber-punk city), cinematic lighting, close-up picture, extremely realistic ((ninja con luces de neón negras geniales) en la (ciudad ciber-punk con luces de neón futuristas, niebla oscura), iluminación cinematográfica, imagen de primer plano, extremadamente realista)

Generación Stable Diffusion 10 - a (huge Cthulhu ornate grey and black antropomorphic body), (attacking and sinking ship), stormy waves, lots of fine details, cinematic wallpaper, sharp focus, megalophobia, dark tones palette, extremely detailed, bioluminescent details, overcast reflection mapping, movie quality, vfx post production, atmospheric lighting, hyperrealism, insanely detailed, unreal engine 5 ((enorme cuerpo antropomórfico ornamentado gris y negro de Cthulhu), (barco que ataca y se hunde), olas tormentosas, muchos detalles finos, papel tapiz cinematográfico, enfoque nítido, megalofobia, paleta de tonos oscuros, extremadamente detallado, detalles bioluminiscentes, mapeo de reflejo nublado, película calidad, posproducción de efectos visuales, iluminación atmosférica, hiperrealismo, increíblemente detallado, unreal engine 5)

Generación Stable Diffusion 11 - a portrait of an old coal miner in 19th century, beautiful painting with highly detailed face by greg rutkowski and magali villanueve (un retrato de un Viejo minero del carbon en el siglo XIX, hermosa pintura con un rostro muy detallado de Greg Rutworski y Magali Villanueve)

Ahora, con esto, tenemos una idea más clara de la importancia que tiene el prompt al momento de generar diferentes tipos de resultados. También tuvimos la oportunidad de ver algunos ejemplos donde se aplica la ingeniería de prompts para generar los resultados esperados por parte del modelo de inteligencia artificial. Aunque pueda parecer trivial, todos estos conceptos son bastante importantes al momento de generar diversos resultados y hacer que el obtenido se ajuste completamente a lo que deseamos.

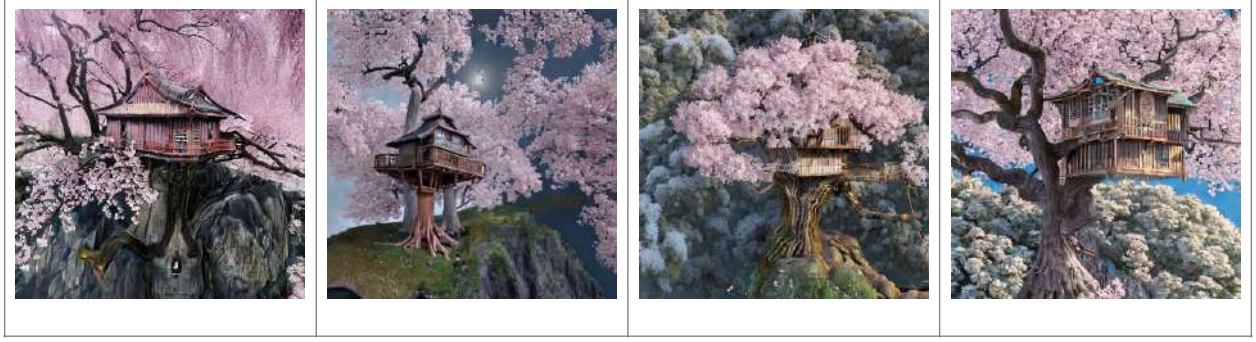
Aunque esto es bastante importante y útil hasta cierto punto, no es la única opción con la que contamos para generar diferentes tipos de resultados usando Stable Diffusion o, para ser más precisos, el espacio asignado para este modelo de inteligencia artificial en el sitio web de Hugging Face. Si bien es cierto que es importante tener en cuenta cómo funciona la ingeniería de prompts y cómo podemos usarla eficientemente dentro de nuestras generaciones de Stable Diffusion para sacarle provecho a este modelo de inteligencia artificial, también es cierto que contamos con más parámetros con los que podemos jugar para obtener diferentes tipos de resultados en las generaciones que nos ofrece el modelo. En este caso, nos referimos a dos otros parámetros con los que contamos para variar nuestros resultados generados en el modelo: el "negative prompt" y el "guidance scale". Por ahora, nos centraremos en explicar el funcionamiento de qué es, cómo funciona y cómo podemos usar a nuestro favor el "negative prompt", debido a que la explicación de "guidance scale" es un poco compleja e involucra más terminología técnica; por lo tanto, se explicará en la sección posterior a esta.

Es probable que ya sea algo trivial para muchos saber esto, pero al momento de usar el modelo de Stable Diffusion en la página oficial de Hugging Face, podemos notar que nos solicitan en dos cajas de texto nuestro "prompt" y el "negative prompt". Podemos ver más en detalle en qué parte se encuentra el negative prompt en la siguiente captura de pantalla:

Ilustración 31 – Ubicación de la caja de texto de "negative prompt" en Stable Diffusion de Hugging Face

Sin embargo, ¿a qué se refiere realmente este modelo con el "negative prompt"? Puede parecer un poco difícil, sin embargo, es todo lo contrario, y de alguna forma, el propio nombre indica cuál es la función principal de este espacio. En términos simples, como podemos inferir de alguna manera, el término "negative prompt" hace referencia a aquellos detalles y características de una imagen que queremos que nuestro modelo no genere u omita. Es decir, en el caso de que estemos generando por medio de un prompt una imagen de, por ejemplo, una ciudad algo nublada, triste y con bastante humo, podríamos usar el parámetro de negative prompt de nuestro modelo para indicar que no queremos ninguna de esas características en el resultado, mostrando así una versión que no tenga las características que hemos descrito dentro de esta caja de texto.

Podemos ver algunos ejemplos de este mismo concepto con las siguientes generaciones de Stable Diffusion, ambas reciben el mismo prompt, sin embargo, a una no se le da un negative prompt mientras que a la otra sí, podemos ver las diferencias a continuación:



Generación Stable Diffusion 12 - (beautiful ornate treehouse) in a (gigantic pink cherry blossom tree), on a high blue grey and brown cliff with light snow and pink cherry blossom, Intricate details, very realistic, cinematic lighting, volumetric lighting, photographic ((hermosa casa del árbol adornada) en un (gigantesco árbol de flor de cerezo rosa), en un alto acantilado gris azulado y marrón con nieve ligera y flor de cerezo rosa, detalles intrincados, muy realista, iluminación cinematográfica, iluminación volumétrica, fotografía)

Sin embargo, podemos notar que, aunque no lo hayamos indicado en nuestro prompt, por algún motivo las imágenes generadas tienen colores algo opacos, se ven oscuras y les falta un brillo especial. Podemos entonces indicar por medio del negative prompt que queremos que nuestro modelo no genere, en efecto, colores opacos, iluminación oscura, entre otros detalles que puedan entorpecer la belleza de la imagen. Al proporcionar estas instrucciones adicionales mediante el negative prompt, es posible refinar los resultados generados por el modelo y obtener imágenes más acordes con nuestras expectativas y preferencias estéticas.

Generación Stable Diffusion 13 – Negative prompt: opaque colors, night, old treehouse, creepy treehouse, moon light (colores opacos, noche, caza de árbol antigua, caza de árbol aterradora, luz nocturna)

¡Voilá! Ahora nuestro resultado definitivamente presenta un aspecto mucho más positivo. Aunque hemos aplicado anteriormente lo que conocíamos como ingeniería de prompts, hemos notado que con solo ingresar dentro de nuestro prompt lo deseado no ha sido suficiente para generar los resultados deseados. Sin embargo, usando el negative prompt hemos podido hacer que el resultado sea mucho más agradable visualmente, con colores mucho más vivos, una casa del árbol más atractiva, una luz de día bastante agradable y, en general, un resultado artístico más profesional y adecuado a lo que queremos obtener. Esto demuestra que la combinación de técnicas, como la ingeniería de prompts y el uso adecuado de los negative prompts, puede ayudarnos a obtener imágenes generadas por modelos de inteligencia artificial que se ajusten mejor a nuestras expectativas y necesidades. A continuación, se presentan los ejemplos previos con diferentes negative prompts aplicados para mejorar la calidad y el enfoque de las imágenes generadas:

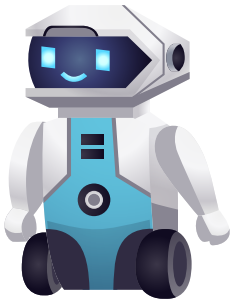
Generación Stable Diffusion 14 – Negative prompt: opaque color, foggy, moon, night light, big breasts, exhibitionism, creepy forest, dark (color opaco, brumoso, luna, luz nocturna, pechos grandes, exhibicionismo, bosque espeluznante, oscuro)

Generación Stable Diffusion 15 – Negative prompt: opaque colors, multiple city colors (colores opacos, ciudad de múltiples colores)

Generación Stable Diffusion 16 – Negative prompt: duplicated, sun light, excesive tentacles (duplicado, luz solar, tentáculos excesivos)

Generación Stable Diffusion 17 – Negative prompt: Photo frame, painting frame, hats, deformed face, helmets, sad face (Marco de fotos, marco de pintura, sombreros, cara deformada, cascos, cara triste)

Conclusión del capítulo



Este capítulo ha ofrecido una visión completa de los términos "prompt" y "negative prompt" en el ámbito de la generación de imágenes, enfocándose en su aplicación dentro del modelo de Stable Diffusion. Hemos visto cómo el "prompt" actúa como una guía detallada para el modelo, permitiendo un control preciso sobre los resultados generados. La ingeniería de prompts ha emergido como un área crucial en la obtención de imágenes detalladas y de alta calidad. Por otro lado, el "negative prompt" ha sido explicado como una herramienta para indicar las características que deben ser excluidas de la imagen generada. A través de ejemplos concretos y análisis comparativos, se ha demostrado cómo estos dos elementos pueden ser empleados de manera efectiva para lograr resultados que se ajusten exactamente a lo deseado, destacando la necesidad de entender y utilizar estos conceptos para sacar el máximo provecho de las capacidadesdes de Stable Diffusion.

¿QUÉ ES EL “GUIDANCE SCALE” Y CÓMO AFECTA LOS RESULTADOS?

Objetivos del capítulo

El propósito de este capítulo es examinar detalladamente dos aspectos fundamentales en la generación de imágenes con Stable Diffusion. Primero, se busca explicar el concepto de "escala de guía" (guidance scale) y su impacto en la generación de imágenes, demostrando cómo afecta la fidelidad y literalidad de la representación visual con respecto a la descripción textual. El capítulo incluirá ejemplos prácticos y experimentos que varían la escala de guía para mostrar cómo influye en los resultados generados. Segundo, se tiene como objetivo introducir las alternativas disponibles a Hugging Face para la implementación de Stable Diffusion en la web, explorando las posibles variantes y adaptaciones del modelo en diferentes plataformas.

Como mencionamos previamente, además del prompt y el negative prompt como únicos parámetros modificables en este modelo para generar diferentes resultados, contamos con un parámetro adicional que, aunque no lo parezca, desempeña un papel crucial en la generación de imágenes. Nos referimos a la ‘escala de guía' (Guidance scale) de nuestras imágenes.

Entonces, ¿qué es exactamente la escala de guía y por qué es un término relevante en nuestro modelo? Para responder adecuadamente a esta pregunta, es necesario considerar varios factores importantes que influyen en la generación de una respuesta por parte de Stable Diffusion.

Para comprender plenamente este fenómeno en la generación de resultados utilizando Stable Diffusion, analicemos un ejemplo práctico. En este caso, optaremos por un ejemplo sencillo sin realizar ingeniería de prompts. Simplemente indicaremos al modelo que deseamos generar algo básico; por ejemplo, un tiburón practicando surf en el mar. A continuación, se presentan las imágenes generadas por el modelo:

Generación Stable Diffusion 18 – a shark surfing in the sea (Un tiburón practicando surf en el mar)

Desde una perspectiva humana, podemos comprender fácilmente que la imagen generada corresponde a la descripción solicitada, pero ¿cómo sabe el modelo qué especificaciones seguir al crear la imagen? ¿Debería mostrar un tiburón en una tabla de surf? ¿Realista? ¿De qué color debe ser el mar?

A pesar de no haber proporcionado estos detalles, el modelo ha generado una respuesta basada en la solicitud. Sin embargo, podemos observar que el modelo realiza cierta inferencia de lo que ocurre o debería mostrarse, aunque no sea totalmente fiel al prompt original que hemos ingresado. Simplemente lo hace de tal manera que sea suficiente para satisfacer nuestras expectativas como usuarios.

En términos simples, la escala de guía es un valor numérico que oscila entre 0 y 50, el cual puede ser modificado en el sitio web de Hugging Face antes de generar una imagen. La función de la escala de guía es bastante sencilla: indica cuán fiel y literal debe ser el modelo respecto a la descripción textual proporcionada. Es decir, cuanto menor sea el número ingresado, menos literal será en relación con la descripción proporcionada, y cuanto mayor sea el número, más literal será con respecto a la descripción textual indicada.

Uno podría preguntarse: ¿Por qué necesitaría que mi modelo no sea fiel a la descripción proporcionada? ¿No debería utilizar el valor más alto de la escala de guía para obtener resultados más acordes con lo que deseo? Aunque este

razonamiento puede parecer válido, en realidad no lo es. Establecer un valor de escala de guía excesivamente alto o excesivamente bajo puede generar resultados contraproducentes en las imágenes generadas por nuestro modelo.

Para comprender de manera más precisa este concepto, a continuación, se presenta una comparación de la generación de imágenes mediante Stable Diffusion, utilizando distintos valores de valor de guía en un mismo prompt. En este caso, todos ellos corresponden al mismo prompt que describe un tiburón practicando surf. Es importante señalar que, por lo general, el valor predeterminado del coeficiente guía es de 9.0.

Valor de guía: 0.0	Valor de guía: 5.5	Valor de guía: 11.0	Valor de guía: 16.5	Valor de guía: 22.0
				
Valor de guía: 27.5	Valor de guía: 33.0	Valor de guía: 38.5	Valor de guía: 44.0	Valor de guía: 50.0
				

Al introducir distintos valores de guía, obtenemos variados resultados, con diferencias significativas en algunos casos. Valores extremadamente pequeños o grandes pueden afectar negativamente el resultado, como se aprecia en los valores 0.0, 33.0, 38.5 y 50.0. La calidad de las imágenes es adecuada en el rango numérico [5.5 – 27.5], pero a partir del valor de guía 33.0, se deteriora, mostrando ruido. Sin embargo, en el valor específico de 44.0, la calidad se recupera y se encuentra dentro de los estándares aceptables.

En este caso, los cambios pueden ser sutiles debido al prompt general y no específico. Podemos experimentar con un prompt más complejo, aplicando ingeniería de prompts para evaluar el comportamiento del valor de guía en contextos más elaborados. Consideremos un ejemplo de generación de imágenes con el siguiente prompt (imágenes con valor de guía predeterminado de 9.0):

Generación Stable Diffusion 19 – Prompt: painted portrait of rugged zeus, god of thunder, greek god, white hair, upper body, fantasy, intricate, elegant, highly detailed, artstation, sharp focus, art by gaston bussiere / Negative prompt: duplicated, opaque colors, blurry, bad quality, deformed face (Prompt: retrato pintado de zeus robusto, dios del trueno, dios griego, cabello blanco, parte superior del cuerpo, fantasía, intrincado, elegante, muy detallado, estación de arte, enfoque nítido, arte de gaston bussiere / Negative prompt: duplicado, colores opacos, borroso, mala calidad, cara deformada)

Al repetir el experimento modificando los valores de guía para este prompt, obtenemos los siguientes resultados:

Valor de guía: 0.0	Valor de guía: 5.5	Valor de guía: 11.0	Valor de guía: 16.5	Valor de guía: 22.0
				
Valor de guía: 27.5	Valor de guía: 33.0	Valor de guía: 38.5	Valor de guía: 44.0	Valor de guía: 50.0
				

Como podemos observar, en este caso, los valores en los que la imagen no coincide o pierde calidad son 0.0, 16.5, 22.0, 38.5, 44.0 y 50.0. Aunque no hay un valor óptimo específico, es recomendable utilizar valores cercanos al predeterminado, preferentemente entre 3 y 30.

Entonces, ¿cuándo es importante usar el valor de guía si el predeterminado ya genera buenos resultados? Generalmente, el valor predeterminado es suficiente; sin embargo, esto varía según el prompt y la calibración del modelo, pues no siempre se garantiza que los resultados generados sean los solicitados. Hemos presenciado este fenómeno, especialmente al generar imágenes de Cthulhu en la página 33, que se muestran a continuación:

En este ejemplo, utilizaremos la imagen que aplica ingeniería de prompts, pero no el uso de un negative prompt. Pedimos a Stable Diffusion generar a Cthulhu atacando un barco; sin embargo, aunque se generó a Cthulhu, no se muestra atacando al barco especificado en el prompt. Es decir, el modelo interpretó "parcialmente" incorrectamente la solicitud.

En casos así, es pertinente modificar el valor de guía para que el modelo se apegue más a la descripción y genere la imagen esperada. Valores de guía más altos conducen a una mayor fidelidad a la descripción. Repitamos el experimento de modificación del valor de guía utilizando el mismo prompt a continuación:

Valor de guía: 0.0	Valor de guía: 5.5	Valor de guía: 11.0	Valor de guía: 16.5	Valor de guía: 22.0
				
Valor de guía: 27.5	Valor de guía: 33.0	Valor de guía: 38.5	Valor de guía: 44.0	Valor de guía: 50.0

Obtenemos diversos resultados, la mayoría con buena calidad de imagen, excepto en los valores 0.0, 22.0 y 33.0. En este caso, las imágenes que coinciden con lo solicitado tienen valores de guía de 38.5, 44.0 y 50.0. Este es un caso particular donde se requiere un valor de guía alto para obtener los resultados esperados del modelo. A continuación, se presentan los tres elementos generados por el modelo que cumplen con las características deseadas:

Resultados acordes modificando el valor de guía		
		

En estas tres imágenes, a diferencia de las obtenidas previamente, se aprecia claramente que cumplen con la característica solicitada, donde Cthulhu ataca o intenta hundir al barco. De esta forma, ahora comprendemos mejor la importancia de modificar el valor de guía cuando el modelo no produce una salida esperada.

Alternativas a Hugging Face de Stable Diffusion en la web

Como mencionamos previamente, una de las principales ventajas de Stable Diffusion radica en su naturaleza de proyecto de código abierto. Esto implica que cualquier persona puede estudiar en profundidad este modelo, utilizarlo e implementarlo en diversos proyectos. Además, como hemos experimentado de primera mano, Stable Diffusion permite la implementación de su modelo y la generación de resultados a través de sitios web.

Esta circunstancia ha propiciado el surgimiento de numerosos sitios web, tanto de compañías como de individuos o grupos independientes, que ofrecen sus "propias versiones" o variantes del modelo de inteligencia artificial Stable Diffusion. De esta forma, Hugging Face no es la única plataforma disponible para generar imágenes a partir de descripciones textuales utilizando Stable Diffusion.

No obstante, esto no implica que los modelos de cada uno de estos sitios web sean idénticos y, en consecuencia, proporcionen resultados similares a los obtenidos al utilizar la plataforma de Hugging Face. Por el contrario, la esencia de Stable Diffusion radica en brindar a los usuarios la posibilidad de personalizar el modelo y generar distintos tipos de imágenes según las necesidades y capacidades del mismo, ofreciendo así un amplio abanico de opciones en cuanto a las posibilidades que se pueden lograr con diferentes sitios web. A continuación, presentamos algunos ejemplos breves de resultados generados por distintos sitios web, cada uno de ellos con enfoques diversos entre sí:

Generación Stable Diffusion 20 - Misterious girl dancing under the rain, black and white, intricated, rain (https://stablediffusionweb.com/#demo)

Generación Stable Diffusion 21 - Misterious girl dancing under the rain, black and white, intricated, rain (https://replicate.com/stability-ai/stable-diffusion) – K_EULER_ANCESTRAL scheduler.

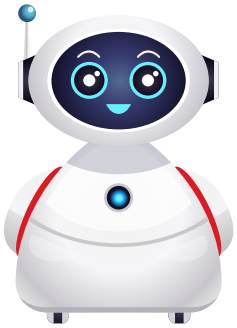
Generación Stable Diffusion 22 - Misterious girl dancing under the rain, black and white, intricated, rain (https://creator.nightcafe.studio/studio) – Candy V2

Generación Stable Diffusion 23 - Misterious girl dancing under the rain, black and white, intricated, rain (https://hotpot.ai/art-generator?s=stable-diffusion-api) – Sculpture General 1

Generación Stable Diffusion 24 - Misterious girl dancing under the rain, black and white, intricated, rain (https://dezgo.com/) – Stable Diffusion 2.1

Generación Stable Diffusion 25 - Misterious girl dancing under the rain, black and white, intricated, rain (https://www.mage.space/) – Default

Resulta intrigante observar cómo las aplicaciones y los resultados obtenidos mediante el uso de Stable Diffusion pueden variar en diferentes sitios web, evidenciando una gran diversidad. Esto sugiere que, aunque estos sitios web se fundamenten en el mismo modelo de inteligencia artificial, no siempre proporcionan idénticos resultados. Al contrario, es posible hallar una variedad de soluciones en línea, de modo que la opción seleccionada para nuestro proyecto se adecue y satisfaga nuestras necesidades y expectativas.



Conclusión del capítulo

Este capítulo ha proporcionado una comprensión exhaustiva del papel que desempeña la escala de guía en la generación de imágenes a través de Stable Diffusion. Se han presentado múltiples experimentos y ejemplos para ilustrar cómo la variación de este valor numérico puede influir en la literalidad y fidelidad de las imágenes generadas. Asimismo, se ha esclarecido por qué no existe un valor óptimo universal y cómo la selección de la escala de guía puede requerir experimentación y ajuste basado en el prompt específico. Además, se ha presentado una breve introducción a las alternativas a Hugging Face en la implementación de Stable Diffusion en la web, sentando las bases para una exploración más profunda de las diferentes opciones disponibles en el campo de la generación de imágenes mediante aprendizaje profundo.

CAPÍTULO 5: OTRAS FUNCIONES DE STABLE DIFFUSION

Objetivos del capítulo

En este capítulo, se pretende expandir la comprensión del lector sobre las capacidades multifacéticas del modelo Stable Diffusion, más allá de su función principal de generación de imágenes a partir de descripciones textuales. Los objetivos principales son presentar la funcionalidad de "imagen a imagen" dentro de Stable Diffusion, explicar su aplicación práctica mediante ejemplos concretos, y explorar la personalización y ajustes avanzados que el usuario puede realizar en el proceso de generación de imágenes. También se propone proporcionar una introducción a conceptos más avanzados como la escala de guía y la semilla de generación, ayudando al lector a entender cómo estos parámetros influyen en los resultados finales.

Hasta el momento, hemos explorado el uso de Stable Diffusion para generar imágenes a partir de descripciones textuales: le proporcionamos una idea al modelo y este responde con una representación visual. Este enfoque ha demostrado ser útil en la solución de problemas y en la generación de diversas ideas y proyectos.

Es importante resaltar que las capacidades de Stable Diffusion no se limitan únicamente a la generación de imágenes a partir de descripciones textuales. Si bien esta es su función más conocida y popular, este modelo de inteligencia artificial también brinda otras posibilidades para la creación de imágenes que

van más allá de la utilización de descripciones textuales.Por ejemplo, además de la generación de imágenes a través de descripciones textuales (conocido como "texto a imagen"), Stable Diffusion ofrece la funcionalidad de "imagen a imagen". En este escenario, en lugar de usar una descripción textual como punto de partida, se utiliza una imagen ya existente.

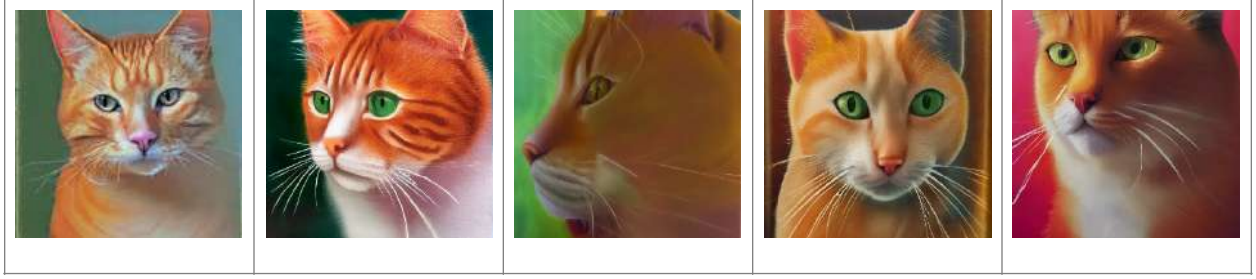
Esto puede parecer un poco confuso al principio, especialmente si no se está familiarizado con los modelos generativos de contenido digital. Sin embargo, con una explicación conceptual sólida, es sencillo de comprender. Un ejemplo hipotético puede ayudar a entender mejor cómo funciona este proceso.

Caso ejemplar: Modelo “Stable Diffusion Image Variation”

En un ejemplo sencillo, usemos imágenes generadas por el propio modelo Stable Diffusion, sin recurrir a recursos externos. Imaginemos que usamos Stable Diffusion para crear un retrato de un gato en primer plano, tarea fácil para el modelo. Así, podríamos obtener resultados como los siguientes (usaremos el espacio en línea de Hugging Face para este ejemplo):

Generación Stable Diffusion 26 - (orange adult cat) in a (photo portrait), first plane, realistic, fine details

Imaginemos que queremos modificar o trabajar con la primera imagen generada por el modelo. Podríamos tener varias razones para hacerlo, como agregar detalles, eliminar elementos no deseados, obtener variaciones, cambiar el estilo, etc. Las posibilidades con Stable Diffusion son amplias, lo que ofrece un gran abanico de resultados potenciales. En este caso, para facilitar el aprendizaje y evitar temas complejos, usaremos nuevamente un espacio de Hugging Face con el modelo Stable Diffusion, mostrando que sin conocimientos técnicos avanzados ni mucho tiempo invertido, se pueden disfrutar las funciones de este potente modelo de forma gratuita. El espacio que usaremos está en el sitio web de Hugging Face y se llama "Stable Diffusion Image Variations", que, como su nombre indica, genera variaciones de una imagen. Veamos qué ocurre si ingresamos al modelo la primera imagen generada previamente:



Generación Stable Diffusion 27 – Variaciones de: (orange adult cat) in a (photo portrait), first plane, realistic, fine details

En este caso, como mencionamos previamente y conforme a su nombre y funcionalidad, podemos observar que la primera imagen en la tabla corresponde a la imagen original. Esta fue creada utilizando el modelo Stable Diffusion de StabilityAI en el proceso de conversión de texto a imagen. Las otras cuatro imágenes mostradas son las generadas por el modelo "Stable Diffusion Image Variations". Aquí podemos apreciar que ninguna es idéntica a las demás y, efectivamente, cada una representa diferentes resultados generados por el modelo al intentar crear variaciones de la misma imagen, conservando el mismo estilo y características principales.

Es cierto que las imágenes variantes generadas por el modelo poseen un matiz de "variante artística", es decir, parecen más caricaturescas y con un estilo pictórico. Esto puede deberse a la naturaleza estilizada de la imagen original o también porque este modelo no está optimizado para ofrecer un rendimiento completo, de manera que pueda ser ejecutado a través del navegador web.

En esta sección del sitio web de Hugging Face, también podemos personalizar de forma más avanzada los distintos parámetros de generación de nuestro modelo, tal como lo hacíamos anteriormente con el modelo en línea de StabilityAI. En este caso, podemos apreciar que los parámetros que podemos modificar son mucho más extensos, lo que nos permite, en teoría, personalizar aún más las generaciones de nuestro modelo.

El uso de los diferentes parámetros que podemos emplear en este modelo de inteligencia artificial son temas que por ahora no hemos tenido la oportunidad de ver; sin embargo, realmente podemos decir que no es un tema del todo complejo y también tendremos la oportunidad de estudiar los diferentes parámetros con los que contamos dentro de Stable Diffusion para poder modificar y hacer que las imágenes que generamos por medio de este modelo de inteligencia artificial se ajusten cada vez más a nuestras necesidades al momento de generar imágenes. Es decir, si bien es cierto que por ahora no los hemos estudiado, tendremos la oportunidad de verlos más detenidamente y entender cómo es que los diferentes tipos de parámetros que podemos ver dentro de esta página, que nos permite usar Stable Diffusion de forma gratuita, pueden ayudar a que las generaciones que hagamos por medio de este modelo se ajusten a lo que buscamos.

La siguiente captura de pantalla muestra los diferentes parámetros que podemos ajustar dentro de nuestro modelo, lo cual nos permite variar las salidas y adaptarlas a nuestras necesidades específicas en cuanto a lo que deseamos que el modelo genere como resultado.

Ilustración 32 – Captura de pantalla de los parámetros modificables de "Stable Diffusion Image Variations"

En esta captura de pantalla, se muestran de manera más directa los diversos parámetros avanzados que se pueden ajustar en el modelo para obtener distintos tipos de resultados. Aunque estos parámetros puedan parecer complejos al principio, no hay motivo para preocuparse. De hecho, ninguno de ellos es difícil de comprender, y pronto analizaremos cada uno, su función y los diferentes resultados que se pueden obtener al modificarlos.

Stable Diffusion Image Variations: Guidance Scale

Ahora analizaremos cada uno de los distintos parámetros que se pueden ajustar en el modelo para obtener diferentes resultados. En este caso, exploraremos el funcionamiento del parámetro "Guidance Scale" en la generación de nuestras imágenes.

Efectivamente, el parámetro "Guidance Scale" no es nuevo, ya lo analizamos cuando estudiamos el modelo de generación de imágenes "texto a imagen" de Stable Diffusion subido en Hugging Face por StabilityAI. Llegamos a la conclusión de que la función principal de este parámetro en la generación de texto a imagen era, en términos simples, determinar qué tan fiel debía ser la generación a la descripción textual proporcionada al modelo. Vimos que modificar el valor de la escala de guía (o "Guidance Scale") a valores muy pequeños o grandes podía tener efectos contraproducentes en la generación de contenido del modelo.

No obstante, como se ha observado previamente, este no es el caso en esta situación. Esta discrepancia se debe principalmente al hecho de que el enfoque

ya no se centra en un modelo de generación de contenido de texto a imagen, sino en un modelo de imagen a imagen. ¿Cuál es la relevancia de este cambio? Aunque pueda parecer insignificante, es bastante significativo. En realidad, el concepto del valor guía en la generación del modelo no es completamente desconocido en comparación con lo que se conocía anteriormente. Es decir, la generación de este modelo basado en el valor que se especifique para la escala de guía también dependerá y variará considerablemente en función de la imagen original proporcionada.

Conceptualmente, la idea subyacente continúa siendo la misma: a menor valor asignado al parámetro guía, menos "fidelidad" presentarán las imágenes generadas con respecto a la imagen original. Por el contrario, a mayor valor asignado, más fiel será la generación de imágenes por parte del modelo. Retomemos el experimento realizado previamente, en el que se modificó el valor guía utilizando la imagen base del gato, generada con anterioridad, como imagen de entrada para el modelo. Cabe destacar que, a diferencia de las pruebas efectuadas en etapas anteriores, en este caso, el valor guía del modelo se representa mediante un valor numérico que fluctúa entre 0 y 25.

Valor de guía: 0.0	Valor de guía: 2.8	Valor de guía: 5.6	Valor de guía: 8.3	Valor de guía: 11.1
				
Valor de guía: 13.9	Valor de guía: 19	Valor de guía: 19.4	Valor de guía: 20	Valor de guía: 25
				

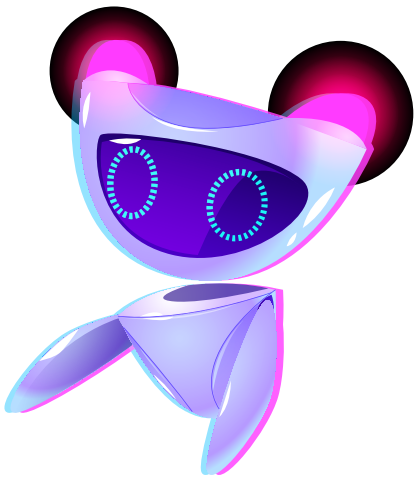
Como se puede apreciar, un valor excesivamente elevado del parámetro guía en el modelo puede provocar efectos negativos, como la generación de imágenes de baja calidad que pierden relación con la imagen original proporcionada. Además, se pueden identificar otros fenómenos en la generación de imágenes, aparte del mencionado. Por ejemplo, las imágenes generadas presentan similitudes entre ellas, a pesar de que las generaciones anteriores eran totalmente aleatorias. Este

fenómeno se debe a un parámetro ajustable en el modelo, llamado "semilla de generación" o "seed".

La función de esta semilla es sencilla de comprender: es un valor numérico que introduce variación en los valores aleatorios generados por el modelo. La semilla actúa como un identificador único y personalizable para diferentes valores aleatorios que pueden ajustarse según las preferencias del usuario. Es importante resaltar que trabajar con números completamente aleatorios es difícil y consume muchos recursos informáticos. Por esta razón, la generación de aleatoriedad en los modelos suele ser pseudoaleatoria en lugar de realmente aleatoria.

Aunque estos conceptos puedan parecer complejos, no hay motivo para preocuparse. A lo largo del libro, se examinarán detalladamente todos los aspectos relacionados con el funcionamiento de este modelo de imagen a imagen en particular. Tal como se mencionó previamente, la importancia de todos estos tipos de parámetros que podemos utilizar y modificar dentro de Stable Diffusion radica en lograr que las imágenes finales generadas se adapten mejor a nuestras necesidades personales, permitiendo una mayor variedad en las generaciones de imágenes y la obtención de diferentes tipos de resultados. Estos parámetros pueden modificarse fácilmente a través de un sitio web en línea que se puede usar en nuestro navegador. También se ofrece la capacidad de utilizar el modelo y modificar todos sus parámetros de forma local, con el modelo instalado en nuestra propia computadora.

Conclusión del capítulo



El capítulo 5 ha abordado de manera exhaustiva las diferentes posibilidades y funcionalidades de Stable Diffusion, más allá de la conversión de "texto a imagen". A través de una exploración detallada y ejemplos concretos, hemos visto cómo se pueden lograr variaciones de imágenes, adaptar los resultados según nuestras necesidades y comprender los diferentes parámetros que permiten una personalización avanzada. La presentación del caso ejemplar "Stable Diffusion Image Variation" ha ofrecido una visión práctica sobre cómo manipular y aprovechar las diversas capacidades del modelo. Además, se ha proporcionado una introducción a conceptos más avanzados y técnicos, sin perder la accesibilidad para los lectores que pueden no estar familiarizados con ellos. La conclusión clave de este capítulo es la versatilidad y la adaptabilidad del modelo Stable Diffusion en el campo de la generación y manipulación de imágenes, abriendo una amplia gama de posibilidades creativas y aplicaciones prácticas.

STABLE DIFFUSION IMAGE VARIATIONS: NUMBER IMAGES

Objetivos del capítulo

El objetivo de este capítulo es explorar en detalle dos parámetros esenciales del modelo Stable Diffusion que se utilizan para la variación de imágenes: "Number images" y "Steps". Se pretende explicar el concepto de "Number images" y cómo se utilizan diferentes valores para generar múltiples imágenes a partir de una misma semilla. Además, el capítulo busca examinar a fondo la función del parámetro "Steps" en el proceso de generación de imágenes y cómo afecta la calidad y eficiencia del modelo. Se utilizarán ejemplos prácticos, gráficos e ilustraciones para facilitar la comprensión de estos conceptos.

Este parámetro es uno de los más fáciles de entender y, conceptualmente, uno de los más simples. Consiste en un valor numérico que oscila entre 1 y 4, el cual indica la cantidad de imágenes que se desea generar y que el modelo debe crear. Es importante destacar que cada una de estas 4 imágenes utiliza como dato de entrada la imagen proporcionada por el usuario. Sin embargo, cada una de las imágenes generadas es completamente diferente de las otras. A pesar de que las imágenes varían entre sí, todas se originan a partir de la misma semilla de generación.

Al principio, este concepto puede parecer un poco confuso; sin embargo, se analizarán más a fondo las distintas funcionalidades que ofrece la semilla de generación en la creación de imágenes por parte del modelo en secciones posteriores de este libro. Por ahora, es suficiente entender que la función principal

de este parámetro es definir la cantidad de imágenes que el modelo generará como resultado. Si el valor es 1, el modelo generará una única imagen; si el valor es 2, generará dos imágenes, y así sucesivamente.

Stable Diffusion Image Variations: Steps

A continuación, se explorará el tercer parámetro que puede ajustarse al generar imágenes utilizando este modelo de inteligencia artificial basado en Stable Diffusion para la variación de imágenes. En este caso, se trata del parámetro "steps" del modelo, el cual es probablemente uno de los más importantes que deben tenerse en cuenta al desarrollar un modelo. ¿Por qué? Se explicará con más detalle la razón detrás de esto a continuación.

Stable Diffusion, como se puede inferir de alguna manera a partir del nombre del modelo, es un modelo de inteligencia artificial que genera imágenes utilizando una técnica conocida como "Diffusion", o simplemente técnica de difusión en español.

Esta técnica se explorará con mayor profundidad en otras secciones de este libro. Por el momento, es suficiente comprender que es el nombre que recibe la técnica mediante la cual este modelo de inteligencia artificial crea imágenes. Una característica clave y bastante importante de esta técnica es que se trata de un proceso iterativo. ¿Qué significa esto realmente y qué relación tiene con el parámetro de pasos que debe seguir nuestra generación de Stable Diffusion? Se puede intentar entender mejor este concepto de la siguiente manera:

Imaginemos, por ahora, que el proceso de generación de imágenes de Stable Diffusion consiste en un ciclo repetitivo. Es decir, un proceso en el cual nuestro modelo, inicialmente, genera una imagen; luego, repite el proceso con esta imagen, mejorándola un poco; después, repite el proceso y mejora aún más la imagen previamente generada, y así sucesivamente.

Se puede visualizar de forma más clara el proceso que lleva a cabo nuestro modelo de inteligencia artificial mediante el siguiente gráfico:

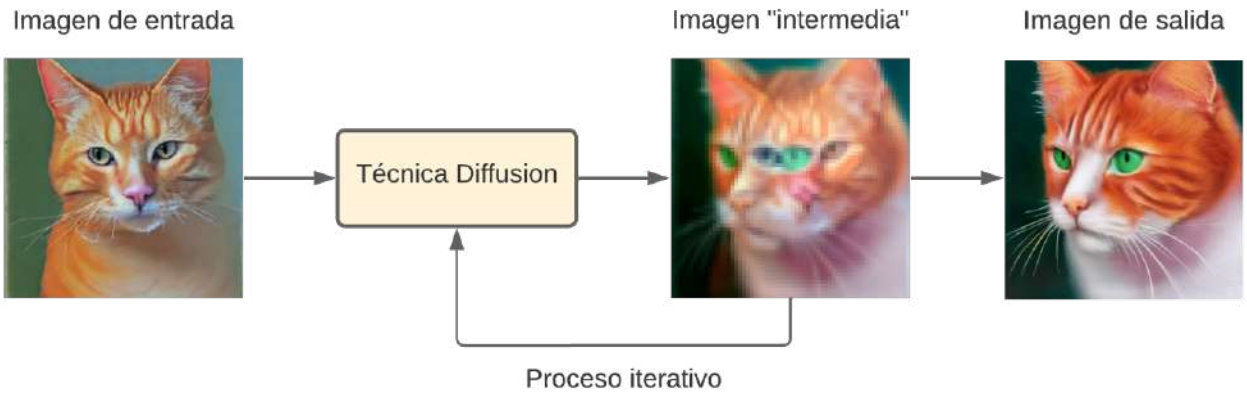


Ilustración 33 – Proceso iterative de generación de imágenes en Stable Diffusion

Es fundamental aclarar también que, en realidad, el proceso interno que se lleva a cabo dentro de Stable Diffusion, así como lo que hemos señalado en la descripción del gráfico como nuestra "imagen intermedia", es un proceso mucho más complejo.

Sin embargo, se puede afirmar que conceptualmente, esta es la forma más sencilla y fácil de visualizar para poder entender, de manera superficial, el proceso que ocurre dentro de un modelo de imagen a imagen de Stable Diffusion.

De esta manera, mediante la descripción del gráfico, podemos observar a qué nos referimos cuando decimos que la generación de imágenes dentro de Stable Diffusion es un proceso iterativo.

Como mencionamos, nuestro modelo recibe la imagen de entrada, luego aplica la técnica de difusión, la cual genera una "imagen". Después, "repite" el proceso de difusión, actualizando la imagen generada. Este paso de "repetir" el proceso es a lo que nos referimos específicamente como iteración.

De este modo, teniendo una comprensión más clara del concepto de iteración en el proceso de generación de imágenes utilizando la técnica de difusión, podemos afirmar que el valor de "steps" o los pasos que debe seguir nuestro modelo corresponde, efectivamente, a la cantidad de iteraciones que nuestro modelo debe realizar antes de mostrarnos la imagen generada.

Pero ¿cómo puede afectar este parámetro en las diferentes generaciones que nuestro modelo puede crear? Como mencionamos anteriormente, de hecho, este es uno de los parámetros más importantes que podríamos necesitar usar al generar nuestros modelos, especialmente si lo que requerimos es un resultado de "alta calidad".

Expliquemos un poco más con un ejemplo práctico y visual, donde podremos ver los diferentes resultados que podemos obtener al modificar los valores de los pasos que debe seguir nuestro modelo:

Steps: 5	Steps: 10	Steps: 15	Steps: 20	Steps: 25
				
Steps: 30	Steps: 35	Steps: 40	Steps: 45	Steps: 50
				

Ahora, dentro de estas generaciones realizadas por el modelo de inteligencia artificial, podemos observar varios puntos importantes que debemos considerar al analizar los resultados. Uno de estos aspectos principales que debemos destacar y que es importante notar en las generaciones es que, notoriamente, el cambio en la imagen es poco o incluso nulo después de un valor numérico de 15 en este parámetro. Es decir, los cambios dentro de la imagen realmente no son significativos ni relevantes como para marcar una gran diferencia en las generaciones producidas por el modelo.

Como se explicó anteriormente, cuanto mayor sea el valor de este parámetro, el modelo realizará más iteraciones en el proceso de difusión para generar imágenes de mejor calidad.

Cada proceso de iteración implica, entonces, que el modelo se encargará de mejorar progresivamente la imagen, de tal forma que haya un avance significativo. En pocas palabras, cuantas más veces se repita el proceso, más se perfeccionará la imagen.

Podemos apreciar mejor este proceso si hacemos que el valor oscile entre 5 y 15, donde se puede ver un cambio más notorio e importante en el impacto que tienen los pasos en la generación de las imágenes.

