

APLICACIÓN DE LA INTELIGENCIA ARTIFICIAL EN LA ENSEÑANZA DE LA INGENIERÍA

APPLICATION OF ARTIFICIAL INTELLIGENCE IN ENGINEERING TEACHING

S. Martínez Díaz¹
I. Castro Liera²
I. M. Santillán Méndez³

RESUMEN

La inteligencia artificial es un área de las ciencias computacionales que ha crecido rápidamente en los últimos años. El desarrollo de herramientas computacionales, tales como los chats inteligentes, han revolucionado el mundo en muchos aspectos. Uno de los grandes retos para los docentes es la incorporación de este tipo de herramientas en el proceso de enseñanza-aprendizaje. Más allá de los debates que se han generado en torno al tema, dichas herramientas ya se encuentran a la disposición de todos, y pueden ser utilizadas de forma benéfica en la educación. Para aprovechar al máximo este tipo de tecnología, es necesario un conocimiento básico de qué son y cómo funcionan. Esto puede reducir también los riesgos de hacer mal uso de ella. En este trabajo se presentan de forma básica los aspectos teóricos del diseño de los sistemas generativos, así como, el funcionamiento de los chats inteligentes, y algunos ejemplos de su uso en la preparación de material de clase.

ABSTRACT

Artificial intelligence is an area of computer science that has grown rapidly in recent years. The development of computational tools, such as intelligent chats, have revolutionized the world in many aspects. One of the great challenges for teachers is the incorporation of this type of tools in the teaching-learning process. Beyond the debates that have been generated around the topic, these tools are now available to everyone and can be used beneficially in education. To get the most out of this type of technology, it is necessary a basic understanding of what it is and how it works. This can also reduce the risks of misusing it. In this work, the theoretical aspects of the design of generative systems are presented in a basic way, as well as the operation of intelligent chats, and some examples of their use in the preparation of class material.

ANTECEDENTES

La preparación de material de clase, especialmente para algunas materias del área de ingeniería, presenta varios retos como son la extensión del programa de estudios, la obsolescencia de la bibliografía incluida en el temario y la falta de equipamiento adecuado en algunas escuelas (Hernández, 2023).

Como un ejemplo, en este trabajo se utilizará la materia de Lenguajes de Interfaz que se imparte en la carrera de Ingeniería en Sistemas Computacionales del Instituto Tecnológico de La Paz. En el temario de dicha materia se incluyen temas diversos como son la arquitectura del procesador, programación en lenguaje ensamblador y aplicaciones de este para el desarrollo de programas de bajo nivel. Dichos contenidos claramente están orientados a la enseñanza de lenguaje ensamblador. Sin embargo, la bibliografía sugerida es incongruente,

¹ Profesor de tiempo completo. Tecnológico Nacional de México/Instituto Tecnológico de La Paz.
saul.md@lapaz.tecnm.mx

² Profesora de tiempo completo. Tecnológico Nacional de México/Instituto Tecnológico de La Paz.
iliana.cl@lapaz.tecnm.mx

³ Profesor de tiempo completo. Tecnológico Nacional de México/Instituto Tecnológico de La Paz.
israel.sm@lapaz.tecnm.mx

ya que propone algunos libros de programación en lenguaje Java y otros recursos de programación .NET con C#.

En ese temario también se proponen libros de lenguaje ensamblador, pero dichos libros no están actualizados y no contienen información de las arquitecturas de computadoras modernas. Por otro lado, la sección de prácticas propuestas dice simplemente que se genere un glosario, se desarrollen mapas conceptuales de cada tema y se desarrollen programas en ensamblador.

Por todo lo antes mencionado, la preparación de los temas y el material de clases, especialmente cuando se impartirá la materia por primera vez, resulta bastante complicado. En este contexto, el uso de las herramientas de inteligencia artificial (IA), tales como, los chats inteligentes, pueden resultar de gran ayuda en la preparación de los temas clase y la elaboración del material pertinente. Sin embargo, es importante tener en cuenta que dichas herramientas pueden ofrecer resultados incompletos o, incluso, incorrectos debido a sesgos y fallas en el proceso de entrenamiento de dichas herramientas. Por todo lo anterior, en este trabajo, se presentan los conceptos básicos del funcionamiento de estas tecnologías, así como algunos ejemplos de su utilización en la preparación del material de clases.

