



INTELIGENCIA ARTIFICIAL

6CV2

CAMACHO VAZQUEZ VANESSA
ALEJANDRA

PRACTICA 1
Agente inteligente “Adivina el número”

LOPEZ PEREZ ALBERTO ANDREI
Yael ALMANZA RIOS
CAMPOS DURAN FABRIZIO

Práctica 1. Agente inteligente “Adivina el número” en Python

Inteligencia artificial DCC. Vanessa Alejandra Camacho Vázquez

Introducción

¿Qué es un agente inteligente?

Un agente inteligente es un sistema computacional o entidad que tiene la capacidad de percibir su entorno, procesar información, tomar decisiones y realizar acciones con el objetivo de alcanzar metas o resolver problemas de manera autónoma. Estos agentes están diseñados para actuar de manera inteligente en situaciones específicas y pueden ser programados o entrenados para realizar tareas específicas o aprender de su experiencia para mejorar su desempeño con el tiempo.

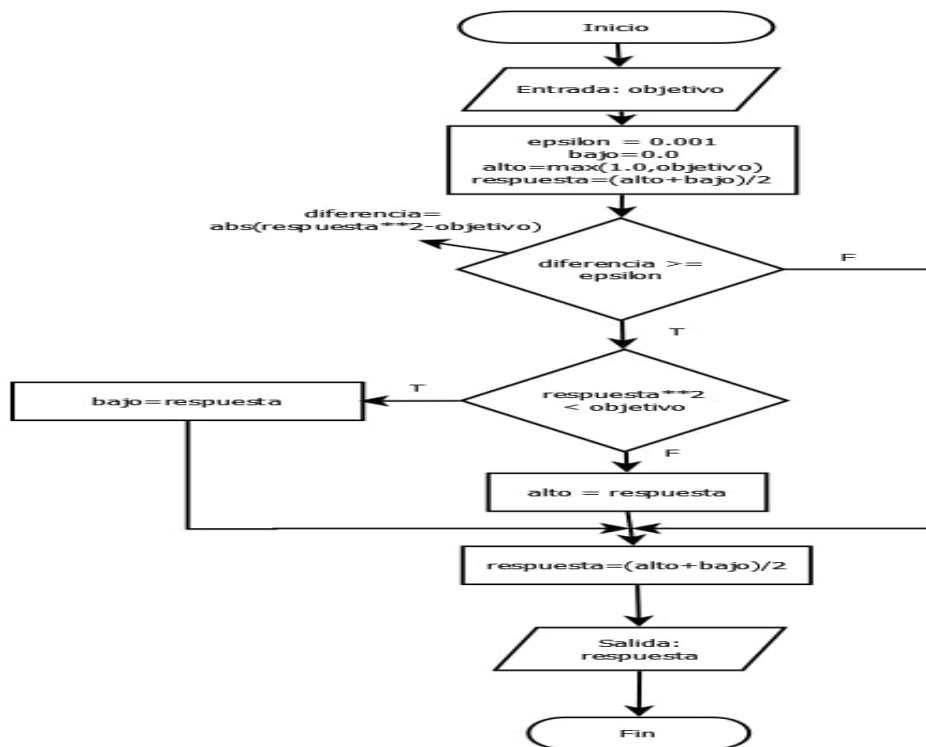
¿Cómo funciona un agente inteligente?

El funcionamiento de un agente inteligente puede variar significativamente según su diseño y su aplicación específica. Sin embargo, en términos generales, un agente inteligente sigue un ciclo de funcionamiento que implica las siguientes etapas:

- ❖ **Percepción:** El agente recopila información del entorno utilizando sensores, cámaras, micrófonos u otras fuentes de datos. La percepción implica capturar datos relevantes que le permitan entender su entorno y tomar decisiones.
- ❖ **Procesamiento de la información:** Una vez que el agente ha recopilado datos del entorno, estos datos se procesan para extraer características relevantes y realizar análisis. Esto puede implicar el procesamiento de imágenes, reconocimiento de patrones, procesamiento de lenguaje natural o cualquier otro tipo de análisis necesario según la tarea del agente.
- ❖ **Razonamiento y toma de decisiones:** Basándose en la información procesada y en sus objetivos predefinidos, el agente utiliza algoritmos y modelos para razonar sobre la situación actual y tomar decisiones. Esto puede incluir la planificación de acciones futuras o la selección de una acción específica en respuesta a la situación.
- ❖ **Acción:** Una vez que el agente ha tomado una decisión, realiza una acción en el entorno. Esta acción puede ser física, como moverse en el caso de un robot, o virtual, como generar una respuesta en el caso de un chatbot.
- ❖ **Aprendizaje (opcional):** Algunos agentes inteligentes tienen la capacidad de aprender de su experiencia. Esto implica ajustar sus algoritmos y modelos en función de la retroalimentación que reciben del entorno o de datos históricos. El aprendizaje puede mejorar su desempeño con el tiempo y adaptar sus acciones a nuevas situaciones.
- ❖ **Interacción continua:** El agente inteligente opera en un entorno dinámico y puede requerir interacción continua para alcanzar sus objetivos. Esto implica repetir el ciclo de percepción, procesamiento, razonamiento, toma de decisiones y acción en intervalos regulares o en respuesta a cambios en el entorno.