Steps: 5	Steps: 6	Steps: 7	Steps: 8	Steps: 9
				
Steps: 10	Steps: 11	Steps: 12	Steps: 13	Steps: 14
				

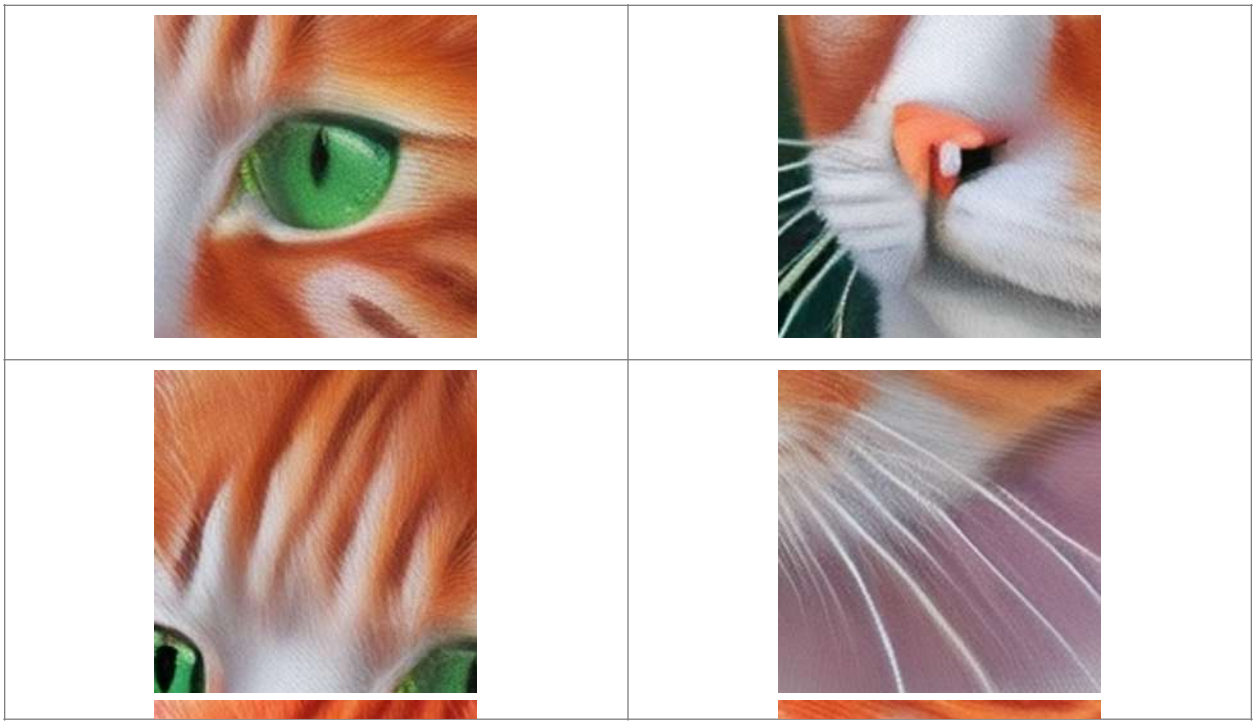
De esta forma, es posible apreciar con mayor claridad el proceso que se lleva a cabo para generar la imagen. De hecho, se podría afirmar que, incluso a partir de la generación realizada en el valor de 10 pasos, ya se tiene una idea del resultado final que el modelo presentará. Después de este valor, realmente no hay mucho margen para mejorar, es decir, se llega a un punto en el que el modelo intenta perfeccionar la imagen de manera constante, pero debido a que ya ha alcanzado una versión casi óptima, los cambios realizados en la imagen son mínimos y no generan una diferencia significativa.

Por lo tanto, no siempre es necesario asignar la mayor cantidad de pasos posibles al modelo. Si bien es cierto que, teóricamente, una mayor cantidad de pasos indicados al modelo podría mejorar la calidad de la imagen, no siempre es la mejor opción por considerar. En ciertos casos, puede ser preferible evaluar la eficiencia y el rendimiento del proceso, equilibrando la calidad de la imagen con la cantidad de recursos y tiempo empleados.

En ejemplos previos, se pudo observar claramente cómo asignar un valor numérico de 10 pasos resultaba más que suficiente. Indicar una cantidad mayor de pasos implica, en realidad, un consumo innecesario de recursos computacionales, lo que conlleva a que el modelo tarde mucho más tiempo en generar la imagen de lo que sería estrictamente necesario. En este contexto, es importante evaluar la relación entre la calidad de la imagen y el rendimiento del proceso, a fin de optimizar el uso de los recursos disponibles y mejorar la eficiencia del modelo de inteligencia artificial en cuestión.

Sin embargo, no se puede negar que existe una diferencia notoria en la calidad entre las imágenes generadas con distintas cantidades de pasos. Aunque ambas presentan similitudes considerables, la imagen generada con la mayor cantidad de pasos posible ofrece una calidad superior en comparación con aquella que recibió la cantidad mínima de pasos necesarios para alcanzar el resultado esperado, en este caso, un valor numérico de 10.

Este fenómeno puede analizarse más detenidamente al observar las diferentes características de las imágenes, como el tipo de pelaje del gato, la forma de los ojos, los colores, entre otros aspectos. Así, aunque es importante considerar la eficiencia y el rendimiento del proceso, también resulta fundamental evaluar el nivel de calidad deseado en función de las necesidades y expectativas específicas para cada aplicación del modelo de inteligencia artificial.



Al comparar las imágenes generadas, se pueden apreciar diferencias en distintos elementos. En cada una de las comparaciones, se presenta, en el lado izquierdo, la imagen generada con un valor numérico de 10 pasos, mientras que, en el lado derecho, se muestra la misma escena exacta pero generada con un valor de 50 pasos. Aunque las diferencias no son drásticas, se puede observar una mejora en cuanto a la forma de las partes generadas, los pequeños detalles y los colores, entre otros aspectos.

No obstante, es posible que no siempre sea necesario contar con todos estos detalles adicionales. En ocasiones, es más importante priorizar un valor numérico que proporcione un resultado suficientemente adecuado, en lugar de utilizar valores excesivamente altos que únicamente conlleven un mayor tiempo de

ejecución. Así, la selección del número de pasos óptimo dependerá de las necesidades y prioridades específicas de cada aplicación del modelo de inteligencia artificial.

Existen varios detalles adicionales relacionados con el parámetro numérico de pasos que vale la pena mencionar. Aunque no se ha abordado de manera directa, es posible observar que, a medida que disminuye el valor numérico, la generación resultante se vuelve más similar a la primera imagen proporcionada, es decir, la imagen de entrada. No hay que preocuparse si este fenómeno no es evidente de inmediato, ya que sería más notorio si se pudieran utilizar valores inferiores a 5. Sin embargo, este modelo en particular no lo permite, por lo que será necesario adoptar enfoques más creativos para demostrar y analizar este comportamiento.

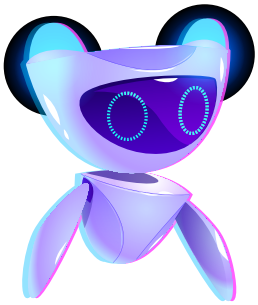
Para apreciar este fenómeno con mayor claridad, se puede emplear nuevamente el modelo y modificar el valor de la semilla (seed) utilizando números aleatorios, manteniendo el resto de los parámetros en sus valores predeterminados, excepto la cantidad de pasos. Estos se establecerán en el valor mínimo posible; en este caso, el valor mínimo aceptado por el modelo corresponde al valor numérico de 5.

A continuación, se presentan algunas imágenes que ilustran lo mencionado anteriormente:

Es cierto que las imágenes generadas pueden resultar un tanto desconcertantes; sin embargo, al dejar esto de lado, se puede observar de manera más directa el fenómeno mencionado anteriormente. En efecto, todas estas imágenes presentan cierto grado de similitud visual con la imagen principal proporcionada, ya que fueron generadas utilizando la misma entrada. Además, debido a que el modelo no ha realizado demasiados pasos de generación iterativos, la similitud con la imagen original se aprecia con mayor claridad, mostrando así una notable correspondencia entre ellas.

De este modo, se puede considerar concluido el estudio de la función que desempeña el parámetro de pasos (steps) en la generación de imágenes llevada a cabo por el modelo de Stable Diffusion. A lo largo del análisis, se ha observado cómo este parámetro influye en la calidad, la similitud con la imagen de entrada y el tiempo de ejecución del modelo. Entender y ajustar adecuadamente este parámetro permitirá optimizar el rendimiento del modelo en función de las

necesidades y objetivos específicos de cada aplicación en el campo de la inteligencia artificial.



Conclusión del capítulo

En este capítulo, hemos abordado dos parámetros clave en la generación de imágenes utilizando el modelo Stable Diffusion: "Number images" y "Steps". Hemos descubierto cómo el parámetro "Number images" permite controlar la cantidad de imágenes generadas, y cómo todas ellas se originan a partir de la misma semilla. Además, se ha explorado el concepto de iteración dentro del parámetro "Steps" y cómo el ajuste de este valor puede influir en la calidad, eficiencia, y detalle de las imágenes generadas. Se ha proporcionado una visión clara de cómo equilibrar la calidad con la eficiencia, resaltando la importancia de la selección cuidadosa del número de pasos en función de las necesidades y expectativas de cada aplicación del modelo. Los ejemplos e ilustraciones presentados han contribuido a una mejor comprensión de estos conceptos complejos, subrayando que no siempre es necesario asignar la mayor cantidad de pasos posible al modelo, y que una evaluación cuidadosa puede conducir a una utilización óptima de los recursos.

STABLE DIFFUSION IMAGE VARIATIONS: SEED

El objetivo principal

de este capítulo es explorar y desglosar el último parámetro modificable en el modelo de Stable Diffusion, denominado "seed" o "semilla". Se pretende explicar en detalle el papel y la función que desempeña la semilla en la generación de imágenes, con un enfoque especial en su importancia para obtener resultados reproducibles y variados. Asimismo, el capítulo busca ilustrar mediante ejemplos prácticos cómo la semilla influye en la generación de imágenes y cómo puede ajustarse para experimentar con diferentes resultados en el modelo Stable Diffusion.

En este punto, es crucial examinar el último parámetro modificable en el modelo de Stable Diffusion: el parámetro "seed" o "semilla". A lo largo del libro, se han presentado ejemplos en los cuales se mencionaba o modificaba este parámetro, pero no se había explicado aún su relevancia en el proceso de generación de imágenes. Es fundamental comprender el papel que desempeña este parámetro en la producción de resultados.

De hecho, el concepto de semilla de generación en el modelo de Stable Diffusion no es particularmente complejo. De hecho, se encuentra entre los más fáciles de comprender dentro de los parámetros del modelo, junto con el número de

imágenes generadas. La semilla es un elemento clave para entender y controlar en el proceso de generación de imágenes.

Tal como se mencionó previamente, los resultados generados por el modelo de Stable Diffusion no son completamente aleatorios. En el ámbito informático, el uso de una semilla para generar resultados es una práctica común, aunque a menudo se pasa por alto o se relega a un segundo plano. La semilla de valores aleatorios en casi cualquier contexto se refiere a un valor numérico que "inicializa" un proceso pseudoaleatorio. En otras palabras, funciona como un identificador principal de inicialización en elementos que, por lo general, se generan de manera aleatoria.

En el caso del modelo de Stable Diffusion, la semilla proporciona la capacidad de generar resultados "aleatorios" con diferentes inicializaciones y valores aleatorios, según se especifique. ¿Parece complicado? Para facilitar la comprensión, analicemos algunos ejemplos concretos que ilustren el funcionamiento de este parámetro.

En primer lugar, se utilizará el modelo de Stable Diffusion proporcionado por StabilityAI en la plataforma de Hugging Face, con el objetivo de variar las imágenes empleadas en los ejemplos y no centrarse exclusivamente en la misma imagen de un gato. Para este caso, se proporcionará al modelo un prompt sencillo y conciso: la palabra "park", que se traduce como "parque" en español. No es necesario que la imagen sea altamente descriptiva para ilustrar adecuadamente este ejemplo explicativo.

Generación Stable Diffusion 28 – Park (Parque)

En este ejemplo, se seleccionará la cuarta imagen resultante de la generación para llevar a cabo el experimento en el modelo "Stable Diffusion Image Variations". Este experimento se centrará en explorar el efecto del valor del parámetro de la semilla de generación en la producción de imágenes a partir de la generación del modelo.

Al inicializar una generación utilizando este modelo, el parámetro de semilla se asigna por defecto como un valor numérico de cero (0). Este valor puede modificarse libremente, adoptando cualquier número, desde valores muy grandes

hasta incluso negativos. La semilla es el único parámetro que se puede ajustar sin restricciones.

A continuación, se pueden observar las imágenes generadas por el modelo con el valor predeterminado de la semilla. Es importante señalar que todos los demás parámetros del modelo se mantendrán en sus valores predeterminados:




Los resultados obtenidos hasta el momento muestran que las imágenes generadas son variantes de la misma idea. Aunque no presentan diferencias significativas, es posible observar ciertas variaciones entre ellas, lo que indica la presencia de diversidad en las imágenes generadas a pesar de su similitud.

A continuación, se modificará el valor de la semilla a un número distinto al predeterminado. Para este ejemplo, se utilizará un valor numérico de -17 para la semilla, aunque el valor preciso no es crítico, siempre y cuando sea diferente del valor predeterminado.




Efectivamente, se puede apreciar que el concepto de generar variantes de la misma imagen se mantiene, a pesar de haber modificado el valor de la semilla. Las imágenes obtenidas con la semilla de -17 son diferentes a las generadas con el valor numérico de cero, el valor predeterminado del parámetro. No obstante, esto no implica que exista una relación directa entre el valor de la semilla y las características específicas de las imágenes generadas. Al regresar al valor original de la semilla, es decir, el valor numérico de cero, se obtienen las siguientes imágenes (Nota: Las imágenes mostradas NO fueron copiadas y pegadas de la página anterior. Para mantener la fidelidad al concepto, se generaron nuevamente las imágenes en el modelo por segunda vez):

Tal como se puede apreciar, al regresar el valor de la semilla a cero, se obtienen las mismas imágenes que se generaron previamente antes de modificar el valor de la semilla predeterminado al valor numérico de -17.

Por lo tanto, como se mencionó anteriormente, el propósito de la semilla es proporcionar un valor numérico a partir del cual el modelo generará valores aleatorios. Sin embargo, surge la pregunta: ¿cuál es el motivo principal detrás de esto? Podría parecer que generar imágenes completamente aleatorias o mantenerlas constantes no tiene un propósito claro. No obstante, esta suposición no es del todo cierta, ya que existen numerosas ventajas al considerar este enfoque.

Una de las principales ventajas al ajustar el valor de la semilla en las generaciones realizadas por el modelo es, por ejemplo, la capacidad de obtener resultados reproducibles. Esto permite llevar a cabo experimentos variados y modificar los valores cuando sea necesario. En otras palabras, resulta útil poder regenerar la misma salida del modelo para modificar valores y realizar pruebas con estas salidas ajustando otros parámetros. Asimismo, facilita la implementación de modificaciones pertinentes que se ajusten a los requisitos específicos del usuario.

Aunque podríamos proporcionar ejemplos específicos, el concepto de la semilla en nuestros modelos para determinar la inicialización de una generación creada por el modelo resulta bastante claro. Por lo tanto, en este caso particular, no se presentará un ejemplo para ilustrar que, al cambiar la semilla a un valor utilizado previamente, efectivamente se obtienen los mismos resultados que se lograron anteriormente.

Conclusión del capítulo

La exploración del parámetro "semilla" en el modelo de Stable Diffusion ha arrojado una comprensión detallada de su función en la generación de imágenes. A través de explicaciones teóricas y ejemplos prácticos, se ha mostrado cómo este parámetro simple pero fundamental permite controlar y reproducir los resultados generados. La semilla es un aspecto clave en el proceso de generación de imágenes, y su correcto manejo es esencial para llevar a cabo experimentos precisos y modificaciones específicas. La comprensión de la semilla no solo enriquece nuestra visión general del modelo Stable Diffusion sino que también subraya la importancia de los elementos que a menudo se dan por sentado en los procesos de aprendizaje profundo y generación de imágenes.

CASO EJEMPLAR: CONTROLNET

Objetivos del capítulo

El propósito de este capítulo es proporcionar un análisis en profundidad de ControlNet, una herramienta avanzada que funciona como una extensión de Stable Diffusion. Los objetivos específicos incluyen explicar qué es ControlNet, cómo funciona, y cómo se relaciona con Stable Diffusion. Además, el capítulo busca ilustrar cómo ControlNet puede interpretar imágenes e inferir aspectos como la profundidad y los elementos sobresalientes en ellas. Finalmente, se ofrece una guía práctica sobre cómo acceder y utilizar ControlNet, con ejemplos visuales que demuestren su funcionamiento y capacidades únicas en la modificación de imágenes.

Anteriormente, se tuvo la oportunidad de analizar los diversos usos que se pueden dar a Stable Diffusion mediante la aplicación de un modelo llamado "Stable Diffusion Image Variations". En dicho modelo, se pudo apreciar la utilidad de emplear enfoques de "imagen a imagen" para diversas tareas.

No obstante, aunque se pudo observar de primera mano esta interesante función de Stable Diffusion, en realidad no representa las innovaciones más significativas y las capacidades relevantes que ofrece el modelo de inteligencia artificial. De hecho, esta función ha estado disponible desde hace algún tiempo y no muestra las novedades más importantes que el modelo puede ofrecer. Para abordar esta situación, al momento de redactar este libro, se presenta una de las herramientas más recientes basadas en Stable Diffusion.

Esta herramienta refleja de manera natural algunas de las funciones más interesantes y novedosas que ofrece el modelo. Además, esto se logra sin la necesidad de instalar ningún programa o el modelo en sí en las computadoras, permitiendo a los usuarios disfrutar de estas importantes actualizaciones utilizando únicamente su navegador.

Concretamente, nos referimos a ControlNet, un modelo de inteligencia artificial basado en Stable Diffusion que aprovecha al máximo una de las innovaciones más importantes y relevantes que puede ofrecer Stable Diffusion en la actualidad. Esta innovación consiste en la capacidad de interpretar imágenes y, a partir de dicha interpretación, inferir con gran precisión la profundidad de las imágenes generadas, identificar qué elementos sobresalen más que otros en el plano, entre otros aspectos destacados.

Es posible que todo esto suene algo confuso; sin embargo, se examinará en profundidad todo lo relacionado con este tema, de modo que no queden dudas acerca del funcionamiento básico de este modelo de inteligencia artificial. Cabe destacar que ControlNet representa un avance muy importante y significativo dentro de este tipo de modelos y para la inteligencia artificial en general.

ControlNet: ¿Qué es?

Antes de abordar ControlNet y trabajar con este modelo de inteligencia artificial, es fundamental comprender qué es realmente este modelo y, específicamente, qué lo hace tan especial. Además, es crucial analizar qué relación tiene con Stable Diffusion y los modelos de inteligencia artificial que se han estudiado hasta el momento.

En términos simples, ControlNet es un modelo de inteligencia artificial que funciona como una extensión de Stable Diffusion. Es decir, no se trata del mismo modelo, sino de uno independiente que se basa en Stable Diffusion para generar sus resultados. Sin embargo, esto se lleva a cabo con una modificación particular en la generación que realiza este modelo.

Antes de explicar en detalle esta generación y todo el proceso que ocurre detrás de escena para obtener los resultados, es importante tener una visión general de ControlNet para comprender las posibilidades que ofrece este modelo de inteligencia artificial.

Para utilizar ControlNet, se puede acceder al espacio gratuito proporcionado por el sitio web "Stable Diffusion Online" (https://stablediffusionweb.com/), donde se encuentra el espacio dedicado al modelo ControlNet. En este caso, el sitio web se denomina "ControlNet – Control Diffusion Models". Generalmente, se puede encontrar el sitio web realizando una búsqueda en un motor de búsqueda, como Google, utilizando los términos "ControlNet Stable Diffusion". Es importante destacar que, después de la publicación de este libro, es probable que haya

surgido una variedad de sitios web que también ofrezcan la posibilidad de utilizar este modelo de inteligencia artificial. Por lo tanto, el sitio web específico que se utilice no es crucial, siempre que contenga el modelo en cuestión.

Cabe señalar que algunos modelos, como Stable Diffusion, están diseñados para ser utilizados por cualquier persona y, en consecuencia, se pueden encontrar en diversas variantes a lo largo de Internet y en diferentes sitios web.

A continuación, se presenta una captura de pantalla del sitio web mencionado:

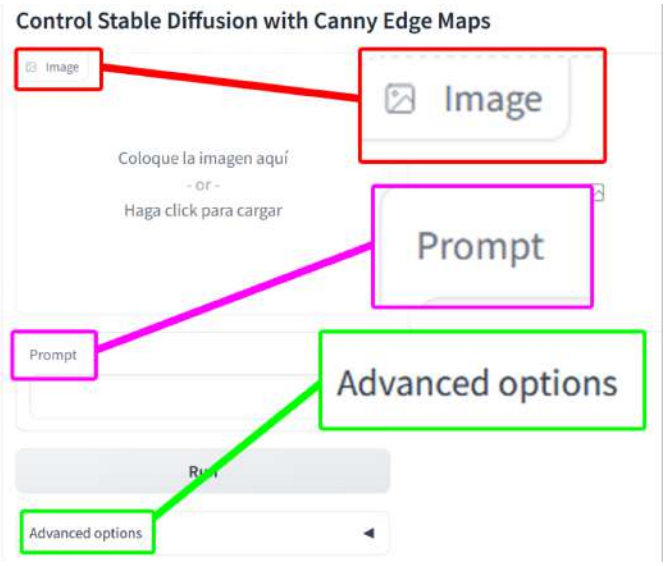
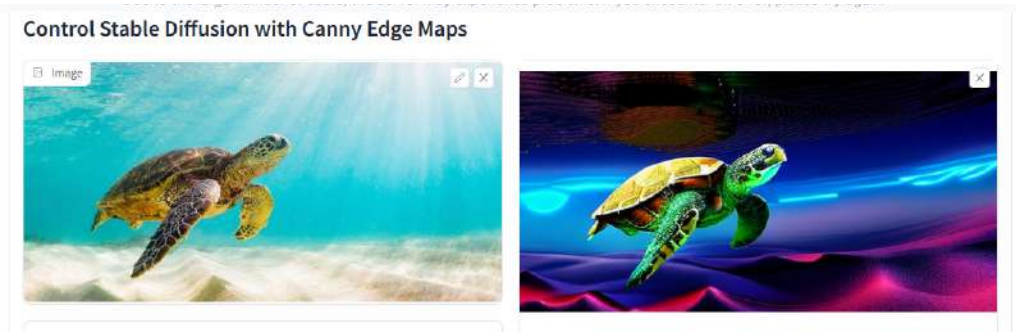


Ilustración 34 - Captura de pantalla de ControlNet

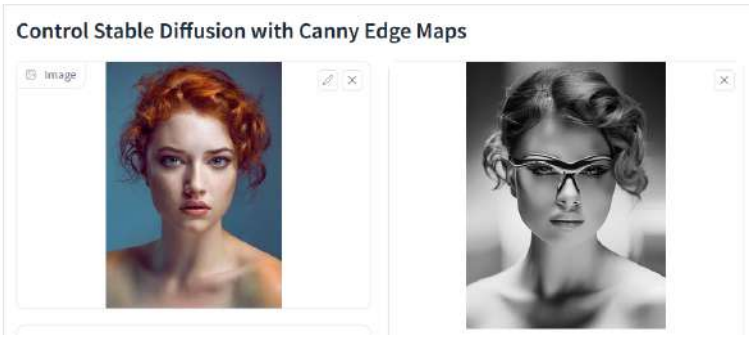
En la captura de pantalla, se pueden apreciar los tres elementos diferentes que se pueden utilizar al trabajar con ControlNet. En este caso, el modelo solicita una imagen, un prompt y, finalmente, permite modificar la sección de opciones avanzadas. Para comprender el funcionamiento de ControlNet, se comenzará buscando la imagen que se ingresará en el modelo. En este ejemplo, se utilizará una simple imagen de una tortuga, que corresponderá a la imagen que se proporcionará al modelo en la sección de imagen. Por otro lado, en la sección de "prompt", se solicitará al modelo que dibuje una "Tortuga animada en una fiesta techno", como se muestra a continuación:



Generación Stable Diffusion 29 – Imagen: Tortuga / Prompt: Animated turtle in a techno party

Como podemos observar, la función de ControlNet en este caso resulta bastante peculiar. La imagen obtenida muestra un gran parecido con la original proporcionada, pero presenta variaciones acordes a la descripción textual que indicamos.

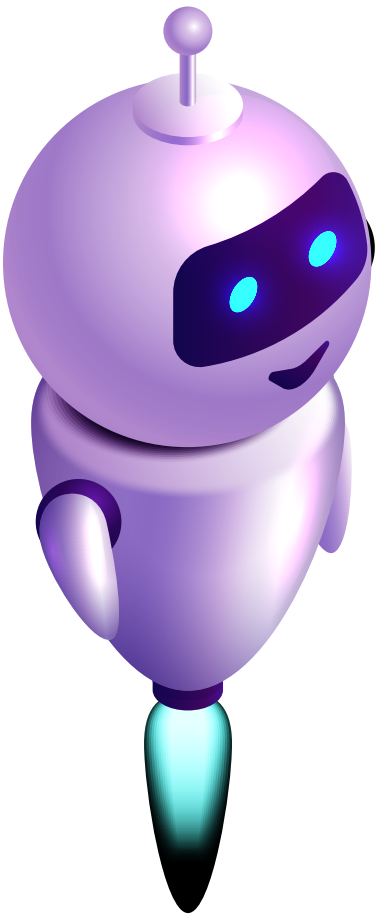
Es posible que la función de ControlNet en la generación de la imagen aún no esté del todo clara o pueda resultar confusa. Por ello, a continuación, presentaremos algunos ejemplos adicionales para facilitar su comprensión:



Generación Stable Diffusion 30 – Imagen: Retrato de una mujer / Prompt: Cool glasses

Generación Stable Diffusion 31 – Imagen: Dibujo de un ave / Prompt: A bird

Ahora, gracias a estos ejemplos, podemos obtener una comprensión más clara y visual del rol que desempeña ControlNet, y específicamente, cómo interactúa con las imágenes que utilicemos en el modelo. Este modelo es particularmente eficaz en la modificación de las imágenes que le proporcionemos, asegurando que las generaciones que cree sean lo más cercanas posible a las imágenes originales que le indiquemos. No obstante, aunque esta explicación pueda parecer bastante evidente, surge la pregunta: ¿Qué sucede realmente detrás de este modelo de inteligencia artificial y por qué es tan crucial para los avances y habilidades que Stable Diffusion puede ofrecer?



Conclusión del capítulo

Este capítulo ha presentado ControlNet como un modelo innovador y relevante dentro del campo de la inteligencia artificial y como una extensión natural de Stable Diffusion. A través de una explicación detallada y la presentación de ejemplos concretos, se ha ilustrado cómo ControlNet aprovecha las capacidades de Stable Diffusion para interpretar y modificar imágenes de manera precisa. La introducción de ControlNet en este libro refleja algunos de los avances más significativos que la inteligencia artificial puede ofrecer, demostrando no solo la flexibilidad de Stable Diffusion sino también las posibilidades en constante expansión en el dominio de la generación y manipulación de imágenes. Además, se ha proporcionado una visión práctica y accesible de cómo utilizar ControlNet, asegurando que los lectores puedan comprender y explorar esta herramienta por sí mismos.

CONTROLNET: ¿CÓMO FUNCIONA?

Objetivos del capítulo

El propósito de este capítulo es analizar y desglosar el funcionamiento interno de ControlNet y su interacción con el modelo de aprendizaje profundo Stable Diffusion en la generación de imágenes. El capítulo pretende explicar de manera detallada cómo ControlNet procesa las imágenes, interpretando diversos elementos y colaborando en la creación de imágenes coherentes y fieles a la original. Además, se enfocará en entender las aplicaciones y posibilidades que ofrece ControlNet en diferentes industrias como la ilustración, el desarrollo de videojuegos, la animación, entre otras, poniendo de relieve su potencial en la automatización de procesos.

Ahora que tenemos una idea más precisa de lo que ControlNet hace con nuestras imágenes, es momento de profundizar en su funcionamiento y cómo se relaciona directamente con otros modelos de inteligencia artificial, como, por ejemplo, Stable Diffusion.

Antes de poder explicar el funcionamiento de ControlNet, debemos comprender el flujo de procesos que Stable Diffusion lleva a cabo para generar imágenes. Podemos observar el flujo al que nos referimos a continuación en el siguiente gráfico:

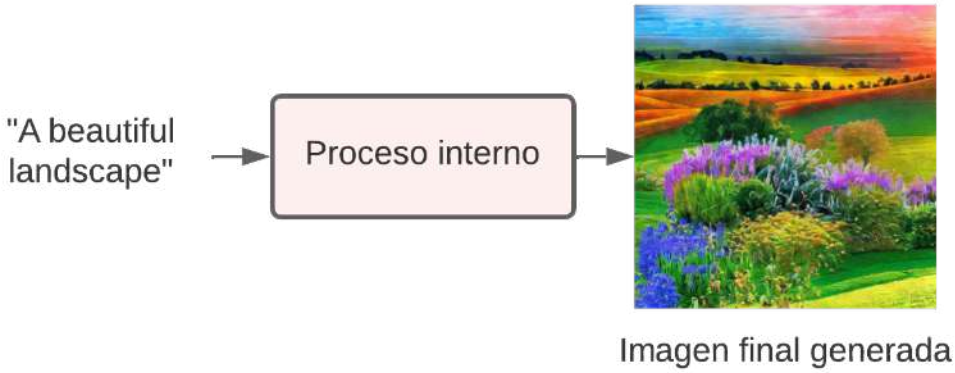


Ilustración 35 – Proceso de generación de Stable Diffusion

Es ampliamente conocido que el "proceso interno" representado en la imagen es en realidad mucho más complejo e involucra elementos como la técnica de generación de imágenes llamada Diffusion. No obstante, con el objetivo de facilitar una explicación más abstracta y sencilla de comprender, se ha resumido todo el proceso mostrado en la imagen bajo la etiqueta de "Proceso interno".

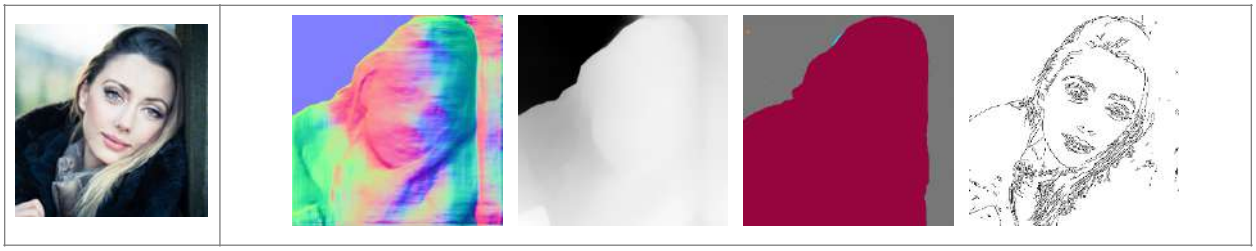
El rol de ControlNet en el proceso de generación de imágenes mediante Stable Diffusion se centra en la entrada del modelo y en su funcionamiento interno. En primer lugar, ControlNet recibe dos entradas esenciales: una imagen y un texto. La imagen actúa como guía para la generación que llevará a cabo Stable Diffusion, mientras que el texto representa el resultado que deseamos obtener. La imagen proporcionada al modelo de ControlNet orienta el proceso de generación de Stable Diffusion de tal manera que la imagen final creada sea coherente y similar a la imagen original suministrada.

Este proceso puede parecer confuso; no obstante, lo analizaremos detenidamente y paso a paso para comprender todo el desarrollo que se lleva a cabo.

Además, es importante destacar que los modelos de ControlNet presentan variantes para realizar diversas tareas, como identificar los bordes alrededor de una imagen, determinar la profundidad de una imagen tridimensional, entre otras. Profundizaremos en estas variantes del modelo y sus distintas funciones en secciones posteriores. Ahora bien, para comprender de manera más profunda el proceso que emplea ControlNet en la generación de imágenes, es necesario entender el procedimiento que se desarrolla dentro del modelo. De este modo, eventualmente podremos utilizar la información y lograr que la imagen generada se ajuste adecuadamente a la imagen original proporcionada al modelo.

Para comprender mejor este concepto, analicemos una imagen que ilustre visualmente el funcionamiento interno de ControlNet. Así, observaremos cómo el modelo identifica los componentes de la imagen. De hecho, al utilizar el modelo, podemos apreciar directamente su interpretación de los diversos elementos

presentes en la imagen, así como el proceso mediante el cual los interpreta. A continuación, presentamos algunas interpretaciones realizadas por el modelo en distintas imágenes.



Así, obtenemos una representación visual más clara de la interpretación del modelo ante las imágenes proporcionadas. Como se mencionó previamente, ControlNet cuenta con diversas variantes que cumplen distintos tipos de tareas. Concretamente, podemos apreciar que las interpretaciones realizadas por ControlNet corresponden a distintos aspectos, como un filtro que identifica elementos de la imagen según sus temperaturas de color, la profundidad tridimensional, las diversas secciones de la imagen, los bordes, los negativos y muchos otros elementos más.

Una vez que el modelo ha interpretado la imagen original, se encarga de asegurar que la salida generada corresponda y esté en concordancia con la imagen interpretada. De esta forma, el modelo crea una especie de "plano" a partir de dicha interpretación. Sobre este plano, el modelo de ControlNet procederá a dibujar la imagen final siguiendo el prompt indicado.

Este concepto puede sonar algo complejo, pero no es necesario entenderlo por completo. Con tener una idea general del funcionamiento de la interpretación de los elementos de la imagen en el modelo ControlNet, es suficiente para comprender cómo se generan imágenes con Stable Diffusion. Podemos observar el proceso que se lleva a cabo en la generación de imágenes en ControlNet y cómo el modelo se asegura de que las imágenes sean fieles a la original mediante la siguiente ilustración.

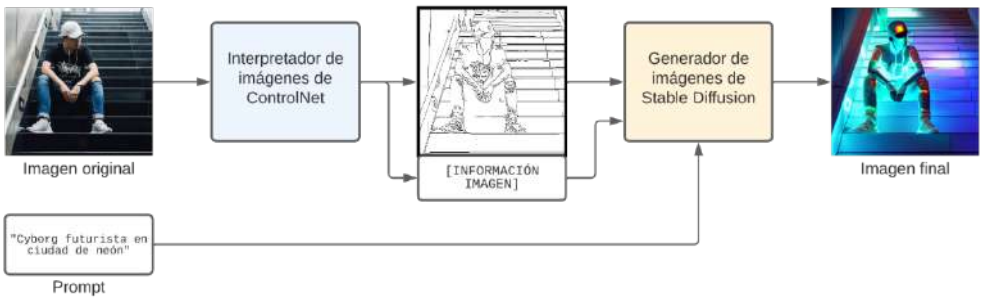


Ilustración 36 - Proceso y papel de generación de imágenes en ControlNet

En este gráfico podemos obtener una idea más clara del proceso general que sigue ControlNet en la creación de imágenes, así como comprender su relación con Stable Diffusion. Podemos observar que el proceso es similar al que habíamos mencionado previamente. Sin embargo, la imagen que se presentó anteriormente es una versión simplificada que no muestra la complejidad y el poder computacional requerido en todo el proceso. En realidad, la imagen original es analizada en detalle para cada uno de sus elementos, generando información que se utiliza como entrada para la generación de Stable Diffusion, junto con el prompt correspondiente. La interpretación de los diferentes elementos de la imagen se convierte en un molde para la creación de la imagen final.

ControlNet: ¿Qué se puede hacer con este modelo?

Ahora tenemos una idea más sólida acerca del funcionamiento general de ControlNet, así como de los diferentes pasos que sigue para generar imágenes a partir de una imagen original que sirve como molde y un prompt. Sin embargo, surge la pregunta importante: ¿De qué nos sirve todo esto y cómo podemos aprovechar las distintas posibilidades que ofrece ControlNet? Aunque las opciones son ilimitadas, en esta sección se brindarán algunas sugerencias e ideas sobre las distintas aplicaciones de ControlNet.

Como se mencionó previamente, Stable Diffusion tiene diversas aplicaciones en varios campos, como la industria del dibujo y la animación. ControlNet no es la excepción.

Veamos un ejemplo sencillo: supongamos que tenemos un boceto de un dibujo o simplemente una idea plasmada en un dibujo que necesitamos detallar y pintar. Sin embargo, este proceso suele ser bastante largo.

Una alternativa inteligente para mejorar la velocidad y eficiencia al añadir detalles a una imagen es minimizar el trabajo mediante el uso de modelos. Esto puede resultar muy beneficioso en industrias como la ilustración, el desarrollo de videojuegos, la animación y la creación de efectos especiales.

A continuación, mostraremos un ejemplo de cómo funciona este proceso. Supongamos que somos administradores de una compañía dedicada a crear arte digital para publicidad, diseños personalizados y más. Como administradores, nuestra tarea consiste en utilizar un modelo de inteligencia artificial con la imagen proporcionada por el cliente o el equipo de diseño artístico para crear una imagen que cumpla con las expectativas del usuario.

Recientemente, recibimos nuestro primer encargo, que es bastante sencillo: el cliente desea una ilustración de un personaje en una pose específica, enviada en una foto. Para completar este trabajo, utilizaremos ControlNet para identificar la pose y luego generaremos una descripción que cumpla con las expectativas del

cliente. Aunque pueda parecer confuso, ControlNet también nos permite interpretar poses de imágenes y generar diferentes resultados basados en la misma pose original.

A continuación, podemos ver un ejemplo visual de esta interpretación:

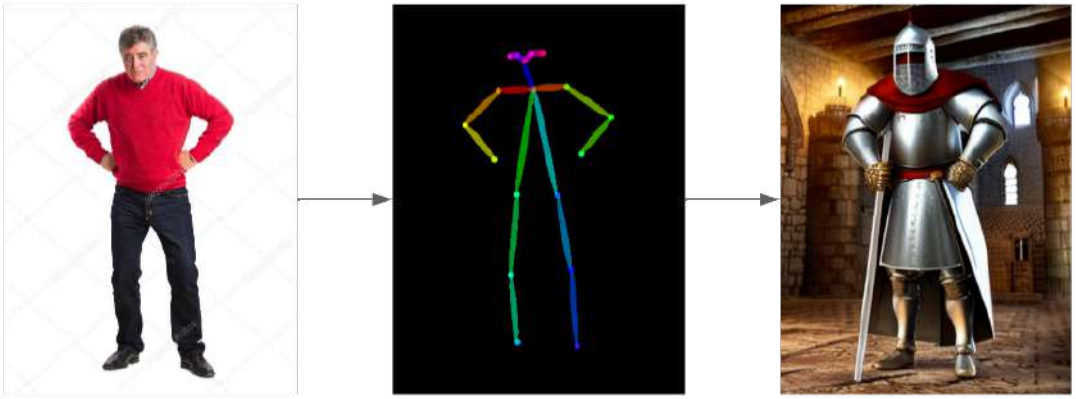


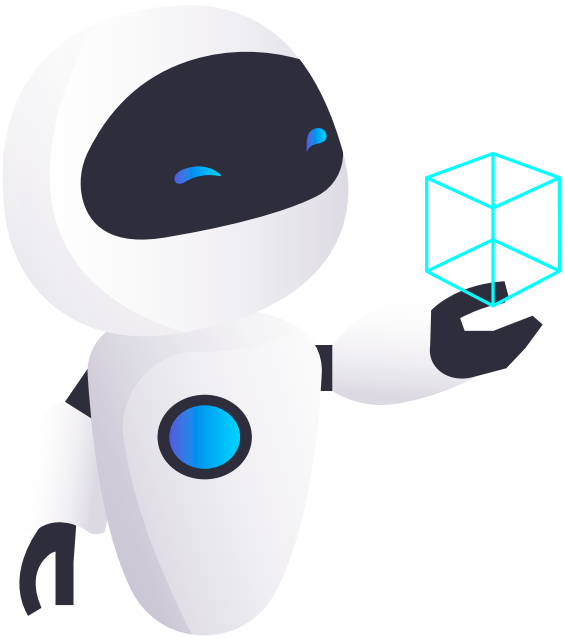
Ilustración 37 – Identificación de poses usando ControlNet

Se puede observar cómo ControlNet interpreta una imagen en busca de una pose humana, la cual será utilizada para generar una imagen basada en el prompt proporcionado. En la imagen mostrada, se aprecia la generación de un caballero medieval. Si esto corresponde a la solicitud del cliente mencionado, ya tendríamos una idea básica para la ilustración futura. Este proceso de identificación de imágenes, basado en solicitudes de usuario, resulta útil, por ejemplo, al proporcionar la imagen al equipo de producción para trabajar en ella.

Aunque la inteligencia artificial no siempre produce resultados totalmente precisos, podemos usar la imagen generada por el modelo como base para una ilustración. Este ejemplo, aunque parezca trivial, demuestra cómo ControlNet puede ayudar a controlar y mejorar el proceso de producción en industrias que adopten esta herramienta para agilizar su trabajo.

Es importante destacar que, más allá del impresionante avance en inteligencia artificial y sus capacidades, el uso de estas herramientas no se limita a simples "juguetes", sino que también pueden automatizar procesos complejos. Existen múltiples ejemplos sobre cómo utilizar ControlNet para diversas tareas. No obstante, la intención de este libro no es profundizar en un modelo en particular, sino ofrecer diferentes perspectivas de cómo estos modelos de inteligencia artificial pueden ser herramientas beneficiosas para las industrias y sus trabajadores. Como se ha observado en los diversos ejemplos presentados, el límite con este tipo de modelos de inteligencia artificial prácticamente no existe. La creatividad y la implementación cuidadosa de estos modelos pueden potenciar la automatización de numerosos procesos.

Conclusión del capítulo



En este capítulo, hemos explorado profundamente el funcionamiento de ControlNet, un componente esencial en el proceso de generación de imágenes mediante Stable Diffusion. Hemos descrito su papel en la interpretación de las imágenes originales, generando una especie de "plano" que orienta la creación de la imagen final. Se abordaron también las diversas variantes de ControlNet, mostrando su capacidad para identificar bordes, profundidad tridimensional y otras tareas relacionadas con la imagen. Mediante ejemplos prácticos, se ha demostrado cómo ControlNet puede aplicarse en diferentes campos como la industria del arte digital y la animación, destacando su potencial para agilizar y mejorar procesos. Este análisis detallado brinda una comprensión sólida de cómo la combinación de ControlNet con Stable Diffusion abre nuevas puertas en la generación y manipulación de imágenes, un avance tecnológico con aplicaciones prácticamente ilimitadas.

CAPÍTULO 6: VERSIONES DE STABLE DIFFUSION

Objetivos del capítulo

El presente capítulo tiene como objetivo principal profundizar en los aspectos técnicos y diversas versiones de Stable Diffusion, destacando la evolución y contribuciones a lo largo del tiempo. Se busca brindar una comprensión clara de las diferencias clave entre las versiones, incluyendo elementos como los Diffusers y su impacto en la implementación del modelo. También se enfocará en el contexto histórico, comparando Stable Diffusion con modelos previos, y destacando el proceso de investigación y desarrollo. Se pretende que el lector adquiera una comprensión profunda de cómo los avances en Stable Diffusion han afectado la generación de imágenes y la comunidad científica en general.

Ahora es el momento de sumergirnos en los aspectos más técnicos de Stable Diffusion. Aunque a simple vista no lo parezca, Stable Diffusion ha existido por un tiempo considerable. A pesar de que este período pueda considerarse corto, en el mundo de la inteligencia artificial, las cosas avanzan a un ritmo vertiginoso. En tan solo unos meses, algunos modelos pueden mejorar significativamente, y surgen nuevos modelos que revolucionan aún más el campo de la inteligencia artificial, dejando al mundo asombrado.

El propósito de esta sección es explorar las diferentes versiones de Stable Diffusion que han sido lanzadas hasta la fecha y detallar las nuevas características que estos modelos ofrecen en comparación con sus versiones anteriores. El objetivo es comprender mejor el progreso de la IA y los cambios significativos que se han producido con el tiempo en varios aspectos, los cuales tienen un impacto cada vez más positivo en cómo la inteligencia artificial está transformando el mundo y la vida de las personas.

Por ahora, puede parecer algo trivial entender por qué las versiones de un mismo modelo de inteligencia artificial pueden generar resultados distintos. Sin embargo, todo esto adquirirá mucho más sentido cuando realmente comprendamos las diferencias principales que existen entre las diferentes versiones de este modelo de inteligencia artificial. Después de todo, dependiendo de la versión del modelo, la forma en la que el modelo haya sido entrenado puede variar, junto con otras características, como por ejemplo el poder computacional que consume el modelo y, en consecuencia, los resultados mismos que está generando.

Con este conocimiento, ahora tendremos una mejor oportunidad de entender las distintas versiones que existen de este modelo y comprender las variaciones de cada uno de ellos. De esta manera, podremos apreciar el impacto de estos avances en la aplicación práctica de la inteligencia artificial en diversos sectores, así como reconocer la importancia de seguir investigando y desarrollando nuevas versiones y mejoras que puedan aportar aún más valor al campo de la IA y a la sociedad en general.