METODOLOGÍA

Funcionamiento de los chats inteligentes

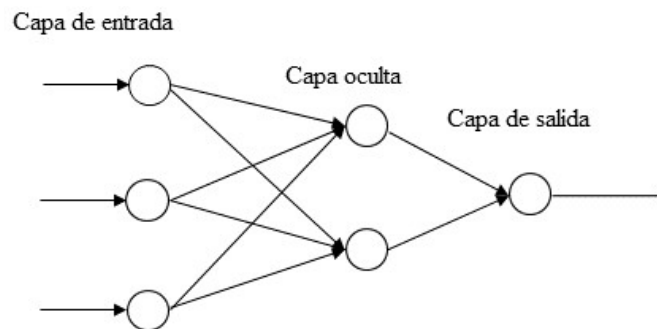
Para aplicar de forma adecuada las herramientas que ofrece la inteligencia artificial es necesario comprender su funcionamiento básico. Por esta razón, en esta sección se introducen los conceptos básicos del funcionamiento de los chats inteligentes, así como la manera de acceder a ellos.

Según la Real Academia Española, la inteligencia artificial es una disciplina científica que se ocupa de crear programas informáticos que ejecutan operaciones comparables a las que realiza la mente humana, como el aprendizaje o el razonamiento lógico (RAE, 2014). En particular, los chats inteligentes son modelos generativos capaces de reconocer e interpretar el lenguaje, ya sea escrito o hablado. Para ello, dichos modelos son entrenados con una gran cantidad de datos, en computadoras de alto rendimiento. De dichos datos el sistema “aprende” las características más importantes para, posteriormente, reconocer e interpretar las frases que se le introducen y así poder dar una respuesta adecuada.

Generalmente, estos modelos se basan en las llamadas redes neuronales artificiales (RNA). Una RNA trata de simular el mecanismo de aprendizaje de los organismos biológicos (Aggarwal, 2018). En el caso biológico, las neuronas se conectan entre sí formando red. Cada neurona genera una respuesta cuya intensidad depende del estímulo externo que recibe. En el caso artificial existen unidades de procesamiento, llamadas neuronas artificiales, que se comunican entre sí a través de conexiones llamadas pesos. En cada unidad de procesamiento, cada estímulo externo se multiplica con su respectivo peso y los resultados se suman y se pasan por una función no lineal, para obtener la salida de dicha unidad. Básicamente, la red neuronal mapea el estímulo de entrada al valor de salida más cercano que corresponda, según el conjunto de datos con el que fue entrenada.

La Figura 1 muestra la estructura de una RNA con una capa de entrada que contiene 3 unidades, una capa oculta con dos unidades y una capa de salida con una unidad. Las flechas representan los pesos que unen cada unidad. Estos pesos se ajustan durante el entrenamiento, para generar la respuesta adecuada a cada señal de entrada.

Figura 1. Red neuronal artificial con una capa oculta



Durante el entrenamiento se presentan muestras etiquetadas (patrones) de los objetos que la red debe identificar. La red calcula un valor de salida y el error que se cometió, comparado con la etiqueta correspondiente. Con este valor de error se utiliza un algoritmo de descenso de gradiente para minimizar el error, ajustando los pesos. Este procedimiento se repite hasta que el error de salida tenga un valor aceptable. Aunque la Figura 1 muestra una red con una sola capa oculta, es posible diseñar redes con muchas capas ocultas. Este último tipo de redes forma parte de las técnicas conocidas como aprendizaje profundo (Deep learning).

En el año 2012, en el “ImageNet Large Scale Visual Recognition Challenge,” se presentó una arquitectura de red profunda llamada AlexNet, la cual ganó con amplio margen el reto de identificar una gran variedad de objetos contenidos en imágenes (Krizhevsky et al., 2017). AlexNet es un tipo de red que, a diferencia de las RNA, realizan una operación de convolución entre la señal de entrada y sus respectivos pesos. Por tal motivo son llamadas Redes Neuronales Convolutivas (CNN, por sus siglas en inglés). Desde entonces, cada año han surgido nuevas arquitecturas de CNN’s, que han superado su desempeño.

Basándose en las CNN’s, se han propuesto las Redes Generativas Antagónicas (GAN, por sus siglas en inglés), las cuales permiten generar patrones similares a aquellos con los que fueron entrenadas. Las GAN utilizan un algoritmo generador y otro algoritmo discriminador (Goodfellow et al., 2014). El generador toma ruido gaussiano como entrada y produce una salida, similar a los datos de entrenamiento. El discriminador suele ser un clasificador que trata de distinguir las muestras reales de entrenamiento y la muestra generada. Durante el entrenamiento, en cada iteración, el generador se ajusta para tratar de generar salidas que “engañen” al discriminador. Con el entrenamiento adecuado, es posible obtener un sistema que genere imágenes realistas o que completen la información faltante en imágenes reales (Schawinski et al., 2017). También es posible entrenar la red con extensas secuencias de texto (en lugar de utilizar imágenes) para generar sus respuestas en lenguaje natural (Hossam et al., 2021).