¿En qué consiste el algoritmo de búsqueda binaria?

La búsqueda binaria es un algoritmo de búsqueda eficiente utilizado para encontrar un elemento específico en una lista ordenada. Su eficiencia se debe al hecho de que divide repetidamente el espacio de búsqueda a la mitad en cada paso, reduciendo drásticamente el número de comparaciones necesarias en comparación con otros métodos de búsqueda, como la búsqueda lineal.



Coloca un ejemplo de búsqueda binaria en Python.

```
1 def binary_search(arr, target):
2     left, right = 0, len(arr) - 1
3
4     while left <= right:
5         mid = left + (right - left) // 2 # Calcular el índice medio
6
7         if arr[mid] == target:
8             return mid # El elemento fue encontrado, devuelve su índice
9         elif arr[mid] < target:
10            left = mid + 1 # El objetivo está en la mitad derecha
11        else:
12            right = mid - 1 # El objetivo está en la mitad izquierda
13
14    return -1 # Si no se encontró el elemento, devuelve -1
15
16 # Ejemplo de uso
17 sorted_list = [2, 4, 7, 10, 13, 18, 23, 27, 33, 40]
18 target_number = 33
19
20 result = binary_search(sorted_list, target_number)
21
22 if result != -1:
23     print(f"El número {target_number} fue encontrado en el índice {result}.")
24 else:
25     print(f"El número {target_number} no fue encontrado en la lista.")
```

Objetivo

Implementar un agente que juega al juego "Adivina el número". El agente intentará adivinar un número que el usuario haya pensado, y el usuario indicará si el número del agente es demasiado alto, demasiado bajo o correcto.

El agente inteligente adivina el número mediante la estrategia de búsqueda binaria. El agente ajusta su rango de búsqueda en función de las respuestas del usuario para acercarse al número correcto.

Materiales y Métodos

Laptop – Instalar el lenguaje de programación Python.

El código base para comenzar será el siguiente:

```
// Ustedes deberán realizar los cambios en caso de ser necesarios
// para quitar los errores y obtener la ejecución del agente inteligente

import random

class GuessingAgent:
    def __init__(self, min_range, max_range):
        self.min_range = min_range
        self.max_range = max_range
    def guess(self):
        return random.randint(self.min_range, self.max_range)

def main():
    print("Bienvenido al juego 'Adivina el número'!")
    min_range = 1
    max_range = 100
    number_to_guess = random.randint(min_range, max_range)
    agent = GuessingAgent(min_range, max_range)
    while True:
        agent_guess = agent.guess()
```

```

print(f"El agente adivina: {agent_guess}")

user_response = input("Es demasiado alto (A), demasiado bajo (B) o correcto (C)?")

user_response = user_response.upper()

if user_response == "A":
    agent.max_range = agent_guess - 1
elif user_response == "B":
    agent.min_range = agent_guess + 1
elif user_response == "C":
    print("El agente ha adivinado el número correctamente!")
    break
else:
    print("Respuesta no válida. Por favor, responde con A, B o C.")

if __name__ == "__main__":
    main()

```

Desarrollo

Mostrar su código fuente y señalar si realizaron algunos cambios.

Explicar lo que va realizando cada una de las partes del código del agente inteligente ya sea mediante comentarios en el código fuente o aparte en unos párrafos abajo del código.

```

1  import random # Importa la librería random para generar números aleatorios
2
3  class GuessingAgent: # Define la clase GuessingAgent, que representa al agente adivinador.
4      def __init__(self, min_range, max_range):
5          self.min_range = min_range
6          self.max_range = max_range
7
8      def guess(self): # El agente se inicializa con un rango mínimo y máximo en el que adivinará el número secreto.
9          return random.randint(self.min_range, self.max_range)
10
11 def main(): # Configura el rango de números entre los que el agente adivinará y se genera un número secreto aleatorio.
12     print("Bienvenido al juego 'Adivina el número!')
13     min_range = 1
14     max_range = 100
15     number_to_guess = random.randint(min_range, max_range)
16     agent = GuessingAgent(min_range, max_range)
17
18     while True: # El agente hace una suposición, muestra el número adivinado y solicita la respuesta del usuario.
19         agent_guess = agent.guess()
20         print(f"El agente adivina: {agent_guess}")

```

```

20     print(f"El agente adivina: {agent_guess}")
21     user_response = input("Es demasiado alto (A), demasiado bajo (B) o correcto (C)? ").upper()
22
23     if user_response == "A":
24         agent.max_range = agent_guess - 1
25     elif user_response == "B":
26         agent.min_range = agent_guess + 1
27     elif user_response == "C":
28         print("El agente ha adivinado el número correctamente!")
29         break
30     else:
31         print("Respuesta no válida. Por favor, responde con A, B o C.")
32
33     # Basado en la respuesta del usuario (A, B o C), el agente ajusta su rango de búsqueda para la próxima adivinanza.
34     # Si la respuesta es "C", el juego termina.
35
36 if __name__ == "__main__":
37     main()

```

Resultados

Muestra los resultados obtenidos después de realizar la práctica.