Primeras apariciones de Stable Diffusion

Es importante aclarar que el estudio e investigación de un modelo de inteligencia artificial tan impactante y relevante en el mundo como lo es Stable Diffusion, requiere de un gran proceso de investigación que generalmente lleva mucho más tiempo del que podría parecer antes de su lanzamiento. Aunque se mencionarán algunas fechas y tiempos guía en estas secciones para entender mejor el contexto temporal del avance de este modelo, estas fechas no son precisas, ya que como se mencionó, el proceso de investigación y desarrollo previo puede llevar meses o incluso años.

Antes de que Stable Diffusion se popularizara en Internet, existían otros modelos de inteligencia artificial que realizaban tareas similares. Aunque Stable Diffusion es uno de los más populares, estos modelos aún se utilizan actualmente y pueden ofrecer resultados de alta calidad dependiendo de la tarea en cuestión.

Uno de los modelos que ganó popularidad en su lanzamiento y que se desarrolló antes de Stable Diffusion es el modelo DALL-E y DALL-E 2, desarrollado por OpenAI. Estos modelos ofrecían un enfoque similar al de Stable Diffusion en la actualidad. Sin embargo, había una limitación significativa en la naturaleza de

pago de estos modelos, lo que es una ventaja para Stable Diffusion, ya que es un modelo open-source completamente gratuito.

El grupo de investigación CompVis (Computer Vision and Learning LMU Munich) publicó un artículo titulado "Síntesis de imágenes de alta resolución con modelos de difusión latente" (High-Resolution Image Synthesis with Latent Diffusion Models) (Rombach, Blattmann, Lorenz, Esser, & Ommer, 2021), en el cual se menciona una de las primeras y más importantes aplicaciones de Stable Diffusion.

En este artículo, publicado en agosto de 2022, se presenta una de las primeras versiones de Stable Diffusion hasta ese momento. La idea conceptual de Stable Diffusion era la misma, ofreciendo diferentes funciones que podían ser realizadas con este modelo de inteligencia artificial. Además, se compararon los resultados generados por este modelo con otros modelos, como DALL-E y VQGAN. El artículo presenta diferentes elementos y resultados de un proceso de investigación innovador y riguroso que se llevó a cabo dentro del modelo.

En este artículo se detallan no solo las funciones principales que ofrece Stable Diffusion actualmente, como la generación de "texto a imagen", sino también varias otras ventajas y funcionalidades que proporciona el modelo. Además, se destaca que los resultados obtenidos mediante este modelo son de alta calidad y sorprendentes, siendo un modelo de inteligencia artificial de código abierto.

También se ofrece una breve descripción de las diversas bases de datos y elementos utilizados en el proceso de desarrollo. Por ahora, puede parecer algo trivial, sin embargo, realmente es demasiado importante para la comunidad científica entender desde un punto de vista más interno y profundo el funcionamiento de este tipo de modelos de inteligencia artificial. Después de todo, estamos hablando de que este modelo de inteligencia artificial ha dado la vuelta al mundo, especialmente gracias a sus increíbles resultados y generaciones de imágenes.

Además, tal y como hemos mencionado anteriormente, Stable Diffusion también tiene la enorme ventaja de ser un modelo de inteligencia artificial completamente de código abierto. Es decir, realmente cualquier persona tiene la capacidad de poder usar este modelo de inteligencia artificial y, además, todo el proceso y la información, como las bases de datos utilizadas en el proceso de desarrollo de este modelo, siguen siendo completamente libres. Esto permite que realmente toda la comunidad científica conozca el proceso entero que se llevó a cabo en el desarrollo de este modelo de inteligencia artificial. Además, en base a esto, se permiten diferentes aplicaciones y modelos de inteligencia artificial, e incluso mejoras constantes por parte de la comunidad, que se aplican al modelo y generan resultados positivos para el avance del proyecto del modelo.

Diffusers y su papel en Stable Diffusion

Ahora, vamos a hablar sobre otro elemento clave en el proceso de desarrollo de Stable Diffusion y las distintas versiones que han surgido a lo largo del tiempo. Aunque no se trata de una versión en sí misma, es importante mencionarla como preámbulo para comprender mejor qué es Stable Diffusion y el impacto que tiene en la generación de resultados según el tipo de modelo. Es fundamental destacar que algunas versiones de Stable Diffusion cuentan con dos variantes: una con los Diffusers incluidos y otra sin ellos, lo cual afecta significativamente el modelo. La mayoría de las versiones de Stable Diffusion que se verán en este proceso aplican los Diffusers para mejorar la calidad de la generación. Sin embargo, es posible que ocasionalmente realicemos una breve comparación entre las versiones con y sin esta herramienta.

En términos sencillos, Diffusers es una biblioteca desarrollada por Hugging Face. La importancia de esta biblioteca radica en que permite el uso de diferentes modelos de difusión de generación de manera más fácil y a través de diferentes tipos de tuberías (pipelines). Lo más destacado de Diffusers es que no sólo permite el proceso de entrenamiento de modelos de difusión estable, sino también de otros tipos de modelos, como ControlNet.

Aún podemos acceder a la información de los modelos originales de Stable Diffusion que no utilizan Diffusers. Sin embargo, esta práctica se ve principalmente en las versiones más antiguas de Stable Diffusion. Realmente no hay una diferencia significativa que debamos destacar al comparar las versiones con la librería Diffusers y las versiones originales de Stable Diffusion. Simplemente debemos entender que el uso de la librería de Diffusers puede facilitar el proceso de uso de diferentes modelos de generación basados en difusión.

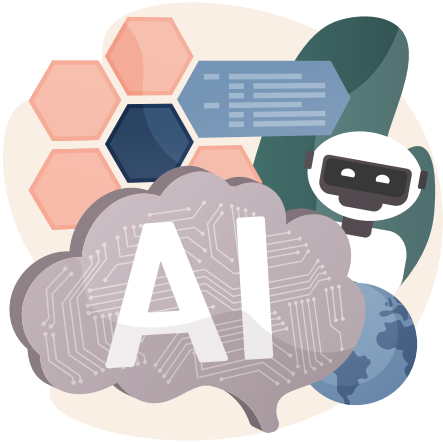
Ahora, puede parecer algo complejo, pero veremos que todo esto tiene mucho sentido cuando estemos haciendo ejemplos directos de cómo podemos usar Stable Diffusion de forma local en nuestra computadora. Allí, tendremos la oportunidad de ver que realmente el proceso de usar este tipo de modelos de inteligencia artificial desde cero, donde tendremos que hacer diferentes pasos que pueden llevar tiempo, es realmente tedioso. Algunos pasos, como por ejemplo, tener que descargar los pesos del modelo de forma manual. Sin embargo, podemos decir que no tenemos que preocuparnos del todo por este tipo de cosas en el caso de que estemos deseando usar los modelos que vienen dentro de los Diffusers, pues realmente lo único por lo que tendremos que preocuparnos será en tener una cuenta en la página web de Hugging Face.

Antes de que Stable Diffusion se convirtiera en un modelo realmente popular y con tanto apoyo económico, podemos decir que el desarrollo de este modelo de inteligencia artificial y su uso consistía meramente en el método clásico, es decir, la instalación de las diferentes versiones del modelo de inteligencia artificial manualmente. Sin embargo, las versiones más recientes del modelo de

inteligencia artificial no ofrecen la opción de instalarlos y usarlos usando la forma antigua de hacerlo, sino que realmente la única forma de poder usarlos es por medio de los Diffusers. Puede parecer algo un poco centralizado, pero realmente no es así, pues el propósito principal de esto es permitir que las personas tengan acceso libre a este modelo de una forma completamente fácil de usar, ahorrando bastante tiempo.

Además del ejemplo que presentaremos, en el cual veremos cómo podemos utilizar este modelo de inteligencia artificial en nuestra computadora de forma libre y desde cero, empleando el método antiguo no relacionado con los Diffusers, también tendremos la oportunidad de aprender a utilizar la Stable Diffusion mediante el uso de los Diffusers.

De esta manera, comprenderemos por qué son tan importantes y cómo agilizan significativamente el proceso para emplear este modelo de inteligencia artificial. Así, podremos tener una idea clara y completa de la importancia de los Diffusers dentro de la Stable Diffusion como modelo de inteligencia artificial.



Conclusión del capítulo

Este capítulo ha permitido explorar a fondo las distintas versiones de Stable Diffusion, iluminando la evolución y características únicas que definen cada versión. Se ha mostrado cómo los cambios en el modelo, incluyendo el uso de Diffusers, han afectado la calidad y facilidad de uso en la generación de imágenes. La comparación con modelos anteriores y la inclusión de aspectos como el carácter open-source de Stable Diffusion, han enriquecido el entendimiento sobre su importancia y posición en el mundo de la inteligencia artificial. La sección también ha puesto de manifiesto el rápido ritmo de cambio en el campo de la IA y la necesidad de comprender estos desarrollos para aprovechar plenamente sus aplicaciones y contribuir a futuras innovaciones.

STABLE DIFFUSION V1.1

Objetivos del capítulo

En este capítulo, se busca realizar un análisis exhaustivo y comparativo de las diferentes versiones de Stable Diffusion, desde la versión 1.1 hasta la versión 2.1. Se enfatizará en las características, funcionalidades y críticas de cada versión, identificando tanto los avances como las deficiencias presentes en cada iteración del modelo. A través de este análisis, los lectores obtendrán una perspectiva clara de la evolución de Stable Diffusion, su impacto en la generación de imágenes, y su papel en el campo de la inteligencia artificial. Además, se presentarán ejemplos visuales y técnicas específicas que han surgido con las sucesivas actualizaciones.

Vamos a abordar las distintas versiones de Stable Diffusion y las funcionalidades que ofrecen. Comenzaremos con la versión 1.1 de este modelo de inteligencia artificial, que se encuentra en la página oficial del grupo de investigación CompVis en Hugging Face. A pesar de que el artículo de investigación sobre Stable Diffusion se publicó en agosto, la versión a la que nos referimos aquí se lanzó en junio de 2022. Esto nos proporciona una perspectiva más amplia de lo que mencionamos anteriormente.

Al momento de escribir este texto, aún no ha pasado un año desde su lanzamiento y, sin embargo, este modelo de inteligencia artificial ha generado mucho interés, lo que demuestra el rápido avance en el campo de la IA. En

cuanto a las funciones y características del modelo, es importante destacar su capacidad para generar imágenes a partir de las primeras versiones de Stable Diffusion.

En el momento de su lanzamiento, este modelo ofrecía dos opciones de generación de imágenes con diferentes resoluciones. Sin entrar en detalles técnicos, la primera opción generaba imágenes de alta calidad con una resolución de 256 x 256 píxeles, mientras que la segunda generaba imágenes de 512 x 512 píxeles, aunque eran de buena calidad, posiblemente no superaban a las de 256 x 256 píxeles.

Es esencial destacar que esta versión de Stable Diffusion representa una de las primeras iteraciones del modelo. A medida que el modelo avanzó con el tiempo, las imágenes generadas mejoraron en calidad y ofrecieron un panorama cada vez más prometedor en el mundo de la IA. Esto no implica que esta versión fuera obsoleta o insuficiente, sino que, naturalmente, las versiones posteriores del modelo fueron mejorando progresivamente.

Stable Diffusion v1.5

Actualmente existen varias versiones de Stable Diffusion. La versión previamente presentada fue la v1.1, sin embargo, en este momento nos enfocaremos en la versión v1.5. ¿Por qué? ¿No deberíamos considerar otras versiones como la v1.3? En realidad, sí existen estas versiones, pero el motivo principal por el que estamos avanzando directamente a la v1.5 es porque si dedicáramos un capítulo para cada versión de Stable Diffusion, el libro ocuparía más espacio del necesario. Recordemos que el objetivo de este libro es aprender sobre Stable Diffusion y cómo se puede aplicar este modelo de inteligencia artificial para diversas tareas de generación de imágenes.

Esta versión presenta mejoras significativas en comparación con la versión anterior que revisamos, incluyendo una mayor optimización. Aunque no hay grandes diferencias en los resultados generados por este modelo de inteligencia artificial, es importante destacar que las versiones posteriores a la v1.2 se basan principalmente en esta versión, aunque con algunas nuevas mejoras.

Otro aspecto importante de esta versión es que, al igual que la v1.4, es una de las versiones más populares y utilizadas de Stable Diffusion en diferentes modelos de inteligencia artificial. Incluso algunas versiones de ControlNet también están basadas en Stable Diffusion. Por alguna razón, estas dos versiones del modelo son las preferidas para el desarrollo de modelos de inteligencia artificial basados en Stable Diffusion. En este libro, hemos utilizado algunos ejemplos que emplean imágenes generadas por los modelos de la versión 1.4 y 1.5. A partir de esto, podemos concluir que los resultados obtenidos por ambas versiones son de alta calidad.

Stable Diffusion v2.0

Ahora es el momento de dar un paso importante en relación con las versiones de Stable Diffusion. En este caso, exploraremos y veremos las diferencias entre esta nueva versión y las anteriores. Es importante aclarar que esta versión no es la más reciente, ya que, al momento de escribir este libro, la última versión de Stable Diffusion es la v2.1. Sin embargo, es importante estudiar a fondo esta nueva versión, ya que ha habido muchos cambios en comparación con las versiones anteriores. Algunos cambios han sido bien recibidos, mientras que otros han sido criticados, pero analizaremos todos ellos en esta sección.

Algunas de las críticas más importantes que se hicieron en contra de este modelo de inteligencia artificial son, por ejemplo, que aparentemente esta nueva versión de Stable Diffusion ofrecía un rendimiento mucho peor en comparación con las imágenes generadas por el modelo. Es decir, hubo bastantes críticas debido a que las imágenes generadas por el modelo eran de una calidad considerablemente inferior a las que solían generar los modelos anteriores de Stable Diffusion.

Sin embargo, a pesar de estas críticas, no necesariamente significa que los avances de este modelo de inteligencia artificial fueran negativos. Todo lo contrario, este modelo también ofreció bastantes mejoras en otros aspectos muy importantes que tenían como falencias las versiones anteriores de Stable Diffusion en las versiones v1.

Si hablamos de las mejoras que ofrece este modelo de inteligencia artificial en comparación con sus versiones anteriores, podemos mencionar que el proceso de generación de imágenes se ha mejorado al cambiar la base de datos principal a una llamada "LAION-5B". Esta base de datos es enormemente grande y contiene una gran cantidad de imágenes, lo que permite al modelo entrenarse con ellas y generar imágenes a partir de los prompts indicados. Además, esta mejora, junto con otras novedades importantes del modelo, permite tener un modelo entrenado mucho más robusto.

De hecho, en la misma página de StabilityAI podemos encontrar imágenes que fueron generadas usando este modelo de inteligencia artificial, y podemos ver también que estas mismas imágenes generadas corresponden a imágenes de muy alta calidad, a continuación, podemos ver dos imágenes que se encuentran en el sitio web StabilityAI que fueron generadas usando este nuevo modelo de difusión de texto a imagen.



Generación Stable Diffusion 32 – Ejemplos de imágenes generadas usando Stable Diffusion v2.0 (StabilityAI, 2022).

Una de las ventajas y novedades que ofrece este modelo de inteligencia artificial, en comparación con sus versiones anteriores, es una herramienta nueva llamada "Modelos de difusión de aumento de resolución de imágenes" (Super-resolution Upscaler Diffusion Models en inglés) que ofrece Stable Diffusion. Como su nombre indica, esta herramienta permite a los modelos mejorar la relación y el nivel de detalle de las imágenes al recibir una imagen como entrada.

Es sabido por muchas personas que existían muchas herramientas que realizaban la misma tarea antes del lanzamiento de Stable Diffusion v2.0, utilizando modelos de inteligencia artificial y otras técnicas. Sin embargo, lo que hace que Stable Diffusion se destaque en esta tarea es el nivel de detalle increíblemente alto que ofrece, y su capacidad para aumentar la resolución de las imágenes desde una calidad muy baja. Por ejemplo, en el anuncio oficial de este modelo de inteligencia artificial, se muestra cómo se toma una imagen pequeña de 128 x 128 píxeles y se aumenta a una imagen precisa de 512 x 512 píxeles. Aunque las capacidades de Stable Diffusion son mucho más altas, permitiendo incluso aumentar la resolución de las imágenes hasta 2048 x 2048 píxeles o más.

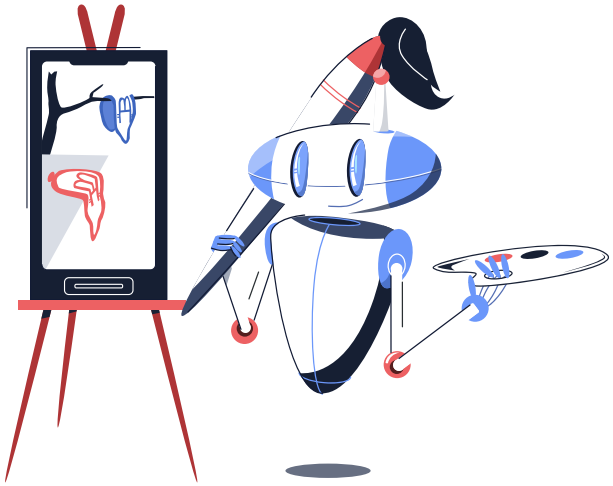
Este modelo ha sido actualizado con numerosas mejoras, como una mayor precisión en la detección de profundidad de imágenes y su generación, así como mejoras en el proceso de pintura mediante Stable Diffusion, entre otras novedades. Sin embargo, a pesar de estas mejoras, gran parte de la comunidad ha expresado su descontento con esta actualización, ya que aparentemente el rendimiento de esta nueva versión es inferior al de las versiones anteriores del modelo de inteligencia artificial.

Stable Diffusion v2.1

Hablaremos ahora de una de las versiones más reciente de este modelo de inteligencia artificial. Como se ha mencionado en distintas secciones de este libro, la investigación y el desarrollo en el campo de la inteligencia artificial avanzan a un ritmo vertiginoso. Es probable que, al momento de leer este libro, ya exista una versión más actualizada del modelo. Por ello, es importante estar pendiente de

las fuentes principales y conocer las diferentes versiones que van surgiendo de Stable Diffusion, así como las novedades que estas últimas ofrecen.

En la sección donde se explican las novedades que trajo la versión v2.0 del modelo, se mencionó que varios usuarios afirmaban que esta versión ofrecía un rendimiento y resultados inferiores en comparación con las versiones anteriores. Sin embargo, la intención principal de Stable Diffusion v2.0 era ofrecer una versión base con importantes innovaciones para mejorar el rendimiento y los resultados en versiones posteriores. Con la llegada de Stable Diffusion v2.1, se solucionaron la mayoría de las quejas relacionadas con el rendimiento y los resultados generados por la versión anterior. De esta manera, los resultados y las imágenes generadas con Stable Diffusion v2.1 superaron las expectativas de muchos usuarios, proporcionando un modelo sorprendente que marcó una gran diferencia en comparación con la versión anterior. Así, se logró ofrecer imágenes y resultados de mayor calidad.



Conclusión del capítulo

El capítulo ha brindado una exploración completa de las versiones sucesivas de Stable Diffusion, destacando la constante innovación y adaptación que caracteriza este campo. A partir de la versión 1.1, se observó cómo las iteraciones subsiguientes del modelo han trabajado para mejorar la resolución, calidad y eficiencia en la generación de imágenes. A pesar de las críticas y desafíos enfrentados, como la disminución de la calidad en la versión 2.0, la constante evolución y mejora se reflejaron en la versión 2.1. Los desarrollos tecnológicos detallados, como la herramienta de aumento de resolución de imágenes, ejemplifican las posibilidades y el potencial que Stable Diffusion continúa ofreciendo. Este análisis enfatiza la importancia de mantenerse actualizado en un campo en constante cambio y destaca la contribución significativa de Stable Diffusion en el mundo de la IA y la generación de imágenes.

STABLE DIFFUSION XL

Objetivos del capítulo

Este capítulo tiene el propósito de presentar y analizar Stable Diffusion XL, la versión más reciente y avanzada de Stable Diffusion en el momento de redacción del libro. Se pretende brindar una explicación detallada de las innovaciones y mejoras que este modelo ofrece en comparación con su predecesor, Stable Diffusion 2.1. A través de una descripción técnica y ejemplos visuales, se busca ofrecer una perspectiva clara de las capacidades y aplicaciones de este nuevo modelo, enfocándose en su mayor fotorrealismo y facilidad de acceso. Además, el capítulo busca destacar la relevancia de esta innovación para el avance general de la tecnología de inteligencia artificial, y cómo su naturaleza gratuita y de código abierto juega un papel crucial en su impacto en la comunidad científica y tecnológica.

Actualmente, al escribir este libro, se ha lanzado un nuevo modelo de Stable Diffusion, el cual representa una innovación considerable. El rendimiento de esta nueva versión del modelo implica un progreso significativo y relevante en el desarrollo de Stable Diffusion. En concreto, nos referimos a Stable Diffusion XL, una versión mejorada de este modelo de inteligencia artificial que ofrece un avance realmente destacado e importante en comparación con su predecesor, Stable Diffusion 2.1.

Al momento de redactar este libro, el modelo es bastante reciente, pero una cosa es segura: ofrece resultados simplemente asombrosos. Para utilizar este modelo de inteligencia artificial, basta con visitar la página oficial de Stable Diffusion en StabilityAI. Allí, encontraremos la herramienta Dream Studio, donde podremos aprovechar este innovador y sorprendente modelo. Es importante señalar que, debido a la novedad del modelo, algunos aspectos pueden cambiar en el futuro.

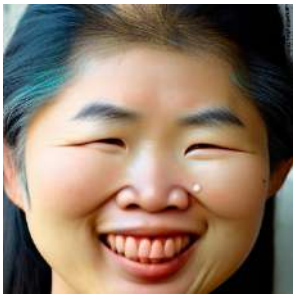
No obstante, hasta la fecha, esta es la forma más sencilla de acceder y utilizar este modelo en nuestro navegador web.

Una imagen vale más que mil palabras: Stable Diffusion XL

Debido a que este modelo representa la versión más reciente de Stable Diffusion publicada, es momento de analizar algunas de las novedades que ofrece. Para este ejemplo, utilizaremos la versión del modelo disponible en DreamStudio, una herramienta diseñada para trabajar con diversos modelos de Stable Diffusion desarrollados por StabilityAI. Esta plataforma permite experimentar con diferentes modelos y versiones de Stable Diffusion.

Particularmente, en esta sección veremos algunas comparaciones de los resultados que ofrece esta nueva versión del modelo en relación con la versión de Stable Diffusion 2.1, y de esta manera, apreciaremos los avances significativos e importantes que representa este modelo. También tendremos la oportunidad de mencionar algunas especificaciones técnicas del modelo, lo cual nos permitirá entender mejor lo que ocurre detrás del telón y tener una idea más precisa y clara de los diferentes aspectos que han cambiado para mejorar y hacer más eficiente este modelo.

A continuación, se presentan algunas imágenes que muestran una comparación del rendimiento entre el nuevo modelo Stable Diffusion y su versión anterior, el 2.1. De esta manera, podemos obtener una idea más clara de lo que ocurre dentro del modelo y los mejores resultados que ofrece. En la parte superior de las imágenes, se muestra la versión generada por Stable Diffusion 2.1, mientras que, en la parte inferior, se encuentra la versión generada por Stable Diffusion XL. Ambas imágenes se generaron utilizando el mismo prompt y los mismos parámetros del modelo.

Stable Diffusion 2.1			
			
Stable Diffusion XL			

Generación Stable Diffusion 33 – Upclose portrait of an asian woman (Portrato de cerca de una mujer asiática)

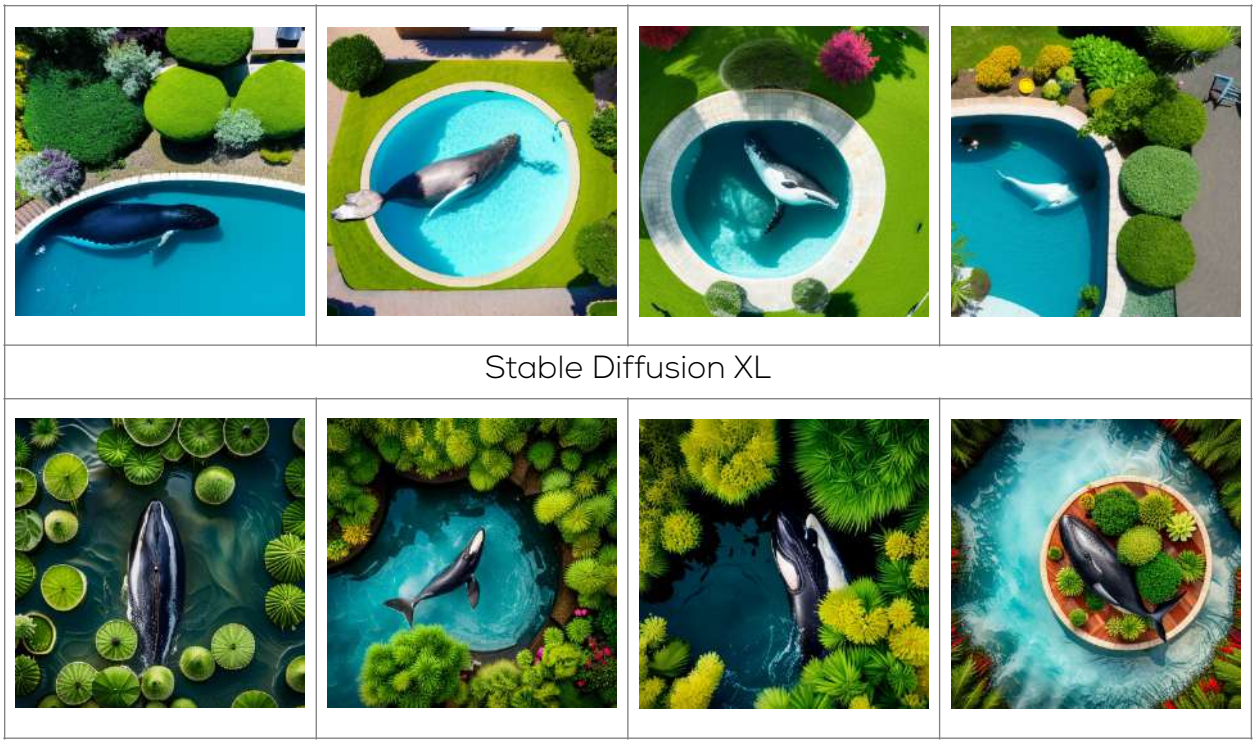
Este es un ejemplo que ilustra y demuestra claramente lo que estábamos diciendo anteriormente. Sin embargo, en realidad podemos encontrar muchos más ejemplos de lo mencionado, donde podemos observar que esta nueva versión del modelo ofrece resultados mucho mejores.

Stable Diffusion 2.1

Stable Diffusion XL

Generación Stable Diffusion 34 – Nice neon city landscape (Bonito paisaje de ciudad de neón)

Stable Diffusion 2.1



Generación Stable Diffusion 35 - Top view of a whale swimming in a garden pool (Vista superior de una ballena nadando en una piscina de jardín)

Podemos observar que las generaciones e imágenes nuevas generadas usando Stable Diffusion XL representan un enorme avance en comparación con la versión anterior del modelo, Stable Diffusion v2.1. Esto queda bastante claro al observar las imágenes mostradas arriba. Tal como mencionamos anteriormente, podemos hacer una breve comparación entre las imágenes generadas por ambos modelos. Es importante destacar que las imágenes fueron creadas usando DreamStudio, y las generaciones relacionadas emplean las mismas configuraciones y valores de parámetros. El único parámetro que se modificó corresponde a la versión del modelo.

Desde el lanzamiento de esta nueva versión, podemos afirmar que ahora es posible comparar el rendimiento del nuevo modelo (que algunos consideran como una versión 2.2) con otros modelos que generan resultados mucho más realistas y profesionales. Sin embargo, la gran diferencia radica en que, a diferencia de esos otros modelos que suelen ser de pago, Stable Diffusion es gratuito. Esto posiciona a Stable Diffusion como uno de los mejores, o potencialmente el mejor, modelo de generación de imágenes basado en la técnica de difusión.

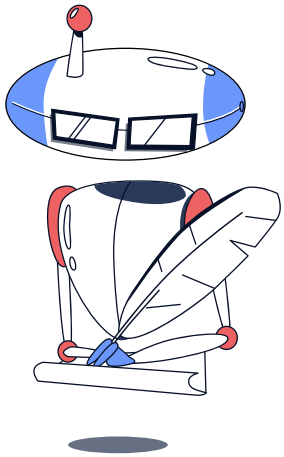
Este avance es significativo para la tecnología de inteligencia artificial en general, ya que es crucial destacar nuevamente la importancia de que estos modelos sean de código abierto. Desde una perspectiva técnica, la nueva versión de Stable Diffusion brinda una capacidad de generar fotorrealismo significativamente más avanzada y sorprendente en comparación con sus

modelos anteriores. Esta mejora es particularmente notable en la composición y generación de imágenes que utilizan rostros. Además, se pueden emplear prompts más breves para obtener resultados superiores. En general, también proporciona la capacidad de generar texto más legible, lo cual representa un gran avance por razones que detallaremos más adelante.

El progreso en modelos de inteligencia artificial como Stable Diffusion, que poseen gran potencial y capacidades avanzadas, representa uno de los pilares fundamentales en la generación de imágenes mediante inteligencia artificial.

Además, este avance es crucial para la ciencia, la investigación y el desarrollo humano. Profundizar en nuestra comprensión y exploración de estos novedosos y sorprendentes modelos de inteligencia artificial nos permite entender de manera más detallada el funcionamiento interno que ocurre en el proceso, lo cual es vital frente a los últimos y más relevantes avances en el campo del estudio de la inteligencia artificial. Comprender cómo funcionan estos modelos detrás de escena se convierte en un aspecto crucial a tener en cuenta.

Conclusión del capítulo



Stable Diffusion XL marca un hito importante en la evolución de la tecnología de generación de imágenes por difusión, presentando avances significativos en cuanto a realismo y versatilidad en comparación con las versiones anteriores. La presentación de comparaciones visuales entre la versión XL y la 2.1 permite apreciar claramente las mejoras y la calidad superior de las imágenes generadas por el nuevo modelo. La novedad de Stable Diffusion XL, junto con su disponibilidad gratuita a través de DreamStudio, establece una posición única y prometedora en el campo de la inteligencia artificial. Al profundizar en las especificaciones técnicas y aplicaciones prácticas, este capítulo subraya el papel fundamental que desempeñan estos modelos en la investigación y desarrollo humano, y cómo su continua innovación sigue impulsando los límites de lo que es posible en el mundo de la generación de imágenes basada en técnicas de difusión.

LIMITACIONES DE STABLE DIFFUSION: ANTES DE XL

Objetivos del capítulo

En este capítulo, nos proponemos analizar y comprender las limitaciones principales de los modelos de Stable Diffusion antes de la versión XL. Con el objetivo de entender las áreas de oportunidad y los desafíos, se abordará con detenimiento la problemática en la generación de imágenes de manos y textos, dos limitaciones significativas que caracterizan a estos modelos. Además, se buscará identificar los aspectos técnicos y las razones detrás de estas limitaciones, proporcionando ejemplos claros y gráficos para una mejor comprensión. Por último, el capítulo aspira a presentar un primer vistazo a los avances en limitaciones en la versión Stable Diffusion XL y su impacto en la generación de imágenes con texto.

Es importante mencionar que, a pesar de los sorprendentes resultados que estos modelos de inteligencia artificial han logrado, captando la atención mundial, también presentan ciertas limitaciones. Aunque en ocasiones generan buenos resultados y son útiles para diversas tareas, exhiben fallas en actividades que podrían parecer sencillas en un principio. De este modo, se evidencia que, por muy eficientes que sean en la generación de imágenes, aún tienen áreas de oportunidad para mejorar.

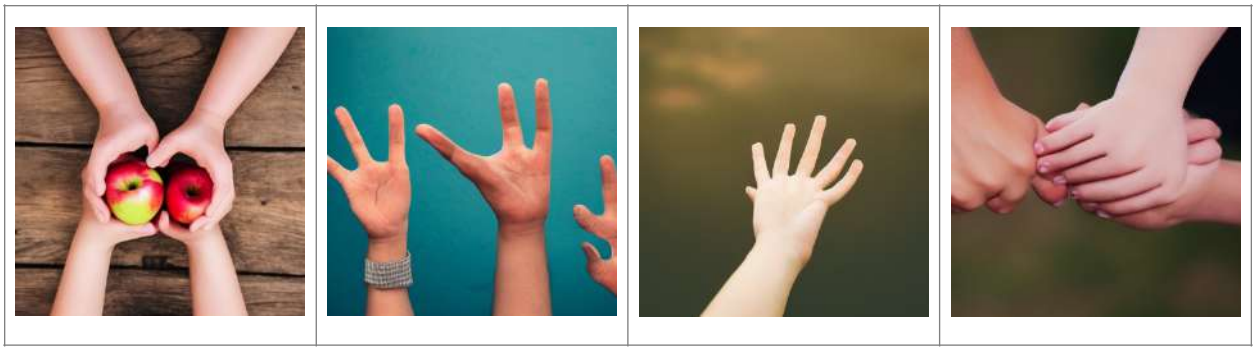
Cuando estos modelos de inteligencia artificial comenzaron a emerger y a ganar popularidad en internet, presentaban principalmente tres limitaciones significativas: la capacidad para generar caras, manos y texto. Con el tiempo, Stable Diffusion ha experimentado un avance considerable, ofreciendo una nueva

forma de generar rostros. Anteriormente, el modelo presentaba problemas en la generación de caras, lo que resultaba en imágenes deformes y sin sentido. Sin embargo, estos inconvenientes fueron solucionados.

Actualmente, los modelos de generación basados en técnicas de difusión enfrentan principalmente dos limitaciones en lugar de tres: las manos y los textos. Existen varias razones detrás de estas limitaciones en la generación de este tipo de modelos de inteligencia artificial, las cuales exploraremos en secciones posteriores.

Manos: ¿Por qué son una limitante?

Cualquier persona que haya intentado dibujar manos se percatará de que, en general, es una tarea bastante complicada. Esto no es una excepción para modelos de inteligencia artificial grandes y potentes, como el que estamos analizando, llamado Stable Diffusion. Por lo general, estos modelos enfrentan dificultades para dibujar y crear imágenes de manos, principalmente debido a que suelen estar compuestas por muchos dedos, y es complicado entender cómo y cuántos de ellos deben estar presentes para que se vean bien. Además, las proporciones entre los dedos y su relación en función del ángulo pueden variar considerablemente. En resumen, aunque el modelo Stable Diffusion puede generar imágenes sorprendentes de muchos objetos, en el caso de las manos, los resultados suelen ser bastante pobres y negativos. Por ahora, esto puede parecer confuso sin una ilustración, pero podemos observar con más detalle lo que estamos mencionando en el siguiente gráfico:



Generación Stable Diffusion 36 – Hands holding an apple, Hands doing a peace sign, Woman showing hands, Pair of hands holding together.

Para comprender esto en mayor detalle, primero debemos tener una idea clara de cómo funcionan estos modelos generadores de imágenes basados en técnicas de difusión. Además, es esencial entender el proceso de entrenamiento que se lleva a cabo en este tipo de modelos. Como sabemos, entrenar un modelo de inteligencia artificial, especialmente uno tan grande como Stable Diffusion, requiere que el modelo considere una gran cantidad de datos diversos.

Por ejemplo, si tenemos la capacidad de generar imágenes de sombrillas utilizando un modelo, significa que, en algún momento, el modelo debió haber sido entrenado con una gran cantidad de imágenes de sombrillas. De esta manera, el modelo puede entender a qué se refiere cuando se menciona "sombrilla" y, efectivamente, no consumir demasiado poder computacional buscando imágenes nuevas para entrenar en ese instante. Esto facilita la atención a las necesidades del usuario y la generación de imágenes de manera eficiente.

En términos sencillos, podemos decir que la razón por la cual estos modelos suelen generar resultados pobres al mostrar imágenes que contienen manos, se debe a que, generalmente, las imágenes con las que fueron entrenados estos modelos pueden no incluir muchas imágenes de manos o, por otro lado, no se les enseña el funcionamiento de estas. De este modo, el modelo no tiene la información necesaria para comprender el comportamiento de las imágenes de manos. Además, dependiendo del ángulo, el modelo puede generar, por ejemplo, una mano con más o menos dedos de lo normal. Esto se debe principalmente a que, debido al ángulo de la imagen, pueden aparecer o no diferentes elementos. Aunque el modelo entiende qué es una mano, en realidad no posee el conocimiento sobre cómo funcionan los dedos en una imagen para generar una representación adecuada.

No necesariamente implica que este fenómeno ocurra únicamente en las manos al momento de generar imágenes. De hecho, se trata de un concepto que sucede frecuentemente cuando intentamos crear algo que, por lo general, presenta muchos patrones a considerar, como, por ejemplo, los dientes, las manos, las pecas, entre otros elementos con patrones secuenciales complejos para un modelo.

Texto: ¿Por qué es un limitante?

Ahora intentaremos explicar por qué los modelos basados en la técnica de difusión, como el de Stable Diffusion, generan texto de baja calidad y qué novedades presenta esta nueva versión en el ámbito de la inteligencia artificial.

Los modelos de generación de imágenes basados en difusión, como Stable Diffusion, no son eficientes al generar imágenes. Estos modelos de IA se enfocan en detectar patrones en imágenes y, a partir del prompt y base de datos, crean imágenes relacionadas. No comprenden lo indicado, sino que se basan en imágenes de entrenamiento para ajustarse a la solicitud.

Es posible que no todos estén al tanto de que estos modelos de inteligencia artificial no son buenos generando texto. Aunque generar texto en nuestras imágenes usando inteligencia artificial puede no ser el principal motivo para utilizarlos, veamos en detalle este aspecto con algunas imágenes de ejemplo.

Generación Stable Diffusion 37 – Board that says hello, Sign that says text, KFC logo and text, Plain text

Como mencionamos anteriormente, nuestro modelo se enfoca en identificar diferentes patrones dentro de la imagen y relacionarlos con las imágenes originales utilizadas en el entrenamiento. Por lo tanto, cuando se le solicita escribir texto, no comprende ni ejecuta correctamente la tarea. Esto se debe principalmente a que el modelo solo ha aprendido a identificar patrones, como los colores y la iluminación de las imágenes originales con las que se entrenó. De esta manera, no entiende en profundidad lo que ocurre dentro del texto que se le indica. Nuestro modelo se centra en crear imágenes, pero no sabe escribir. Esto resulta en un rendimiento bastante bajo al generar texto y la incapacidad de producir texto de calidad.

De hecho, podemos observar que las imágenes y los resultados finales que ha generado el modelo en las versiones anteriores realmente no corresponden del todo a lo que nosotros le pedimos al modelo. Además, también podemos decir que, si bien es cierto que el modelo de alguna forma hace el esfuerzo de generar las imágenes en base a la solicitud inicial que se le haya dado, la realidad es que realmente el texto dentro de este modelo como generación es bastante pobre y realmente no produce el tipo de resultados que nosotros podemos esperar al respecto.

La parte positiva de todo esto es que, generalmente, este tipo de modelos de inteligencia artificial, a pesar de que generan resultados bastante pobres y malos al momento de realizar texto dentro de sus generaciones, la realidad es que actualmente este mismo tipo de modelos de inteligencia artificial ha podido sobrepasar este tipo de limitantes. Haciendo que las generaciones de imágenes actuales con la nueva versión de este modelo de inteligencia artificial puedan y tengan la capacidad de generar imágenes con una fidelidad simplemente increíble. Podemos ver de una forma más detallada y puntual lo que estamos mencionando en la sección posterior a esta, donde veremos cómo esta nueva versión del modelo ha sobrepasado y dado grandes resultados en sus generaciones como un avance positivo, lo cual representa un nuevo tipo de logro positivo para la comunidad de la inteligencia artificial.

Stable Diffusion XL y sus avances en limitaciones

Ahora bien, como mencionamos anteriormente, los modelos de Stable Diffusion suelen presentar limitaciones al generar imágenes. No obstante, con el lanzamiento de este nuevo modelo de inteligencia artificial, se ha logrado un avance significativo en estos aspectos. Primero, es importante destacar que, aunque el progreso en cuanto a las imágenes generadas no es tan amplio, sí podemos resaltar un avance sumamente relevante en la generación de contenido que incluye texto.

Es decir, una de las novedades más destacadas e importantes de este nuevo modelo, en comparación con capacidades técnicas previas, es su habilidad para generar imágenes con texto de manera muy fiel a lo solicitado inicialmente.

Este avance representa un progreso significativo en general para el campo de la inteligencia artificial y los modelos que emplean un proceso de difusión para generar imágenes, ya que demuestra un avance constante en los desafíos que estos tipos de modelos presentaban. En resumen, este desarrollo es de gran importancia para el mundo de la IA.

Esta novedad es de las más importantes en este nuevo modelo de inteligencia artificial, ya que las imágenes que genera, además de ofrecer resultados profesionales y realistas, llegan a tal punto que parecen auténticas, siendo difícil creer que han sido creadas por una IA.

Además, proporciona la nueva capacidad de generar imágenes con texto en su interior, asegurando que este último corresponda efectivamente a lo solicitado previamente. En otras palabras, el modelo se encarga de que las imágenes generadas tengan una resolución y fidelidad excelentes, así como un óptimo resultado con el texto en particular que se desea agregar a las imágenes.

Podemos analizar este tema con mayor detalle mediante las siguientes imágenes, en las cuales se muestra cómo se utiliza Stable Diffusion XL para generar imágenes que contienen carteles y diversos elementos. De hecho, las imágenes finales producidas por este modelo son bastante precisas y se asemejan a lo solicitado inicialmente.

Generación Stable Diffusion 38 – A racoon holding a sign that says hello, Photo of a woman wearing a t-shirt that says SDXL, A photo of a gorilla wearing a hat that says kfc, A pice of paper that says hello

Es crucial enfatizar que, dado que este nuevo modelo es una versión preliminar en desarrollo de Stable Diffusion, se espera que su rendimiento mejore considerablemente a medida que evolucione.

Al momento de escribir este artículo, este innovador modelo de inteligencia artificial aún no se ha lanzado como un proyecto de código abierto. Sin embargo, aunque actualmente solo podemos acceder a él a través de la plataforma DreamStudio de StabilityAI, la idea es que eventualmente se libere como código abierto para el beneficio de toda la comunidad.

Esto permitirá el surgimiento de diversas aplicaciones y sitios web que ofrezcan versiones actualizadas del modelo, cada una con avances específicos. En resumen, esta nueva versión de Stable Diffusion representa un progreso extremadamente significativo e importante en el ámbito de la inteligencia artificial.

Conclusión del capítulo

En la sección actual, hemos identificado y examinado profundamente las limitaciones que enfrentaba Stable Diffusion en la generación de imágenes, especialmente en las áreas de manos y texto. La complejidad en la representación de manos se explicó a través del análisis de la estructura y la falta de información detallada en los modelos de entrenamiento, mientras que la generación de texto se discutió como un aspecto desafiante por la naturaleza del enfoque en patrones. A pesar de estas limitaciones, el capítulo concluyó con un análisis prometedor de los avances significativos en la generación de contenido con texto en la nueva versión de Stable Diffusion XL, destacando la importancia de la constante evolución y el compromiso con la mejora en el campo de la inteligencia artificial.

PROYECTO DEEP FLOYD IF

Objetivos del capítulo

El propósito de este capítulo es explorar y analizar las recientes innovaciones en modelos de inteligencia artificial para la generación de imágenes, incluyendo tanto el Proyecto Deep Floyd IF como Stable Diffusion. Se busca comparar y contrastar estas tecnologías, enfocándose en sus capacidades para generar imágenes con texto y fotorrealismo. Además, se pretende abordar el avance acelerado en el desarrollo de modelos generativos, la competencia en el mercado, y la naturaleza única y abierta de Stable Diffusion en comparación con otros modelos de IA. La intención también es proporcionar una perspectiva sobre el futuro incierto pero prometedor de estos modelos, reconociendo su potencial en diversos campos y su contribución al progreso científico y tecnológico.