Todo lo anterior se puede utilizar para que, al leer un texto de entrada que contenga algún cuestionamiento, éste sea interpretado y se genere una respuesta adecuada (con la mayor probabilidad, de acuerdo con los datos de entrenamiento), como es el caso de los chats inteligentes. Muchas empresas están invirtiendo en este tipo de sistemas, para incrementar sus ventajas competitivas. También, pueden ser un buen apoyo en el proceso de enseñanza aprendizaje. Por esas razones, en este trabajo se propone su uso en la preparación de material de clase.

Algunos chats inteligentes

Actualmente, existen varias plataformas que ofrecen el servicio de chats inteligentes de forma gratuita, en la cual se ofrece acceso a solo una parte de la capacidad total del sistema. Aunque su funcionalidad está limitada con respecto a las versiones de paga, pueden ser muy útiles en la preparación de material de clase. Algunos de los principales servicios de chats inteligentes son (Ayala et al., 2024):

- Chat GPT. Su nombre proviene de Chat Generative Pre-Trained Transformer (Transformador pre-entrenado generativo de chat) y fue lanzado por la compañía OpenAI. Para acceder a esta herramienta es necesario registrarse en su sitio web.
- Perplexity AI. utiliza una técnica llamada grandes modelos de lenguaje (LLM, por sus siglas en inglés) para responder de forma precisa a preguntas complejas. Esta herramienta proporciona además las fuentes bibliográficas que utilizó en la generación de su respuesta.
- Bing Chat. Es una herramienta creada por la compañía Microsoft y se encuentra disponible en la página del buscador Microsoft Bing.
- Elcit. Esta herramienta ayuda a automatizar flujos de trabajo y encontrar documentos relevantes, aunque no haya una coincidencia exacta de las palabras clave. Se puede acceder a la herramienta registrándose en su sitio web.
- Gemini. Esta herramienta fue desarrollada por Google y permite generar texto, hacer traducciones y escribir contenidos creativos. Se puede acceder a la herramienta utilizando una cuenta de Gmail.

Para utilizar las herramientas, basta introducir en forma de texto las preguntas o requerimientos que se tengan y, casi de forma inmediata, las herramientas presentan uno o varios resultados en respuesta a lo solicitado. Sin embargo, si las preguntas no son planteadas de forma clara, la respuesta generada puede no ser la esperada. También es importante mencionar que, aunque estos chats suelen ser certeros, en ocasiones proporcionan respuestas incorrectas. Por dichas razones, es obligación del docente asegurarse de no incluir información errónea en su material de clase. En la siguiente sección se presentan algunos ejemplos y resultados del tipo de consultas que pueden hacerse.

RESULTADOS

Para probar la eficacia de los chats inteligentes se introdujeron algunas preguntas relacionadas con el contenido de la materia Lenguajes de Interfaz, mencionada anteriormente. Algunos ejemplos sencillos de las consultas que se pueden realizar para la preparación de una clase son:

¿Qué es el lenguaje ensamblador?

Respuesta de Google-Gemini:

“El lenguaje ensamblador (también conocido como assembler) es un lenguaje de programación de bajo nivel que se utiliza para comunicarse directamente con el hardware de un computador. Se caracteriza por ser:

Simbólico: En lugar de usar código binario (ceros y unos), el lenguaje ensamblador utiliza mnemónicos, que son palabras o abreviaturas que representan instrucciones y operaciones específicas del procesador.

Orientado al hardware: Cada instrucción en ensamblador se traduce directamente a una instrucción en código máquina que el CPU puede entender y ejecutar. Esto permite un control preciso sobre el hardware y un uso eficiente de los recursos del sistema.

De bajo nivel: El lenguaje ensamblador está muy cerca del código máquina, por lo que, requiere un conocimiento profundo de la arquitectura del computador para poder utilizarlo correctamente.

Específico de la arquitectura: A diferencia de los lenguajes de alto nivel, el lenguaje ensamblador es específico para cada tipo de procesador. Esto significa que un programa escrito en ensamblador para un tipo de CPU no funcionará en otro.

Usos del lenguaje ensamblador:

Programación de sistemas embebidos: Se utiliza para programar microcontroladores y otros dispositivos con recursos limitados.

Desarrollo de sistemas operativos: Es fundamental para el desarrollo de las partes del sistema operativo que interactúan directamente con el hardware.

