5 SPRINT 2 Analizar datos con Python

Utilizar la IA generativa para el desarrollo de software

**Visión general del módulo**

Después de completar este módulo, debería ser capaz de:

* •

Identificar los beneficios del uso de la IA generativa para el desarrollo de software

* •

Describir cómo se utiliza la IA generativa para escribir código

* •

Identificar problemas comunes con código generado por la IA

* •

Utilizar IBM watsonx en una simulación para generar código

Si bien la IA generativa ofrece numerosas ventajas en el desarrollo de software, también presenta varios desafíos Una de las mayores preocupaciones es la posibilidad de generar código incorrecto o de mala calidad. Los modelos de IA se basan en datos existente y, si esos datos contienen errores o malas prácticas de codificación, la IA podría reproducir estos problemas. Además, el código generado por IA puede no entender el contexto y los requisitos que posee un desarrollador humano, lo que da lugar a código con errores o, peor aún, **alucinaciones**.

**Las alucinaciones**se producen cuando la IA produce información o códigos incorrectos, engañosos o totalmente inventados. Por ejemplo, la IA podría sugerir un fragmento de código que en realidad no resuelve su problema o incluso puede crear errores porque no comprende el contexto.

PRECISIÓN Y ERRORES

A veces, el código no funciona como se espera o tiene errores. La IA puede generar código que parece correcto pero no realiza la tarea correctamente.

**Ejemplo:**Si le solicita a la IA que escriba una función para calcular la media de una lista de números, el código podría funcionar en la mayoría de los casos, pero podría fallar si no maneja situaciones especiales, como una lista vacía. Si el código no comprueba si hay una lista vacía, podría dar un error o un resultado incorrecto , como dividir por cero. Esto puede causar problemas inesperados y errores que deben corregirse para garantizar que el código funcione correctamente.

MALENTENDIDO DE CONTEXTO

Es posible que la IA no comprenda completamente los detalles de su solicitud o el contexto general, lo que puede dar lugar a un código que no satisfaga sus necesidades.

**Ejemplo:**Si le pide a la IA que genere código para procesar los comentarios de los clientes pero no especifica que necesita gestionar tanto las valoraciones de texto como las numéricas, la IA podría crear código que sólo procese los comentarios de texto. Esto significa que las valoraciones numéricas se perderían, lo que daría lugar a un tratamiento incompleto de los datos y a la pérdida de información importante.

CALIDAD DEL CÓDIGO

El código generado por IA podría funcionar, pero podría no estar bien organizado o no ser fácil de entender. Esto puede dificultar el mantenimiento o la actualización posterior del código.

**Ejemplo:**Puede que el código funcione correctamente, pero no esté escrito con claridad ni utilice convenciones de nomenclatura coherentes. Esto puede resultar confuso para alguien que necesite trabajar con él más adelante. Por ejemplo, el código puede utilizar nombres de variables imprecisos o carecer de comentarios que expliquen qué hace cada parte, lo que dificulta su comprensión o modificación en el futuro.

PROBLEMAS DE SEGURIDAD

El código generado por la IA podría tener fallos de seguridad si no sigue las mejores prácticas para proteger los datos y evitar accesos no autorizados.

**Ejemplo:**Si le pide a la IA que cree un sistema de inicio de sesión para un sitio web, es posible que el código generado permita a los usuarios iniciar sesión pero podría no utilizar el cifrado para proteger las contraseñas. También es posible que no compruebe si los datos introducidos son seguros, lo que permitiría a los usuarios malintencionados aprovecharse de estos puntos débiles y obtener acceso no autorizado a información confidencial.

|  |  |
| --- | --- |
| Examine el código en busca de errores evidentes, como puntos y comas o erratas. | Ejecute el código en un entorno de desarrollo que resalte los errores de sintaxis. Muchos entornos de desarrollo integrados (IDE) incorporan herramientas que ayudan en esta tarea. |

|  |  |
| --- | --- |
| Ejecute el código con diferentes entradas para ver si se comporta como se espera. | Cree casos de prueba que cubran una serie de escenarios, incluidos casos extremos como números muy grandes y entradas vacías. |

|  |  |
| --- | --- |
| Lea cualquier comentario en el código o documentación externo para comprender qué se supone que debe hacer. | Busque comentarios dentro del código o en la documentación externa que explique el propósito y uso del código. |

|  |  |
| --- | --- |
| Si el código es demasiado complejo, divídalo en partes más pequeñas y pruebe cada una por separado. | Aísle secciones del código y pruébelas por separado para identificar dónde podrían estar fallando las cosas. |

|  |  |
| --- | --- |
| Si está atascado, pida ayuda en comunidades o foros en línea | Publique una descripción clara del problema, incluido el código y lo que intenta conseguir, para obtener asesoramiento de programadores experimentados. |

**Conceptos clave**

1. 1

La IA generativa es un tipo de inteligencia artificial que puede generar nuevo contenido.

1. 2

Beneficios de utilizar la IA generativa para el desarrollo de software:

* + Puede tomar unas pocas instrucciones suyas y convertirlas en un programa completo.
  + Puedo sugerir cómo completar ciertas secciones del código o incluso escribir nuevo partes por usted.
  + Puede encontrar y corregir errores en el código.
  + Puede ahorrar tiempo encargándose de tareas de codificación tediosas.
  + Facilita aprender a programar.

1. 3

Una solicitud es como dar a la IA una serie de instrucciones claras sobre lo que quiere que cree.

1. 4

Proceso de escritura de código mediante la IA generativa:

* + Usted proporciona una solicitud clara.
  + La IA interpreta la petición.
  + La IA genera el código.

1. 5

Razones para convertir código de un lenguaje de programación en otro:

* + Le brinda la flexibilidad de trabajar con el lenguaje de su elección.
  + Permite que el software se ejecute en una nueva plataforma que requiere un lenguaje diferente.
  + Permite la integración con otros sistemas o software construidos utilizando un lenguaje de programación diferent.
  + Mejora el mantenimiento, el rendimiento y la seguridad de los sistemas heredados.

1. 6

Problemas comunes encontrados en el código generado por IA

* + La IA puede generar código que parece correcto pero no realizar la tarea correctamente.
  + Es posible que la IA no comprenda completamente los detalles o el contexto general de su solicitud.
  + El código generado por IA podría funcionar, pero podría no estar bien organizado o no ser fácil de entender.
  + El código generado por IA podría tener fallos de seguridad.

**Objetivos del curso**

Ahora que ha completado este módulo, debería poder:

* •

Identificar los beneficios del uso de la IA generativa para el desarrollo de software

* •

Describir cómo se utiliza la IA generativa para escribir código

* •

Identificar problemas comunes con código generado por la IA

* •

Utilizar IBM watsonx en una simulación para generar código

Python para ciencia de datos

La duración es20 horas