Ahora, en relación con los temas que hemos discutido recientemente, mencionamos que la nueva versión de Stable Diffusion nos ofrece la capacidad de generar imágenes que contienen texto. Es importante recordar que esto solía ser una de las mayores limitaciones en los modelos de generación de imágenes basados en la técnica de difusión. Antes del lanzamiento de esta nueva versión, ya existían proyectos en desarrollo que intentaban lograr lo mismo. En otras palabras, ya había modelos de inteligencia artificial generativa, similares a Stable Diffusion, enfocados en crear imágenes con texto en su interior.

Uno de estos modelos es el proyecto "IF" desarrollado por DeepFloyd Lab, que también es un modelo generativo de imágenes parecido a Stable Diffusion. Sin embargo, su importancia y diferencia radican en su capacidad para generar imágenes con texto y enfocarse en el fotorrealismo. ¿Cómo es posible que un

modelo de inteligencia artificial como el IF pueda realizar tareas de generación de imágenes que Stable Diffusion no ha podido lograr hasta ahora?

Para entender esto, es fundamental señalar que, a diferencia de Stable Diffusion, el proyecto IF de DeepFloyd Lab puede comprender el lenguaje natural. Esto es crucial, ya que implica que este modelo de inteligencia artificial puede entender realmente lo que se solicita dentro del modelo. Aunque otros modelos, como Stable Diffusion, pueden generar resultados sorprendentes, no significa necesariamente que comprendan lo que se les solicita. Por otro lado, IF afirma tener la capacidad de entender profundamente lo que se le pide, lo que le da cierta ventaja en la generación de contenido con texto o fotorrealismo en sus imágenes.

A pesar de los desafíos, la comunidad de inteligencia artificial ha logrado avances significativos en la generación de texto a partir de imágenes creadas por modelos de IA. Esto ha contribuido a que otros proyectos de modelos más grandes, como Stable Diffusion, tengan un punto de partida para implementarse en sus propias estructuras. Ahora, podemos observar que la capacidad de generación de este modelo es realmente sorprendente. A continuación, podemos ver algunas imágenes que fueron generadas utilizando este modelo:

Generación Deep Floyd IF 1 – Ejemplos de capacidad de generación de IF (DeepFloyd Lab, 2023).

Con el avance de proyectos como estos y de modelos de generación populares como Stable Diffusion, nos acercamos cada vez más a la perfección de este tipo de modelos. Aunque todavía existen limitaciones y desafíos futuros, el desarrollo y progreso de estos modelos apenas está comenzando. Estos proyectos han sido objeto de estudio e interés para muchas personas, y el rápido avance en la investigación de la inteligencia artificial muestra que la humanidad está progresando a pasos agigantados en este campo.

El futuro de los modelos generativos

En efecto, como mencionamos anteriormente, el progreso en el desarrollo de modelos de inteligencia artificial es asombrosamente rápido, dejando a muchos sorprendidos. El lapso entre la aparición de modelos con innovaciones relevantes

se acorta cada vez más. Sin embargo, esto dificulta mantenerse al tanto del avance en el estado del arte de dichos modelos, debido a su evolución vertiginosa.

Modelos de generación de imágenes destacados y potentes, como Stable Diffusion, ofrecen una extensa gama de funcionalidades. Estos modelos no se limitan únicamente a generar imágenes a partir de prompts iniciales, sino que también proporcionan capacidades para casi cualquier tipo de manipulación de imágenes que podamos imaginar. Entre estas habilidades, se encuentran la transformación de una imagen base en otra completamente diferente, como en el caso del modelo ControlNet. Además, estos modelos pueden integrarse en herramientas de edición de imágenes ampliamente utilizadas en la industria, como Photoshop, permitiendo la edición en tiempo real de imágenes y la generación de videos basados en ellas, entre otras funciones.

El futuro de estos modelos es incierto debido a su avance acelerado y al surgimiento constante de nuevos modelos. Existe toda una industria detrás, con numerosos investigadores y desarrolladores esforzándose por crear y ofrecer al mundo modelos superiores a los anteriores. Esto convierte a la inteligencia artificial en un área de enfoque significativo en el progreso científico, teniendo en cuenta diversos aspectos importantes a lo largo del desarrollo de más modelos. A pesar de la incertidumbre, una cosa es segura: estos modelos seguirán mejorando con el tiempo, y en pocos años presenciaremos modelos aún más avanzados. Aquello que hoy nos asombra palidecerá en comparación con lo que emergerá tras un periodo de mayor investigación y desarrollo.

La inteligencia artificial avanza a un ritmo vertiginoso, y es evidente que la integración de estos modelos en nuestra vida cotidiana y laboral será cada vez más notoria. Esto no necesariamente implica consecuencias negativas, sino que puede ser una herramienta que potencie aún más nuestro trabajo. Es crucial reconocer que el avance y desarrollo de la inteligencia artificial no depende únicamente de Stable Diffusion o la técnica de difusión. Diferentes modelos con enfoques, precios, accesibilidad y rendimiento computacional variados están en constante evolución.

Independientemente del modelo, ya sea Stable Diffusion u otro, su progreso siempre representará un avance positivo en la investigación de la inteligencia artificial y su historia. Por lo tanto, podemos concluir que el futuro de los modelos de generación de imágenes es en realidad incierto, pero seguramente sorprendente y lleno de progreso. Incluso en la actualidad, ya existen avances notables, como la capacidad de generar videos en cuestión de segundos o minutos. El futuro de la inteligencia artificial es prometedor, y solo queda esperar y ver cómo evoluciona este campo de estudio y las nuevas tecnologías que emerjan en relación con él.

El potencial de la inteligencia artificial va más allá de la generación de imágenes y videos, incluyendo una amplia variedad de aplicaciones en diversos campos

como medicina, finanzas, educación y medio ambiente, por mencionar algunos. Esto también abarca diversos modelos con diferentes aplicaciones que utilizan, incluso, la técnica de difusión. A medida que los modelos de inteligencia artificial sigan mejorando, también aumentará nuestra capacidad para enfrentar desafíos complejos y hallar soluciones innovadoras en estos ámbitos. La colaboración entre investigadores y la comunidad en general será esencial para asegurar el desarrollo ético y responsable de la inteligencia artificial. En última instancia, el futuro de la inteligencia artificial promete transformar nuestra sociedad de maneras sorprendentes e impredecibles, abriendo nuevas oportunidades y desafíos en la era digital.

Stable Diffusion: No es un modelo único

Ahora, es crucial resaltar un aspecto que puede parecer obvio al principio, pero a menudo se pasa por alto u olvida. Como mencionamos antes, Stable Diffusion es un modelo de inteligencia artificial que genera imágenes utilizando una técnica previamente mencionada. Esta técnica empleada por Stable Diffusion se conoce como difusión. Sin embargo, no podemos afirmar que sea el único modelo generador de imágenes; de hecho, existen numerosos modelos que también utilizan esta técnica.

La popularidad y ventaja de Stable Diffusion frente a sus competidores se debe a su facilidad de uso y su naturaleza de código abierto. Entrenar modelos de inteligencia artificial suele ser un proceso costoso, ya que requiere unidades de procesamiento gráfico (GPU) especializadas. Las tarjetas gráficas comunes generalmente no son lo suficientemente potentes para proporcionar un rendimiento óptimo en estos casos. Incluso las tarjetas gráficas comerciales de alto rendimiento son insuficientes frente a otras diseñadas específicamente para entrenar modelos de inteligencia artificial de gran tamaño.

Estas GPU especializadas se centran en el entrenamiento de modelos de alta capacidad y no necesariamente en videojuegos. Por lo general, no son comerciales y están destinadas exclusivamente a este propósito. Adquirir una tarjeta gráfica comercial de alto rendimiento para entrenar modelos de inteligencia artificial puede ser costoso y tedioso. Si esto es cierto para una tarjeta gráfica convencional, adquirir una tarjeta gráfica dedicada exclusivamente a la inteligencia artificial puede ser aún más costoso y difícil debido a la demanda alta y la oferta baja.

A pesar de contar con una tarjeta gráfica diseñada específicamente para inteligencia artificial, el proceso de entrenamiento de estos modelos suele ser lento. Esto se debe a que, en general, los modelos tienen que entrenarse utilizando millones de datos, independientemente de la capacidad computacional de nuestra computadora o de la tarjeta gráfica utilizada. Por lo general, el entrenamiento adecuado requiere muchas horas o incluso días y el modelo debe

someterse a diferentes pruebas y escenarios de entrenamiento para alcanzar su punto óptimo.

No obstante, Stable Diffusion no es el único modelo de inteligencia artificial en el mercado de las inteligencias artificiales generativas. De hecho, existen otros modelos que ofrecen un rendimiento similar, e incluso, algunos argumentan que dicho rendimiento puede ser mejor de lo esperado. Esto podría parecer sorprendente, pero en realidad, no lo es si se considera que la investigación de modelos generativos de inteligencia artificial es un tema muy popular en el mundo científico.

Por consiguiente, diversos grupos de investigación y empresas están desarrollando modelos de inteligencia artificial en una especie de carrera competitiva, donde el objetivo es determinar qué grupo puede producir y lanzar al mercado un modelo avanzado lo más rápido posible. Esta competencia fomenta el rápido avance de la inteligencia artificial, pero también presenta desafíos, como la limitación de la colaboración y retroalimentación entre grupos de investigación, quienes pueden optar por no compartir sus conocimientos y avances para mantener una ventaja comercial.

A pesar de esto, han surgido modelos exitosos en el campo de la generación de imágenes utilizando técnicas de difusión, como DALL-E de StabilityAI, que se considera un pionero en la industria. Aunque DALL-E y otros modelos similares han ganado popularidad, muchos de ellos requieren suscripciones o la compra de créditos para acceder a sus servicios.

Stable Diffusion, en cambio, ha logrado gran importancia y popularidad debido a su enfoque de código abierto, lo que permite a la comunidad participar en su evolución y contribuir al progreso del modelo. La colaboración de los usuarios en el desarrollo de este tipo de proyectos hace que estos modelos sean especialmente interesantes.

En el mercado existen numerosos modelos generativos de inteligencia artificial, lo cual ofrece diversidad y permite encontrar aquel que mejor se adapte a nuestras necesidades. Esta variedad posibilita la identificación de modelos con enfoques y resultados distintos en la generación de imágenes y otros elementos multimedia relacionados.

Conclusión del capítulo

Este capítulo ha ilustrado la rápida evolución en el campo de la generación de imágenes utilizando inteligencia artificial, con un enfoque particular en el Proyecto Deep Floyd IF y Stable Diffusion. Hemos explorado cómo el entendimiento del lenguaje natural por parte del Proyecto IF y la naturaleza de código abierto de Stable Diffusion les han otorgado ventajas únicas en la industria. Además, se ha subrayado el avance acelerado en la investigación y desarrollo, lo que lleva a una competencia feroz y a una continua innovación en la generación de imágenes. La incertidumbre en cuanto al futuro de estos modelos no eclipsa su promesa y potencial en una variedad de campos, desde la medicina hasta el medio ambiente. La conclusión resalta que, a pesar de los desafíos y limitaciones actuales, la inteligencia artificial seguirá transformando nuestra sociedad en formas sorprendentes, y modelos como Stable Diffusion continuarán desempeñando un papel vital en este emocionante viaje.

CAPÍTULO 7: MODELOS DE DIFUSIÓN

Objetivos del capítulo

Este capítulo tiene como objetivo principal brindar una comprensión detallada de los modelos de difusión en la inteligencia artificial, con un enfoque en Stable Diffusion. Los lectores serán introducidos a los fundamentos de los modelos de difusión, su relación con Stable Diffusion, y la importancia y funcionamiento profundo de esta técnica. Se explorarán conceptos tales como el proceso de generación en Stable Diffusion, la aplicación de la técnica de difusión a elementos multimedia más allá de las imágenes, y una explicación específica de la técnica de difusión. Se pretende que el lector adquiera una comprensión profunda de cómo la difusión y los modelos de eliminación de ruido se aplican en diferentes campos, como la generación de imágenes, audio y más.

Ahora que poseemos una comprensión básica del funcionamiento de Stable Diffusion y las diversas aplicaciones que podemos lograr con este modelo de inteligencia artificial, es momento de adentrarnos más en su funcionamiento. Para ello, primero debemos comprender qué son los modelos de difusión y por qué Stable Diffusion se clasifica como tal. En esta sección, se abordará la importancia de los modelos de difusión y su relación con Stable Diffusion. Además, se explorará en detalle la relevancia y el funcionamiento profundo de esta técnica para entender completamente los procesos involucrados.

La importancia de este tipo de modelos no se limita únicamente a la generación de imágenes y otros elementos multimedia, sino que también abarca la idea detrás de las técnicas de difusión para la creación de contenido. A pesar de que el enfoque principal es la generación de contenido, la realidad es que las aplicaciones que este modelo de inteligencia artificial puede ofrecer son mucho más amplias, por ejemplo, limpiar señales y eliminar ruido en diferentes elementos.

La aplicación de técnicas de difusión es un concepto relativamente nuevo; sin embargo, los modelos que utilizan esta técnica y las diferentes aplicaciones que ofrecen continúan en constante desarrollo e investigación. Esto hace que sea solo cuestión de tiempo para que aparezcan variantes de la técnica o incluso aplicaciones mucho más significativas.

¿Qué son los modelos de difusión?

Antes de avanzar, es necesario comprender mejor qué son los modelos de difusión. Es importante aclarar que Stable Diffusion no es el único modelo existente. Aunque el nombre "Stable Diffusion" incluya la palabra "difusión" en inglés, no implica necesariamente que la existencia de este modelo de inteligencia artificial tenga una relación directa con la técnica de difusión en sí misma.

En términos sencillos, se conoce como modelos de difusión a una variedad de modelos generativos de inteligencia artificial que utilizan una técnica de generación de contenido denominada "Difusión". Aunque esta definición podría parecer trivial, en este artículo profundizaremos en el funcionamiento y los distintos aspectos relacionados con esta técnica, así como en la forma en que los modelos pueden emplearla.

Como se mencionó anteriormente, Stable Diffusion no es el único modelo de inteligencia artificial que utiliza esta técnica para generar contenido. De hecho, hay muchos modelos de inteligencia artificial que la usan desde hace mucho tiempo antes de la creación de Stable Diffusion. Sin embargo, debido a la novedad de la técnica y a la falta de un desarrollo profundo en diferentes modelos, estos no presentaban resultados tan impresionantes como la mayoría de los modelos de generación de imágenes actuales. Algunos de los modelos de generación que utilizan esta técnica son Stable Diffusion, DALL-E, Midjourney, entre otros.

Modelos de difusión: No solamente genera imágenes

Es crucial resaltar un hecho que quizás muchas personas desconocen: aunque estemos estudiando Stable Diffusion y este modelo de inteligencia artificial se enfoque en la generación de imágenes, no significa que los modelos basados en esta técnica estén únicamente enfocados en la generación de imágenes. De

hecho, también se han aplicado a la generación de otros archivos multimedia, como videos y audio, lo que permite que la técnica sea utilizada en diversos aspectos y elementos que queremos generar.

Recordemos el proceso llevado a cabo mediante la técnica de difusión. A pesar de centrarse en la generación de imágenes, este proceso no se limita exclusivamente a ello. Puede expandirse a otros tipos de elementos multimedia, como videos y audio. Más adelante veremos algunos ejemplos, pero es importante mencionar que el proceso principal en los modelos de difusión consiste en agregar ruido a una imagen previamente obtenida. El objetivo es añadir ruido hasta obtener un elemento completamente ruidoso, donde el ruido añadido no tiene una relación directa con el proceso original indicado al modelo.

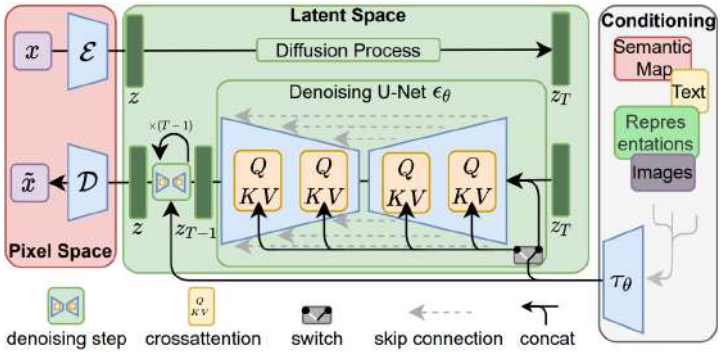


Ilustración 38 – Proceso de generación en Stable Diffusion (Técnica de difusión) (Rombach, Blattmann, Lorenz, Esser, & Ommer, 2021).

Posteriormente, entrenamos el modelo para descubrir el proceso necesario para eliminar el ruido de una imagen. Comenzamos con una imagen completamente ruidosa y, de forma iterativa, el modelo aprende cómo proceder para obtener una imagen similar a la original, pero sin ruido. En resumen, entrenamos al modelo para eliminar el ruido de manera efectiva.

Este proceso es más fácil de visualizar con imágenes, ya que podemos ver gráficamente cómo el modelo añade ruido a la imagen original hasta que el modelo final genera el resultado esperado. Sin embargo, esto no implica que la técnica de difusión se aplique solo a las imágenes. También se puede aplicar al audio, pues este puede contener ruido. Entrenar un modelo de inteligencia artificial para convertir un audio ruidoso en uno limpio es factible mediante esta técnica de difusión. De hecho, ya existen estudios y experimentos que intentan replicar esto. Es importante mencionar que, generalmente, es más fácil identificar el ruido en elementos como imágenes, ya que podemos convertirlas en vectores que podemos usar en el modelo. En cambio, la codificación del audio es más compleja que la decodificación de una imagen, lo que dificulta llevar a cabo modelos con elementos distintos a imágenes, ya que el proceso intermedio es más extenso y complicado.

Por ahora, no es necesario comprender completamente cómo aplicar la técnica de difusión en otros elementos multimedia además de las imágenes. Basta con entender que es posible aplicarla a cualquier elemento que contenga ruido y que podemos entrenar el modelo de tal forma que aprenda a eliminar el ruido de dicho elemento. Este es el concepto principal que rige la técnica de difusión en la inteligencia artificial.

¿Qué es la técnica de difusión?

Ahora es momento de examinar con más detalle el funcionamiento de esta técnica y por qué se puede utilizar para generar imágenes. En términos simples, existen muchas variantes de los modelos de difusión debido a la aplicabilidad de la técnica. Por lo general, en el caso de los modelos de inteligencia artificial generativos de contenido, como el "Stable Diffusion", estos modelos se basan en una variante de la técnica de difusión llamada "modelos probabilísticos de eliminación de ruido" (Rombach, Blattmann, Lorenz, Esser, & Ommer, 2021). Si bien esto no es una regla estricta, estos modelos de eliminación de ruido se han convertido en un estándar en la generación de modelos de inteligencia artificial como el "Stable Diffusion" y otros similares, debido a la eficiencia y facilidad que ofrece esta técnica.

El concepto de esta técnica de generación de imágenes es fácil de entender a nivel superficial. En términos simples, se trata de un proceso en el que un modelo de inteligencia artificial recibe una imagen como punto de partida. Como sabemos, las imágenes están compuestas por píxeles, cada uno de los cuales es reemplazado por diferentes etapas de ruido gaussiano.

La tarea principal del modelo de difusión de eliminación de ruido es recrear la imagen original, eliminando el ruido gaussiano y agrupando los píxeles para formar la imagen final. Esperamos que la imagen final generada por el modelo corresponda con la imagen original proporcionada y presente un cierto nivel de similitud natural con ella.

Es muy probable que gran parte de los temas que abordaremos contengan terminología técnica, y es posible que incluyan fórmulas o bloques de código. Sin embargo, no debemos preocuparnos por esto en este momento. Veremos estos elementos a lo largo del libro y en los diferentes temas, de manera que queden claros y concisos, sin dejar ninguna duda.

Para comprender mejor el concepto que hemos explicado anteriormente, que incluye la imagen inicial, el ruido gaussiano y la generación de la imagen final del modelo, podemos recurrir al siguiente gráfico para visualizar este proceso con mayor claridad.

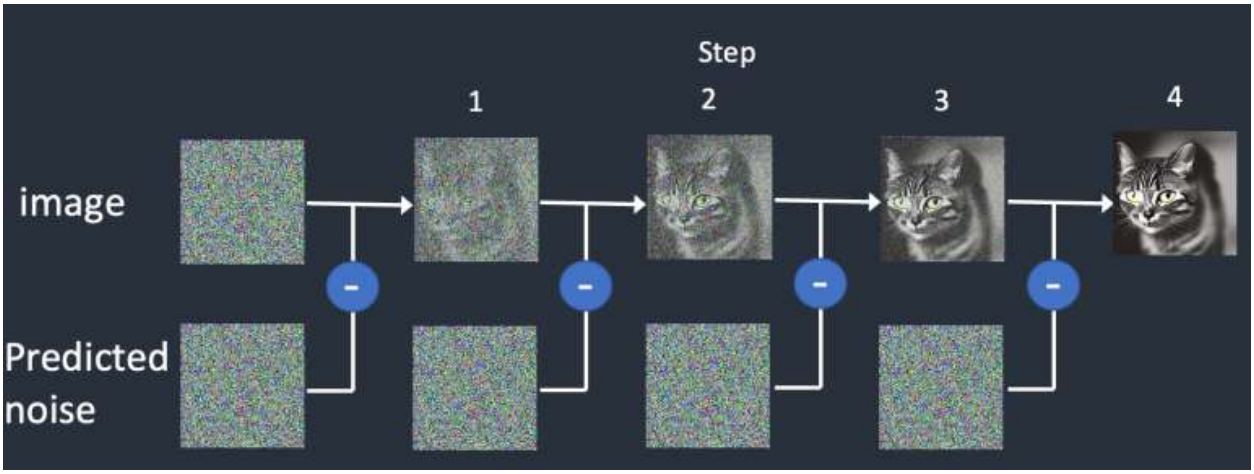
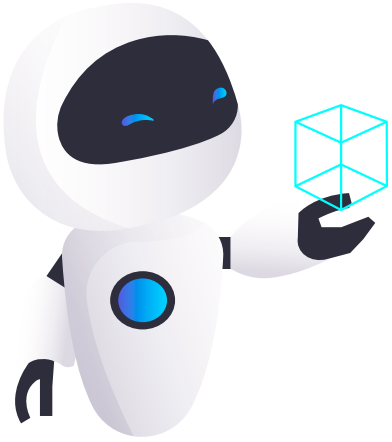


Ilustración 39 – Breve ilustración del proceso de generación de la técnica de difusión inversa (Stable Diffusion Art, 2023).

Para comprender mejor el proceso que se lleva a cabo, dividiremos la explicación en diferentes secciones, cada una de ellas explicando los pasos y secciones que se llevan a cabo en el proceso de generación mediante técnicas de difusión.

El primer paso consistirá en partir de la imagen y su conversión en ruido gaussiano, seguido del entrenamiento de la red y, finalmente, la generación de la imagen final.

Conclusión del capítulo



El capítulo ha desentrañado de manera exhaustiva la naturaleza y aplicabilidad de los modelos de difusión en el contexto de la inteligencia artificial. La presentación en detalle de la técnica de difusión y su relación con modelos como Stable Diffusion, DALL-E y Midjourney ha permitido entender cómo se lleva a cabo el proceso de generación. Hemos visto que la técnica de difusión no se limita únicamente a la generación de imágenes, sino que también puede ser aplicada a otros elementos multimedia. Además, se ha explicado cómo los modelos de eliminación de ruido se han convertido en un estándar en la generación de modelos de inteligencia artificial. A través de explicaciones claras y visualizaciones, este capítulo ha proporcionado una base sólida sobre los conceptos y técnicas asociados con la difusión, preparando al lector para futuras exploraciones y aplicaciones en el campo.

PREÁMBULO: ¿QUÉ ES EL RUIDO GAUSSIANO?

Objetivos del capítulo

El objetivo principal de este capítulo es introducir al lector en el concepto de ruido Gaussiano y explicar su importancia y aplicación en el proceso de generación de imágenes mediante la técnica de difusión. Esto incluye una comprensión detallada de cómo se crea y aplica el ruido Gaussiano en las imágenes, y cómo se relaciona con la técnica de difusión. Además, se pretende explicar el funcionamiento interno del ruido Gaussiano, utilizando ilustraciones para visualizar el proceso y relacionarlo con la distribución gaussiana. Finalmente, el capítulo busca explicar el proceso iterativo necesario para convertir la imagen original en una imagen completamente ruidosa.

Antes de continuar con el primer paso del proceso para convertir imágenes en este tipo de ruido y explicar las diferentes técnicas y herramientas informáticas para hacerlo, es necesario comprender qué es el ruido Gaussiano y su relación con los procesos que detallaremos a continuación. Aunque pueda parecer trivial para algunos, es posible que no resulte completamente claro para muchas personas.

En términos simples, el ruido Gaussiano es un tipo de ruido de imagen causado por interferencias no deseadas. Este tipo de ruido suele estar presente en las imágenes y puede dificultar la visualización de los diferentes elementos que componen la imagen. El objetivo de este libro es centrarse en la generación de imágenes utilizando Stable Diffusion, por lo que no se profundizará demasiado en el funcionamiento y las características técnicas de la definición del ruido

Gaussiano, ya que esto se aleja de la función principal y los temas que se espera cubrir. Sin embargo, para no dejar de lado los temas y elementos que acabo de mencionar, es conveniente proporcionar una explicación visual del ruido Gaussiano mediante el siguiente gráfico.

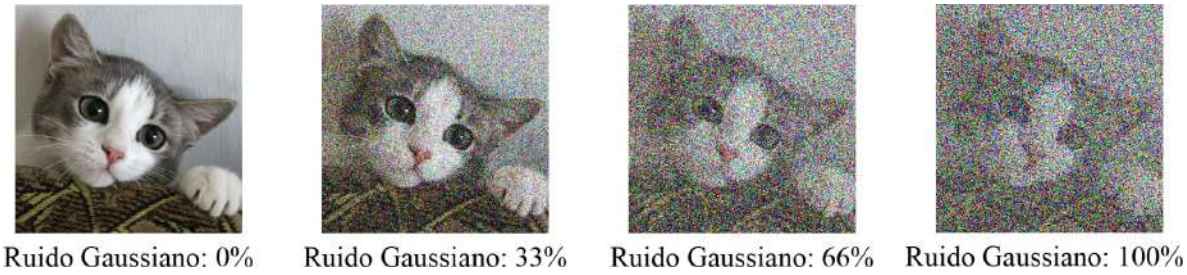


Ilustración 40 – Efectos del ruido Gaussiano en las imágenes.

En esta imagen se observa cómo un patrón de ruido Gaussiano va cubriendo progresivamente una imagen normal, haciendo que este patrón sea más notable e intenso. De esta forma, podemos entender que el ruido Gaussiano se refiere a un tipo de distribución de píxeles aparentemente aleatorios y multicolores que se extienden uniformemente por toda la imagen, dificultando la interpretación de la imagen original. Generalmente, el ruido se genera en los canales de color de la imagen, lo que significa que, si la imagen no utilizara canales de color, como el RGB, sino que se basara en un canal de blanco y negro, entonces el ruido de la imagen se presentaría en los tonos de blanco y negro.

Con esto podemos tener un poco más claro de qué se trata el ruido Gaussiano en las imágenes, por ahora, puede parecer un poco trivial tener que entender este concepto, sin embargo, tendrá mucho más sentido y podremos ver más a detalle cómo se relaciona este ruido con el proceso que se lleva a cabo para poder generar imágenes usando la técnica de difusión.

Funcionamiento: Conversión a ruido Gaussiano

Como mencionamos anteriormente, dividiremos el proceso de generación de difusión en varias etapas para poder explicar detalladamente el funcionamiento de cada uno de los elementos que intervienen en cada una de ellas. En este caso, nos referimos al primer paso que consiste en convertir la imagen original a ruido Gaussiano, el cual es esencial para construir la imagen final utilizando esta técnica.

Agregar ruido Gaussiano a una imagen no es conceptualmente difícil. De hecho, este proceso es bastante sencillo y se puede llevar a cabo utilizando diferentes técnicas y herramientas, como Photoshop y otras aplicaciones de manipulación de imágenes.

Sin embargo, aunque el proceso en sí no es complicado, es importante señalar que, desde una perspectiva informática, programar este tipo de técnicas puede representar un desafío para utilizar estas herramientas. Por lo tanto, en este caso particular, no se pueden utilizar herramientas de manipulación de imágenes para realizar esta tarea.

Ya que no podemos utilizar herramientas de manipulación de imágenes para añadir ruido Gaussiano a nuestras imágenes, ¿cómo podemos lograrlo mediante diferentes técnicas de programación y ciencias de la computación? La forma de hacerlo puede variar según la interpretación que se le dé al problema y la aplicación final que se desee para la imagen resultante. En este caso, buscamos utilizar este proceso para llevar a cabo la generación de imágenes mediante la técnica de difusión.

Para poder comprender este proceso a fondo, vamos a empezar analizando la composición de una imagen. En este caso, utilizaremos una imagen sencilla con canales de color blanco y negro para facilitar la comprensión. La imagen constará de solo 3x3 píxeles, lo que significa que hay un total de 9 píxeles. Este ejemplo es solo con fines de simplificación, ya que se puede aplicar a imágenes más grandes y con más canales de color. También es importante destacar que la imagen que usaremos trabaja con escalas de colores de 8 bits, un detalle que será relevante más adelante.

Podemos interpretar esta imagen como una matriz que contiene valores numéricos del 0 al 255. Este valor corresponde a la profundidad de color de la imagen, que anteriormente mencionamos tiene una profundidad de 8 bits, lo que significa que hay $2^8 = 256$ posibles valores, incluyendo el cero. Por lo tanto, podemos obtener valores que van desde 0 hasta 255, donde 0 corresponde al color negro y 255 corresponde al color blanco. Cada valor intermedio corresponde a la intensidad del color blanco en el píxel.

Cuanto mayor sea el valor, mayor será la intensidad presente del color blanco en el píxel. Podemos observar este efecto en la siguiente imagen:

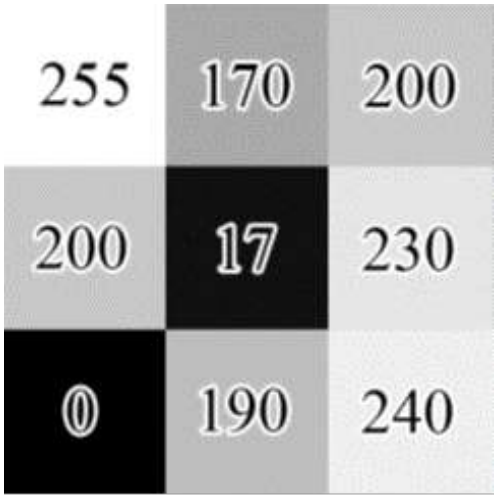


Ilustración 41 – Imagen de 3 x 3 píxeles con sus respectivos valores de escala de grises en 8 bits.

Con esto, podemos visualizar una imagen de 3 x 3 píxeles como una matriz de 9 celdas. Cada celda contiene un valor numérico que representa la intensidad de la luz en la imagen. De esta manera, podemos interpretar nuestra imagen como una matriz que contiene valores numéricos que representan los distintos colores que se pueden obtener. Es importante destacar que este concepto es escalable a imágenes de dimensiones mayores.

Una vez que entendemos claramente este proceso, debemos comprender cómo se aplica el ruido gaussiano a nuestras matrices o imágenes. Aunque esto no explica directamente el funcionamiento del proceso de difusión, se explicará con más detalle en secciones posteriores. Es importante comprender cómo funciona el proceso para agregar ruido gaussiano a nuestras imágenes en este momento.

Partiremos de un ejemplo sencillo en el que añadimos ruido a nuestra imagen. En este ejemplo, explicaremos el proceso interno que se lleva a cabo. El tipo de ruido que debemos agregar a nuestra imagen es el "ruido gaussiano". Este nombre se refiere al uso de una distribución estadística gaussiana, que tiene un nivel relativamente bajo de aleatoriedad y se representa gráficamente en forma de campana de Gauss.

Aunque puede parecer complejo, detallaremos su funcionamiento a continuación: Nuestra tarea es bastante sencilla, cada uno de los valores independientes de nuestra imagen recibirá un poco de ruido Gaussiano, es decir, este valor numérico que tiene cada una de las celdas independientes (Las cuales representan píxeles), se le sumará o restará algún valor, de tal forma que ahora el valor de dicha celda cambiará a uno diferente.

Por ejemplo, podemos ver en el siguiente gráfico un ejemplo que describe lo que debemos de hacer, podemos observar que los valores numéricos cambian a valores diferentes, sea aumentando o restando su valor original.

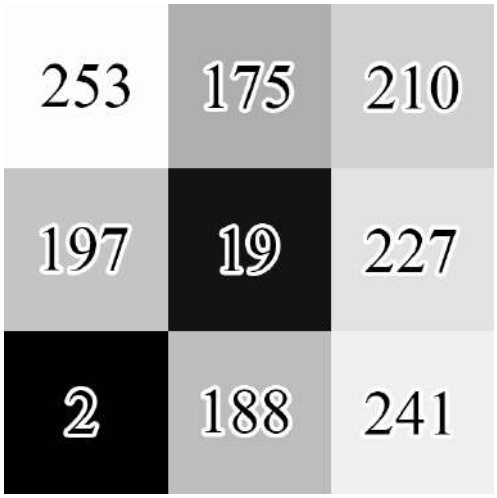


Ilustración 42 – Ilustración de 3 x 3 con un poco de ruido Gaussiano.

Aún no comprendemos completamente el proceso interno que regula el aumento y disminución de los valores almacenados en los píxeles. Es crucial analizar cómo funcionan y qué reglas rigen estos cambios. Los valores se ajustan según una distribución gaussiana, lo que implica que la mayoría se ubicará cerca del centro, con una probabilidad decreciente de alejarse de él. Por ejemplo, en un rango de valores de [-10,10], la distribución gaussiana se centrará en cero, y los valores se dispersarán en un rango de 10 unidades, con mayor probabilidad de estar cerca del centro. Un gráfico ilustra este concepto a continuación:

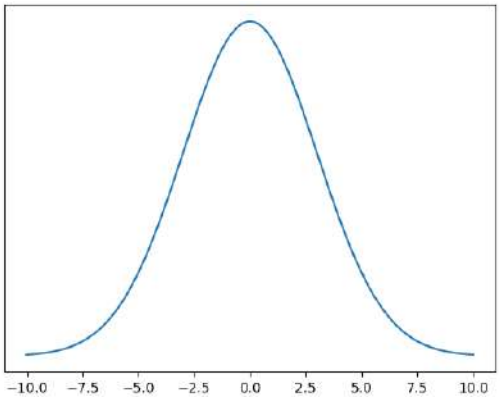


Ilustración 43 – Distribución Gaussiana en rango [-10, 10]

De esta forma, los valores empezarán a generarse dentro del rango [-10, 10]. Estos nuevos valores afectarán a los valores originales que se encuentran dentro de las matrices, incrementando o reduciendo los valores obtenidos en función de las probabilidades de la distribución. Como resultado, los valores dentro de las matrices cambiarán en cada uno de los pasos.

Este es el proceso que se utiliza para convertir las imágenes originales en imágenes con ruido gaussiano. Como se puede apreciar, la generación de estos

valores dentro de un rango definido no es completamente aleatoria, sino que se basa en una distribución probabilística de Gauss.

De este modo, podemos entender que la generación de las imágenes es relativamente predecible al considerar el aumento o disminución de los valores en aquellas secciones con mayor probabilidad de aparecer, como las que se encuentran cerca del valor cero. Es importante destacar que el centro de los valores no siempre debe ser cero, ya que esto puede variar y depender del tipo de variación gaussiana que se esté utilizando para los valores.

Otro punto que debemos considerar en este proceso para entender cómo añadir ruido Gaussiano a nuestras imágenes es que, por lo general, un solo proceso de modificación de valores utilizando la distribución de Gauss puede no ser suficiente para obtener una imagen completamente ruidosa. De hecho, las imágenes resultantes tendrían una pequeña variación en comparación con la imagen original. Por lo tanto, para obtener una imagen completamente ruidosa, se requiere repetir iterativamente el proceso, cambiando los valores de los píxeles una y otra vez hasta que la imagen esté completamente llena de ruido.

Podemos analizar este fenómeno en el gráfico adjunto, en el cual se aprecia que, con cada paso efectuado para introducir ruido a la imagen, se añade únicamente una pequeña cantidad adicional. Sin embargo, no es suficiente como para afirmar que las imágenes están completamente inundadas de ruido.

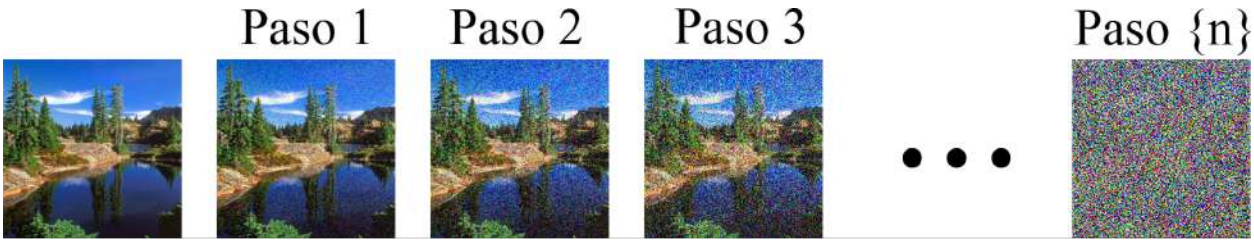


Ilustración 44 – Pasos de entrenamiento para añadir ruido Gaussiano en la técnica de difusión.

Para lograr esto, el proceso para obtener una imagen completamente ruidosa suele ser iterativo. En cada etapa, se añade gradualmente un poco más de ruido a las imágenes. Algunos estudios sugieren que se requieren aproximadamente 1,000 iteraciones para completar este proceso (Rombach, Blattmann, Lorenz, Esser, & Ommer, 2021).

Ruido Gaussiano: ¿Para qué es?

Aunque es normal preguntarse para qué sirve todo esto, es decir, si bien comprendemos el proceso actual para añadir ruido gaussiano a una imagen, también es importante indagar el propósito y la relevancia de este proceso en nuestras imágenes. Principalmente, los motivos más importantes para añadir ruido a nuestras imágenes en el funcionamiento de los modelos de difusión

radican en que la dispersión de los datos es generalmente predecible mediante estadísticas y la probabilidad de dispersión de datos de una distribución gaussiana. Esto significa que la distribución de los datos obtenidos no consiste en valores completamente aleatorios, sino en valores dispersos a través de una distribución probabilística. Por lo tanto, podemos considerar que es posible intentar predecir, mediante ciertas probabilidades, la variación utilizada por la imagen para generar el ruido gaussiano.

Puede parecer confuso y no estar del todo claro cuál es el propósito de agregar ruido gaussiano a las imágenes. Sin embargo, por ahora, basta con saber que estas imágenes con ruido nos serán bastante útiles.

Desde un punto de vista informático, podemos hacer que nuestro programa almacene en memoria cada una de las imágenes generadas con ruido, lo que nos permite tener un registro lineal de la conversión que se llevó a cabo durante el proceso. Podemos visualizar esto mejor con el siguiente gráfico.

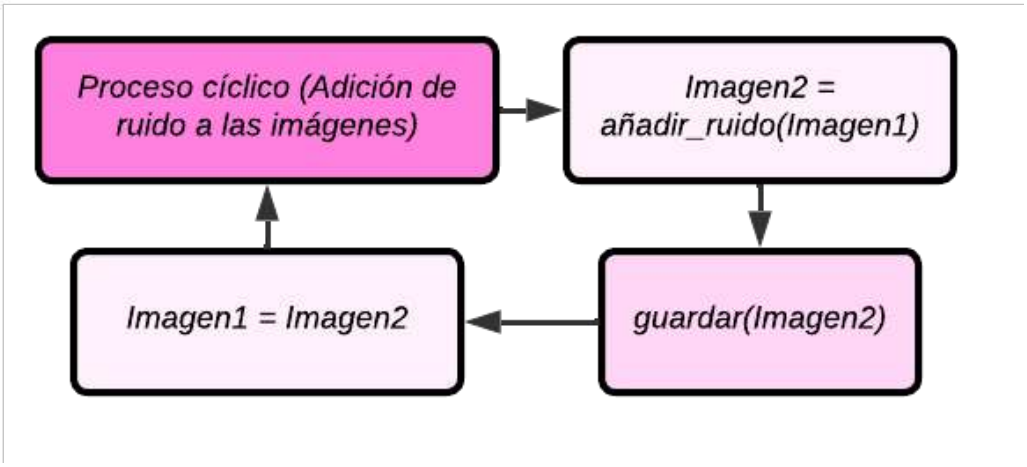


Ilustración 45 – Proceso de guardado de estados de la imagen con ruido en la técnica de difusión.

Ahora, para comprender en profundidad el proceso asociado con imágenes que contienen ruido gaussiano, presentamos una breve descripción del procedimiento informático que se aplica a este tipo de imágenes. En primer lugar, cuando se genera una imagen final completamente llena de ruido, sin relación directa con la imagen original proporcionada, es el momento de entrenar un modelo de inteligencia artificial que aprenda y comprenda cómo convertir dicha imagen ruidosa en la imagen original a través de un "proceso inverso" de generación de ruido gaussiano. Es decir, debemos entrenar a un modelo de IA que sepa cómo transformar una imagen con ruido gaussiano en una que se ajuste a ciertas descripciones proporcionadas.

Para lograr esto, podemos emplear una técnica en la que el modelo analice de manera inversa el proceso que se llevó a cabo para generar la imagen original a

partir de la imagen ruidosa. Así, el modelo entenderá el proceso y la probabilidad de cambio en la dispersión gaussiana, de tal manera que pueda convertir la imagen ruidosa en otra con menos ruido. Luego, repetirá este proceso iterativamente, generando imágenes con menos ruido en cada paso. De hecho, ya hemos visto este proceso anteriormente, el cual se relaciona con los "pasos" mencionados en modelos de Stable Diffusion. Cada paso es un valor numérico que indica la cantidad de veces que se debe eliminar ruido de la imagen.

Así, entonces, podemos entender de una forma más clara cuál es el proceso que se lleva a cabo con las imágenes generadas de ruido y cuál es el papel que toma en el proceso de entrenamiento del modelo. Esto, conceptualmente, es todo el proceso que se lleva a cabo y la técnica como tal que se espera aplicar por medio de las técnicas de difusión, donde se intenta que el modelo aprenda cómo puede avanzar poco a poco para poder generar una imagen que originalmente está conformada de solamente ruido Gaussiano, haciendo que de forma regresiva el modelo aprenda a generar imágenes reales en base del mismo ruido Gaussiano.

Ciertamente, existen numerosos elementos relevantes en el proceso de generación de imágenes de un modelo basado en la técnica de difusión que aún no hemos mencionado. Sin embargo, podemos afirmar que la explicación del concepto central, en el cual se entrena un modelo de inteligencia artificial para aprender progresivamente a eliminar el ruido presente en una imagen, es la parte más importante del proceso. Así, la imagen final generada resulta ser una representación realista, todo ello gracias a la implementación de la técnica de difusión.

Funcionamiento: Etiquetas en el proceso Diffusion

Es probable que muchas personas están familiarizadas con los modelos de inteligencia artificial basados en entrenamiento supervisado, que utilizan "etiquetas" o "labels" en su proceso de aprendizaje. En este contexto, Stable Diffusion es un modelo de inteligencia artificial que también se entrena a través de la supervisión. Sin embargo, no se limita a un simple proceso de etiquetado.

Aunque la técnica de Diffusion se aplica de manera similar en el entrenamiento, hay diferencias notables en el proceso de generación de imágenes. Stable Diffusion no solo utiliza una imagen como punto de partida, sino también una descripción textual de su contenido. En otras palabras, el modelo se entrena a partir de un texto que relaciona la imagen con su descripción.

