



Universidad: Instituto Tecnológico de Culiacán

Evidencia-03 Sistema Experto

Carrera: Ingeniería en Sistemas Computacionales

Alumnos:

- Jose Manuel Urias Escobar
- Flores Beltrán Jesús Guadalupe

Carrera: Ing. en Sistemas Computacionales

Materia: Inteligencia Art

Docente: ZURIEL DATHAN MORA FELIX

## INTRODUCCION

El proyecto consiste en el desarrollo de un Sistema Experto para el diagnóstico de enfermedades respiratorias. El objetivo principal con este programa es ofrecer una herramienta que, a partir de cómo se sienta el paciente, calcule la probabilidad de padecer una de las enfermedades predefinidas.

Se estuvo viendo al inicio del proyecto el uso de Prolog o Python. Al final decidimos usar Python porque nos sentíamos más familiarizados con este y teníamos más experiencia lo cual nos ayudaría a un avance sin tantas complicaciones al proyecto.

Las enfermedades predefinidas que estamos manejando son las siguientes:

- Asma
- Neumonía
- Bronquitis Aguda
- COVID-19
- Gripe (Influenza)
- Resfriado Común

Para determinar cuál de las enfermedades puede afectar al paciente, el sistema hace preguntas relacionadas a estos y hace un cálculo de porcentajes de certeza en base a estas.

La respuesta del paciente se traduce a un valor Booleano (True para Sí o False para No), para evitar que el usuario llegue a cometer errores a la hora de teclear los datos nos encargamos de guiarlo de una manera clara y eficiente y de corregir algún error que pueda cometer a la hora de llenar estos.

Para determinar la probabilidad de cada enfermedad, utilizamos un sistema de porcentajes de certeza.

## ARQUITECTURA GENERAL DEL CODIGO

El proyecto se compone de dos archivos:

1.- Modulo de conocimiento, motor de diagnostico y explicación de diagnostico(información.py):

Contiene la base de conocimiento: enfermedades y sus listas de síntomas habituales.

Implementa el motor de inferencia mas simple (encadenamiento hacia adelante por coincidencia de síntomas):

- Metodo diagnosticar(paciente): calcula un porcentaje de match por enfermedad.
- Metodo mejor\_diagnostico(diagnósticos): escoge el diagnostico con mayor porcentaje.
- Metodo explicar.todos(paciente): detalla los síntomas coincidentes y faltantes por enfermedad.

2.- Interfaz en línea de comandos (CLI) (interfaz\_cons.py):

- Se encarga de preguntar datos del paciente y síntomas.
- Llama a diagnosticar(paciente) y muestra resultados.
- Presenta la explicación del diagnostico mas probable usando explicar.todos(paciente)

## Base de conocimiento

En información.py se define un diccionario enfermedades donde cada enfermedad tiene la clave "síntomas" con una lista de llaves booleanas que luego se buscaran en el diccionario paciente.

```
enfermedades = {
    "Asma": {
        "sintomas": [
            "sibilancias", "tos_noche", "antecedentes_alergia",
            "opresion_pecho", "silbido_respirar", "mejora_broncodilatadores" ]
    },
    "Neumonía": {
        "sintomas": [
            "fiebre_alta", "tos_con_flemas", "dificultad_respirar",
            "crepitantes", "dolor_toracico", "escalofrios" ]
    },
}
```

Decisiones del diseño

- ✓ Claves booleanas: cada síntoma se representa como true/false en el diccionario del paciente.
- ✓ Y para agregar una enfermedad solo hay que añadir una entrada con su lista de síntomas.

## Diagnóstico (encadenamiento hacia adelante)

### Diagnóstico:

Se utiliza Método diagnosticar(paciente), mandado a llamar de información.py a interfaz\_cons.py

Entrada: paciente, es un diccionario con datos y síntomas en forma booleana.

Para cada enfermedad, cuenta cuantos datos de sus síntomas están en true en el paciente.

Calcula el porcentaje, con la siguiente operación:  
porcentaje=coincidencias/total\_sintomas\*100.