Optimización del rendimiento: Se puede utilizar para optimizar el rendimiento de programas críticos en cuanto a tiempo o recursos.

Comprensión del funcionamiento del hardware: Es una herramienta útil para comprender cómo funciona el hardware a bajo nivel.

Ventajas del lenguaje ensamblador:

Eficiencia: Permite un uso más eficiente de los recursos del sistema que los lenguajes de alto nivel.

Velocidad: Los programas escritos en ensamblador pueden ser más rápidos que los de alto nivel.

Control: Ofrece un control preciso sobre el hardware.

Desventajas del lenguaje ensamblador:

Complejidad: Es un lenguaje complejo y difícil de aprender.

Falta de portabilidad: Los programas escritos en ensamblador no son portables a otras arquitecturas de computador.

Mantenimiento: El código ensamblador es difícil de leer y mantener.

En resumen, el lenguaje ensamblador es una herramienta poderosa que permite un control preciso sobre el hardware del computador. Sin embargo, es un lenguaje complejo y difícil de aprender, por lo que no es la mejor opción para todos los proyectos de programación.”

Como se puede apreciar, el sistema devuelve una respuesta muy completa a la pregunta planteada. Además, es posible solicitar que el sistema nos sugiera la estructura de una presentación de clase, por ejemplo:

¿Cómo preparar una presentación del tema arquitectura del microprocesador x86?

Respuesta ChatGPT:

“Preparar una presentación sobre arquitectura del microprocesador x86 puede ser una tarea emocionante. Aquí se presenta un esquema general para estructurar una presentación de manera efectiva:

Introducción

- Explicar qué es un microprocesador y su importancia en la informática moderna.
- Presentar el microprocesador x86 como una de las arquitecturas de procesadores más importantes y ampliamente utilizadas en el mundo de la computación.

Historia del microprocesador x86:

- Dar una breve reseña histórica desde sus inicios hasta el presente.
- Es conveniente destacar hitos importantes, como el lanzamiento del Intel 8086 en 1978 y sus sucesores, incluyendo el 80286, 80386, Pentium, etc.

Arquitectura básica del microprocesador x86:

- Explicar los conceptos fundamentales como registros, unidades funcionales, y buses.
- Describir la estructura interna del procesador, incluyendo la unidad de control, la ALU (Unidad Aritmético-Lógica), y la unidad de ejecución.

Set de instrucciones x86:

- Introducir los diferentes tipos de instrucciones, como aritméticas, lógicas, de control de flujo, etc.
- Hablar sobre las extensiones del set de instrucciones x86 a lo largo del tiempo, como MMX, SSE, AVX, etc.

Modos de operación:

- Explicar los diferentes modos de operación del procesador x86, como el modo real, el modo protegido, y el modo de compatibilidad.

Pipeline y rendimiento:

- Describir el concepto de pipeline y cómo ayuda a mejorar el rendimiento del procesador.
- Hablar sobre técnicas de optimización de rendimiento, como la predicción de saltos, la ejecución fuera de orden, y la caché de instrucciones y datos.

Evolución reciente:

- Discutir los desarrollos más recientes en la arquitectura x86, como los procesadores de múltiples núcleos, la virtualización, y las características de seguridad.

Aplicaciones y usos:

- Explorar las diversas aplicaciones y casos de uso del microprocesador x86, desde ordenadores personales hasta servidores y dispositivos integrados.

Conclusiones:

- Recapitular los puntos clave de la presentación.
- Se recomienda destacar la importancia continua del microprocesador x86 en la informática moderna y su papel en la evolución tecnológica.

Preguntas y respuestas:

- Abrir una sesión para preguntas de la audiencia y ofrecer respuestas aclaratorias.

Libros de texto recomendados: "Fundamentos de arquitectura de computadores" de John L. Hennessy y David A. Patterson, "Arquitectura de computadores: un enfoque cuantitativo" de John L. Hennessy y David A. Patterson.

Sugerencias adicionales: Es recomendable utilizar gráficos, diagramas y ejemplos prácticos para hacer la presentación más visual y comprensible. Además, practicar la presentación varias veces para asegurarse de que fluya suavemente y transmita efectivamente la información. ¡Buena suerte!”

Como se observa en la respuesta anterior, el sistema hace una sugerencia de contenido adecuada para el nivel de estudios que se solicitó en la pregunta. También, sugiere los temas básicos que aparecen en los libros clásicos de texto, por ejemplo, los procesadores 286, 386 y Pentium, así como temas actualizados, por ejemplo, los procesadores multinúcleo. Además, ofrece referencias bibliográficas adecuadas para el tema.