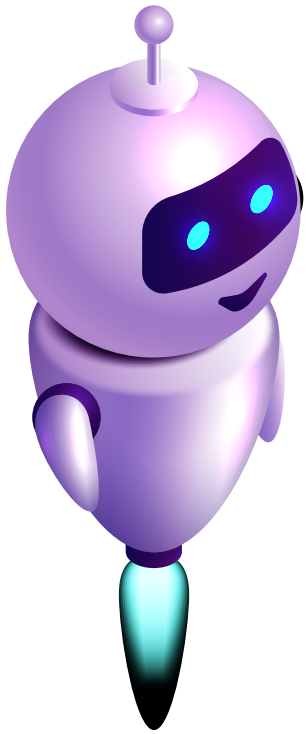
El proceso de entrenamiento de un modelo de inteligencia artificial que utilizar una técnica de generación de imágenes como Stable Diffusion implica entrenar el modelo con descripciones textuales previas de lo que debe generar. Al principio, se parte de la imagen original y se añade ruido gaussiano progresivamente hasta obtener una imagen completamente ruidosa. En esta etapa, las descripciones

textuales no son muy relevantes, debido a que el modelo se centra en añadir ruido a la imagen.

Una vez que se añade el ruido iterativamente y se almacenan las imágenes resultantes, comienza el proceso de entrenamiento. Durante esta fase, el modelo aprende a eliminar progresivamente el ruido de la imagen en cada iteración. Es en esta etapa donde la descripción textual del contenido original se vuelve esencial, ya que permite que el modelo se entrene para ser consciente de la imagen final a la que intenta llegar.

Cuando el modelo esté completamente entrenado, proporcionar un prompt con la descripción textual de lo que deseamos obtener permitirá que el modelo sepa cuál es el proceso que debe llevar a cabo. Sin embargo, si el modelo ha sido entrenado solo con imágenes de automóviles y descripciones textuales sobre su tipo, ante una solicitud hipotética de generar la imagen de un edificio, el modelo probablemente no sabrá cómo proceder y generará una imagen incoherente o simplemente generará otra imagen de un automóvil.

Para que un modelo como Stable Diffusion pueda generar una amplia variedad de elementos con numerosas variaciones, estilos artísticos y especificaciones particulares, debe ser entrenado con una gran cantidad de datos, incluyendo imágenes y sus descripciones textuales correspondientes. Sin embargo, este proceso implica desafíos significativos en términos de tiempo y recursos computacionales. Entender la importancia de entrenar el modelo de esta manera permite comprender cómo, al contar con una mayor cantidad y variedad de resultados, el modelo podrá generar imágenes con una mayor variación y capacidad de resultados.



Conclusión del capítulo

Este capítulo ha presentado un análisis exhaustivo del ruido Gaussiano y su papel esencial en la generación de imágenes a través de la técnica de difusión. Hemos explorado la definición, la naturaleza y el funcionamiento del ruido Gaussiano, así como su aplicación específica en la conversión de una imagen original a una imagen con ruido. Se han discutido las etapas y métodos para aplicar el ruido Gaussiano, incluyendo una explicación del proceso iterativo y su importancia en la técnica de difusión. Mediante ejemplos, ilustraciones y una descripción detallada, este capítulo ha ofrecido una visión completa del ruido Gaussiano, estableciendo una base sólida para comprender cómo se relaciona este concepto con el proceso global de generación de imágenes usando Stable Diffusion.

CAPÍTULO 8: STABLE DIFFUSION: USANDO EL MODELO LOCALMENTE

Objetivos del capítulo

El objetivo principal de este capítulo es introducir y explorar la manera de emplear Stable Diffusion localmente en los dispositivos del usuario. Se pretende demostrar cómo se puede migrar de la utilización de este modelo a través de la nube a una implementación más personalizada y controlada en la computadora personal. Se busca proporcionar una comprensión detallada de cómo usar el modelo tanto en una forma simplificada, a través de aplicaciones de abstracción, como en una forma más detallada y personalizable a través del código fuente. Este capítulo también tiene como objetivo explicar las consideraciones prácticas, como los requisitos de hardware y las aplicaciones disponibles, que permiten una implementación exitosa de Stable Diffusion en un entorno local.

Hasta ahora, hemos tenido la oportunidad de utilizar Stable Diffusion para generar imágenes. Este proceso no consume mucho poder computacional de nuestros dispositivos, ya que existen numerosas páginas web que ofrecen acceso a Stable Diffusion en la nube y permiten usar su capacidad computacional para crear imágenes. Solo necesitamos un navegador web para aprovechar este modelo de inteligencia artificial.

Esta característica ofrece varias ventajas, como la posibilidad de ver las imágenes generadas con Stable Diffusion sin instalar ningún software en nuestros dispositivos. Esta ha sido la principal ventaja que hemos experimentado.

Sin embargo, es importante mencionar que el proceso de entrenamiento y modificación de parámetros del modelo puede ser menos versátil, debido a que estamos utilizando un modelo previamente entrenado por las empresas que gestionan estas páginas web y herramientas que permiten el uso de Stable Diffusion sin instalación previa.

Ahora, en esta sección intentaremos explorar cómo utilizar Stable Diffusion en nuestras computadoras. En lugar de depender del modelo entrenado por los administradores del sitio web, tendremos la oportunidad de aprender cómo emplear el código fuente de Stable Diffusion para implementar un modelo en nuestros dispositivos y comenzar a trabajar con él. La idea de ejecutar un modelo de inteligencia artificial tan grande y potente como Stable Diffusion puede parecer confusa o complicada al principio. Sin embargo, descubriremos que gran parte del modelo ya está preconfigurado y lo único que necesitamos es instalar el modelo a partir del código fuente original, lo que implica que no es necesario programar demasiado.

Ahora, antes de continuar, es importante aclarar que en esta sección exploraremos diversas alternativas. Es decir, el uso de Stable Diffusion en nuestra computadora no implica necesariamente que debamos emplear el modelo desde un punto de vista de bajo nivel, donde debamos descargarlo por completo y programar para poder utilizarlo. De hecho, también existen diferentes herramientas que nos permiten utilizar Stable Diffusion localmente sin la necesidad de tener conocimientos avanzados de programación. Estas herramientas facilitan el disfrute de Stable Diffusion y la modificación de sus parámetros principales en nuestro equipo, sin depender de los sitios web que utilizábamos anteriormente.

En esta sección, analizaremos ambas formas de emplear Stable Diffusion. Por un lado, veremos cómo podemos utilizarlo de manera simplificada y con un proceso de abstracción. Por otro lado, examinaremos cómo podemos emplear Stable Diffusion desde su código fuente, utilizando código escrito en Python para personalizar diversos aspectos del modelo de forma más detallada y comprender cómo podemos construir este tipo de aplicaciones desde cero.

Stable Diffusion local: Aplicaciones de abstracción

La primera forma que exploraremos para utilizar Stable Diffusion localmente en nuestras computadoras será la más sencilla, es decir, mediante aplicaciones de abstracción que facilitan el uso de Stable Diffusion. Simplemente instalamos una aplicación y, voilà, el modelo ya está integrado en ella. No se trata realmente de magia, sino de aplicaciones diseñadas para simplificar el uso de Stable Diffusion sin tener que preocuparnos por aspectos tediosos, como el entrenamiento del modelo, especialmente si no tenemos mucho conocimiento sobre el tema.

Generalmente, estas aplicaciones requieren únicamente ser descargadas e instaladas, y el modelo de Stable Diffusion se integrará en nuestro sistema. Sin embargo, es importante tener en cuenta que, como se mencionó en varias ocasiones en este libro, ejecutar o entrenar modelos como Stable Diffusion suele requerir un gran poder computacional. Por lo tanto, es recomendable considerar el rendimiento y el hardware de la computadora antes de ejecutar el modelo mediante estas aplicaciones. Por lo general, las propias aplicaciones incluyen las descripciones de los requisitos mínimos y recomendados para ejecutar el modelo correctamente. Así que es importante revisar esta información antes de empezar a trabajar con ellas.

La lista de aplicaciones que permite utilizar esta función es bastante amplia, y generalmente, las aplicaciones pueden variar en disponibilidad según el sistema operativo. No obstante, es cierto que existen numerosas herramientas basadas en código abierto que facilitan esta tarea y son compatibles con diversos sistemas operativos. Un ejemplo de ello es FusionKit, una aplicación que proporciona acceso a una amplia gama de funciones de Stable Diffusion en distintas plataformas y sistemas operativos, lo cual resulta muy conveniente.

Uno de los ejemplos más populares entre los seguidores de la comunidad de desarrollo de Stable Diffusion es Easy Diffusion, una herramienta ampliamente utilizada que facilita el proceso de implementación del modelo. Easy Diffusion es una herramienta escrita en Google Colab Notebooks y se encuentra alojada en GitHub. Esta herramienta, desarrollada en Colab, permite utilizar el modelo de manera sencilla al ejecutar las celdas que contiene. Si no se está familiarizado con Google Colab o su función, no hay problema, ya que lo abordaremos con más detalle en otras secciones del libro. Esta herramienta será de gran utilidad para realizar experimentos con este modelo de inteligencia artificial sin consumir excesivos recursos computacionales en nuestro dispositivo.

Existen numerosas herramientas que facilitan el uso de un modelo de inteligencia artificial tan potente como Stable Diffusion en diversos sistemas operativos. Solo basta con buscar estas herramientas para encontrar ejemplos y apreciar la facilidad que ofrecen en comparación con el uso del modelo en su forma básica. A continuación, se pueden observar algunas capturas de pantalla de dichas aplicaciones:

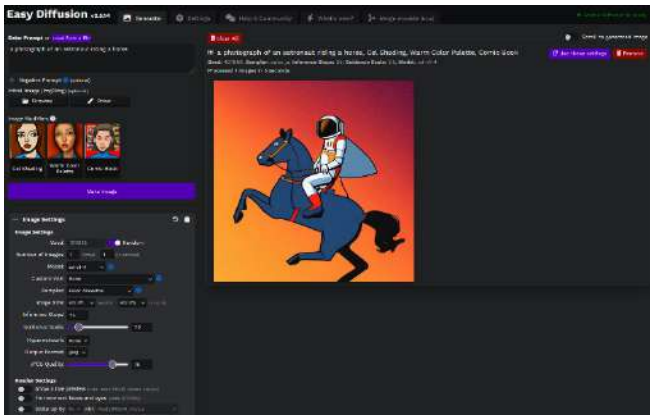
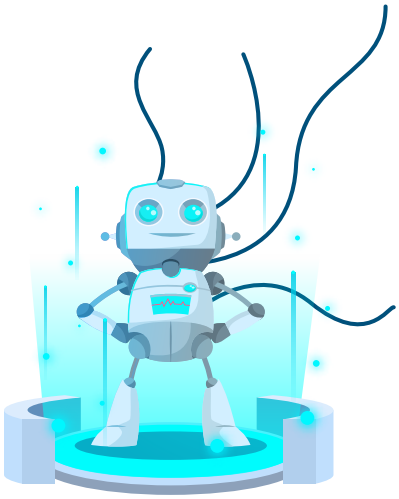


Ilustración 46 – Captura de pantalla de Easy Diffusion, una de las aplicaciones más populares para usar Stable Diffusion localmente de una forma sencilla y rápida (cmdr2 (GitHub), 2023).

Con estas herramientas, podremos experimentar modificando diversos valores y parámetros del proceso de generación de imágenes. Aunque esto se ejecuta en nuestra computadora y consume una cantidad significativa de recursos computacionales, proporciona la posibilidad de personalizar en mayor medida la experiencia de generación de imágenes mediante Stable Diffusion.



Conclusión del capítulo

Este capítulo ha presentado una guía completa sobre cómo usar Stable Diffusion localmente en nuestras computadoras, brindando al lector opciones y flexibilidad en la implementación. Se ha destacado la importancia de considerar los requisitos de hardware y las aplicaciones disponibles que pueden facilitar el proceso. Desde las soluciones más sencillas y accesibles, como Easy Diffusion, hasta la implementación detallada utilizando el código fuente en Python, este capítulo ha ofrecido una perspectiva detallada sobre cómo utilizar Stable Diffusion sin depender de servicios en la nube. La descripción de estas metodologías pone de relieve la versatilidad y adaptabilidad de Stable Diffusion, permitiendo a los lectores experimentar y explorar el modelo en su propia maquinaria, ya sea para fines educativos, de experimentación o de desarrollo.

PREÁMBULO: ¿QUÉ ES GOOGLE COLAB?

Objetivos del capítulo

El propósito de este capítulo es introducir al lector a Google Colab, un servicio esencial para el desarrollo y ejecución de modelos de aprendizaje profundo en la nube, como el Stable Diffusion. Comenzando con una descripción general y las razones para emplear esta herramienta, el capítulo se enfocará en explicar los aspectos básicos de su interfaz, la ejecución de código en bloques y la manipulación de texto no ejecutable. Adicionalmente, el capítulo también se adentrará en los entornos de ejecución, destacando las diferencias y beneficios de la utilización de CPU, GPU y TPU en el contexto de Google Colab. El objetivo es proporcionar una base sólida para aquellos que son nuevos en la plataforma, así como una referencia útil para aquellos que ya tienen experiencia en ella.

Ahora, antes de avanzar a la siguiente etapa del libro, donde veremos cómo construir nuestro modelo, necesitamos entender en términos simples qué es Google Colab y por qué estamos utilizando esta herramienta en este libro. Por ahora, en términos simples, aunque mencionamos anteriormente que íbamos a usar Stable Diffusion e instalarlo en nuestro sistema local, en realidad no lo estamos haciendo localmente, sino montando el modelo en un servicio en la nube.

Entonces, ¿cuál es el motivo de esto? ¿Por qué debemos utilizar un servicio en la nube para emplear el modelo cuando nuestro objetivo es tener un modelo de Stable Diffusion construido localmente en nuestra computadora? Aunque Google Colab sea un servicio en la nube, esto no significa que no podamos replicar los elementos y todo lo que se ejecute del modelo en nuestra computadora.

La principal razón para hacerlo es que tendremos la capacidad de replicar todo lo que veamos en Colab en cualquier dispositivo e incluso, si no deseamos utilizar

Google Colab, podremos replicar todo el código en nuestra computadora mediante un intérprete de Python.

Google Colab: Principios básicos de su interfaz

Ahora bien, es posible que algunas personas encuentren el uso de herramientas como Google Colab bastante sencillo; no obstante, asumiremos que la mayoría de los lectores de este libro tienen, al menos, cierto conocimiento previo y experiencia en programación, aunque no necesariamente en Python, sino en cualquier lenguaje de programación. Dado esto, es probable que muchos de los lectores ya hayan aprendido a programar en Python de manera local, es decir, utilizando los recursos computacionales de sus propias computadoras.

Por lo tanto, esta sección está dedicada a aquellos que no han tenido la oportunidad de utilizar Google Colab anteriormente. Aun así, si este fuera el caso y no se tuviera experiencia previa con Google Colab, realmente no debería presentar ningún problema, ya que el uso de los Notebooks y Google Colab en general es bastante simple y no requiere de demasiada formación al respecto. Si ya se cuenta con experiencia en el uso de Google Colab, entonces esta sección puede ser omitida, puesto que únicamente enseñará los elementos más básicos de esta herramienta para programar en Python en la nube.

El concepto de Google Colab es bastante sencillo: se trata de archivos que contienen bloques ejecutables en su interior. Estos bloques, por lo general, cumplen diversas funciones, pero su propósito principal es ser ejecutados por un intérprete de Python. En términos simples, podemos programar en Google Colab utilizando Python y ejecutar el código por bloques. Además, una de las funciones más importantes que ofrece Google Colab es la capacidad de manejar texto no ejecutable, como el formato markdown. Esto nos permite editar y dar formato al texto, utilizando notaciones que faciliten la comprensión de nuestro código.

Para acceder a la plataforma Google Colab, se puede utilizar cualquier motor de búsqueda. Primero, ingrese los términos "Google Colab" en el motor de búsqueda para localizar la página oficial y comenzar a utilizar esta herramienta. Aunque el enlace oficial de Google Colab es https://colab.research.google.com/, es más conveniente buscar la herramienta a través de un motor de búsqueda. Al ingresar a la página oficial de Colab, encontrará la pantalla de inicio, que se asemeja a la siguiente captura de pantalla:

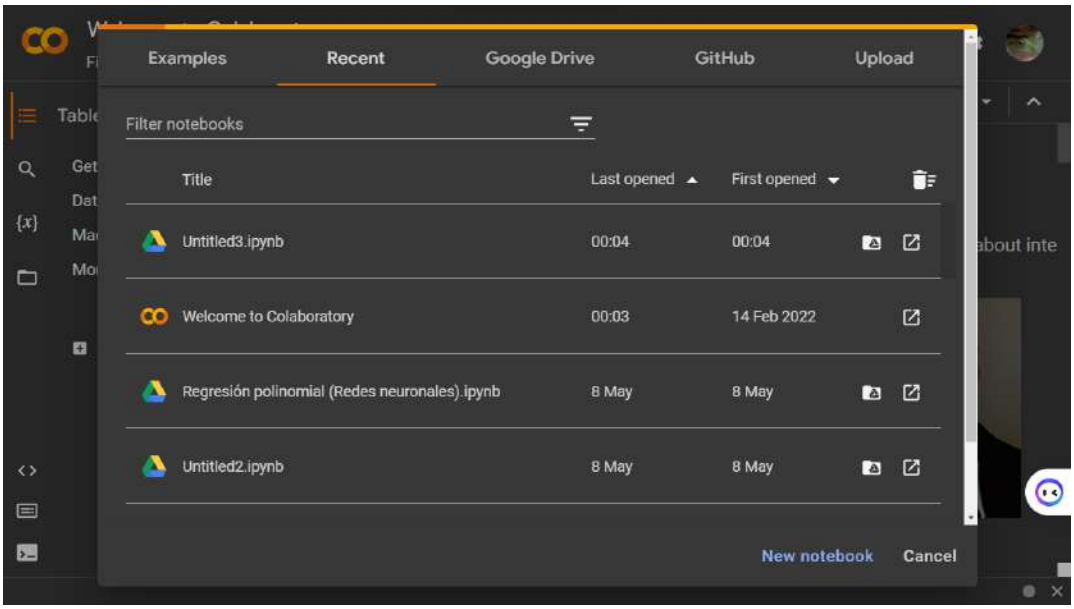


Ilustración 47 – Captura de pantalla de la página de inicio de Google Colab.

Una vez que ingresamos a esta plataforma, podemos comenzar a trabajar con ella. Como mencionamos previamente, las celdas de Google Colab se dividen en dos partes: la primera parte corresponde a las celdas que podemos utilizar para ejecutar código o comandos; por otro lado, el segundo tipo de celda es aquel en el que podemos agregar texto dentro de nuestro archivo. Este texto no se ejecuta, sino que sirve únicamente para tomar apuntes y notas relevantes sobre la ejecución del programa. A continuación, examinaremos con más detalle estos dos tipos de celdas y cómo podemos utilizarlas con la siguiente imagen:

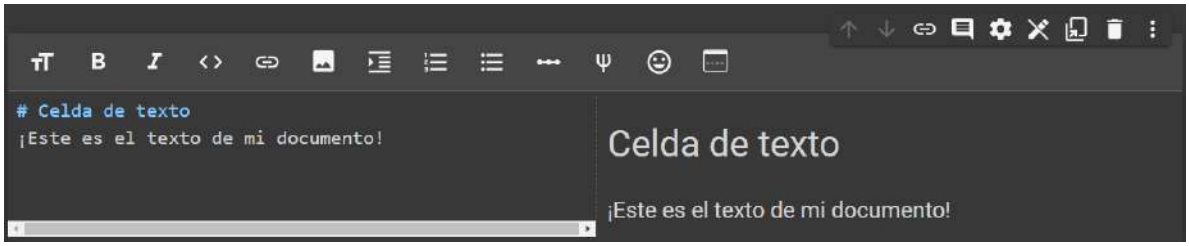


Ilustración 48 – Celda de texto en Google Colab, el código se escribe en Markdown.

Para ejecutar el código y los diversos comandos en este documento, el proceso es bastante simple. Lo único que debemos hacer es hacer clic en el botón de ejecutar. Si es nuestra primera vez utilizando Google Colab, la inicialización del sistema y la computadora remota en los servidores de Google puede tardar un poco. Para ejecutar un bloque, simplemente presionamos el botón en la esquina superior izquierda correspondiente a esta acción.

Es importante destacar que la posición de las celdas no influye significativamente en el orden de ejecución de los bloques. En cambio, el orden en que se ejecutan

se determina por el orden en que hayamos ejecutado las celdas. Podemos observar con más detalle lo mencionado anteriormente en la siguiente imagen:

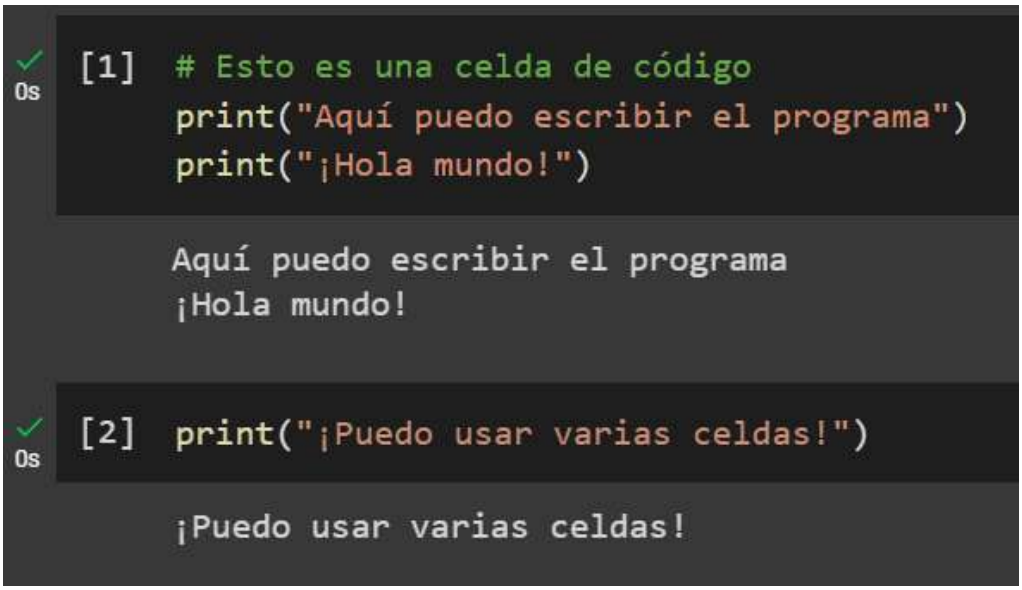


Ilustración 49 – Celda de código en Google Colab, el código está escrito en Python.

Así, podemos comenzar a escribir código utilizando Google Colab. Es importante destacar que el lenguaje de programación que emplea Google Colab es Python. Escribiremos código dentro de las celdas de la misma forma en que normalmente lo haríamos al ejecutar un archivo de Python; sin embargo, en lugar de archivos, utilizaremos celdas de código.

Google Colab: Entornos de ejecución

Una de las características más importantes que probablemente utilizaremos en el proceso de desarrollo con Google Colab como herramienta, es la capacidad que ofrece Stable Diffusion para controlar el tipo de entorno en el que estamos trabajando. Es decir, Google Colab nos brinda la posibilidad de definir cuál será el elemento de ejecución principal en el que se basará nuestro programa, ya sea, por ejemplo, una simple CPU, una GPU (tarjeta gráfica) o incluso el uso de elementos llamados TPUs (Unidades de procesamiento de tensores). Como es posiblemente conocido por muchos, los modelos de inteligencia artificial generalmente se ejecutan y entrenan mediante tarjetas gráficas, esto se debe principalmente a la velocidad que ofrecen estas unidades de procesamiento en comparación con las CPUs, por ejemplo.

Por defecto, la unidad de procesamiento predeterminada al utilizar Stable Diffusion es una CPU. Es decir, si no modificamos nuestro entorno de trabajo en Google Colab, nuestro programa se ejecutará principalmente con una unidad de procesamiento central de CPU. Sin embargo, cambiar el entorno de ejecución

dentro de Google Colab es bastante sencillo. A continuación, se presenta una imagen que ilustra los pasos a seguir para cambiar el entorno de ejecución en nuestro proyecto. En el caso de Stable Diffusion, se recomienda encarecidamente considerar el cambio del entorno de ejecución en Google Colab a uno basado en una tarjeta gráfica (o simplemente GPU).

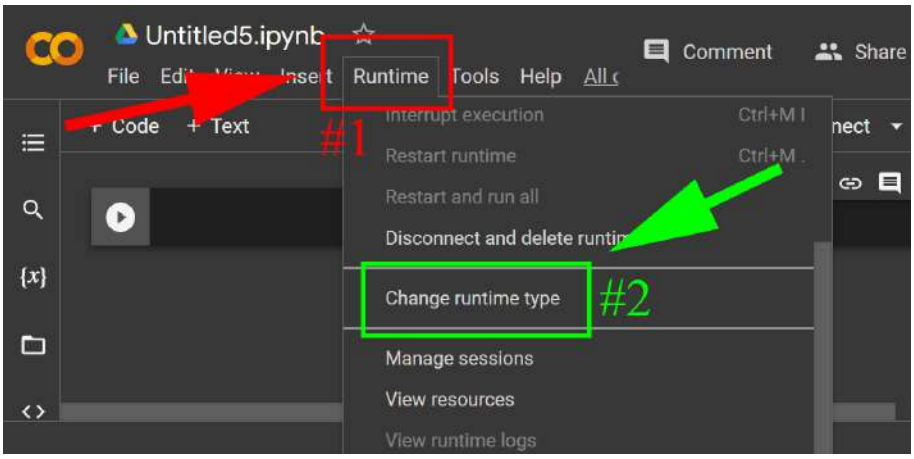
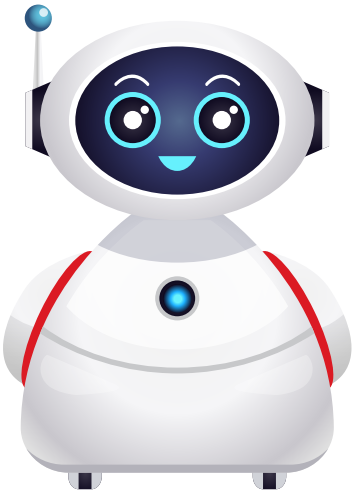


Ilustración 50 – Sección para campiar entorno de ejecución en Google Colab

Ahí se mostrará una ventana con diversas opciones desplegables, en la cual podremos seleccionar el entorno de ejecución que necesitamos. En este caso, corresponde al entorno de ejecución de GPUs, ya que estas unidades de procesamiento son bastante rápidas en el mundo de la inteligencia artificial.



Conclusión del capítulo

En este capítulo, hemos explorado a fondo Google Colab, una herramienta poderosa y versátil que facilita el desarrollo y la ejecución de modelos como Stable Diffusion en la nube. Hemos detallado los principios básicos de su interfaz, incluyendo cómo crear, editar y ejecutar bloques de código en Python, así como manipular texto en formato markdown. También hemos examinado los entornos de ejecución y cómo seleccionar el más adecuado para nuestras necesidades, especialmente en el contexto de aprendizaje profundo. Con esta introducción, el lector debe estar equipado con el conocimiento y la comprensión necesarios para utilizar Google Colab de manera eficiente en sus futuros proyectos, así como replicar el trabajo realizado en esta plataforma en su propio sistema local. La exploración de Google Colab establece una base sólida para los siguientes capítulos, donde el modelo Stable Diffusion será el enfoque principal.

STABLE DIFFUSION LOCAL: CONSTRUYENDO DESDE LA FUENTE

Objetivos del capítulo

El presente capítulo tiene como propósito explorar en profundidad el proceso de construcción del modelo Stable Diffusion desde la fuente. Los objetivos son claros y divididos en varias etapas para facilitar la comprensión. Se comenzará introduciendo el lector a la construcción local del modelo Stable Diffusion, destacando la flexibilidad y control que ofrece esta aproximación. A continuación, se abordará el acceso al código fuente a través de los repositorios disponibles, con énfasis en el de CompVis en GitHub. Se describirá el proceso de trabajo con Google Colab, explicando las ventajas de su uso en términos de accesibilidad y costos computacionales. Finalmente, se tratarán aspectos técnicos importantes como las diferentes versiones del modelo, los repositorios, y el acceso a información específica en el repositorio oficial de Stable Diffusion. La idea es proporcionar una guía completa y detallada para que los lectores puedan desarrollar su propio modelo desde cero y entender los conceptos subyacentes.

Ahora, examinaremos la otra forma de trabajar con Stable Diffusion. Esta vez, quizás no sea tan sencillo como la primera opción que vimos, donde simplemente necesitábamos instalar una aplicación compatible con el modelo o usar un cuaderno de Google Colab que nos permitiera experimentar de manera fácil con el modelo. En esta ocasión, intentaremos construir el modelo de Stable Diffusion desde el código fuente, es decir, utilizaremos los repositorios disponibles para desarrollar nuestro propio modelo desde cero. Esto podría llevar a la pregunta: ¿Cuál es la ventaja principal de hacer esto y por qué alguien querría emplear un modelo construido localmente en su computadora desde cero? En términos simples, esto significa que trabajaremos con la versión más pura del modelo y tendremos la oportunidad de utilizarlo de manera más flexible,

incluso modificando parámetros que generalmente no podríamos cambiar mediante una aplicación sencilla. Además, podremos comprender mucho más a fondo el funcionamiento complejo de este tipo de modelos de inteligencia artificial.

Como mencionamos anteriormente, para lograr esto necesitamos utilizar el código fuente del modelo Stable Diffusion. En ocasiones pasadas, hemos mencionado este modelo, pero no hemos tenido la oportunidad de trabajar con él hasta ahora. Existen diversas maneras de acceder al código fuente de Stable Diffusion, incluso dependiendo de la versión que queramos obtener. Por el momento, basta con tener en mente que podemos desarrollar este tipo de modelos recurriendo a la página oficial de GitHub de CompVis o utilizando el modelo e información almacenada en Hugging Face.

En este caso, utilizaremos el modelo de Stable Diffusion alojado en el repositorio de GitHub de CompVis. Sin embargo, el proceso sería el mismo o muy similar si se tratara de otro repositorio que contenga el modelo. Es importante mencionar que, en realidad, no trabajaremos directamente en nuestra computadora para los ejemplos. Como hemos señalado previamente, haremos uso de una herramienta llamada Google Colab. Esta herramienta permite programar en Python a través de celdas de código en la nube, lo que significa que toda la capacidad de cómputo será proporcionada por las computadoras en los servidores de Google.

Sin embargo, el uso de herramientas como Google Colab, que se explorará en este libro, tiene como objetivo principal facilitar el acceso al código y la implementación de este modelo a un mayor número de personas. Esto se debe, en gran parte, al elevado costo computacional asociado a este tipo de modelos de inteligencia artificial. Al utilizar una herramienta en la nube como Google Colab, podemos emplear las herramientas necesarias para trabajar con el modelo sin preocuparnos por si nuestra computadora cumple con los requisitos. Además, no solo se limita a nuestra computadora, sino que al utilizar Google Colab, también tenemos la oportunidad de programar y ejecutar el código en prácticamente cualquier lugar con un navegador web, incluso en nuestros propios dispositivos móviles.

A pesar de esto, es importante destacar que todos los procesos que veremos en los ejemplos donde usamos Google Colab pueden replicarse de igual manera y sin limitaciones en nuestras computadoras locales. Por supuesto, el uso de Google Colab tiene la intención de hacer que el proceso sea más fácil y con menos limitaciones en términos de hardware.

Ahora, es importante destacar otro aspecto relevante: los distintos tipos de cuentas disponibles en Google Colaboratory y los dispositivos que podemos utilizar para llevar a cabo este proceso. Generalmente, la versión gratuita de Google Colab nos brinda los recursos necesarios para entrenar el modelo; no obstante, es útil conocer que tenemos la opción de acceder a una capacidad

computacional aún mayor ofrecida por Google para ejecutar y entrenar nuestros modelos. Por el momento, para los ejemplos presentados en este libro, la versión gratuita del hardware proporcionado por Google Colab resulta más que suficiente.

Stable Diffusion local: ¿Qué debemos tener en cuenta?

Antes de trabajar directamente con Stable Diffusion en Google Colab, es crucial conocer de antemano las especificaciones e información necesarias para utilizar el modelo adecuadamente. Por el momento, debemos enfocarnos en aspectos relacionados con el uso del modelo, tales como determinar qué versión del modelo emplearemos en nuestro proyecto, comprender los distintos aspectos que podemos modificar dentro del mismo, evaluar si podemos alterar el proceso de entrenamiento y, además, considerar si es posible utilizar modelos pre-entrenados para agilizar este proceso. Aunque esto pueda parecer trivial, en realidad es un paso fundamental que nos permitirá ahorrar tiempo y esfuerzo en el futuro.

Versiones

Ahora, tal y como dijimos, empezaremos por ver uno de los primeros elementos que debemos tener en cuenta al momento de poder desarrollar el modelo por medio de Google Colab. El primer aspecto que debemos considerar es la versión que queremos usar de Stable Diffusion. Como sabemos, este modelo presenta diferentes versiones y cada una de ellas presenta mejoras o aspectos importantes a tener en cuenta del modelo. Realmente no podemos decir que debemos usar una versión en particular de nuestro modelo para obtener mejores resultados, todo depende de las necesidades que tengamos para poder usar el modelo.

Por ahora, debido a que realmente no presentamos una necesidad importante y que solamente haremos uso de Stable Diffusion para poder aprender el funcionamiento y los aspectos básicos más importantes de este mismo, entonces realmente no tendría sentido trabajar con una de las versiones más recientes del modelo, como lo es, por ejemplo, el uso de la versión v2.1 de Stable Diffusion.

Ahora, si tendremos la oportunidad de poder experimentar un poco con este modelo, pero solamente es importante tener en cuenta por ahora la versión que utilizaremos para poder trabajar con Stable Diffusion dentro de Google Colab. Para que el modelo y el proceso de entrenamiento no sea demasiado largo, haremos uso de la versión v1.5 del modelo. Dentro de esta versión, los resultados de las imágenes siguen siendo de muy buena calidad. Pero esto no significa que debamos tener en cuenta que obtendremos malos resultados, sino que debemos entender y usar una versión que nos permita comprender los conceptos básicos con los que contamos para poder usar este modelo.

Además, de alguna forma podemos decir que, si usamos una versión anterior del modelo, entonces el proceso de entrenamiento y el tiempo que nos llevará usar este tipo de modelos realmente será muy corto. Por lo que tendremos que esperar muy poco tiempo para poder volver a usar el modelo.

Repositorios

Otro aspecto que debemos considerar en este proceso es el repositorio que utilizaremos. Como vimos al inicio de este libro, tuvimos la oportunidad de revisar dos opciones: en primer lugar, el repositorio del modelo alojado en GitHub; y en segundo lugar, el repositorio dentro de la página de Hugging Face, donde encontramos las versiones actualizadas y actuales de los modelos. Para hacer este proceso más sencillo y familiar para la mayoría de los lectores, haremos uso del repositorio que se encuentra dentro de GitHub, específicamente dentro de CompVis. Aun así, esto no significa que no podamos hacer uso de los repositorios que se encuentran dentro de Hugging Face. De hecho, tendremos la oportunidad de utilizar ambos repositorios en el proceso de aprendizaje. Sin embargo, en este ejemplo en particular, especificaremos y haremos uso del repositorio de GitHub. Por lo tanto, realmente no debemos preocuparnos por el modelo almacenado dentro del repositorio de Hugging Face. Aunque podríamos utilizarlo, no lo haremos, al menos no por ahora.

Repositorio de GitHub

De momento, podemos explorar el menú principal que se encuentra dentro del repositorio. Para acceder al menú principal, basta con ingresar al GitHub oficial de Stable Diffusion en la página de CompVis en GitHub. Podemos hacer esto siguiendo el enlace https://github.com/CompVis/stable-diffusion. Una vez hayamos ingresado a la página principal, podremos observar los siguientes elementos:

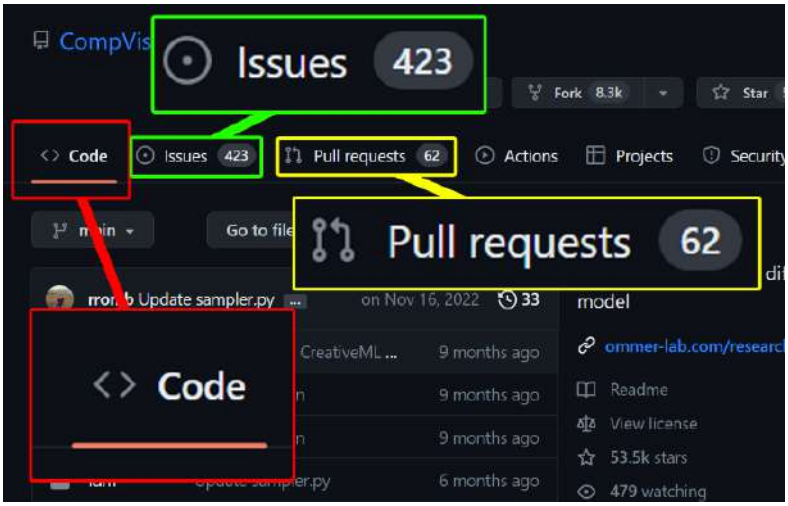


Ilustración 51 – Página principal de GitHub de Stable Diffusion (CompVis).

En esta imagen, podemos observar cómo se destacan los colores de las diferentes partes del repositorio, como los issues y una lista de los distintos pull requests que existen dentro del repositorio. Sin embargo, por ahora, podemos afirmar que ninguno de estos elementos nos llama la atención, ya que nuestro propósito principal al utilizar este modelo es aprender los conceptos básicos y saber cómo hacer uso de este.

Primero, navegaremos un poco más abajo en la página principal del repositorio de GitHub de Stable Diffusion para encontrar la información que necesitamos. Específicamente, buscamos detalles relacionados con los diferentes pasos introductorios que podemos seguir para utilizar Stable Diffusion.

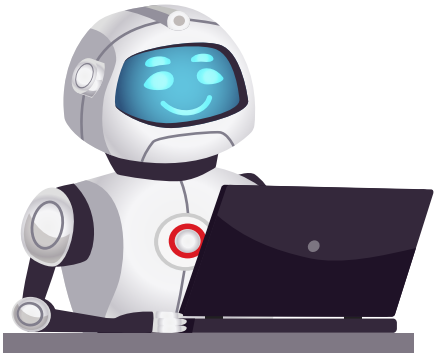
Podemos observar a qué nos referimos en la siguiente imagen:

Ilustración 52 – Sección de requerimientos del repositorio de Stable Diffusion.

En la captura de pantalla, podemos observar la sección que muestra los requisitos para utilizar el modelo. Más abajo, encontraremos diversos recursos relacionados con el uso del modelo, incluyendo una introducción y los pasos

principales para su implementación. A lo largo de este proyecto, la mayoría de los ejercicios y códigos que emplearemos serán proporcionados en este libro. Por lo tanto, quizás no necesitemos recurrir al repositorio para resolver todas nuestras dudas. Sin embargo, es útil saber que, si encontramos un error o tenemos una pregunta que no esté claramente explicada en el libro, podemos acudir al repositorio para obtener aclaraciones.

Una vez que tengamos una idea más clara sobre el repositorio y cómo utilizarlo, será momento de comenzar a programar y emplear Google Colab para implementar el modelo en nuestras computadoras. Es importante mencionar que, al usar Google Colab, todo se ejecutará en la nube de Google. No obstante, todos los procesos realizados en Google Colab también pueden replicarse localmente en nuestra computadora. Por ello, afirmamos que esta práctica trata sobre el uso de Stable Diffusion en nuestras computadoras locales. Aunque en los ejemplos emplearemos el servicio en la nube de Google Colab, todo este proceso puede llevarse a cabo de manera local en nuestras computadoras. Con esto en mente, es momento de comenzar a programar y trabajar directamente con el modelo de Stable Diffusion.



Conclusión del capítulo

En este capítulo, hemos profundizado en el proceso de construir el modelo Stable Diffusion desde la fuente, permitiendo una mayor comprensión y control sobre el mismo. Se ha abordado la ventaja principal de construir localmente el modelo, la accesibilidad al código fuente y la flexibilidad que ofrece trabajar con Google Colab. Se ha examinado detenidamente la elección de versiones y repositorios, así como el uso de herramientas en la nube que facilitan el proceso de implementación. Este enfoque brinda a los usuarios una manera de experimentar con el modelo sin las restricciones comunes en aplicaciones simplificadas y brinda una comprensión profunda del funcionamiento de la inteligencia artificial en este contexto. Con esto, los lectores están ahora equipados con el conocimiento y las herramientas necesarias para desarrollar y experimentar con su propia versión de Stable Diffusion, lo que refuerza la comprensión integral del modelo y sus aplicaciones prácticas.

ENTORNOS DE EJECUCIÓN

Objetivos del capítulo

El objetivo principal de este capítulo es proporcionar una comprensión completa de los entornos de ejecución en Google Colab, especialmente en el contexto de la implementación y el entrenamiento del modelo de Stable Diffusion. Se enfocará en los diferentes tipos de unidades de procesamiento, incluyendo CPUs, GPUs y TPUs, así como sus roles en la ejecución del modelo. Se abordará cómo configurar el entorno de ejecución en Google Colab, verificar las especificaciones técnicas, y clonar el repositorio necesario. También se explicará la importancia de seleccionar la unidad de procesamiento adecuada y los aspectos relacionados con la versión gratuita del servicio, y las opciones disponibles para aquellos que no tienen tarjetas gráficas Nvidia.

Ahora, antes de continuar con la siguiente etapa en nuestro proceso para implementar Stable Diffusion, es necesario considerar el entorno de ejecución adecuado para nuestro modelo. Como mencionamos previamente, Google Colab ofrece diversas opciones de entornos de ejecución según el modelo que deseemos utilizar o la tarea específica que queramos llevar a cabo en este entorno en la nube.

Los entornos de ejecución clave a considerar al emplear Stable Diffusion incluyen, por ejemplo, las CPUs (Unidades Central de Procesamiento), las GPUs (Unidades de Procesamiento Gráfico, también conocidas como tarjetas gráficas) y las TPUs

(Unidades de Procesamiento de Tensores). Estos últimos suelen ser de gran utilidad en diversos modelos de aprendizaje automático; sin embargo, en este caso particular, no haremos uso de estas unidades de procesamiento. Principalmente, emplearemos GPUs, también conocidas como tarjetas gráficas o simplemente unidades de procesamiento gráfico.

Para configurar el entorno de ejecución en Google Colab, primero debemos especificar qué tipo de unidad de procesamiento queremos que utilice el modelo.

Así, la plataforma se adaptará y nos proporcionará el hardware en la nube correspondiente a nuestras necesidades, aunque esto puede tener ciertas limitaciones, especialmente en la versión gratuita del servicio. Una de las principales restricciones al usar, por ejemplo, una GPU en Google Colab, es que, aunque obtenemos una tarjeta gráfica potente, nuestro rendimiento en la CPU podría verse afectado negativamente.

Después de todo, de no ser así, no tendría mucho sentido emplear esta división de las unidades de procesamiento principales disponibles en Google Colab para aplicar en nuestros modelos. Si todos los usuarios exigieran el máximo rendimiento, simplemente se proporcionarían dispositivos en la nube capaces de realizar cualquier tipo de operaciones, similares a una supercomputadora. Esto eliminaría la necesidad de especificar qué entorno de ejecución requerimos, lo que teóricamente nos ahorraría tiempo, ya que la CPU es la unidad de procesamiento predeterminada proporcionada por Google Colab.

Para cambiar el tipo de entorno de ejecución que deseamos utilizar en Google Colab, simplemente debemos seguir una serie de pasos sencillos. Esto implica cambiar el tipo de entorno de ejecución mediante la interfaz gráfica de Google Colab.

Podemos observar con más detalle a qué nos referimos en la siguiente captura de pantalla, donde se muestran los diferentes elementos que podemos utilizar para modificar nuestro entorno de ejecución. En este caso, utilizaremos la función indicada en esta imagen.