Devuelve los resultados de el paciente.

```
def diagnosticar(paciente):
```

```
    resultados = {}
```

```
    for enfermedad, datos in enfermedades.items():
```

```

sintomas = datos["sintomas"]
coincidencias = sum(1 for s in sintomas if paciente.get(s, False))
porcentaje = int((coincidencias / len(sintomas)) * 100)
resultados[enfermedad] = porcentaje

return resultados

```

### **Modulo de explicación:**

Se incorporo el metodo explicar\_todos(paciente), el cual se encarga de que para cada enfermedad separa la lista de síntomas en:

- Coincidentes: los que el paciente si presenta.
- Faltantes: los que no presenta.

```

def explicar_todos(paciente):
    explic = {}

    for enfermedad, datos in enfermedades.items():

        lista = datos["sintomas"]

        coinc = [s for s in lista if paciente.get(s, False)]
        falt = [s for s in lista if not paciente.get(s, False)]

        explic[enfermedad] = {"coincidentes": coinc, "faltantes": falt}

    return explic

```

este se muestra después de imprimir el diagnostico mas probable.

### **Interfaz de Usuario Por Consola**

Se captura y validan las entradas, con los siguientes métodos:

- **Pedir\_booleano(pregunta):** obliga a responder s/n y devuelve true/false.

```
def pedir_booleano(pregunta):
```

```

    while True:
        respuesta = input(f"{pregunta} (s/n): ").lower()
        if respuesta == "s":
            return True
        elif respuesta == "n":
            return False
        else:
            print("Entrada inválida. Por favor, responda con 's' para Sí o 'n' para No.")

```

- **Pedir\_tipo\_tos():** restringe a “seca” o “con flemas”

```

def pedir_tipo_tos():

    while True:
        tipo_tos = input("¿Tos seca o con flemas?: ").lower()
        if tipo_tos in ["seca", "con flemas"]:
            return tipo_tos
        else:
            print("Entrada inválida. Por favor, ingrese 'seca' o 'con flemas'.")
```

- **Pedir\_datos\_paciente():** pregunta si es H/M, su edad y su nombre, pregunta síntomas, controla dependencias.

```

def pedir_datos_paciente():
    paciente = {}
    paciente["nombre"] = input("Nombre del paciente: ")

    # Validar sexo
    while True:
        sexo = input("Sexo (M/F): ").upper()
        if sexo in ["M", "F"]:
            paciente["sexo"] = sexo
            break
        else:
            print("Por favor, ingrese M o F.")

    #convertimos el string a int
    paciente["edad"] = int(input("Edad: "))

    print("\nResponda con 's' para Sí o 'n' para No:\n")

    # Síntomas comunes
    paciente["tos"] = pedir_booleano("¿Tiene tos?")

    if paciente["tos"]:
        # Se llama a la función de validación del tipo de tos
        tipo_tos = pedir_tipo_tos()

        # Se usan los strings validados para asignar los booleanos
        paciente["tos_seca"] = tipo_tos == "seca"
        paciente["tos_con_flemas"] = tipo_tos == "con flemas"

        paciente["tos_noche"] = pedir_booleano("¿Tose más por la noche o con ejercicio?")

```

```

paciente["tos_persistente"] = pedir_booleano("¿La tos ha durado más
de una semana?")
else:
    # Se asigna False a todas las variables de tos si el paciente no tiene
    tos
        paciente["tos_seca"] = paciente["tos_con_flemas"] =
paciente["tos_noche"] = paciente["tos_persistente"] = False

paciente["fiebre"] = pedir_booleano("¿Tiene fiebre?")

if paciente["fiebre"]:
    paciente["fiebre_alta"] = pedir_booleano("¿La fiebre es mayor a
38.5°C?")
else:
    paciente["fiebre_alta"] = False

# El resto de las preguntas usan pedir_booleano
paciente["fatiga"] = pedir_booleano("¿Siente fatiga o cansancio?")
paciente["dificultad_respirar"] = pedir_booleano("¿Tiene dificultad para
respirar?")
paciente["sibilancias"] = pedir_booleano("¿Tiene silbidos al respirar
(sibilancias)?")
paciente["antecedentes_alergia"] = pedir_booleano("¿Tiene antecedentes
de alergias o asma?")
paciente["opresion_pecho"] = pedir_booleano("¿Siente opresión en el
pecho?")
paciente["silbido_respirar"] = pedir_booleano("¿Escucha silbidos al
respirar?")
paciente["mejora_broncodilatadores"] = pedir_booleano("¿Mejora con
broncodilatadores?")
paciente["crepitantes"] = pedir_booleano("¿Escucha crepitaciones al
respirar?")
paciente["dolor_toracico"] = pedir_booleano("¿Tiene dolor torácico al
respirar?")
paciente["escalofrios"] = pedir_booleano("¿Tiene escalofríos?")
paciente["infeccion_respiratoria_previa"] = pedir_booleano("¿Tuvo una
infección respiratoria reciente?")
paciente["duracion_menor_3_semanas"] = pedir_booleano("¿Los
síntomas han durado menos de 21 días?")

radiografia_previa = pedir_booleano("¿Se ha hecho una radiografía de
tórax antes de esta consulta?")
if radiografia_previa:

```

```
paciente["sin Consolidacion_rx"] = pedir_booleano("¿La radiografía no muestra consolidación pulmonar?")
else:
    paciente["sin Consolidacion_rx"] = False

paciente["irritacion_garganta"] = pedir_booleano("¿Tiene irritación de garganta?")
paciente["perdida_olfato_gusto"] = pedir_booleano("¿Ha perdido el olfato o el gusto?")
paciente["dolor_cabeza"] = pedir_booleano("¿Tiene dolor de cabeza?")
paciente["dolor_muscular"] = pedir_booleano("¿Tiene dolor muscular?")
paciente["congestion_nasal"] = pedir_booleano("¿Tiene congestión nasal?")
paciente["estornudos"] = pedir_booleano("¿Tiene estornudos frecuentes?")
paciente["dolor_garganta"] = pedir_booleano("¿Tiene dolor de garganta?")

return paciente
```

## EJEMPLOS DE EJECUCION

Ejemplo 1:

```
PS C:\Users\franc\8puzzle> & C:/Users/franc/8puzzle/.venv/Scripts/Activate.ps1
(.venv) PS C:\Users\franc\8puzzle> & C:/Users/franc/8puzzle/.venv/Scripts/python-3.7.0/interfaz_cons.py
● Nombre del paciente: Manuel
Sexo (M/F): M
Edad: 20

Responda con 's' para Sí o 'n' para No:

¿Tiene tos? (s/n): s
¿Tos seca o con flemas?: seca
¿Tose más por la noche o con ejercicio? (s/n): s
¿La tos ha durado más de una semana? (s/n): s
¿Tiene fiebre? (s/n): n
¿Siente fatiga o cansancio? (s/n): n
¿Tiene dificultad para respirar? (s/n): s
¿Tiene silbidos al respirar (sibilancias)? (s/n): n
¿Tiene antecedentes de alergias o asma? (s/n): s
¿Siente opresión en el pecho? (s/n): n
¿Escucha silbidos al respirar? (s/n): s
¿Mejora con broncodilatadores? (s/n): n
¿Escucha crepitaciones al respirar? (s/n): s
¿Tiene dolor torácico al respirar? (s/n): n
¿Tiene escalofríos? (s/n): s
¿Tuvo una infección respiratoria reciente? (s/n): n
¿Los síntomas han durado menos de 21 días? (s/n): s
¿Se ha hecho una radiografía de tórax antes de esta consulta? (s/n): n
¿Tiene irritación de garganta? (s/n): s
¿Ha perdido el olfato o el gusto? (s/n): n
¿Tiene dolor de cabeza? (s/n): s
¿Tiene dolor muscular? (s/n): n
¿Tiene congestión nasal? (s/n): s
¿Tiene estornudos frecuentes? (s/n): n
```

Nombre del paciente: Manuel  
¿Tuvo una infección respiratoria reciente? (s/n): n  
¿Los síntomas han durado menos de 21 días? (s/n): s  
¿Se ha hecho una radiografía de tórax antes de esta consulta? (s/n): n  
¿Tiene irritación de garganta? (s/n): s  
¿Ha perdido el olfato o el gusto? (s/n): n  
¿Tiene dolor de cabeza? (s/n): s  
¿Tiene dolor muscular? (s/n): n  
¿Tiene congestión nasal? (s/n): s  
¿Tiene estornudos frecuentes? (s/n): n  
¿Tiene dolor de garganta? (s/n): s