Para obtener buenos resultados, es importante destacar que se debe evitar hacer preguntas ambiguas, demasiado vagas o fuera de contexto, pues el sistema puede interpretarla de manera incorrecta. En ese caso, es posible que se tenga que invertir más tiempo depurando las consultas hechas. Por ejemplo, ante la siguiente pregunta:

¿Qué es un registro?

Respuesta Google-Gemini:

“Un registro puede tener varios significados, dependiendo del contexto en el que se use. Aquí se presentan algunos de los más comunes:

Registro (estructura de datos): Es una colección de datos relacionados que se agrupan en una sola unidad. Un registro típicamente contiene varios campos, cada uno de los cuales almacena un tipo de dato específico. Por ejemplo, un registro de cliente podría tener campos para el nombre, la dirección, el número de teléfono y la dirección de correo electrónico.

Registro (base de datos): Es una fila en una tabla de una base de datos. Cada registro representa una entidad individual, como un cliente, un producto o una transacción.

Registro (CPU): Es una unidad de almacenamiento de datos en la CPU que se utiliza para almacenar temporalmente datos e instrucciones durante la ejecución de un programa.

Registro (documento): Es un documento que contiene información sobre una transacción o un evento. Los registros se pueden utilizar para fines legales, históricos o de auditoría.

Registro (contabilidad): Es un asiento en un libro de contabilidad que documenta una transacción financiera.

Registro (lingüístico): Es una variedad de una lengua que se caracteriza por su uso en un contexto social específico. Por ejemplo, el registro formal se utiliza en contextos académicos o profesionales, mientras que el registro informal se utiliza en conversaciones entre amigos.

Registro (música): Es una de las diferentes secciones de la tesitura de un instrumento o de una voz. Los diferentes registros se caracterizan por su timbre y su intensidad.

Registro (de audio o video): Es una grabación de sonido o video.

Registro (de marca): Es el proceso de registrar una marca comercial ante una oficina de patentes y marcas.”

En este caso, dada la ambigüedad de la pregunta, la respuesta dada por el sistema no es puntual y se debe depurar. Es por lo que, es recomendable plantear los cuestionamientos de la forma más clara y concisa posible.

CONCLUSIONES

Como se puede observar en los ejemplos presentados, los chats inteligentes basados en redes generativas son una herramienta muy poderosa como apoyo en la preparación de material para las clases. Estos pueden proporcionar información del contenido de los temas, así como, ejemplos de aplicación y referencias bibliográficas. Además, si se desea utilizar la versión de paga, algunos chats permiten la generación de imágenes y gráficos para complementar la presentación.

Sin embargo, teniendo en cuenta la forma en que son diseñados, se debe tener cuidado con la información que brindan. Esto se debe a que, como se mencionó anteriormente, la respuesta de dichos sistemas depende de los datos con los que fueron entrenados. Por tal motivo puede haber sesgos en la información proporcionada. De ahí que es responsabilidad del profesor verificar que dicha información sea correcta y adecuada para propósitos de servir como material de clase.

BIBLIOGRAFÍA

Aggarwal, C. (2018). *Neural Networks and Deep Learning*. Springer

Ayala, M., Manríquez, S., y Acevedo, A. (2024). *Introducción a la Inteligencia Artificial para Profesores*. Editorial Instituto Tecnológico de La Paz

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., & Bengio, Y. (2014). Generative Adversarial Nets. *arXiv*, 1-9. <https://arxiv.org/abs/1406.2661>

Hernández, C. (2023). Actividades de Enseñanza-Aprendizaje para la Materia de Arquitectura de Computadoras. *Revista Electrónica ANFEI Digital*, núm. 8. <https://anfei.mx/revista/index.php/revista/article/view/889>

Hossam, M., Le, T., Papasimeon, M., Huynh, V., & Phung, D. (2021). Text Generation with Deep Variational GAN. *arXiv*, 1-6. <https://arxiv.org/abs/2104.13488>

Krizhevsky, A., Sutskever, I., & Hinton, G. (2017). ImageNet Classification with Deep Convolutional Neural Networks. *Communications of the ACM*, vol. 60(6), 84-90. <https://dl.acm.org/doi/10.1145/3065386>

Real Academia Española [RAE] (2014). Inteligencia artificial. *Diccionario de la lengua Española*. <https://dle.rae.es/inteligencia>

Schawinski, K., Zhang, C., Zhang, H., Fowler, L., y Santhanam, G. (2017). Generative Adversarial Networks recover features in astrophysical images of galaxies beyond the deconvolution limit. *arXiv*, 1-6. <https://arxiv.org/abs/1702.00403>