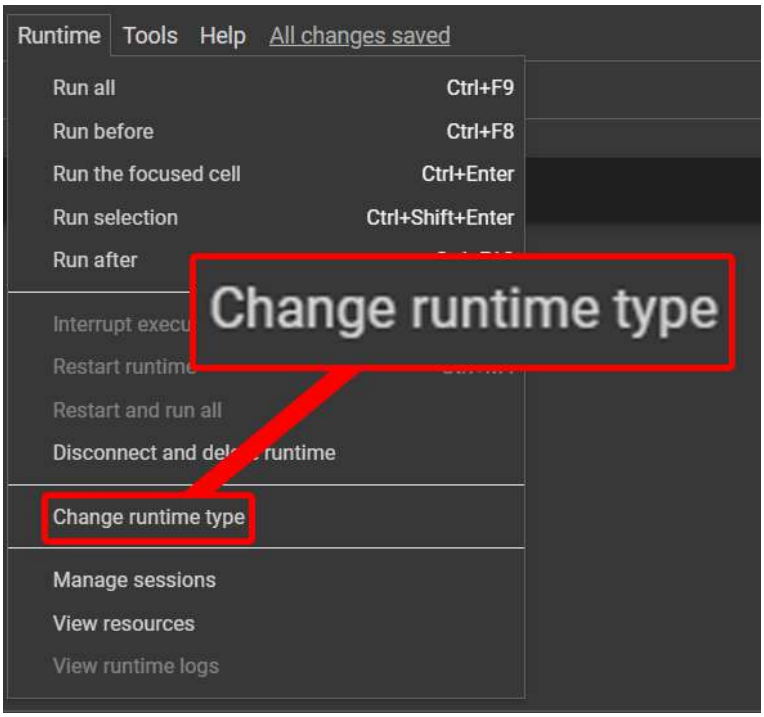


Ilustración 53 - Opciones de configuración del entorno de ejecución en Google Colab.

Una vez dentro de esta ventana, se nos pide que especifiquemos el tipo de entorno de ejecución que queremos emplear en nuestra sesión de Google Colab. Entre las diferentes opciones disponibles, como CPU, GPT y TPU, debemos hacer la elección del entorno de ejecución que deseamos utilizar. Podemos ver las distintas opciones presentadas en esta ventana en la siguiente imagen a continuación:

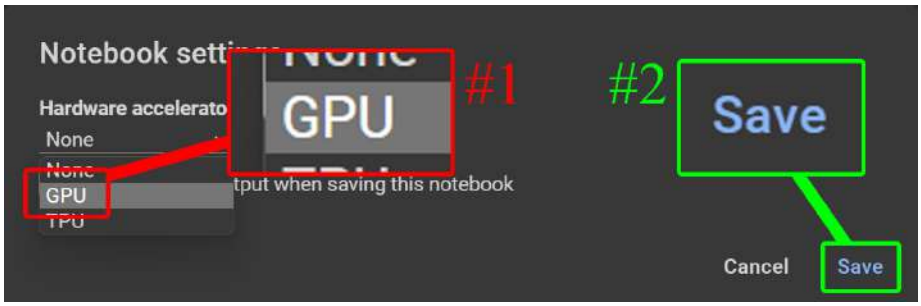


Ilustración 54 – Selección de unidad de procesamiento en Google Colab.

En este caso específico, nuestro objetivo es aprender cómo funciona Stable Diffusion y cómo utilizarlo localmente en sus distintas variantes. Para ello, es suficiente señalar que queremos usar GPUs, ya que son la unidad de procesamiento principal en este tipo de ejemplos. Una vez que indiquemos a nuestro programa el uso de las GPUs, nuestro cuaderno de Colab se inicializará con tarjetas gráficas capaces de realizar procesos más complejos relacionados

con gráficos y otras técnicas de cómputo, como CUDA. Por esta razón, utilizaremos este entorno de ejecución para entrenar y ejecutar nuestro modelo de Stable Diffusion.

Una vez que hayamos habilitado y asignado la opción para utilizar GPUs en nuestro entorno, es necesario confirmar que, efectivamente, estemos usando la GPU dentro de nuestro ambiente y no cualquier otro componente principal de procesamiento. Esto se debe a que la mayoría (o todas) las GPUs que podemos usar en Google Colab son de la marca Nvidia. Por lo tanto, podemos verificar que nuestra tarjeta gráfica esté activa con la siguiente línea de comando:

```
1 !nvidia-smi
```

Una vez ejecutado este comando, nos aparecerá una ventana que nos muestra la información de la tarjeta gráfica que estamos utilizando en nuestro entorno. Al mismo tiempo, podremos observar las diferentes especificaciones técnicas de dicha tarjeta gráfica, lo que nos permitirá evaluar si estas características son adecuadas o no, según el modelo que vayamos a emplear.

```
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 525.85.12    Driver Version: 525.85.12    CUDA Version: 12.0     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================+======================|
|   0  Tesla T4            Off  | 00000000:00:04.0 Off |                    0 |
| N/A   53C    P8    10W /  70W |      0MiB / 15360MiB |      0%      Default |
|                               |                      |                  N/A |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
|  No running processes found                                                 |
+-----------------------------------------------------------------------------+
```

Ilustración 55 – Pantalla de salida del comando nvidia-smi en Google Colab.

Generalmente, aunque las personas puedan pensar que debido a que nuestra cuenta y entorno de ejecución de Google Colab son gratuitos, esto implica un bajo rendimiento en nuestro entorno, realmente, esta idea no es del todo cierta. A pesar de que el servicio que estamos utilizando en Google Colab corresponde a un entorno gratuito, esto no significa que Google nos vaya a proveer con hardware en la nube de muy bajo rendimiento. Contrario a esta creencia, la realidad es que el rendimiento que nos ofrece Google para ejecutar nuestros

modelos es de muy buena calidad y se trata de hardware moderno y de alto rendimiento.

Por lo tanto, en términos de capacidad computacional, podemos afirmar que no debemos preocuparnos en exceso. Al menos para el uso de Stable Diffusion y el aprendizaje de este modelo con el fin de comprender su funcionamiento, es más que suficiente utilizar las especificaciones técnicas de nuestro entorno sin necesidad de requerir más recursos.

Ahora, otro aspecto que también debemos destacar en todo este proceso es el hecho de que el uso de la tarjeta gráfica como unidad de procesamiento principal dentro de los modelos de Stable Diffusion realmente no se trata de una necesidad que deba ser cumplida a toda costa. De hecho, en la actualidad existen modelos que han sido entrenados y optimizados para ser ejecutados únicamente dentro de la CPU del computador.

Sin embargo, no por eso debemos desprestigiar el rendimiento tan grande y la facilidad que ofrecen las tarjetas gráficas. Si bien es cierto que es recomendable dejar nuestra tarjeta gráfica como unidad de procesamiento principal, también es importante tener en cuenta que, generalmente, este tipo de modelos, o al menos en el caso particular de Stable Diffusion, gran parte de la comunidad se ha encargado de optimizar los procesos para que se lleven a cabo sin usar una tarjeta de video de Nvidia.

Esto no solo es beneficioso para las personas que no cuentan con una tarjeta gráfica de alta calidad, sino que también permite y facilita el uso de estas herramientas para aquellos individuos con una tarjeta gráfica que, aunque pueda ser potente, quizás no sea completamente compatible con el modelo de inteligencia artificial. Esto se debe principalmente a que estos modelos suelen utilizar CUDA (un elemento exclusivo de las tarjetas gráficas de Nvidia). Por lo general, las personas que utilizan otros tipos de tarjetas gráficas, como las AMD Radeon, no pueden aprovechar fácilmente estos modelos. Saber que podemos emplear distintos elementos de procesamiento además de una tarjeta gráfica resulta bastante reconfortante.

En el caso hipotético de que esta situación nos aplique, también podemos verificar cuál es la unidad de procesamiento principal que estamos utilizando en nuestro entorno de Google Colab. Dado que no estaremos empleando una tarjeta gráfica en este caso, el comando que vimos anteriormente no resulta del todo útil para nuestra situación. Para determinar qué CPU estamos utilizando dentro de nuestra sesión de Google Colab, podemos ejecutar la siguiente línea de comando que se muestra a continuación:

```
1 !sudo apt-get install neofetch && neofetch
```

Al ejecutar este comando, obtendremos una salida similar a la siguiente. Esto se debe principalmente a que estamos utilizando el entorno de Google Colab dentro de un servidor que emplea Ubuntu. Por lo tanto, la salida y el comando que acabamos de utilizar corresponden a una terminal de Ubuntu.

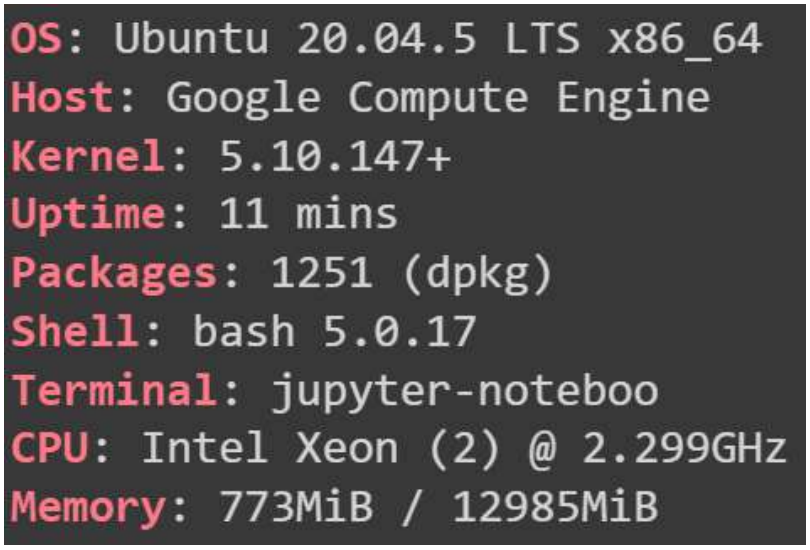


Ilustración 56 – Captura de pantalla de la salida de Neofetch en Google Colab.

En esta ventana, podremos encontrar información diversa relacionada con la computadora que estamos empleando. Dentro de esta información, también podemos observar la descripción específica de la CPU en uso. En el caso de la ilustración, podemos ver que la CPU que estamos usando es un "Intel Xeon (2) @ 2.299GHz".

Configuración: Clonación de repositorio

Ahora, es momento de ver cómo podemos clonar el repositorio dentro de nuestro entorno de Google Colab. Hasta el momento, solamente hemos realizado algunos pasos básicos para configurar nuestro entorno, como por ejemplo, configurar la unidad de procesamiento principal. Sin embargo, debido a que realmente necesitamos utilizar el modelo y, dado que no existen datos del modelo en nuestro entorno de Google Colab, es importante clonar el repositorio dentro de nuestro sistema de Colab. Pero, ¿cómo podemos hacer esto? ¿Y qué es exactamente clonar un repositorio? Es probable que esta sección genere muchas dudas y preguntas, pero no debemos preocuparnos, ya que son conceptos sencillos.

El proceso de clonar el repositorio solo tendremos que hacerlo una vez, la primera vez que ejecutemos nuestro entorno de Google Colab.

Sin embargo, esto no significa que los datos y el repositorio se queden almacenados; en cambio, debemos entender que cada vez que nuestro programa termine su ejecución y entorno, todos los datos relacionados serán eliminados. Esto significa que, cuando deseemos volver a utilizar el modelo después de que el entorno haya sido terminado, tendremos que repetir este proceso.

Hay varias formas de lograr que los datos del repositorio de Stable Diffusion estén dentro de nuestro entorno de Google Colab, cada uno puede variar dependiendo de las necesidades individuales de la persona que desea usar el modelo. Por eso, veremos las diferentes formas en las que podemos clonar el repositorio dentro de nuestro entorno de Google Colab de manera rápida y sencilla, con la intención de no perder mucho tiempo en esta tarea. Realmente no hay una forma de decir que un método sea mejor que otro; cualquiera de los que veremos a continuación servirá para cumplir con la tarea final que queremos realizar, que es clonar el repositorio.

Sin embargo, aún no hemos resuelto una duda, que podría parecer trivial para algunos, pero realmente no podemos partir del supuesto de que todas las personas cuentan con la misma experiencia trabajando con repositorios de código abierto. Por ahora, en términos simples, podemos decir que un repositorio es solamente un conjunto de archivos que se mantienen actualizados a medida que un proyecto dentro de un repositorio se actualiza. Es como si se tratara de una carpeta con muchos archivos, donde estos archivos son los recursos libres del programa.

La idea detrás de un repositorio es que generalmente se encuentran en la nube y permiten a muchos usuarios cooperar en el desarrollo de un proyecto, haciendo que cada uno añada, modifique, elimine y administre los diferentes archivos del proyecto. Para poder ofrecer este servicio de cooperación, los repositorios se encuentran en servidores externos a los que los diferentes integrantes de cada equipo pueden acceder y luego intentar hacer solicitudes de cambio. Para que cada usuario y participante del proyecto pueda colaborar, debe contar con una copia actualizada de los archivos del repositorio en su computadora, de tal forma que un programa se encargue de escanear dentro de esta copia los diferentes cambios que se hayan hecho en los documentos con el tiempo. Esto se conoce generalmente como control de versiones.

Con esto en mente, para resumir, el proceso de clonar un repositorio es simplemente guardar una copia en nuestra computadora local, de tal forma que podamos modificarla sin que el repos itorio original se vea directamente afectado.

Es como si tuviéramos un montón de archivos dentro de una carpeta en la nube, luego descargáramos esa carpeta con todos sus archivos dentro de nuestra computadora local. Podríamos modificar los archivos y hacer lo que quisiéramos

con ellos, pero incluso así, los archivos originales dentro del repositorio no tendrían ningún tipo de cambio.

Entonces, al clonar el repositorio en nuestro entorno de Google Colab, estamos obteniendo una copia de todos los archivos del proyecto para trabajar con ellos. Dependiendo de nuestras necesidades, elegiremos el método más adecuado para clonar el repositorio y así poder utilizar el modelo de Stable Diffusion en nuestro entorno de Google Colab de manera eficiente y efectiva.

Conclusión del capítulo

Este capítulo ha dado una visión detallada y práctica de cómo preparar y configurar los entornos de ejecución en Google Colab para trabajar con Stable Diffusion. Se han examinado las opciones disponibles, las limitaciones y los beneficios de cada tipo de unidad de procesamiento, y se ha presentado un método paso a paso para configurar y verificar el entorno. También se discutió cómo clonar el repositorio dentro de Colab, una tarea esencial para trabajar con el modelo. La información presentada en este capítulo establece una base sólida para cualquier persona que quiera implementar y experimentar con Stable Diffusion, asegurando que se aproveche al máximo el hardware en la nube proporcionado por Google Colab.

DESCARGA DEL REPOSITORIO

Objetivos del capítulo

El objetivo de este capítulo es describir dos métodos esenciales para clonar el repositorio en el entorno de Google Colab, con énfasis en la vinculación de Google Drive y la subida manual. Se busca proporcionar una comprensión detallada de cómo estos métodos interactúan con el entorno de Google Colab y cómo se pueden aplicar en el contexto del trabajo con el repositorio de Stable Diffusion. Los lectores aprenderán los pasos necesarios para vincular su Google Drive, cómo subir archivos manualmente, y las ventajas y desventajas de ambos métodos. Las ilustraciones y ejemplos prácticos se incluyen para ayudar en la comprensión del proceso.

Ahora, antes de poder empezar a trabajar con el modelo en sí y explorar cómo podemos utilizar el repositorio para llevar a cabo las diferentes tareas que se pueden realizar con Stable Diffusion, es importante recordar lo que mencionamos anteriormente sobre qué son los repositorios. Entonces, es momento de aprender cómo podemos descargarlos de tal forma que podamos utilizarlos en nuestro programa. Realmente, podemos decir que este paso es uno de los más sencillos. Aunque existen dos de los diferentes métodos que usaremos para importar este repositorio en nuestro entorno, podemos decir que el tercer método que veremos no requiere descargarlo. Sin embargo, en esta sección veremos cómo podemos descargarlo en caso de que deseemos utilizar el repositorio en nuestro entorno de Google Colab usando alguno de los dos primeros métodos.

Descargar el repositorio es realmente bastante sencillo. Lo único que tenemos que hacer para descargarlo es dirigirnos al repositorio oficial de Stable Diffusion de CompVis. Luego, veremos un botón que nos permite clonar el repositorio bajo el nombre de "code", luego, tendremos que hacer clic a la opción de "Descargar como zip", lo cual descargará un archivo .zip con el repositorio dentro, podemos ver la opción de descarga con la siguiente imagen:

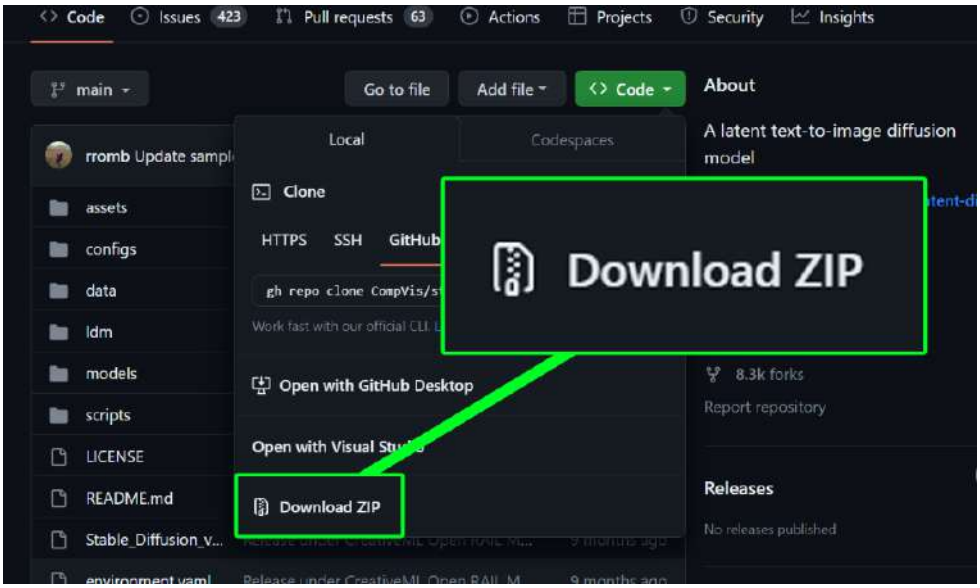


Ilustración 57 – Captura de pantalla de la opción para descargar un repositorio como archivo Zip.

Con esto, un archivo con extensión .zip se descarga en nuestro sistema. Este archivo comprimido puede ser descomprimido manualmente en nuestras computadoras. Sin embargo, no realizaremos este paso por ahora, ya que nuestro objetivo es automatizar este proceso mediante código en nuestro entorno de Google Colab. Por el momento, dejaremos el archivo .zip descargado en nuestro sistema para importarlo posteriormente a uno de los dos métodos que explicaremos en Google Colab.

De esta manera, ya tenemos descargado el repositorio en nuestra computadora local. Aunque la idea principal es utilizar este repositorio en nuestro entorno de Google Colab, tenerlo en nuestra computadora local puede ser útil para importarlo fácilmente en Google Colab más adelante. No obstante, como se mencionó al inicio, si no podemos o no deseamos descargar este repositorio en nuestra computadora local, y preferimos hacerlo todo dentro de Google Colab, también ofreceremos un método para descargarlo directamente en ese entorno.

Primer método: Google Drive

El primer método que utilizaremos para clonar el repositorio en nuestro código será el de vinculación de una cuenta de Google Drive en nuestro entorno de

Google Colab. Como podemos deducir por el nombre, ambas herramientas fueron desarrolladas por Google, lo que crea un pequeño ecosistema de archivos que podemos aprovechar para nuestros proyectos.

Para las personas que no estén familiarizadas con conceptos como Google Drive, en términos simples, Google Drive es un servicio en la nube que nos permite guardar archivos, que pueden ser de cualquier tipo. Además de guardarlos, también nos permite compartir dichos archivos con otras personas, quienes pueden interactuar con ellos, descargarlos, modificarlos, comentarlos, etc. Todo dependerá de cómo configuremos nuestro entorno de Google Drive.

Ahora, ¿qué relación puede tener Google Drive con nuestro entorno de Google Colab? Es una muy buena pregunta, pero la respuesta es más sencilla de lo que parece: podemos subir todo lo que se encuentre en nuestro Google Drive a Google Colab, permitiéndonos hacer uso de los diferentes archivos que encontramos en nuestra unidad de Drive. Además, tenemos la oportunidad de importarlos en nuestro sistema o incluso realizar una copia temporal de ellos.

Usar Google Drive es muy sencillo y el único requisito para hacerlo es contar con una cuenta de Google. Para vincular nuestra unidad de Google Drive con nuestro proyecto de Google Colab, simplemente debemos, como primer paso, dirigirnos a la sección izquierda de nuestro Google Colab. Allí encontraremos un ícono con forma de carpeta, como podemos ver a continuación en el siguiente gráfico:

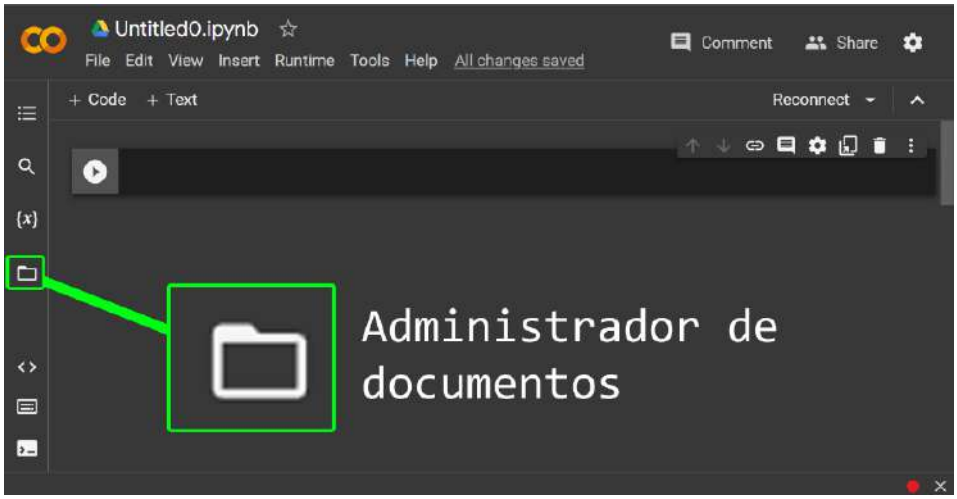


Ilustración 58 – Ubicación del administrador de documentos en Google Colab.

Esta sección nos permite explorar los diferentes archivos que tenemos dentro de nuestro entorno de Google Colab, el cual, como hemos repetido anteriormente, es solamente una computadora que se encuentra en la nube proporcionada por Google. También podemos ver el espacio que tenemos disponible, tendremos la oportunidad de incluso crear nuevos archivos y administrar los que ya tenemos dentro de nuestro entorno. En esta sección tenemos también las diferentes

opciones que podemos usar, de tal forma que podemos vincular nuestra cuenta de Google Drive. Así, cada vez que necesitemos usar un archivo dentro de nuestra unidad de Drive, entonces, podremos importarlo fácilmente a nuestro Google Colab.

Para vincular nuestra cuenta de Drive, simplemente debemos hacer clic en el ícono de Google Drive después de haber pulsado en el ícono del administrador de archivos, tal como se muestra a continuación:

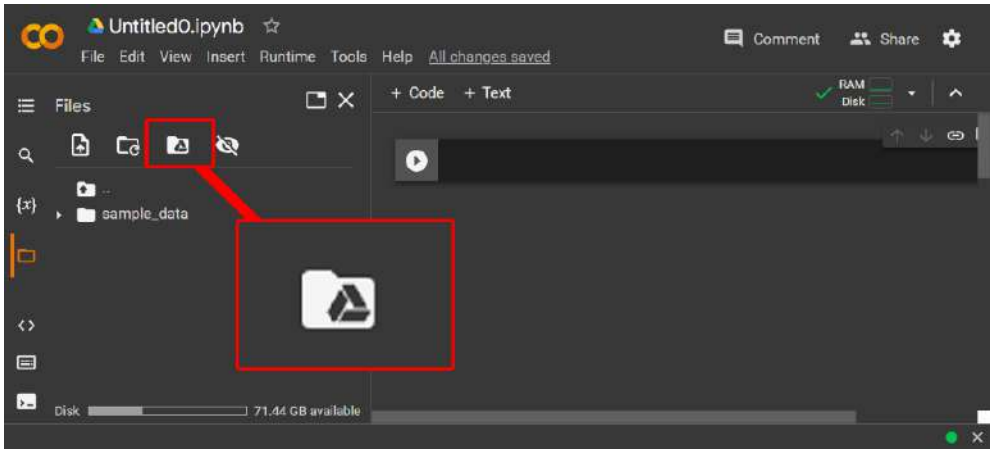


Ilustración 59 – Ícono de vinculación de cuenta de Google Drive.

Una vez que hagamos clic en esta sección, se nos solicitará seleccionar una cuenta de Google Drive y otorgar los permisos necesarios para que Google Colab pueda acceder a nuestra unidad de Drive. Después de conceder todos los permisos requeridos, nuestro entorno de Google Colab se vinculará con nuestra unidad de Google Drive.

Esto creará una nueva carpeta en nuestro administrador de documentos, como se puede observar a continuación:

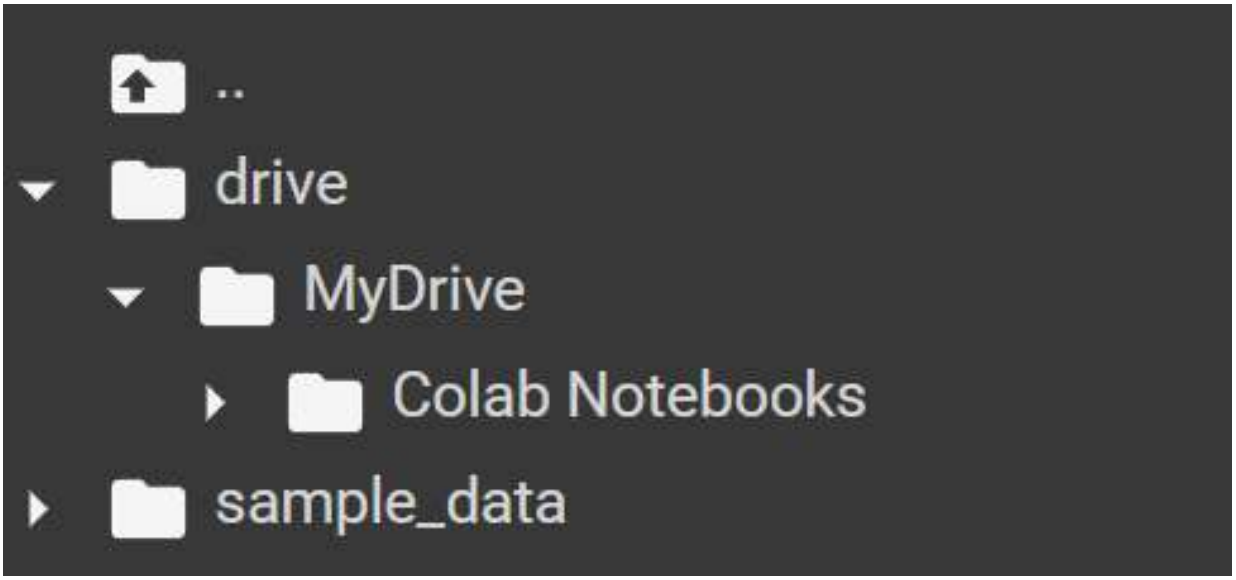


Ilustración 60 – Nueva carpeta creada en Google Colab.

¡Y listo! En el caso de la ilustración, podemos ver que realmente el espacio de Google Drive está vacío. Sin embargo, debemos tener en cuenta el hecho de que dentro de la carpeta "MyDrive" aparecerán todos los elementos que tengamos guardados en nuestra cuenta de Google Drive que hayamos vinculado con nuestro entorno de Colab. Con esto, finalmente, podemos decir que ya tenemos nuestro entorno de Google Drive vinculado a nuestro Google Colab, lo cual permitirá acceder a todos los datos de nuestra unidad fácilmente, es decir, podremos guardar dentro de nuestra unidad todos los archivos del repositorio. Luego, podremos importar estos mismos archivos dentro de nuestro Google Colab para poder usarlos en nuestros proyectos con el modelo.

Ya que finalmente contamos con nuestro entorno de Google Drive dentro de nuestro Google Colab, es el momento de guardar en nuestro Google Drive el archivo que previamente tuvimos la oportunidad de guardar como .zip. Es decir, el archivo comprimido que contiene todos los archivos del repositorio de Stable Diffusion. Ahora, con esto, podemos ver que cobra un poco más de sentido el paso que realizamos anteriormente, al guardar el repositorio en nuestra computadora localmente para después importarlo a nuestro entorno de Google Colab y poder utilizarlo. El proceso realmente no es demasiado complejo, solamente debemos encargarnos de subir el archivo .zip a nuestra unidad de almacenamiento de nuestra cuenta de Google Drive.

Para ello, primero debemos ingresar a la página oficial de Google Drive. Esto lo podemos hacer simplemente indicando en nuestro motor de búsqueda los términos "Google Drive" o ingresando al enlace oficial de este servicio de Google, el cual es https://drive.google.com. Con esto, podremos acceder a nuestra unidad

de almacenamiento y guardar los archivos que consideremos necesarios en nuestro entorno, de tal forma que luego podamos usarlos dentro de nuestro Google Colab.

Una vez que tengamos todos estos archivos en nuestra unidad de Google Drive, solamente debemos ingresar a la carpeta "MyDrive" que se formó dentro de nuestro entorno de Google Drive, como vimos anteriormente en la captura de pantalla de la ilustración.

Con esto, podemos ver que todos los archivos que subamos a nuestro Google Drive serán subidos directamente a esta carpeta de nuestro entorno de Google Colab, tal como podemos observar a continuación.

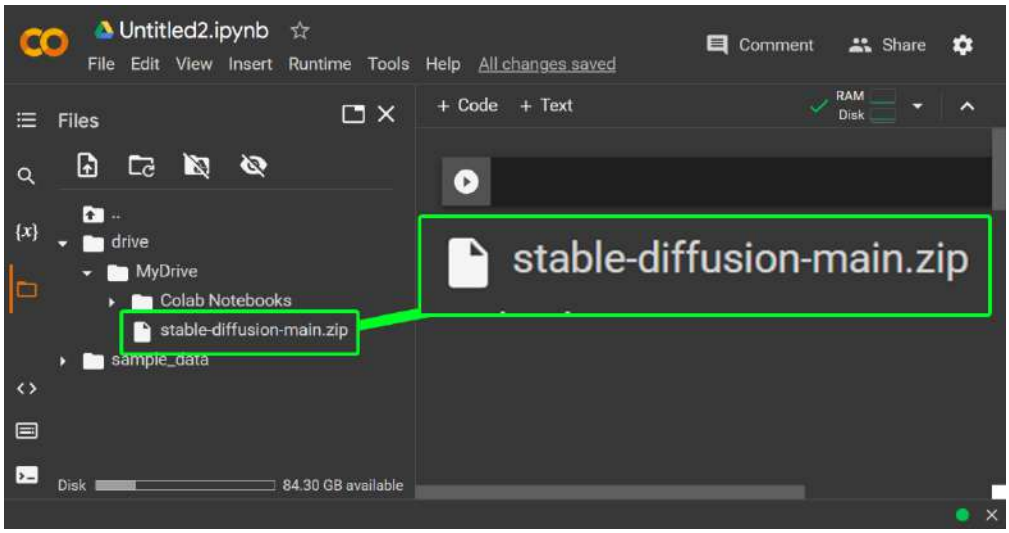


Ilustración 61 – Repositorio como archivo .zip en una unidad de Google Drive accediendo desde Google Colab.

Y con esto, ahora podemos decir que finalmente tenemos el repositorio dentro de nuestro entorno de Google Colab. Con esto, podremos hacer todas las pruebas que consideremos necesarias para que nuestro entorno funcione adecuadamente. Pero por ahora, lo único que hemos hecho es tener que pasar el archivo .zip del repositorio dentro de nuestro Google Colab. Sin embargo, todavía no hemos tenido la oportunidad de descomprimir todos los archivos que se encuentran dentro de este archivo .zip. Tendremos la oportunidad de hacer esto más adelante. Por ahora, solamente debemos preocuparnos por indicar a nuestro programa que ya tiene dentro de sí los archivos necesarios para usar el repositorio. En secciones posteriores, veremos cómo podemos descomprimir este archivo .zip. Esto tiene la ventaja de que nuestra unidad de Google Drive no se vea afectada por los varios cambios que haremos en la unidad que se encuentra vinculada a nuestro entorno de Google Colab.

Segundo método: Subida manual

Ahora, veremos otro método que podemos utilizar para subir nuestro archivo .zip, que contiene el repositorio descargado previamente de Stable Diffusion en GitHub. Podemos decir que este método es más sencillo y no requiere vincular nuestra cuenta de Google Drive, como hicimos antes. Sin embargo, este método sí requiere que el repositorio esté guardado como un archivo .zip en nuestra computadora, lo cual presenta ventajas y desventajas.

La primera ventaja que podemos mencionar es que, de esta manera, trabajamos de forma local sin depender de los servicios de Google. Esto significa que nuestro proceso para manejar el repositorio y los diferentes archivos dentro de él será más autónomo y nos permitirá tener un mayor control. Además, nos ahorramos el paso de vincular nuestra unidad de Google Drive a nuestro entorno de trabajo de Google Colab, como hicimos con el método anterior.

Sin embargo, hay algunas desventajas que debemos tener en cuenta al usar este método. Por ejemplo, necesitamos contar con el archivo .zip guardado en nuestra computadora, mientras que el método anterior nos permitía hacer uso del archivo .zip del repositorio sin necesidad de esto, al almacenarlo en nuestra unidad de Google Drive.

La ventaja de utilizar el repositorio sin necesidad de guardarlo en nuestra computadora y optar por los servicios en la nube es realmente positiva, ya que podemos acceder al repositorio desde cualquier lugar, siempre y cuando tengamos acceso a internet, incluso desde un dispositivo móvil. Además, el hardware y almacenamiento de nuestra computadora no se verían comprometidos, pues todo ocurre internamente.

Con esto en mente, el paso que debemos realizar es muy sencillo: simplemente debemos indicar a nuestro entorno de Google Colab que queremos importar un archivo, tal y como podemos ver a continuación:

```
1  from google.colab import files
2  repositorio_zip = files.upload()
```

Esto hará que, al ejecutar la celda de código, aparezca un botón que podemos presionar, el cual nos ofrecerá la opción de importar el archivo dentro de nuestro sistema. De esta manera, el archivo quedará guardado en la memoria de nuestro entorno de ejecución de Google Colab, en este caso, bajo el nombre de la variable "repositorio_zip". Así, cada vez que nos refiramos a este nombre, haremos referencia al repositorio en sí. Podemos observar a qué nos referimos con el siguiente gráfico, donde podemos ver que nos aparece una opción con un botón que nos permite seleccionar un archivo. Este archivo lo podremos seleccionar mediante el administrador de documentos de nuestra computadora, como es

habitual. En este caso, elegiríamos la carpeta del repositorio que descargamos previamente.

Ilustración 62 – Uso de la función upload() del entorno de Google Colab.

También existen diversas formas adicionales que podemos utilizar para subir archivos en nuestro entorno de Google Colab. Por ejemplo, el propio Colab ya nos proporciona una opción para realizar esta acción sin necesidad de emplear el bloque de código que utilizamos anteriormente.

Para realizar esto, el paso que debemos seguir es bastante sencillo, que consiste simplemente en subir el archivo mediante el administrador de documentos en nuestro entorno de Google Colab. Este botón se muestra como un ícono de archivo con un símbolo de flecha en el medio. Podemos entender mejor a qué nos referimos y ver una indicación del botón en cuestión con la siguiente captura de pantalla:

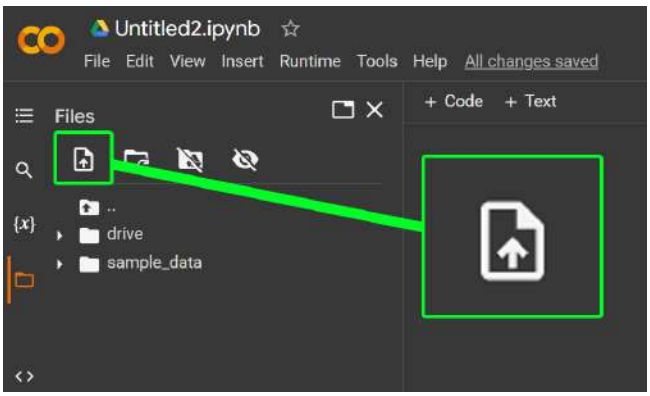


Ilustración 63 – Opción nativa de subida de archivos en Google Colab

Así pues, el proceso para subir el archivo será exactamente el mismo que hicimos anteriormente; es decir, ambos métodos realizan la misma tarea, la cual es solamente subir este archivo dentro de nuestro entorno de Google Colab. Por lo tanto, podemos decir que no importa mucho cuál de las dos formas estemos usando para poder subir el archivo, ya que realmente ambos se encargan de realizar exactamente la misma tarea. La diferencia radica en el hecho de que uno se realiza gracias a la capacidad nativa de Google Colab para importar archivos, mientras que el otro también aprovecha esta capacidad. Sin embargo, la principal diferencia es que el proceso ocurre por medio de una línea de código en la terminal y no a través de la interfaz gráfica, pero ambos cumplen la misma función.

Podemos confirmar que, efectivamente, ambos métodos han sido cargados en nuestro entorno de Google Colab. Si al momento de utilizar el administrador de archivos podemos ver que el archivo se encuentra en la primera página que podemos observar en este mismo, tal y como podemos ver en la siguiente imagen:

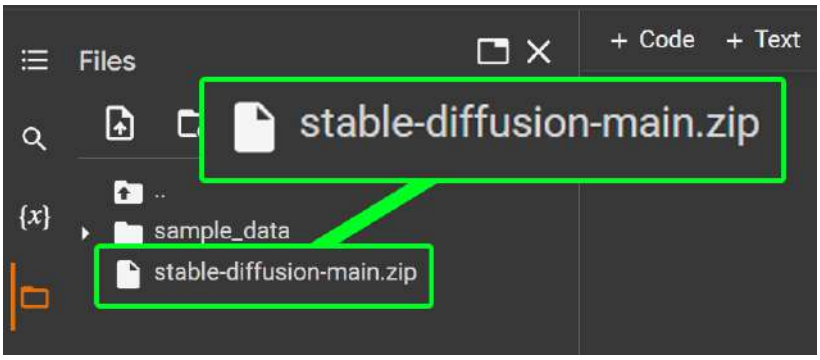


Ilustración 64 – Archivo del repositorio .zip comprimido dentro de nuestra ventana principal de archivos de Google Colab

Con esta información, ahora podemos aprender cómo subir el archivo del repositorio a nuestro entorno de Google Colab de manera sencilla. Así, tendremos todos los archivos necesarios para trabajar con el repositorio. Recordemos que, para utilizar este método, es necesario tener el archivo en nuestra computadora local y que, al finalizar cada sesión de Google Colab, los archivos dentro de dichas sesiones serán eliminados.

Tercer método: Clonación directa

Ahora, finalmente, abordamos el tercer método que podemos utilizar para acceder a este repositorio. Podemos decir que este método es probablemente el más sencillo de los que hemos visto hasta ahora, ya que realmente lo que haremos será un proceso que lleva muy poco tiempo y que, además, hará que la extracción del archivo .zip no sea del todo necesaria. Esto se debe a que todo el repositorio será extraído directamente en nuestra carpeta principal sin la necesidad de que tengamos que modificar nada después de esto. Para llevar a cabo este proceso, simplemente tenemos que usar el siguiente comando:

```
1  !rm -rf .[!.]* *
2  !git clone https://github.com/compvis/stable-diffusion .
```

Este proceso hará que todos los archivos del repositorio se transfieran directamente a nuestro entorno de Google Colab. Además, podemos observar que antes de clonar el repositorio, se utiliza un comando "rm".

Este comando se encarga de eliminar todos los archivos que se encuentran dentro de la carpeta actual, principalmente porque la carpeta de extracción del repositorio requiere que no haya ningún archivo o carpeta en el lugar donde se desea clonar. Generalmente, Google Colab incluye dos carpetas en cada sesión: "sample_data" y la carpeta oculta ".config". Este comando elimina todos los archivos y carpetas ocultas, así como los archivos normales, para que no haya problemas al clonar el repositorio.

El comando se utiliza con frecuencia para clonar repositorios, y el proceso no es demasiado complicado, ya que simplemente crea una copia del repositorio en nuestro sistema. Por lo general, este comando crea una carpeta adicional que contiene todos los archivos del repositorio. Sin embargo, en este caso, el comando se encarga de guardar todos los archivos directamente en nuestra carpeta principal. Esto se logra gracias al punto final del comando, que indica que queremos que todos los archivos se guarden en la carpeta actual.

Podemos ver que este comando también genera una salida que muestra el proceso de clonación del repositorio, como se puede observar a continuación:

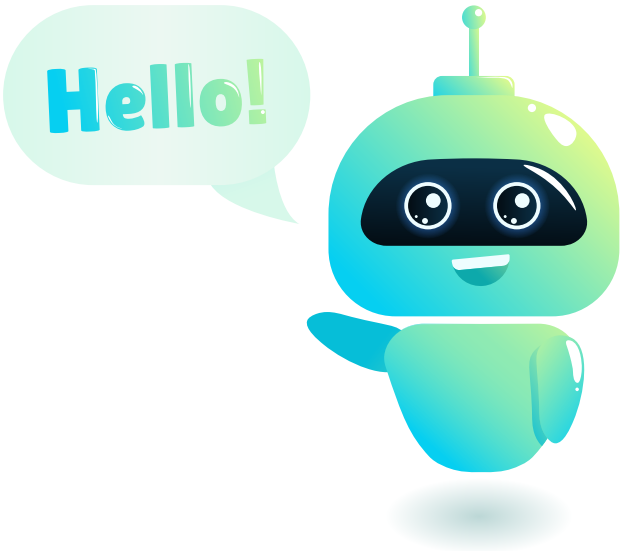
```
Cloning into '.'...
remote: Enumerating objects: 340, done.
remote: Total 340 (delta 0), reused 0 (delta 0), pack-reused 340
Receiving objects: 100% (340/340), 42.65 MiB | 36.37 MiB/s, done.
Resolving deltas: 100% (114/114), done.
```

Ilustración 65 – Salida de terminal generada al clonar el repositorio de Stable Diffusion en Google Cola usando git.

Con esto, finalmente, podemos decir que hemos concluido la clonación del repositorio dentro de nuestro entorno de Google Colab. Como podemos observar, realmente podemos afirmar que este método, en comparación con los otros métodos que vimos anteriormente, puede ser el más sencillo, rápido y con menor costo computacional. Sin embargo, el motivo por el que vimos los otros dos métodos anteriores es únicamente en el caso de que una persona quiera usar una copia del repositorio local dentro de su computadora o unidad de Google Drive, o simplemente para abrir la puerta a diferentes formas en las que podemos realizar el mismo proceso dentro de nuestro entorno de Google Colab. Ahora, teniendo un poco más claro, es probable que para este método realmente no sea del todo aplicable. Sin embargo, es el momento de ver cómo podemos extraer estos archivos .zip dentro de nuestro entorno de Google Colab para poder utilizarlos.

Conclusión del capítulo

Este capítulo ha abordado dos métodos principales para la manipulación de archivos en el entorno de Google Colab, esencial para trabajar con el modelo de Stable Diffusion. La vinculación de Google Drive se presenta como un método integrado y versátil, mientras que la subida manual ofrece una alternativa más autónoma. Cada método viene con sus propias ventajas y desventajas, permitiendo a los usuarios elegir el enfoque que mejor se adapte a sus necesidades y preferencias. La instrucción detallada y las ilustraciones proporcionadas en este capítulo permitirán a los lectores aplicar estos métodos de manera efectiva en sus propios proyectos. La elección del método dependerá de factores como la accesibilidad, la preferencia de trabajo en la nube o localmente, y las necesidades específicas del proyecto.