--- Resultados del Diagnóstico ---  
Paciente: Manuel (20 años, M)

- Asma: certeza 50%
- Neumonía: certeza 50%
- Bronquitis Aguda: certeza 66%
- COVID-19: certeza 42%
- COVID-19: certeza 42%
- Gripe (Influenza): certeza 50%
- Resfriado Común: certeza 75%

Diagnóstico más probable: Resfriado Común (75%)

--- Explicación del porqué ---

○ Reglas activadas (síntomas presentes relevantes para Resfriado Común): tos, congestion\_nasal, dolor\_garganta  
Síntomas faltantes (esperados para Resfriado Común): estornudos  
(.venv) PS C:\Users\franc\8puzzle> []

## Ejemplo 2:

Nombre del paciente: Manuel Urias  
Sexo (M/F): M  
Edad: 20

Responda con 's' para Sí o 'n' para No:

¿Tiene tos? (s/n): s  
¿Tos seca o con flemas?: con flemas  
¿Tose más por la noche o con ejercicio? (s/n): s  
¿La tos ha durado más de una semana? (s/n): s  
¿Tiene fiebre? (s/n): s  
¿La fiebre es mayor a 38.5°C? (s/n): s  
¿Siente fatiga o cansancio? (s/n): s  
¿Tiene dificultad para respirar? (s/n): s  
¿Tiene silbidos al respirar (sibilancias)? (s/n): n  
¿Tiene antecedentes de alergias o asma? (s/n): s  
¿Siente opresión en el pecho? (s/n): s  
¿Escucha silbidos al respirar? (s/n): n  
¿Mejora con broncodilatadores? (s/n): s  
¿Escucha crepitaciones al respirar? (s/n): n  
¿Tiene dolor torácico al respirar? (s/n): s  
¿Tiene escalofríos? (s/n): s  
¿Tuvo una infección respiratoria reciente? (s/n): s  
¿Los síntomas han durado menos de 21 días? (s/n): s  
¿Se ha hecho una radiografía de tórax antes de esta consulta? (s/n): n  
¿Tiene irritación de garganta? (s/n): s  
¿Ha perdido el olfato o el gusto? (s/n): s  
¿Tiene dolor de cabeza? (s/n): n  
¿Tiene dolor muscular? (s/n): n  
¿Tiene congestión nasal? (s/n): n

```
ulio-3y/interraz_cons.py
¿Tiene dolor torácico al respirar? (s/n): s
¿Tiene escalofrios? (s/n): s
¿Tuvo una infección respiratoria reciente? (s/n): s
¿Los síntomas han durado menos de 21 días? (s/n): s
¿Se ha hecho una radiografía de tórax antes de esta consulta? (s/n): n
¿Tiene irritación de garganta? (s/n): s
¿Ha perdido el olfato o el gusto? (s/n): s
¿Tiene dolor de cabeza? (s/n): n
¿Tiene dolor muscular? (s/n): n
¿Tiene congestión nasal? (s/n): n
¿Tiene estornudos frecuentes? (s/n): s
¿Tiene dolor de garganta? (s/n): s

--- Resultados del Diagnóstico ---
Paciente: Manuel Urias (20 años, M)

- Asma: certeza 66%
- Neumonía: certeza 83%
- Bronquitis Aguda: certeza 83%
- COVID-19: certeza 57%
- Gripe (Influenza): certeza 50%
- Resfriado Común: certeza 75%

Diagnóstico más probable: Neumonía (83%)

--- Explicación del porqué ---
Reglas activadas, síntomas presentes relevantes para Neumonía fueron las siguientes: fiebre_alta, tos_con_flema
s, dificultad_respirar, dolor_toracico, escalofrios
y los Síntomas faltantes esperados para Neumonia fueron los siguientes: crepitantes
D:\Users\franc\8puzzle> []
```