Incluir capturas de pantalla de al menos 2 ejecuciones diferentes de su agente inteligente.

```

PS C:\Users\lopez> & C:/Users/lopez/AppData/Local/Programs/Python/Python311/python.exe c:/Users/lopez/OneDrive/Desktop/Septimo/IA/AgenteInteligente.py
Bienvenido al juego 'Adivina el número!'
El agente adivina: 18
Es demasiado alto (A), demasiado bajo (B) o correcto (C)? B
El agente adivina: 67
Es demasiado alto (A), demasiado bajo (B) o correcto (C)? B
El agente adivina: 89
Es demasiado alto (A), demasiado bajo (B) o correcto (C)? A
El agente adivina: 74
Es demasiado alto (A), demasiado bajo (B) o correcto (C)? C
El agente ha adivinado el número correctamente!

```

```

PS C:\Users\lopez> & C:/Users/lopez/AppData/Local/Programs/Python/Python311/python.exe c:/Users/lopez/OneDrive/Desktop/Septimo/IA/AgenteInteligente.py
Bienvenido al juego 'Adivina el número!'
El agente adivina: 41
Es demasiado alto (A), demasiado bajo (B) o correcto (C)? B
El agente adivina: 87
Es demasiado alto (A), demasiado bajo (B) o correcto (C)? A
El agente adivina: 48
Es demasiado alto (A), demasiado bajo (B) o correcto (C)? B
El agente adivina: 59
Es demasiado alto (A), demasiado bajo (B) o correcto (C)? B
El agente adivina: 79
Es demasiado alto (A), demasiado bajo (B) o correcto (C)? A
El agente adivina: 62
Es demasiado alto (A), demasiado bajo (B) o correcto (C)? B
El agente adivina: 67
Es demasiado alto (A), demasiado bajo (B) o correcto (C)? B
El agente adivina: 70
Es demasiado alto (A), demasiado bajo (B) o correcto (C)? B
El agente adivina: 77
Es demasiado alto (A), demasiado bajo (B) o correcto (C)? A
El agente adivina: 76
Es demasiado alto (A), demasiado bajo (B) o correcto (C)? A
El agente adivina: 71
Es demasiado alto (A), demasiado bajo (B) o correcto (C)? B
El agente adivina: 73
Es demasiado alto (A), demasiado bajo (B) o correcto (C)? B
El agente adivina: 74
Es demasiado alto (A), demasiado bajo (B) o correcto (C)? C
El agente ha adivinado el número correctamente!
PS C:\Users\lopez>

```

Análisis y Discusión

Anoten los posibles errores o desafíos encontrados durante la práctica y cómo se resolvieron.

El único error que se nos generó fue a la hora de copiar y pegar el código, ya que se nos juntaba todo y no respetaba la estructura y los espacios, así que solo tuvimos que reacomodar y corregir. El código funcionaba al 100 por sí solo.

Conclusiones

Resuman sus principales conclusiones y aprendizajes de la práctica.

Puede incluir reflexiones sobre la importancia de la práctica en relación con la comprensión de los agentes inteligentes.

Esta práctica nos dejó un poco más claro lo que es un Agente Inteligente, y nos dio una idea muy vaga sobre todas las cosas que se pueden hacer con ellos, a pesar de que el programa que hicimos fue muy sencillo, nos hizo darnos cuenta de toda la capacidad que este puede tener, no pudimos evitar comparar los agentes inteligentes que existen actualmente y tampoco pudimos evitar pensar en las Inteligencias Artificiales que hay, así como imaginar como estos están cambiando actualmente al mundo y como lo van a cambiar en algunos años.

Referencias

<https://www.ceupe.com/blog/agente-inteligente.html>

<https://www.unir.net/ingenieria/revista/agentes-inteligentes/>

[https://es.wikipedia.org/wiki/Agente_inteligente_\(inteligencia_artificial\)](https://es.wikipedia.org/wiki/Agente_inteligente_(inteligencia_artificial))

<https://es.khanacademy.org/computing/computer-science/algorithms/binary-search/a/binary-search>

<https://programacionpython80889555.wordpress.com/2021/12/22/algoritmos-en-python-busqueda-binaria/>