EXTRACCIÓN DE REPOSITORIOS

Objetivos del capítulo

Los objetivos principales de este capítulo son presentar y explicar en detalle el proceso de extracción de archivos .zip en el entorno de Google Colab, específicamente en el contexto de la instalación y configuración del repositorio de Stable Diffusion. Se discutirán las herramientas y comandos disponibles dentro del intérprete de Python de Google Colab para llevar a cabo este proceso. Además, se abordarán las dificultades y soluciones comunes, como la generación de carpetas adicionales durante la extracción y cómo mover los archivos a la carpeta principal. Este capítulo tiene como objetivo equipar al lector con las habilidades y el conocimiento necesarios para preparar y organizar eficientemente el entorno de trabajo para el uso del modelo Stable Diffusion en GoogleColab.

Ahora, finalmente, una vez que hemos visto las tres diferentes formas en las que podemos usar el repositorio dentro de nuestro Google Colab internamente, veremos la forma en la que podemos encargarnos de extraer el archivo .zip que descargamos anteriormente dentro de nuestro entorno de Google Colab para poder hacer que los archivos se encuentren dentro de nuestra carpeta local y poder usarlos de una forma más sencilla. El proceso realmente es bastante sencillo, y podremos usar unas herramientas que ya vienen dentro de nuestro intérprete de Python de Google Colab para poder hacer esto.

Recordemos, esto solamente es en el caso de que se haya decidido hacer uso de algunas de las dos primeras formas en las que podemos conseguir el repositorio dentro de nuestro entorno de Google Colab, esto, principalmente, debido al hecho

de que el tercer método no requiere del todo que tengamos que extraer ningún archivo .zip, pues realmente el proceso que hicimos para poder clonar el repositorio fue bastante sencillo, rápido y directo. El comando que debemos de usar para poder extraer los archivos .zip del repositorio es el siguiente:

```
1  !unzip <ruta del archivo> -d .
```

Así, veremos cómo nuestro entorno de Google Colab se encargará de extraer todos los archivos que se encuentren dentro del archivo .zip en nuestra carpeta actual. Ahora, en la parte del comando donde dice "ruta del archivo", solamente debemos cambiarlo por el nombre de la ruta del archivo y el archivo en sí, dependiendo del caso que se haya usado. Entonces, podría cambiar, por ejemplo, en el caso de que se haya usado el primer método y también el archivo .zip del repositorio se encuentren dentro de la carpeta principal de nuestra unidad de Google Drive. En lugar de "ruta de archivo", podríamos usar la ruta drive/MyDrive/stable-diffusion-main.zip. Mientras que, en el caso de que nuestro archivo .zip se encuentre dentro de nuestra carpeta actual, nuestra ruta sería simplemente el nombre del archivo stable-diffusion-main.zip. Esto hará que todos los archivos del repositorio sean extraídos dentro de nuestra carpeta actual.

No obstante, podemos observar que enfrentamos un problema al momento de extraer los archivos. A pesar de haber indicado en nuestro programa que queremos extraer todos los archivos sin necesidad de crear una carpeta adicional, podemos ver que realmente esta carpeta adicional fue creada.

Esto se evidencia en la siguiente imagen, donde se muestra la salida después de haber ejecutado el comando que mencionamos previamente:

Ilustración 66 – Generación de carpeta extra del repositorio comprimido al realizar la extracción en Google Colab.

Este problema realmente no es demasiado grande, y cumple con las áreas iniciales que deseábamos extraer del comando comprimido. Sin embargo, puede resultar tedioso volver a referenciar cada vez que necesitemos utilizar algo que se

encuentre dentro del repositorio, cuando podríamos simplemente acceder directamente a los archivos dentro de nuestra carpeta principal. Por lo tanto, por conveniencia y para ahorrarnos tiempo en el futuro, es aconsejable mover los archivos.

No debemos preocuparnos realmente, pues este proceso es bastante sencillo de realizar y, en verdad, no toma mucho tiempo. Como dijimos, realmente lo único que tenemos que hacer es mover todos los archivos que se encuentran dentro de esta carpeta a nuestra carpeta principal. Podemos hacer esto de una forma bastante fácil con el siguiente bloque de comando:

```
1  !mv stable-diffusion-main/* .
2  !rmdir stable-diffusion-main
```

Con esto, simplemente le indicamos a nuestro entorno de Google Colab que queremos que primero traslade todos los archivos que se encuentran dentro de la carpeta a la carpeta actual, es decir, la carpeta principal de Google Colab.

Luego, le indicamos que queremos eliminar la carpeta vacía que utilizamos anteriormente, lo cual permite que ahora sí contemos con todos los archivos necesarios para usar el repositorio dentro de nuestro entorno principal de Google Colab, tal y como podemos ver en la siguiente imagen:

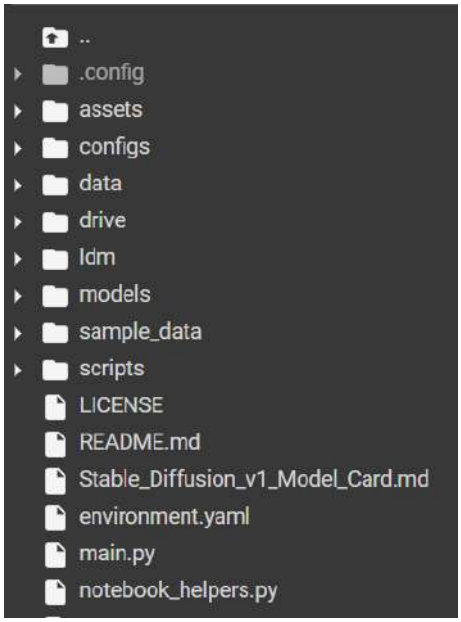


Ilustración 67 – Repositorio de Stable Diffusion extraído en la carpeta principal de Google Colab.

Y con esto finalmente podemos dar por concluida esta parte, ya que podemos afirmar que hemos completado la instalación del repositorio de Stable Diffusion dentro de nuestro Google Colab. Es demasiado importante resaltar el hecho de

que todos los cambios y archivos que guardemos dentro de nuestro entorno no durarán para siempre. Por lo tanto, es crucial que cada vez que iniciemos una nueva sesión en Google Colab, es muy probable que tengamos que repetir este proceso, al menos en el caso de que necesitemos usar Stable Diffusion mientras usamos Google Colab.

Listo, finalmente podemos decir que tenemos todo preparado para poder usar el modelo. Realmente, los pasos que nos faltan para poder empezar a ejecutar el modelo y generar nuestras imágenes son muy pocos. Además, podemos afirmar que el avance que estamos teniendo frente a los diferentes requisitos que tenemos que cumplir para poder utilizar este increíble modelo de inteligencia artificial está cerca de finalizar. Por lo cual, será cuestión de tiempo para que empecemos a ver cada vez más formas en las que podremos usar este modelo de inteligencia artificial de forma local. Este modelo está dando la vuelta al mundo y sorprendiendo a millones con sus resultados.

Conclusión del capítulo

En este capítulo, se logró una comprensión completa de cómo gestionar y extraer repositorios dentro del entorno de Google Colab para facilitar el uso de Stable Diffusion. A través de instrucciones detalladas, ilustraciones y ejemplos de código, se ha guiado al lector en el proceso de extracción de archivos, abordando problemas comunes y brindando soluciones efectivas. La importancia de este proceso en el marco del trabajo con el modelo Stable Diffusion ha sido resaltada, y se ha enfatizado la necesidad de repetir estos pasos en nuevas sesiones de Google Colab. El capítulo concluye con una visión optimista, señalando que el lector está cada vez más cerca de utilizar este revolucionario modelo de inteligencia artificial, preparando el terreno para los próximos pasos en la aplicación práctica de Stable Diffusion.

PRERREQUISITOS

Objetivos del capítulo

El presente capítulo tiene como finalidad explicar los prerrequisitos y configuraciones necesarias para utilizar Stable Diffusion a través de Google Colab. Los objetivos son claros: describir la importancia de los entornos como Miniconda, detallar el proceso de instalación de los paquetes necesarios como PyTorch, torchvision y transformers, y demostrar cómo se configura el entorno de Google Colab para ejecutar el modelo. Además, se destaca la posibilidad de errores y problemas comunes, ofreciendo soluciones alternativas para el uso de Stable Diffusion de una manera más simple y eficiente.

Ahora, daremos el primer paso para empezar a hacer uso de este modelo de inteligencia artificial. Naturalmente, el primer paso es abrir una nueva sesión de Google Colab. En esta sección, es donde escribiremos nuestro código para luego ser ejecutado. Una vez que tengamos listo nuestro Google Colab, entonces, es el momento de comenzar a instalar los prerrequisitos que necesita Stable Diffusion para poder ejecutar correctamente el modelo.

El primer paso para poder ejecutar estos modelos, después de haber creado el archivo de Google Colab donde estaremos programando e indicando por medio de nuestros comandos, es instalar los prerrequisitos del modelo. En este caso, estaremos usando paquetes que reciben el nombre de PyTorch, torchvision y transformers.

Para poder instalar estos prerrequisitos, deberemos indicar por medio de la terminal de ejecución que queremos instalarlos usando el administrador de paquetes pip. Esto podría hacerse fácilmente dentro de una terminal; sin embargo, debido a que no tenemos una terminal dentro de nuestro Google Colab, entonces, debemos ser un poco más creativos.

Cada vez que queramos añadir un comando de terminal dentro de Google Colab, podemos hacer uso de los comandos que deseamos ejecutar dentro de la terminal usando el símbolo "!". Podemos ver un ejemplo de esto mismo a continuación con las siguientes imágenes:

Ilustración 68 – Diferencias entre comandos de instrucción en terminal e intérprete en Google Colab.

Sin embargo, antes de poder empezar a trabajar con los prerrequisitos que necesitamos para poder usar correctamente el modelo, tenemos que hacer que nuestro entorno de Google Colab contenga dentro de sí un entorno de Anaconda, o, para ahorrar algo de tiempo, Miniconda. Esto se debe a que el archivo que tiene los prerrequisitos del modelo se encuentra dentro de un archivo .yaml, es decir, un archivo que es ejecutable dentro de Conda. En términos simples, lo que haremos será modificar nuestra versión e intérprete del entorno de Python actual de tal forma que cumpla con los requisitos que necesitamos para que nuestro entorno de Google Colab pueda correr el modelo y empezar a generar imágenes usando Stable Diffusion.

Para poder hacer esto, entonces, solamente debemos ejecutar los siguientes comandos:

```
1  !curl -L https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
   -o miniconda.sh
2  !bash miniconda.sh -bfp /usr/local
```

Por ahora, no debemos asustarnos si no entendemos lo que hace este comando de Linux. En realidad, es bastante sencillo. Lo único que estamos haciendo con estos bloques de código es configurar nuestro entorno de Google Colab para instalar el administrador Miniconda, el cual nos permitirá instalar los prerrequisitos del modelo.

Esto generará una salida en nuestra pantalla del terminal de Google Colab, la cual solamente mostrará el progreso de la descarga e instalación de Miniconda. Podemos observar en la siguiente imagen la salida a la que nos referimos:

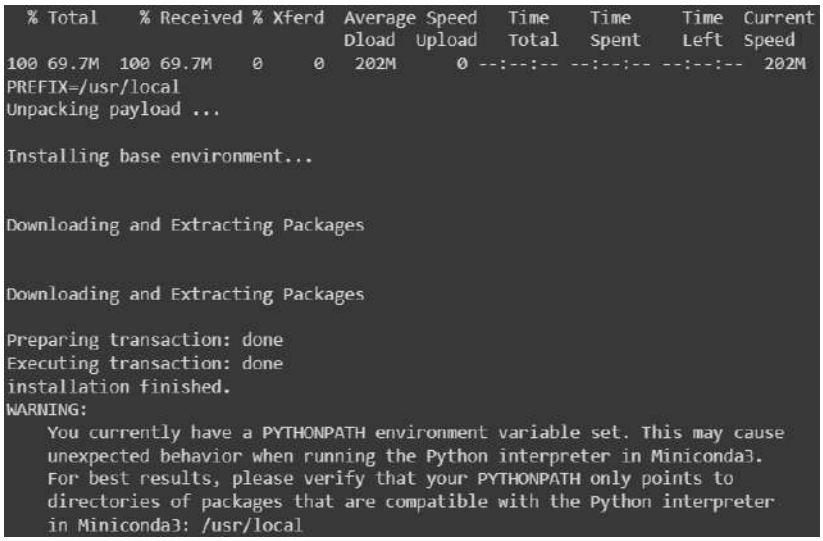


Ilustración 69 – Salida generada al descargar y ejecutar el instalador de Miniconda en Google Colab.

Ya está, con esto tenemos la herramienta principal que utilizaremos para corregir e instalar el archivo que contiene todos los archivos necesarios para el correcto funcionamiento de Stable Diffusion. Ahora, finalmente, es momento de aplicar los prerrequisitos en nuestro Python. Podemos hacerlo con el siguiente comando:

```
1 !conda env update -n base -f environment.yaml
```

Este proceso hará que la lista de los diferentes paquetes que necesitan instalarse, y que se encuentran dentro del archivo "environment.yaml", sean actualizados e incorporados en nuestro entorno principal de ejecución de Python. Este proceso puede ser bastante extenso y llevar mucho tiempo, así que realmente solo es cuestión de esperar hasta que termine la ejecución de este comando para poder continuar. Es importante no interrumpir en ningún momento la ejecución de la celda de código, ya que, como mencionamos, es crucial que los diferentes requisitos y paquetes necesarios para utilizar este modelo se instalen correctamente para no tener problemas más adelante.

Podremos saber que los paquetes se instalaron correctamente cuando nuestra terminal muestre comandos similares a los siguientes. Además, la celda de código dejará de ejecutarse, indicando que la instalación de los paquetes ha finalizado correctamente. Podemos ver a qué nos referimos con la siguiente imagen:

```
Successfully installed MarkupSafe-2.1.2 PyWavelets
#
# To activate this environment, use
#
#     $ conda activate base
#
# To deactivate an active environment, use
#
#     $ conda deactivate
```

Ilustración 70 – Salida en Google Colab después de actualizar el entorno base usando Conda.

Podemos observar que la salida de la terminal del comando que acabamos de escribir nos sugiere que activemos el entorno de Conda. Sin embargo, este paso realmente no es del todo necesario debido al hecho de que estamos actualizando nuestro entorno de Conda actual, es decir, en el caso de que estemos actualizando o creando un nuevo entorno dentro de nuestras diferentes versiones e intérpretes de Python, entonces, podría llegar a ser necesario activar el entorno para que nuestro sistema sepa qué intérprete de Python debe usar. Sin embargo, este raramente es el caso, pues solamente estamos indicando a nuestro programa que actualice el principal y único entorno existente.

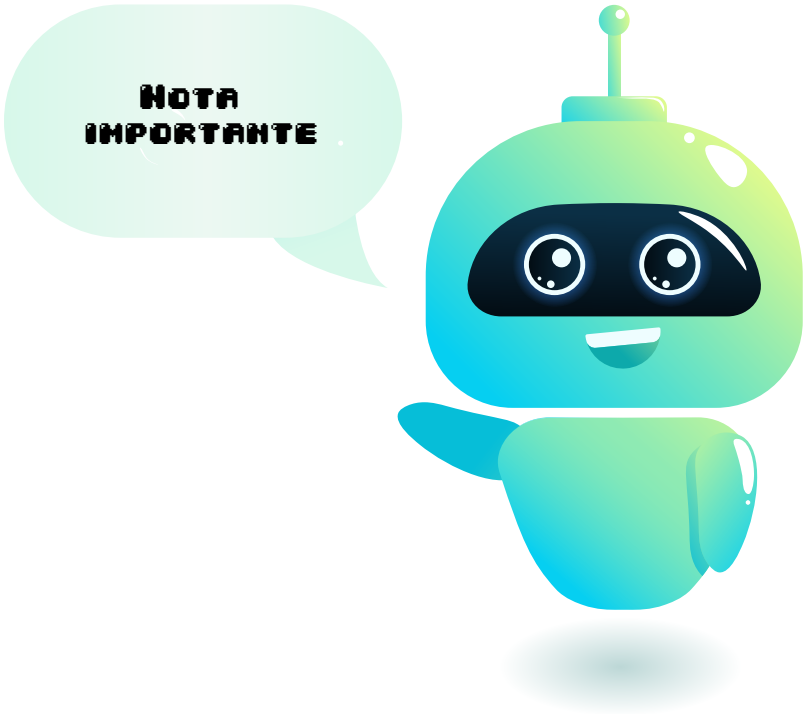
Ahora bien, aunque es cierto que tuvimos que instalar todos estos paquetes y realizar todos los pasos que vimos hasta ahora, en realidad no usaremos este método para hacer uso de Stable Diffusion en este libro, especialmente al considerar varios factores importantes. Primero, partamos del hecho de que queremos que el aprendizaje de los diferentes temas que abordamos en este libro sea lo más sencillo, ameno y fácil de entender posible. Por lo tanto, probablemente entender cómo funcionan los distintos scripts de Python que se encuentran en este repositorio no sea lo que esperamos al intentar comprender de forma sencilla todo lo que ocurre en el modelo y cómo podemos usarlo de una forma simple.

El segundo método que presentaremos ofrece la oportunidad de realizar esto de una manera muy sencilla, rápida y con menos líneas de código. Sin embargo, todos los pasos que vimos hasta ahora nos ayudan a construir el entorno necesario que tendremos que usar para desarrollar nuestras propias aplicaciones de Stable Diffusion, en caso de que deseemos explorar más a fondo el funcionamiento de estos modelos y entender cómo funcionan los diferentes scripts que vienen dentro del repositorio, los cuales pueden ser un poco confusos para aquellos no familiarizados con estos modelos de difusión.

Otro aspecto a tener en cuenta al decidir no usar este método y los scripts que vienen en el repositorio es el hecho de que, generalmente, estos scripts suelen generar errores para aquellas personas que están intentando usar el modelo dentro de sus notebooks de Google Colab. La razón exacta de esto no es del todo

clara, pero es común que estos modelos generativos preprogramados en los scripts de prueba del repositorio generen errores en Google Colab. Esto es más evidente si intentamos ejecutar uno de los scripts de Python del repositorio. Aunque el script se ejecuta, no genera ninguna imagen de salida nueva, lo cual nos indica que no está funcionando correctamente. Podemos ilustrar esto con el siguiente bloque de código.

No es necesario ejecutarlo en nuestros propios Colabs de Google, ya que la idea de esta demostración es evidenciar que los scripts generan errores de funcionamiento en Google Colab, incluso después de haber instalado todas las dependencias necesarias para que el modelo funcione correctamente.



Tal como se mencionó, este paso realmente no es del todo necesario para entender el ejemplo que planeamos ver dentro de este libro. Es solamente para entender una posibilidad de cómo se planea usar el repositorio y empezar a desarrollar nuestros propios códigos avanzados en Stable Diffusion. Esto nos da la oportunidad de ver el código dentro de otros scripts de prueba dentro del modelo. La salida generada por el comando que veremos a continuación solo se produce después de que los pesos del modelo que se desea usar han sido guardados correctamente dentro de nuestro entorno. Los pasos para poder realizar esto se encuentran en la página principal del GitHub que estamos usando, o dentro del archivo README.md. Sin embargo, de nuevo, esto no es del todo necesario debido al error que generará.

El comando que se utiliza para emplear el script de generación de imágenes en el modo de texto a imagen es el siguiente. Recordemos una vez más que los scripts del repositorio corresponden a diversos archivos que ya vienen preprogramados por los creadores del repositorio, con el fin de poder demostrar Stable Diffusion. Esto normalmente podría funcionar en una computadora local. Sin embargo, suele generar un error cuando se utiliza Google Colab.

```
1  !python scripts/txt2img.py --prompt "a photograph of an astronaut riding a
```

El propósito de este comando es ejecutar el script que ya viene previamente definido dentro de nuestro repositorio. Sin embargo, la salida generada por el mismo programa es la que podemos ver en la siguiente imagen. A pesar de intentar realizar las diferentes soluciones que la comunidad ha propuesto para este problema, realmente podemos decir que lidiar con esta advertencia (y el hecho de que después de la ejecución del modelo realmente ninguna imagen es generada) puede llegar a ser un dolor de cabeza. Por esto, es más conveniente usar el método que se explicará más adelante, el cual evita este tipo de errores y permite usar el modelo de una forma más sencilla y rápida. Aunque esto realmente no signifique una versión del modelo completamente personalizable, podemos ver la salida generada de la advertencia en la terminal de Google Colab a continuación con la siguiente imagen:

```
Downloading: 100% 342/342 [00:00<00:00, 345kB/s]
Downloading: 100% 4.44k/4.44k [00:00<00:00, 3.69MB/s]
Downloading: 100% 1.13G/1.13G [00:20<00:00, 60.8MB/s]
Global seed set to 42
Loading model from models/ldm/stable-diffusion-v1/model.ckpt
Global Step: 245000
LatentDiffusion: Running in eps-prediction mode
^C
```

Ilustración 71 – Salida de la terminal después de ejecutar el script de prueba del repositorio de Stable Diffusion en Google Colab.

Ahora, de nuevo, es importante resaltar el hecho de que realmente esto no corresponde al método que se planea usar con este modelo. Es decir, lo que estamos haciendo es intentar ver el motivo de por qué esto falla, el cual realmente no es del todo claro. A pesar de usar los scripts de prueba dentro de nuestro entorno de Google Colab, que ya vienen previamente en el repositorio que usamos anteriormente, la realidad es que no obtendremos el resultado esperado ni ninguna imagen generada.

Sin embargo, la ventaja de poder realizar todos estos pasos radica en el hecho de que esto nos permite construir todo el ambiente que necesitaremos para posteriormente hacer nuestros propios modelos entrenados. Incluso nos brinda la oportunidad de explorar más a fondo los archivos de Python que se encuentran dentro de la carpeta de scripts, lo que nos permite familiarizarnos un poco más con el proceso de generación y control del código necesario para usar este modelo de inteligencia artificial.

Ahora, una vez que tenemos esto claro, es el momento de empezar a desarrollar un breve ejemplo real que nos permita usar de una vez por todas el modelo Stable Diffusion dentro de un entorno de Google Colab. En este caso, haremos

uso de los Diffusers, la librería que se mencionó anteriormente en varias ocasiones y nos permite usar este increíble modelo de inteligencia artificial de una forma bastante sencilla, sin la necesidad de tener que clonar repositorios o hacer pasos extras innecesarios si nuestra intención es solamente darle un uso básico a este modelo de inteligencia artificial, y no modificar o personalizar más a detalle los diferentes parámetros que se pueden modificar dentro de este.

Conclusión del capítulo

En este capítulo, hemos establecido los cimientos para trabajar con el modelo de Stable Diffusion, explorando la instalación de prerrequisitos y la configuración del entorno. A través de explicaciones detalladas y ejemplos visuales, se ha llevado al lector a través de los pasos necesarios para preparar Google Colab para el uso de este modelo de inteligencia artificial. También hemos destacado las posibles complicaciones y cómo abordarlas, ofreciendo una alternativa más sencilla y rápida. Esta sección sirve como una guía esencial para aquellos que deseen explorar más a fondo Stable Diffusion, preparándolos para desarrollar sus propios modelos entrenados y personalizados.

MÉTODO FUNCIONAL: MANOS A LA ACCIÓN

Objetivos del capítulo

En este capítulo, se busca ofrecer una perspectiva detallada y aplicada del método funcional de Stable Diffusion, explicando el proceso de implementación en Google Colab. La sección pretende guiar al lector en la instalación de los prerrequisitos necesarios, detallando la importancia y función de cada paquete, y conduciendo hacia la práctica directa en la generación de imágenes. A través de una exploración cuidadosa de la instalación de paquetes, el uso de la librería Diffusers, y el proceso para elegir y aplicar una versión específica de Stable Diffusion, se busca empoderar al lector con las habilidades y conocimientos necesarios para aplicar este modelo de inteligencia artificial en su entorno local o en Google Colab. La presentación de la metodología de trabajo con el modelo Stable Diffusion apunta a simplificar la experiencia, haciendo accesible la tecnología a una variedad de usuarios.

Ahora, el momento crucial ha llegado. Hemos explorado las diversas aplicaciones que los repositorios nos ofrecen y los distintos recursos que poseemos para utilizar el modelo. Simultáneamente, hemos analizado las ventajas y desventajas de utilizar este modelo de inteligencia artificial, en particular si queremos implementar nuestros propios modelos basados en este, o modificar el proceso de entrenamiento típicamente empleado para generar imágenes.

Sin embargo, aún no hemos tenido la oportunidad de examinar un ejemplo que nos permita aplicar de manera directa el modelo de inteligencia artificial dentro

de Google Colab y visualizar, a su vez, las imágenes generadas que podemos usar en este modelo.

La espera ha terminado, ya que ahora finalmente podemos entender a fondo las capacidades y la importancia de este modelo, así como la conveniencia de utilizar Google Colab como entorno para realizar todas estas operaciones y pruebas dentro de Stable Diffusion. Al final de las distintas subsecciones que abordaremos a continuación, podremos examinar y entender cómo aplicar de manera práctica las diferentes técnicas disponibles en Stable Diffusion para la generación de imágenes.

El principal paso que debemos seguir ahora es sencillo: abordar los prerrequisitos del modelo. Estos los podemos detallar y explicar exhaustivamente en la próxima sección.

Prerrequisitos

Antes, cuando estábamos usando el repositorio de Stable Diffusion, vimos que realmente el proceso para instalar los prerrequisitos del modelo era bastante tedioso. Esto se debía principalmente al hecho de que, en general, el repositorio se utiliza cuando se desea hacer una personalización muy grande de las capacidades que ofrece el modelo. Sin embargo, tal y como hemos mencionado hasta el momento, esta no es la intención de esta sección ni de este libro. En realidad, la intención es que la mayoría de las personas sepan y tengan la oportunidad de entender cómo usar este modelo de inteligencia artificial de forma local en sus computadoras o entornos de Google Colab. Con esto en mente, entonces, en esta sección nos centraremos en instalar solamente aquellos componentes muy específicos que podríamos llegar a requerir al usar el modelo de la manera más sencilla. Es decir, instalaremos todos los paquetes que son estrictamente necesarios para esto. Para poder hacer esto, solamente debemos ejecutar el siguiente bloque de comandos:

```
1  !pip install transformers scipy ftfy accelerate diffusers==0.11.1
```

Este se encargará de instalar todos los paquetes que son necesarios para poder usar correctamente el modelo. Sin embargo, realmente la pregunta importante es, ¿cuál es la importancia de cada uno de los paquetes? No explicaremos del todo a detalle la función completamente interna de estos, sin embargo, veremos unas breves descripciones de lo que hace cada uno de los paquetes para poder entender más a fondo lo que ocurre dentro de nuestro código y el por qué estamos instalando estos paquetes dentro de nuestro entorno de Google Colab. Una vez este comando haya sido ejecutado, la salida de nuestra terminal se debería ver algo así:

```
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.10/dist-packages (f
Requirement already satisfied: mpmath>=0.19 in /usr/local/lib/python3.10/dist-packages (from
Installing collected packages: tokenizers, importlib-metadata, ftfy, huggingface-hub, transf
Successfully installed accelerate-0.19.0 diffusers-0.11.1 ftfy-6.1.1 huggingface-hub-0.14.1
```

Ilustración 72 – Salida de terminal final después de instalar los paquetes necesarios para Stable Diffusion.

La salida puede llegar a ser algo extensa, pero realmente el punto que nos importa es que, finalmente, al ejecutar el código, nos muestre que todos los paquetes fueron instalados correctamente en nuestro entorno. Esto generalmente ocurre por medio de la salida que dice "Successfully installed". En el caso de que no sea así y se genere un error, entonces, es probable que alguno de los pasos o la escritura de los paquetes haya salido mal o con algún error tipográfico que no nos permita instalar correctamente los paquetes necesarios.

En términos simples, todos los paquetes que hemos instalado anteriormente, como Transformers, scipy y ftfy, corresponden a los requisitos mínimos que Stable Diffusion necesita para funcionar correctamente. Desde un punto de vista básico, son los paquetes esenciales para el adecuado funcionamiento de este modelo de inteligencia artificial.

Sin embargo, los otros dos paquetes que estamos instalando no cumplen esta característica, sino que se encargan de cumplir una función más específica dentro del proceso de Stable Diffusion y su generación de imágenes. En términos simples, 'accelerate' es el nombre de una librería que nos permite cargar las imágenes de una forma mucho más sencilla, ya que este proceso puede ser bastante extenso y tomar mucho tiempo. Esta librería optimiza el proceso y agiliza la carga en este ejemplo.

Las versiones actuales de los Diffusers en realidad no presentan ningún problema específico; los inconvenientes suelen surgir solo con los modelos antiguos de Stable Diffusion. Por este motivo, al especificar "diffusers==1.11.1", nos aseguramos de instalar una versión que sabemos de antemano funcionará bien con el modelo. De esta forma, minimizamos la posibilidad de encontrar futuros errores relacionados con la versión de los Difusores. En el caso de que no actualicemos la versión de los Diffusers, es probable que nos encontremos con un error al generar las imágenes. Sin embargo, esto no siempre es así; existe la posibilidad de que la imagen se genere correctamente incluso con la versión actual de los Difusores. Aun así, es recomendable asegurarnos e ir por el camino seguro para lograr nuestros objetivos.

Existen diversas maneras de instalar los Diffusers en nuestro entorno y gestionar las diferentes versiones que podemos usar, por ejemplo, a través del repositorio de esta librería. Sin embargo, realmente no es del todo necesario instalarlo de esta forma si nuestra intención no es desarrollar algo desde un nivel bajo con

esta librería. Por ahora, nuestra intención es simplemente hacer uso de la librería, por lo que utilizar pip será más que suficiente.

Ilustración 73 – Captura de pantalla del repositorio de Diffusers.

Uso del pipeline de Stable Diffusion

Ahora, finalmente llegamos a la parte donde discutiremos la importancia de utilizar la librería Diffusers en nuestra generación de imágenes usando Stable Diffusion. Esta librería nos permite trabajar con las diferentes versiones de Stable Diffusion y descargar los pesos correspondientes del modelo que deseamos usar de una manera sencilla y eficiente, todo esto con tan solo una línea de código. Este nivel de abstracción nos ahorra una considerable cantidad de tiempo, razón por la cual es de gran importancia usar esta librería. Además, presenta una novedad significativa para el modelo en cuestión. Podemos importar el pipeline utilizando la librería Diffusers, como se puede ver en el siguiente bloque de código:

```
1 from diffusers import StableDiffusionPipeline
```

Y listo, ya contamos con la herramienta necesaria que nos permite usar y elegir la versión de Stable Diffusion que queremos, así como generar imágenes con este pipeline de una forma bastante sencilla. Una vez importado el pipeline en nuestro código, es el momento de indicar a nuestro modelo qué versión de Stable Diffusion queremos usar para generar las imágenes.

El administrador del pipeline de Stable Diffusion nos permitirá descargar los pesos correspondientes del modelo que elijamos y generar las imágenes de manera rápida y sencilla.

Anteriormente, tuvimos la oportunidad de ver las diferentes versiones que existen actualmente de Stable Diffusion, así como las diversas funciones que ofrecen. Es

importante aclarar de nuevo que, hasta la fecha en la que se escribe este libro, el modelo de Stable Diffusion XL no ha sido lanzado aún. Esto significa que, para poder usar el modelo de esta forma que estamos utilizando en Google Colab, probablemente no podríamos usar este modelo. Sin embargo, es probable que el modelo se lance posteriormente.

Por ahora, no tenemos acceso a esta versión del modelo, así que tendremos que usar las versiones del modelo a las que ya tenemos acceso actualmente. Podría ser cualquiera, sin embargo, en este caso, dentro de este ejemplo, usaremos la versión del modelo stable-diffusion-v1-4.

Realmente no hay una razón específica para la elección de este modelo, de nuevo, podría ser cualquier otra versión que vimos anteriormente. Podemos indicar por medio del siguiente bloque de código que, en efecto, nuestro pipe corresponde a la versión que mencionamos, como podemos ver a continuación en el siguiente bloque de código:

```
1  pipe = StableDiffusionPipeline.from_pretrained("CompVis/stable-diffusion-
   v1-4")
```

Como podemos observar por el nombre mismo que recibe el método que estamos intentando usar, realmente lo que ocurre es que estamos haciendo uso de un modelo de inteligencia artificial que ya viene previamente entrenado. En realidad, podemos hacer cualquier tipo de modelo, incluso un modelo de inteligencia artificial que haya sido entrenado con las imágenes que nosotros mismos le hayamos indicado.

De nuevo, realmente esto puede llegar a ser bastante interesante para algunas personas, pero puede llegar a ser una tarea algo tediosa, por ende, realmente no lo enseñaremos dentro de este libro. Sin embargo, no está de más mencionar que esto es posible.

Una vez hemos indicado la versión del modelo que iremos a usar, entonces, la librería se encargará de descargar los pesos del modelo necesarios para que todo funcione adecuadamente.

Generalmente, este proceso suele tomar algo de tiempo, pero suele ser rápido debido a la velocidad de conexión de internet que tienen los dispositivos en la nube de Google Colab, así que realmente no debemos preocuparnos y esperar a que el proceso termine.

Una vez el proceso haya terminado, entonces, la salida generada en la terminal de nuestro Google Colab se verá algo así:

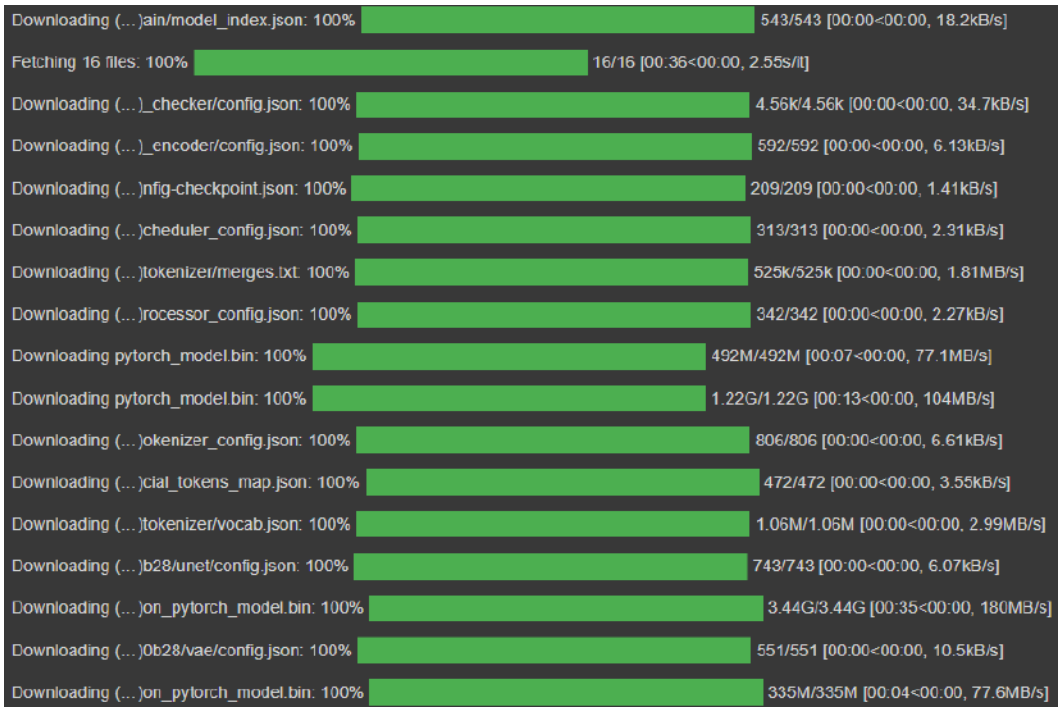


Ilustración 74 – Proceso de descarga del modelo usando Diffusers.

Una vez que todas estas barras estén completas y de color verde, entonces, significará que el modelo ha sido descargado por completo e integrado dentro de nuestro sistema, lo cual significa que finalmente podremos tener la oportunidad de usar el modelo para generar imágenes. Con esto en mente, entonces, el siguiente paso que debemos seguir para poder crear imágenes dentro de nuestro modelo de inteligencia artificial de Stable Diffusion es tener en mente un prompt. Como dijimos, por ahora veremos cómo podemos crear imágenes usando prompts de texto, es decir, es una generación de text-to-image, esto principalmente debido al hecho de que generalmente esto responde a la forma más sencilla en la cual podemos usar Stable Diffusion. Podemos guardar el prompt que deseamos dentro de una variable, como podemos ver a continuación con el siguiente bloque de código:

```
1  prompt = “A cat playing with a beach ball”
```

En este caso, le estamos indicando al modelo que queremos generar una imagen de un gato jugando con una pelota de playa. Sin embargo, no estamos generando ninguna imagen, solamente almacenamos en una variable un texto que contiene la descripción de lo que buscamos generar con el modelo. Con esto, estamos completamente listos para indicarle al modelo que queremos generar una imagen con el 'prompt' que hemos especificado. No obstante, antes de poder generar la imagen como tal, es crucial dar un paso que puede ser esencial para generar las imágenes con esta técnica de inteligencia artificial de una manera bastante sencilla y rápida.

Este paso es configurar el pipeline de Stable Diffusion para que funcione con una tarjeta Nvidia, más específicamente, para que tenga la capacidad de utilizar la tecnología Cuda en la generación de imágenes.

Conclusión del capítulo

Este capítulo ha servido como una guía práctica y detallada para la implementación del Método Funcional de Stable Diffusion en Google Colab. Se ha destacado la importancia de entender y seleccionar cuidadosamente los paquetes necesarios, garantizando una implementación sin problemas del modelo. A través de una descripción exhaustiva, se ha explicado cómo utilizar el repositorio de Stable Diffusion y se ha ilustrado el proceso de selección de la versión adecuada del modelo. Además, se ha mostrado el proceso completo de generación de imágenes a partir de prompts de texto, lo cual refleja la versatilidad y potencia de Stable Diffusion en la tarea de generación de imágenes. La conclusión de este capítulo no solo permite a los lectores comprender la implementación técnica del modelo, sino que también brinda una base sólida para la exploración y aplicación práctica de Stable Diffusion en sus propios proyectos e investigaciones.

PREÁMBULO: ¿QUÉ ES CUDA Y POR QUÉ ES IMPORTANTE?

Objetivos del capítulo

El propósito de este capítulo es profundizar en la comprensión de CUDA, una tecnología desarrollada por Nvidia, que ha sido mencionada superficialmente en la sección anterior. Los objetivos específicos incluyen explicar qué es CUDA (Compute Unified Device Architecture), su importancia y aplicación dentro del campo de la inteligencia artificial, y cómo contribuye a la eficiencia en la generación de imágenes. Además, se explorará cómo CUDA se ha convertido en un estándar en la industria de la inteligencia artificial, y se describirán sus funciones principales, tales como el cálculo en paralelo, la optimización de operaciones matemáticas complejas y la aceleración de hardware.

Ahora, en la sección anterior tuvimos la oportunidad de mencionar un poco superficialmente lo que era CUDA. En términos simples, lo que mencionamos es que se trata de una herramienta que nos permite generar imágenes de una forma mucho más rápida. Pero ¿es esto realmente cierto? Y si de verdad es cierto, ¿qué es lo que tiene de especial y por qué hace que el proceso para generar imágenes sea más rápido? Para poder entender un poco más esto, tenemos que comprender algunos aspectos importantes.

Con este en mente, entonces, esta sección intenta dar a explicar de forma un poco más breve qué es realmente CUDA y por qué es tan importante dentro del campo de la inteligencia artificial. No es algo que solamente ocurre en el caso del

modelo que estamos intentando usar, sino que realmente el uso de las tarjetas de Nvidia y CUDA como tal dentro de los diferentes modelos de inteligencia artificial es algo demasiado común, y se ha convertido en algo así como un tipo de estándar dentro de la industria de la inteligencia artificial.

CUDA es el acrónimo de una tecnología desarrollada por Nvidia, el cual es la abreviatura de "Compute Unified Device Architecture". Naturalmente, esto junto con otras más tecnologías y novedades que ha presentado Nvidia, ha hecho que en gran parte las GPUs de Nvidia se vuelvan el estándar de uso frente a los modelos de inteligencia artificial, ya que esta herramienta y capacidad de tecnología solamente se encuentra presente dentro de las tarjetas de Nvidia. Por lo cual es bastante común que las personas que están investigando dentro de la inteligencia artificial tengan acceso o anhelen el acceso de tarjetas de video de Nvidia. Además, también es la razón principal por la que los Colabs que hemos usado hasta el momento hacen uso de las tarjetas gráficas de Nvidia como vimos anteriormente.

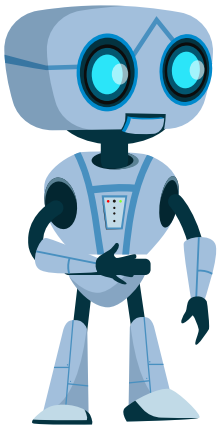
Pero ¿qué es lo que hace a CUDA tan especial? En términos simples, podemos describir que las GPUs actualmente tienen la capacidad de poder realizar operaciones matemáticas complejas a una velocidad simplemente increíble, fuera de solamente calcular gráficos, ha sido demostrado que estas unidades son muy buenas en operaciones matemáticas muy complejas, por lo que usar las GPUs en procesamiento puede llevar a que los procesos que generalmente toman mucho tiempo sean mucho más rápidos con solo usar una GPU.

En términos simples, la importancia de CUDA dentro de las tarjetas gráficas y, especialmente, dentro de la inteligencia artificial es que permite de una manera más sencilla sacar todo el provecho de las tarjetas gráficas frente al procesamiento de operaciones matemáticas demasiado complejas en un plazo de tiempo realmente corto, haciendo que los procesos que generalmente tomarían mucho tiempo usando una CPU ocurran en mucho menor tiempo y mayor eficiencia al usar una GPU.

CUDA es una plataforma de computación que permite varias funciones que permiten sacar todo el provecho a las GPUs de Nvidia, tales como por ejemplo la capacidad de poder realizar operaciones de cómputo en paralelo, optimizar el rendimiento de cálculo de operaciones matemáticas demasiado complejas, aceleración de hardware y también la capacidad de usar muchísimas librerías desarrolladas tanto por Nvidia como por los desarrolladores de CUDA que permiten optimizar varios procesos de una forma muy sencilla.

Por esto, entonces, al momento de usar tarjetas gráficas de Nvidia dentro de nuestro entorno de Google Colab puede llegar a ser bastante beneficioso para nosotros poder usar la capacidad de CUDA y el potencial completo de nuestras tarjetas gráficas que nos da Google para poder realizar la generación de imágenes mucho más rápido y sencillo.

Conclusión del capítulo



En conclusión, este capítulo ha proporcionado una visión detallada de CUDA, destacando su relevancia y aplicación en la inteligencia artificial, en particular en la generación de imágenes. Se explicó cómo CUDA permite una ejecución eficiente de operaciones matemáticas complejas mediante el uso de tarjetas gráficas de Nvidia, lo que reduce significativamente el tiempo de procesamiento en comparación con el uso de CPUs. Además, se enfatizó que las capacidades de CUDA, como la computación en paralelo y la optimización del rendimiento, la han convertido en una tecnología esencial en la industria de la inteligencia artificial. La comprensión de CUDA no solo es vital para aquellos que trabajan en la inteligencia artificial, sino que también ofrece una visión de cómo las tecnologías modernas están siendo utilizadas para aumentar la eficiencia y la velocidad en diversas aplicaciones.

GENERACIÓN DE IMÁGENES

Objetivos del capítulo

Los objetivos de este capítulo se centran en proporcionar una guía completa y accesible para generar imágenes utilizando el modelo de Stable Diffusion en un entorno local. Los lectores aprenderán cómo ejecutar el proceso de generación de imágenes a través de Google Colab, incluyendo la configuración necesaria, la comprensión de las funciones y comandos implicados, y cómo aplicar varias modificaciones y personalizaciones a la generación de imágenes. Se presentarán múltiples ejemplos para ilustrar la funcionalidad y el proceso de la generación de imágenes utilizando este modelo, permitiendo así una aplicación práctica e inmediata de los conceptos explicados.

Y listo, con esto en mente, entonces finalmente podemos ahora sí empezar a generar las imágenes que nosotros deseemos dentro de nuestro código usando el modelo de Stable Diffusion. Realmente, el proceso es bastante sencillo y no toma mucho tiempo, pues tal y como dijimos anteriormente, los Diffusers nos ofrecen la capacidad de poder usar el modelo incluso con solamente una sola línea de código. Sin embargo, antes de poder hacer esto, no podemos avanzar sin realizar el paso que indicamos anteriormente. Como dijimos, debemos indicarle a nuestro modelo que deseamos, en efecto, usar la

tecnología de CUDA dentro de la generación de las imágenes de este modelo. Esto lo podemos hacer sencillamente con el siguiente bloque de código:

```
1  pipe = pipe.to("cuda")
```

Y listo, con esto finalmente le estamos indicando a nuestro programa y entorno de la librería Diffusers que queremos hacer uso de la tecnología de Cuda. Esto, claramente, partiendo del supuesto de que inicialmente dentro de nuestro programa, antes de empezar con el entorno de ejecución, se indicó dentro de Google Colab que se haría uso de la GPU como unidad de procesamiento principal.

Ahora es el momento tan esperado en que podemos usar el modelo. Como sabemos, anteriormente guardamos en una variable el 'prompt' como una cadena de texto que queríamos usar dentro del modelo. Este 'prompt' es simplemente una variable que contiene el texto de lo que queremos que haga el modelo. La variable recibe el nombre de "prompt". Con esto en mente, podemos ejecutar el siguiente bloque de código para que el modelo genere la imagen que nosotros le solicitamos a través de este 'prompt':

```
1  image = pipe(prompt).images[0]
```

Por lo tanto, estamos indicando a nuestro programa que queremos que todo el pipeline previamente configurado de Stable Diffusion genere la imagen específica que hemos seleccionado. Este proceso creará la imagen deseada y la almacenará en la variable "image". Podemos ver la salida que este comando genera en la terminal a continuación:

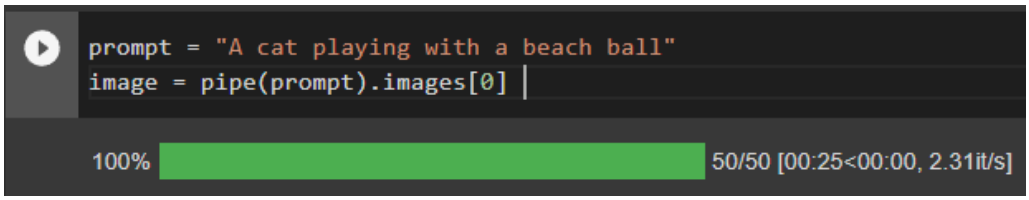


Ilustración 75 – Salida de la generación de imagen usando el pipeline de Stable Diffusion.

¡Listo! Con esto, podemos afirmar que hemos generado nuestra primera imagen. Como podemos observar en la salida de la terminal, la imagen se creó con un total de 50 épocas. Esto significa que el proceso iterativo para eliminar el ruido se llevó a cabo un total de 50 veces. Con este logro, podemos concluir que hemos aprendido a generar imágenes utilizando Stable Diffusion para la creación de imágenes, aplicando inteligencia artificial localmente en Google Colab.

Otro aspecto importante que debemos tener en cuenta respecto al comando que usamos anteriormente es el hecho de que, como podemos observar, al momento de generar la imagen le indicamos a nuestro programa al final del código un

".images[0]". Esto solamente indica que muestre la primera imagen generada. Sin embargo, como sabemos, realmente solo estamos indicando al modelo que queremos generar una única imagen, por lo que si indicamos un índice diferente a 0 simplemente obtendremos un error de índice. En el caso de que tuviéramos varias imágenes, entonces, es muy probable que usemos un valor diferente dependiendo de la imagen en cuestión que queremos ver que ha generado el modelo.

Sin embargo, a pesar de que hemos generado la imagen, aún no la hemos visualizado. Para poder mostrar la imagen generada en nuestra terminal, simplemente podemos llamar a la variable, tal y como se muestra en el siguiente bloque de código:

```
1 image
```

Esto nos mostrará la imagen que ha sido generada por el modelo, tal y como podemos ver a continuación:

Generación Stable Diffusion 39 – A cat playing with a beach ball (Un gato jugando con una pelota de playa)

¡Felicidades! Con esto podemos decir que finalmente hemos aprendido cómo podemos usar Stable Diffusion de una forma local usando Google Colab. Esto sin la necesidad de esperar colas o usar otros servicios ofrecidos por diferentes compañías que brindan la posibilidad de usar Stable Diffusion. No obstante, tenemos la ventaja de poder entender cómo podemos usar este modelo localmente sin ningún tipo de limitante. De nuevo, esta es la forma más sencilla de usar el modelo, y tal como hemos visto a lo largo de este libro, las posibilidades al momento de usar Stable Diffusion son vastas. Por lo tanto, sabemos que hay un mundo mucho más amplio por explorar y estudiar en lo que respecta a la generación de imágenes usando inteligencia artificial.

A partir de aquí, lo que realmente podemos observar es lo que falta para poder entender un poco más cómo podemos generar provecho de Stable Diffusion y su capacidad para generar imágenes. Este hecho se evidencia al ver cómo podemos variar con diferentes prompts, cantidad de imágenes, épocas, etcétera. Sin

embargo, podemos decir que el propósito principal de esta larga sección finalmente concluye, y podemos entender cómo se puede usar este modelo de una forma más local sin la necesidad de otras dependencias.

Ahora, con esto tenemos bastante más posibilidad de lo que podemos hacer con Stable Diffusion. Por ejemplo, podemos cambiar el prompt o indicar de forma directa que queremos mostrar la imagen, tal como podemos hacer con el siguiente bloque de código:

```
1  pipe(“A nice dancing cactus”).images[0]
```

Esto eliminará la necesidad de llamar a una variable, ya que la imagen generada se mostrará inmediatamente sin ningún tipo de espera, tal como se puede apreciar en la siguiente imagen:

Generación Stable Diffusion 40 – “A nice dancing cactus” (Un lindo cactus bailando)

Podemos seguir experimentando con este modelo tanto como deseemos, ya que podemos generar las imágenes que queramos simplemente cambiando el texto del comando, tal y como podemos ver a continuación con los siguientes ejemplos:

```
1 pipe("An illustration of a dog walking in Saturn").images[0]
```

Generación Stable Diffusion 41 – "An illustration of a dog walking in Saturn" (Ilustración de un perro caminando en Saturno)

```
1 pipe("An astronaut riding a horse").images[0]
```

Generación Stable Diffusion 42 – An astronaut riding a horse (Un astronauta montando un caballo)

```
1 pipe("A cat playing the piano").images[0]
```

Generación Stable Diffusion 43 – A cat playing the piano (Un gato tocando el piano)

Como podemos ver, los resultados generados por el modelo corresponden, en efecto, la mayoría de las veces a lo que solicitamos. Sin embargo, también podemos notar que las imágenes generadas por los modelos realmente no son de muy buena calidad; podemos ver que algunas no son cien por ciento exactas, mientras que otras presentan algunas deformaciones al momento de ser generadas. Esto ocurre principalmente debido a que, hasta el momento, no hemos modificado ni personalizado del todo el modelo. Es decir, aún hay muchos más elementos que hay que considerar para poder decir que generamos imágenes de muy buena calidad, como esperamos de este tipo de modelos.

Sin embargo, tener una sola imagen que no podemos modificar puede llegar a ser algo aburrido. Por esto, también podría ser conveniente ver cómo podemos modificar aún más las generaciones de nuestro modelo de tal forma que se ajusten a nuestras necesidades o expectativas. Por eso, a continuación, veremos las diferentes formas que tenemos para poder modificar de una manera bastante sencilla las imágenes que estamos generando. Realmente no es para nada complejo y es muy fácil de entender.

Generación de múltiples imágenes

Ahora, el primer parámetro que veremos que podemos modificar dentro de las generaciones del modelo corresponde a la cantidad total de imágenes. Esto realmente es bastante sencillo, pero como vimos anteriormente puede llegar a ser algo aburrido tener que generar siempre la misma cantidad de imágenes sin tener varias opciones a elegir. Para poder generar varias imágenes con este modelo, solamente tenemos que ejecutar el siguiente comando:

```
1 imagenes = pipe(["An helicopter drinking tea"] * 3)
```

Realmente podemos ver que no cambia mucho con lo que hacíamos anteriormente. Realmente, lo único que cambia es la forma en la que presentamos el prompt. Podemos ver que dentro de una lista de un único elemento se guarda el string del prompt que se desea usar. Luego, se multiplica por 3, lo que hace es que convierte la lista de un único elemento en una lista de 3 elementos. Por lo tanto, el modelo itera por esta lista, generando así 3 imágenes diferentes del mismo prompt. Mientras tanto, esto se guarda dentro de la variable "imagenes". Para poder acceder a las diferentes imágenes, podemos usar los siguientes comandos:

```
1 imagenes.images[0]
2 imagenes.images[1]
3 imagenes.images[2]
```

Es importante notar que esto preferiblemente debería hacerse en celdas diferentes, pues si bien los tres comandos pueden ejecutarse dentro de una misma celda, esto hará que las imágenes no se muestren adecuadamente. Una vez que ejecutemos estos comandos, entonces, obtendremos de salida las imágenes generadas por el modelo, como podemos ver a continuación en este caso con las imágenes generadas:



Generación Stable Diffusion 44 – "An helicopter drinking tea" (Un helicóptero bebiendo té)

Como podemos observar, realmente las imágenes no son del todo acordes a lo que solicitamos al modelo. Esto es principalmente debido a varios factores como, por ejemplo, la versión del modelo y, también, a la falta de optimización del mismo. Sin mencionar el hecho de que la solicitud que se le hace al modelo es poco realista.

Conclusión del capítulo

El capítulo ha demostrado con éxito cómo usar Stable Diffusion para generar imágenes en un entorno local mediante Google Colab. A través de una explicación detallada y numerosos ejemplos, se ha ilustrado cómo se puede aplicar la tecnología de CUDA para optimizar el proceso y cómo se puede generar no solo una imagen, sino múltiples imágenes con diferentes parámetros. A pesar de la sencillez del proceso, se ha destacado que la calidad y exactitud de las imágenes generadas pueden variar, y que hay margen para la optimización y personalización. En resumen, este capítulo ofrece una introducción sólida y práctica a la generación de imágenes con Stable Diffusion, sentando las bases para exploraciones y aplicaciones más avanzadas en futuros estudios y desarrollos.

SEMILLAS PERSONALIZADAS

Objetivos del capítulo

El objetivo de este capítulo es profundizar en la comprensión y la manipulación de parámetros específicos en la generación de imágenes mediante el modelo Stable Diffusion. Se explicará cómo ajustar la semilla de generación para controlar los valores pseudoaleatorios en la generación de ruido gaussiano, utilizando la librería Torch, y se explorará el parámetro correspondiente al número de épocas iterativas en el proceso de generación. Además, se ofrecerá una reflexión sobre el futuro esperado de Stable Diffusion y su relevancia dentro de la inteligencia artificial, extendiendo la discusión hacia el impacto de la IA en el mundo laboral y las capacidades multidimensionales de los modelos de difusión.

Ahora, exploraremos cómo podemos ajustar otro parámetro en la generación de imágenes de este modelo. En este caso, nos referimos al parámetro de la semilla de generación. Tal como mencionamos anteriormente, el papel de la semilla de generación es iniciar un valor numérico que indicará la secuencia de los valores pseudoaleatorios en la generación de ruido Gaussiano. Podemos modificar el valor de la semilla al declarar el uso del pipeline, como se puede observar en el siguiente bloque de código:

```
1  import torch
2  semilla = torch.Generator(“cuda”).manual_seed(727)
```

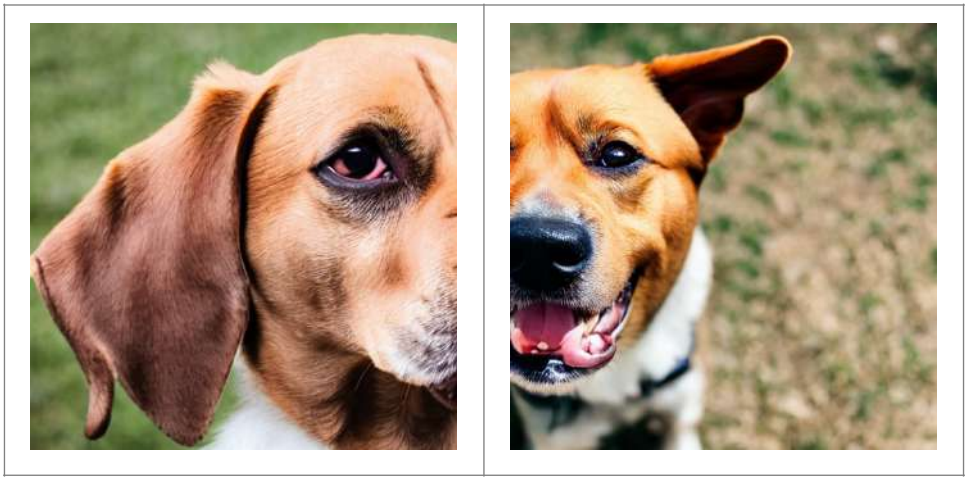
Para poder realizar esto, haremos uso de la librería Torch, que es una biblioteca bastante popular para la aplicación de inteligencia artificial en Python. Una vez que hemos importado la librería, simplemente debemos indicar de nuevo que queremos hacer una generación usando CUDA. Luego, especificamos la semilla de generación que deseamos usar en el modelo. En este caso, podemos ver que el valor que utilizamos para la semilla es el número 727. Pero ¿cómo podemos verificar si la semilla realmente tiene algún efecto? Podemos hacer esto generando dos imágenes, donde ambas imágenes en teoría deberían ser las mismas al recibir la misma semilla. Podemos ver esto con el siguiente bloque de código:

```
1  imagenes = pipe(["A dog smiling"] * 2, generator=semilla)
```

Por lo tanto, el modelo se encargará de generar las dos imágenes. Podemos verificar y visualizar ambas imágenes de la misma forma que usamos anteriormente para ver múltiples imágenes, tal y como podemos ver a continuación con el siguiente bloque de código:

```
1  imagenes.images[0]
2  imagenes.images[1]
```

Lo cual genera a las siguientes imágenes:



Generación Stable Diffusion 45 – "A dog smiling" (Un perro sonriendo)

Como podemos apreciar, el modelo genera imágenes completamente distintas entre sí. Este fenómeno se produce porque ambas imágenes se generan dentro de la misma etapa de generación. En este proceso, el modelo toma una semilla y la lista de números pseudoaleatorios se emplea para la creación de imágenes. Sin embargo, si replicáramos este proceso de generación, incluso tras reiniciar el entorno de ejecución, comprobaríamos que las imágenes resultantes son

idénticas. Por tanto, podríamos afirmar que la semilla ha desempeñado eficientemente su función.

Número de épocas

Ahora, finalmente, examinaremos el último parámetro que modificaremos en estos ejemplos. Como sabemos, en realidad, los parámetros que podemos modificar dentro de este modelo son muchos más. Un ejemplo sería el caso de los parámetros del prompt negativo. Sin embargo, por ahora, este será el último. Este parámetro corresponde efectivamente al número de épocas iterativas del modelo de eliminación de ruido en el proceso de generación de imágenes. Podemos cambiar el número de épocas de una forma bastante sencilla. Solo tenemos que ejecutar el siguiente bloque de comandos:

```
1  pipe("A ballerina dancing in the night", num_inference_steps=30).images[0]
```

Esto generará la siguiente imagen como resultado después de haber realizado un total de 30 iteraciones dentro de la imagen que se desea generar.

Generación Stable Diffusion 46 – "A ballerina dancing in the night" (Una bailarina bailando en la noche)

También es posible experimentar con valores de pasos mucho más pequeños. Por ejemplo, si cambiamos el valor de este prompt de arriba, de iteraciones a un valor de 3, obtendremos lo siguiente:

Generación Stable Diffusion 47 – "A ballerina dancing in the night" (Una bailarina bailando en la noche)

CAPÍTULO 9: SECCIÓN FINAL: EL FUTURO ESPERADO DE STABLE DIFFUSION

Felicidades, con esto podemos decir finalmente que hemos concluido todos los capítulos correspondientes de este libro. Ahora, realmente la intención de las secciones finales es proporcionar una breve reflexión sobre el papel que desempeñan los modelos de difusión latente, como es el caso de Stable Diffusion, por ejemplo. No se trata únicamente de comprender el avance tan positivo que representa para la humanidad y la ciencia en general, el cual se convierte en un tipo de conocimiento común. Después de todo, todos somos conscientes del impacto positivo y lo útil que puede llegar a ser la inteligencia artificial como herramienta.

No solo se trata de entender los aspectos negativos que puede representar la inteligencia artificial, sino también reconocer que este tipo de modelos no son nada nuevo y se sabe que han reemplazado y seguirán reemplazando el trabajo de muchas personas. Esto realmente no es del todo un fenómeno nuevo, pues como sabemos, con el paso del tiempo los avances tecnológicos han cambiado los trabajos de muchas personas, y en este caso, podemos decir que la inteligencia artificial no es la excepción a esto.

Sin embargo, es momento de que también veamos el lado positivo de esto, y es que es muy probable que el cambio radical que presenta la inteligencia artificial en la actualidad frente al reemplazo de los trabajos de las personas realmente no ocurrirá al impacto que nosotros estamos imaginando. Por ejemplo, podemos decir y también notar con los diferentes ejemplos que vimos aquí en este libro y también el conocimiento general de otros tipos de modelos de inteligencia artificial que hasta el momento los resultados presentados por el modelo son muy buenos y beneficiosos para las personas.

Sí, es cierto, y aunque han demostrado una capacidad de poder realizar tareas de una forma simplemente increíble, la realidad es que estos modelos de inteligencia artificial aún se encuentran algo lejanos de poder llegar a un punto donde de verdad consideremos que los modelos de inteligencia artificial se han vuelto absolutamente perfectos, a tal punto que incluso será posible considerar el hecho de que este tipo de modelos de inteligencia artificial son en esencia superiores a los seres humanos.

Sin embargo, la realidad es otra, y tal como mencionamos en secciones pasadas de este libro, pudimos entender que estos modelos requieren aun así el control de un ser humano. Después de todo, sabemos que este tipo de modelos son conexiones digitales que están hechas para meramente realizar una tarea en la

cual hayan sido entrenados, no se trata de seres vivientes, sino de herramientas que realmente pueden ser bastante beneficiosas para potenciar el trabajo de las personas.

Ahora, frente a los diferentes modelos de inteligencia artificial, en este caso, también es importante entender un poco más sobre el futuro de los modelos de inteligencia artificial basados en la técnica de difusión para remoción de ruido. Por ahora, entendimos un poco más a fondo la capacidad que esta técnica presenta para poder generar imágenes. Es decir, pudimos ver que realmente este tipo de modelos son realmente rápidos y eficientes para poder generar cosas partiendo de absolutamente nada, es decir, de solamente ruido Gaussiano. Pero ¿realmente esta técnica se centra solamente en la generación de imágenes como el modelo que vimos actualmente? La realidad es que no, no solamente se centra en la generación de imágenes.

Podemos entender un poco más al respecto de lo que estamos mencionando si nos referimos, por ejemplo, a la teoría de los Diffusers. Aquí podemos ver que la capacidad de los modelos generativos no solamente se centra en las imágenes, sino también en audio e incluso en la capacidad de generar modelos en 3D, lo cual es realmente sorprendente. Sin embargo, el uso de este tipo de modelos de inteligencia artificial para imágenes sigue siendo la aplicación más común.

Podemos observar con más detalle lo que estamos mencionando en la imagen siguiente. En ella, podemos confirmar que, efectivamente, la biblioteca de los Diffusers no solo tiene la capacidad de generar contenido de imágenes, sino también de otras modalidades como audio y modelos en 3D.

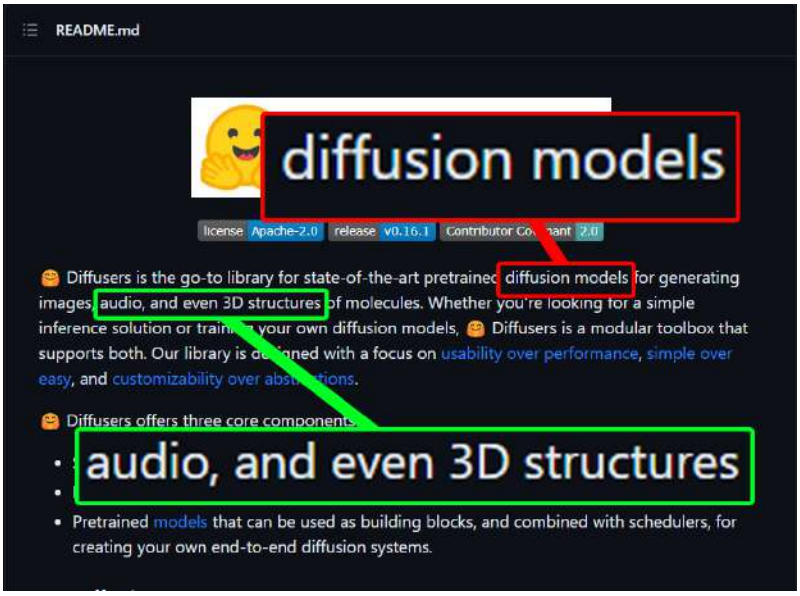


Ilustración 76 – Mención de las capacidades de generación de los modelos de difusión en la librería Diffusers.

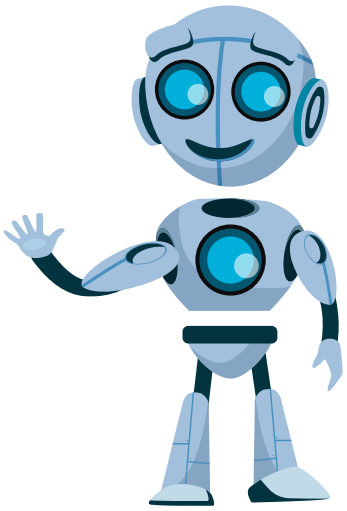
Esto implica que es probable que, con el transcurso del tiempo, surjan numerosas aplicaciones adicionales para los modelos de difusión. Por el momento, no solo debemos enfocarnos en estos modelos, sino también considerar la posibilidad de aplicar esta técnica en la generación de una gran cantidad de contenido, como la generación de video y la limpieza de señales, entre otros. Las oportunidades para explorar esto son, de hecho, infinitas. A pesar de ello, seguir profundizando en las posibilidades de aplicación de estos modelos de inteligencia artificial continúa siendo un campo sumamente interesante. Además de estar a la vanguardia, la capacidad de estos modelos seguirá progresando constantemente, tanto para la comunidad de la inteligencia artificial como para la ciencia en general.

El futuro de los modelos de difusión es incierto y nadie sabe realmente hasta dónde nos llevarán. Sin embargo, una cosa es segura: es un campo fascinante para investigar y explorar. Desde automatizar procesos y generar contenido de alta calidad hasta desarrollar modelos capaces de llevar a cabo múltiples tareas de manera multimodal, las capacidades de la inteligencia artificial son prácticamente ilimitadas y están mucho más allá de lo que podemos imaginar.

Es fundamental recordar que Stable Diffusion no es el único modelo generativo de inteligencia artificial disponible en la actualidad. Aunque es uno de los más populares y eficaces, es esencial resaltar que cada vez están surgiendo modelos nuevos y aún más potentes de inteligencia artificial, los cuales pueden realizar diversas tareas. Este modelo, gracias a su naturaleza de código abierto, nos proporciona la oportunidad de utilizarlo como punto de partida para explorar las capacidades que los modelos de inteligencia artificial pueden ofrecer. Por ende, se trata de mantenernos bien informados y actualizados acerca del lanzamiento de modelos cada vez más novedosos y recientes.

Te felicito por concluir la lectura de este libro. Ha sido un trayecto considerablemente largo, pero podemos afirmar con toda certeza que lo aprendido en este proceso ha resultado sumamente enriquecedor y provechoso. Por ahora, este libro representa tan solo una breve introducción a las capacidades de estos modelos, pero la investigación y el desarrollo que pueden surgir de lo que hemos explorado a lo largo de estas páginas son prácticamente ilimitados; las posibilidades son inmensas. La inteligencia artificial es una potente herramienta que puede realzar el trabajo de las personas. De hecho, no se trata de que la inteligencia artificial sustituya nuestras labores, sino que debemos adaptarnos y acoger las innovaciones y beneficios que el uso de estas herramientas nos brinda.

Conclusión del capítulo



Este capítulo ha permitido desvelar los aspectos técnicos avanzados de la generación de imágenes con el modelo Stable Diffusion, incluyendo la manipulación de la semilla de generación y el número de épocas. A través de ejemplos prácticos y códigos, se ha ofrecido una guía detallada para experimentar con estos parámetros. Además, se ha planteado una reflexión crítica sobre el futuro de los modelos de difusión, resaltando tanto los desafíos como las oportunidades. La conclusión aborda la importancia de la inteligencia artificial como herramienta potenciadora, no solo en la generación de imágenes sino en diversos ámbitos de aplicación, y enfatiza la necesidad de seguir explorando y manteniéndonos informados sobre los avances en este campo fascinante y en constante evolución.

CAPÍTULO 10: EJERCICIOS PRÁCTICOS

Ahora hemos llegado a la sección final del libro. Hasta este punto, hemos explorado en detalle tanto la teoría como las aplicaciones prácticas de Stable Diffusion. Sin embargo, para cerrar la brecha entre el conocimiento teórico y la implementación real, esta última parte se enfocará en ejercicios y ejemplos concretos. El objetivo es permitir una mejor comprensión práctica del modelo y su aplicabilidad en situaciones del mundo real.

Es crucial subrayar que estos ejercicios están diseñados para ofrecer una gama de niveles de dificultad. De este modo, independientemente de su experiencia o familiaridad con los modelos de inteligencia artificial, encontrará desafíos apropiados que le permitirán aplicar de manera efectiva lo aprendido en este libro. Estos ejercicios están pensados para ofrecer una ruta escalonada y gradual hacia una comprensión más profunda, facilitando así su aplicación en entornos reales.

Con esto en mente, esperamos que esta sección le resulte útil para adquirir una experiencia más tangible en el uso de Stable Diffusion en casos de aplicación real.

EJERCICIOS INTRODUCTORIOS (PRIMER NIVEL)

En esta sección ofrecemos una serie de ejercicios preliminares que tienen como objetivo introducir los principios esenciales de Stable Diffusion. Es crucial entender que, aunque estos ejercicios enfatizan los fundamentos del modelo, no abarcan la totalidad de aplicaciones que son viables en el ámbito de la inteligencia artificial. A medida que avance el libro, profundizaremos en las diversas formas en las que Stable Diffusion puede ser empleado de manera efectiva. Es importante señalar que los ejercicios aquí presentados constituyen una introducción práctica, aunque no exhaustiva, al potencial de este innovador modelo de inteligencia artificial.

Adicionalmente, es relevante hacer notar que los ejercicios que se desarrollarán a lo largo del libro no se limitan exclusivamente al código vinculado con Stable Diffusion. También exploraremos el rango de habilidades que este libro puede ofrecer para ampliar su comprensión de los desarrollos actuales en la industria de la inteligencia artificial. Por lo tanto, incluiremos ejercicios que, aunque no estén directamente relacionados con Stable Diffusion, contribuirán a una mejor

comprensión de los desafíos y oportunidades en el campo de la inteligencia artificial.

Ejercicios Prácticos y Evaluativos	
Dificultad: Fácil	
1	**Análisis de Sector** Elija un sector industrial que no se haya abordado en el libro y redacte un breve informe sobre cómo la inteligencia artificial
2	**Identificación de Aplicaciones de IA** Encuentra tres aplicaciones móviles o de software que utilicen inteligencia artificial de alguna forma. Describe brevemente cómo
3	**Interpretación de Imágenes Generadas** Usa una versión online de Stable Diffusion para generar una imagen basada en un 'prompt' de texto extenso y muy detallado. Analiza los resultados, describiendo qué partes del 'prompt' se representaron

EJERCICIOS INTERMEDIOS (SEGUNDO NIVEL)

Ahora que hemos revisado los ejercicios anteriores, podemos observar que su nivel de complejidad es relativamente bajo. Hasta este punto, no hemos explorado en detalle ejemplos o aplicaciones prácticas que empleen los conceptos técnicos de Stable Diffusion discutidos en este libro. Con este contexto en mente, presentaremos a continuación una serie de ejercicios de mayor dificultad en comparación con los que hemos visto hasta ahora.

Ejercicios Prácticos y Evaluativos	
Dificultad: Medio	
1	**Configuración de Parámetros** Utilizando un 'prompt' meticulosamente detallado, experimente con diversas configuraciones del parámetro "guidance scale", así como con textos a libre elección para el "negative prompt" en Stable Diffusion. Documente cómo estas variaciones en los parámetros
2	**Comparación de Generativos** Utilice Stable Diffusion junto con al menos dos modelos generativos adicionales, como por ejemplo DALL-E 2, para generar imágenes a partir de un 'prompt' idéntico. Examine y contraste las diferencias en los resultados, así como las ventajas particulares que cada modelo de inteligencia artificial aporta al proceso de generación

	Configuración de Parámetros (Ampliado)
3	Además de "guidance scale" y "negative prompt", experimente con el parámetro "steps" para generar imágenes en Stable Diffusion. **Estudio de Casos Ejemplares**
4	Elija uno de los casos ejemplares discutidos en el libro y profundice en él. Investigue sobre cómo se podrían aplicar otros modelos de IA, además de Stable Diffusion, para resolver problemas **Implementación de ControlNet**
5	Implemente el modelo ControlNet para guiar o modificar las imágenes generadas mediante Stable Diffusion. Registre meticulosamente tanto los resultados como sus observaciones. Se recomienda llevar a cabo experimentos con imágenes de dibujo vectorial para evaluar cómo las directrices proporcionadas al modelo interactúan con las **Comparativa de Versiones**
6	Utilice al menos dos versiones diferentes de Stable Diffusion (por ejemplo, v1.1 y v2.0) y genere imágenes a partir del mismo 'prompt'. Analice y discuta las diferencias entre las versiones en **Análisis de Etiquetas en el Proceso de Difusión**
7	Utilice un conjunto de etiquetas predefinidas y examine cómo afectan el resultado final al aplicarlas durante el proceso de difusión en diferentes prompts. Por ejemplo, si usas etiquetas como **Integración con Otras Herramientas de IA**
8	Implemente un proyecto pequeño que utilice Stable Diffusion en conjunto con otra herramienta de IA, como un modelo de lenguaje

EJERCICIOS AVANZADOS (TERCER NIVEL)

Hasta este punto, hemos explorado ejercicios que abarcan niveles de dificultad tanto básicos como intermedios en la aplicación y evaluación de Stable Diffusion. A continuación, presentaremos una serie de ejercicios avanzados para profundizar aún más en este tema.

Ejercicios Prácticos y Evaluativos	
Dificultad: Difícil	
	Análisis de Limitaciones y Soluciones
1	Revise cualquier versión de Stable Diffusion y identifique sus limitaciones. Proponga formas de abordar estas limitaciones, ya sea

	Experimento con Ruido Gaussiano
2	Diseñe y lleve a cabo un experimento en el cual se apliquen distintos grados de ruido gaussiano a un conjunto seleccionado de imágenes. Observe y analice el impacto que este ruido tiene en la calidad de las imágenes resultantes generadas por Stable Diffusion. **Instalación y Uso Local**
3	Instale Stable Diffusion en su máquina local o en un entorno como Google Colab. Genere una imagen utilizando los pasos y el código proporcionado en el libro y discuta cualquier desafío que haya **Personalización de Modelo**
4	Investigue cómo podría personalizar Stable Diffusion para una aplicación específica. ¿Qué cambios en el modelo o en los **Optimización de Recursos**
5	Evalúe el rendimiento del modelo Stable Diffusion en diferentes entornos de hardware. Proporcione recomendaciones sobre cómo optimizar el uso de recursos sin sacrificar demasiado la calidad de **Extensión a Otros Tipos de Datos**
6	Aunque Stable Diffusion se emplea principalmente en la generación de imágenes, explore la posibilidad de aplicar este modelo a otros tipos de datos, como audio o texto. Basándose en su investigación, ofrezca un análisis sobre si Stable Diffusion debería considerarse como un método viable para la creación de contenido multimedia más **Modificación del Código Fuente**
7	Clone el repositorio de GitHub de Stable Diffusion y realice modificaciones menores en el código fuente para cambiar el **Exploración Ética**
8	Investigue y redacte un informe acerca de las cuestiones éticas asociadas con el uso de Stable Diffusion, incluyendo temas sensibles como la creación de deepfakes o la apropiación indebida de contenido. Formule estrategias o medidas preventivas para atenuar la incidencia de estos dilemas éticos. ¿Existe en la

Con estos ejercicios, ahora disponemos de un marco práctico que nos permite explorar y profundizar en las aplicaciones concretas de Stable Diffusion. Además, estos ejercicios ofrecen una oportunidad para evaluar y aplicar los diversos conceptos abordados a lo largo del libro.

BIBLIOGRAFÍA Y REFERENCIAS

Alvarado, M., & Meneses-Bautista, F. D. (2017). Pronóstico del tipo de cambio USD/MXN con redes neuronales de retropropagación. Research in Computing Science 113(1), 97-110.

Bennett, J., & Lanning, S. (2007). The Netflix Prize. Netflix.

Britannica. (2023). Tesla, Inc. | History, Cars, Elon Musk, & Facts | Britannica. Obtenido de Encyclopedia Britannica: https://www.britannica.com/topic/Tesla-Motors

Bryant, L. V. (2020). The YouTube Algorithm and the Alt-Right Filter Bubble. Open Information Science, vol. 4, no. 1, 85-90.

Charles Sturt University. (2023). Home - International.

Claughton, D., & Condon, M. (27 de Mayo de 2021). Robots and artificial intelligence to guide Australia's first fully automated farm. Australian Broadcasting Corporation.

cmdr2 (GitHub). (2023). cmdr2/stable-diffusion-ui: Easiest 1-click way to install and use Stable Diffusion on your computer. Provides a browser UI for generating images from text prompts and images. Just enter your text prompt, and see the generated image. Obtenido de GitHub: https://github.com/cmdr2/stable-diffusion-ui

Comunidad de Software Libre Hackem [Research Group]. (3 de Junio de 2020). Computer vision sample using cvlib library of python programming language.

DeepFloyd Lab. (2023). deep-floyd/IF. Obtenido de GitHub: https://github.com/deep-floyd/IF

DeepMind. (2022). A Generalist Agent. Transactions on Machine Learning Research.

DeepMind. (Marzo de 2022). AlphaGo. DeepMing Research.

Deng, L., Bradski, G., Coates, A., & Shah, M. (7 de Julio de 2017). Ask the AI experts: What's driving today's progress in AI? (S. London, Entrevistador)

Ensmenger, N. (2012). Is chess the drosophila of artificial intelligence? A social history of an algorithm. Soc Stud Sci., 5-30.

Forbes Media LLC. (6 de Febrero de 2023). Netflix | NFLX Stock Price, Company Overview & News. Obtenido de Forbes: https://www.forbes.com/companies/netflix/

Gans, J. S. (4 de Octubre de 2022). Artificial intelligence adoption in a monopoly market. Wiley Online Library, págs. 1098-1106.

Glikson, E., & Woolley, A. W. (10 de Agosto de 2020). Human Trust in Artificial Intelligence: Review of Empirical Research. Academy of Management Annals, págs. 627-660.

Government of Jharkhand. (9 de Febrero de 2023). Transport Department, Government of Jharkhand. Obtenido de Causes of Road Accidents | Transport Department, Government of Jharkhand: https://jhtransport.gov.in/causes-of-road-accidents.html

Handelman, G. S., Kok, H. K., Chandra, R. V., Razavi, A. H., Huang, S., Brooks, M., . . . Asadi, H. (2019). Peering Into the Black Box of Artificial Intelligence: Evaluation Metrics of Machine Learning Methods. Neuroradiology/Head and Neck Imaging, 38-43.

Hovsepyan, T. (2 de Mayo de 2022). What Are the Stages of Machine Learning Lifecycle? Plat.AI.

Ingle, S., & Phute, M. (2016). Tesla Autopilot : Semi Autonomous Driving, an Uptick for Future Autonomy. International Research Journal of Engineering and Technology (IRJET).

Injury Facts. (9 de Febrero de 2023). Injury Facts - National Safety Council. Obtenido de Historical Car Crash Deaths and Rates - Injury Facts: https://injuryfacts.nsc.org/motor-vehicle/historical-fatality-trends/deaths-and-rates/

Invisibly. (2023). The Full Guide on Netflix Recommendation Algorithm: How does it work? Obtenido de Invisibly: https://www.invisibly.com/learn-blog/netflix-recommendation-algorithm/

Jakhar, D., & Kaur, I. (2020). Artificial intelligence, machine learning and deep learning: definitions and differences. Clinical and Experimental Dermatology, 131-132.

Jha, K., Doshi, A., Patel, P., & Shah, M. (2019). A comprehensive review on automation in agriculture using artificial intelligence. Artificial Intelligence in Agriculture, Volumen 2, 1-12.

Kaynak, O. (22 de Septiembre de 2021). The golden age of Artificial Intelligence. Discov Artif Intell.

KOROSEC, K. (21 de Diciembre de 2015). Elon Musk Says Tesla Vehicles Will Drive Themselves in Two Years. Fortune.

Krishnamurthy, B. (31 de Octubre de 2022). Chess AI: A Brief History | Built In. Obtenido de Built In: https://builtin.com/artificial-intelligence/chess-ai

Ksenija, B., Šverko, D., Salarić, I., Martinović, A., & Lucijanić, M. (2021). The ABC of linear regression analysis: What every author and editor should know. European science editing, 47.

Kurauchi, A. (8 de Julio de 2016). Means, Medians and Images | Andrew Kurauchi. Obtenido de Andrew Kurauchi: http://kurauchi.com.br/post/means-medians-and-images/

Liao, S.-H. (Enero de 2005). Expert system methodologies and applications—a decade review from 1995 to 2004. Department of Management Sciences and Decision Making, págs. 93-103.

M, C. K., Srinivasulu, S., P, N. R., & B, D. N. (2020). Machine learning model for movie recommendation system. Int. J. Eng. Res. Tech, 9, 800-801.

Mahesh, B. (2018). Machine Learning Algorithms - A Review. International Journal of Science and Research (IJSR).

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics volume 5, 115-133.

Microsoft. (2023). ¿Qué es el aprendizaje profundo? | Microsoft Azure. Obtenido de Microsoft Azure: https://azure.microsoft.com/es-es/resources/cloud-computing-dictionary/what-is-deep-learning/

Mondal, B. (2020). Artificial Intelligence: State of the Art. Recent Trends and Advances in Artificial Intelligence and Internet of Things, págs. 389-425.

Netflix, Inc. (6 de Febrero de 2023). Netflix - Watch TV Shows Online, Watch Movies Online. Obtenido de Netflix: https://www.netflix.com/

Newborn, M. (2012). Kasparov versus Deep Blue: Computer Chess Comes of Age.

Nielsen, M. (29 de Marzo de 2016). Is AlphaGo Really Such a Big Deal? Quanta Magazine.

Nowak, A., Lukowicz, P., & Horodecki, P. (4 de Diciembre de 2018). Assessing Artificial Intelligence for Humanity: Will AI be the Our Biggest Ever Advance ? or the Biggest Threat [Opinion]. IEEE, págs. 26-34.

Oracle. (2023). What is Artificial Intelligence (AI)? | Oracle. Obtenido de Oracle: https://www.oracle.com/artificial-intelligence/what-is-ai/

Ornes, S. (12 de Septiembre de 2022). How Transformers Seem to Mimic Parts of the Brain. Quanta magazine.

Our World In Data. (2021). Annual global corporate investment in artificial intelligence - Our World in Data. Obtenido de Our World In Data: https://ourworldindata.org/grapher/corporate-investment-in-artificial-intelligence-total

Our World In Data. (2022). Affiliation of research teams building notable AI systems - Our World In Data. Obtenido de Our World In Data: https://ourworldindata.org/grapher/affiliation-researchers-building-artificial-intelligence-systems-all

Our World In Data. (2022). Chess ability of the best computers - Our World in Data. Obtenido de Our World In Data: https://ourworldindata.org/grapher/computer-chess-ability

Our World In Data. (2023). Computation used to train notable artificial intelligence systems - Our World In Data. Obtenido de Our World In Data: https://ourworldindata.org/grapher/artificial-intelligence-training-computation

Pandey, D., & Agrawal, M. (2014). Carbon Footprint Estimation in the Agriculture Sector. Assessment of Carbon Footprint in Different Industrial Sectors, Volume 1, 25-47.

Pineda, G. F. (s.f.). Aprendizaje de máquinas | Coursera. Obtenido de Coursera: https://es.coursera.org/learn/aprendizaje-maquinas-test

Reese, H. (17 de Noviembre de 2017). Understanding the differences between AI, machine learning, and deep learning. TechRepublic.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., & Ommer, B. (13 de Abril de 2021). High-Resolution Image Synthesis with Latent Diffusion Models. Computer Vision and Pattern Recognition.

Rosebrock, A. (6 de Mayo de 2021). Implementing the Perceptron Neural Network with Python - PyImageSearch. Obtenido de PyImageSearch: https://pyimagesearch.com/2021/05/06/implementing-the-perceptron-neural-network-with-python/

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. Psychological Review, 65(6), 386–408.

Russell, S., & Norvig, P. (2022). Artificial Intelligence: A Modern Approach, 4th US ed.

Sarker, I. H., Kayes, A. S., & Watters, P. (2019). Effectiveness analysis of machine learning classification models for predicting personalized context-aware smartphone usage. Journal of Big Data.

Shatnawi, A., Al-Bdour, G., Al-Qurran, R., & Al-Ayyoub, M. (7 de Mayo de 2018). A comparative study of open source deep learning frameworks. IEEE, págs. 2573-3346.

Shin, Y. (Octubre de 2019). The Spring of Artificial Intelligence in Its Global Winter. IEEE, págs. 71-82.

Shinde, P. P., & Shah, S. (2018). A Review of Machine Learning and Deep Learning Applications. 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), 1-6.

StabilityAI. (24 de Noviembre de 2022). Stable Diffusion 2.0 Release. Obtenido de StabilityAI: https://stability.ai/blog/stable-diffusion-v2-release

Stable Diffusion Art. (29 de Marzo de 2023). How does Stable Diffusion work? - Stable Diffusion Art. Obtenido de Stable Diffusion Art: https://stable-diffusion-art.com/how-stable-diffusion-work/

Stephens, J. (12 de Mayo de 2022). Getting Started with NVIDIA Instant NeRFs | NVIDIA Technical Blog. Obtenido de NVIDIA Developer: https://developer.nvidia.com/blog/getting-started-with-nvidia-instant-nerfs/

Tesla. (2023). About | Tesla. Obtenido de Tesla: https://www.tesla.com/about

Tesla. (2023). Autopilot | Tesla. Obtenido de Tesla: https://www.tesla.com/autopilot

Tesla. (25 de Abril de 2023). Tesla Vehicle Safety Report | Tesla. Obtenido de Tesla: https://www.tesla.com/VehicleSafetyReport

The World Bank Group. (2023). Population, total | Data. Obtenido de World Bank Group: https://data.worldbank.org/indicator/SP.POP.TOTL

Trimble, M. (2023). Artificial Intelligence and Human Intelligence. GRUR International, Volume 72, Issue 1, 1-2.

Tufekci, Z. (2018). YouTube, the Great Radicalizer. The New York Times.

Umbricht, M. L., & Friend, C. R. (27 de Febrero de 2010). A Symbolics 3640 Lisp machine: an early (1984) platform for expert systems. Rhode Island, Estados Unidos.

World Health Organization. (2022). Road traffic injuries. World Health Organization.