

# Ingeniería del Conocimiento

## Práctica 2: Implementación ID3

### Autores

- Pedro Antonio Martín Peláez
- Álvaro Ray Martínez

### Detalles de Implementación

Para la implementación del algoritmo se ha usado Python, el algoritmo se ha implementado en su totalidad en un único archivo .py, donde se encuentran las diferentes funciones usadas para el desarrollo del ID3.

### **Lenguaje Utilizado y Librerías**

En esta práctica se ha utilizado el lenguaje de programación Python, un lenguaje versátil y muy adecuado para tareas de procesamiento de datos y creación de algoritmos de aprendizaje automático. Las librerías utilizadas en este proyecto son las siguientes:

- csv: Esta librería se utiliza para leer archivos en formato CSV, que en este caso contienen los datos a analizar. Es una librería estándar de Python que permite la lectura y escritura de archivos CSV de manera eficiente.
- math: La librería **math** se utiliza para realizar cálculos matemáticos, en particular para calcular el valor de la entropía, que es un componente clave en el algoritmo ID3. Esta librería es parte de la biblioteca estándar de Python.

### **Procedimiento de Implementación**

1. Lectura de Archivos:
  - Se leen dos archivos: **AtributosJuego.txt** y **Juego.txt**. El primero contiene los nombres de los atributos que se usarán para construir el árbol de decisión, mientras que el segundo contiene los datos de los ejemplos, que se almacenan en una lista de listas.
  - Para leer el archivo **AtributosJuego.txt**, se utiliza el método **readline()** para obtener los atributos, que luego se separan por comas para almacenarlos como una lista.
  - El archivo **Juego.txt** se lee utilizando el módulo **csv** para organizar los datos de manera estructurada.
2. Cálculo de la Entropía:

- Se define la función *infor()* para calcular la entropía o información de un conjunto de ejemplos. Esta función sigue la fórmula estándar de la teoría de la información, donde se calculan las probabilidades de las clases (en este caso, "si" o "no") y se usa la fórmula de Shannon para calcular la entropía.
- 3. Cálculo del Mérito (Ganancia de Información):
  - La función *merito()* evalúa la ganancia de información de cada atributo al calcular la diferencia entre la entropía total del conjunto de ejemplos y la entropía condicional dada la división del conjunto de datos según un atributo. La ganancia de información es el criterio que se utiliza para seleccionar el mejor atributo en cada paso del algoritmo ID3.
- 4. Construcción del Árbol de Decisión (ID3):
  - La función *ID3()* implementa el algoritmo de árbol de decisión utilizando el método de selección de atributos basado en la ganancia de información. En cada paso, se selecciona el atributo con el mayor mérito y se divide el conjunto de ejemplos según los valores posibles de ese atributo. El proceso se repite recursivamente hasta que todos los ejemplos se clasifican correctamente o ya no hay más atributos disponibles.
- 5. Impresión del Árbol de Decisión:
  - La función *imprimir\_arbol\_grafo()* imprime el árbol de decisión generado de manera visual, utilizando caracteres ASCII para representar las ramas y los nodos del árbol.
- 6. Clasificación de Nuevos Ejemplos:
  - La función *clasificar()* permite realizar la clasificación de nuevos ejemplos utilizando el árbol de decisión generado. Esta función toma un ejemplo como entrada, lo compara con los nodos del árbol y, siguiendo las ramas correspondientes, devuelve la clasificación final ("si" o "no").
- 7. Interacción con el Usuario:
  - El programa permite al usuario introducir consultas en un formato específico (atributos y sus valores) y obtiene el resultado de la clasificación utilizando el árbol de decisión generado. Si el usuario desea, puede seguir haciendo consultas hasta que decida salir.

### **Ampliaciones Realizadas**

- Recursión de todos los niveles: el algoritmo implementado realiza todas las llamadas recursivas necesarias para construir el árbol de decisión de ID3 completo.
- Consultas por parte del usuario: el usuario puede realizar consultas siguiendo un formato específico. Formato -> "TiempoExterior=soleado, Temperatura=caluroso, Humedad=alta, Viento=falso"

### **Manual de Usuario**

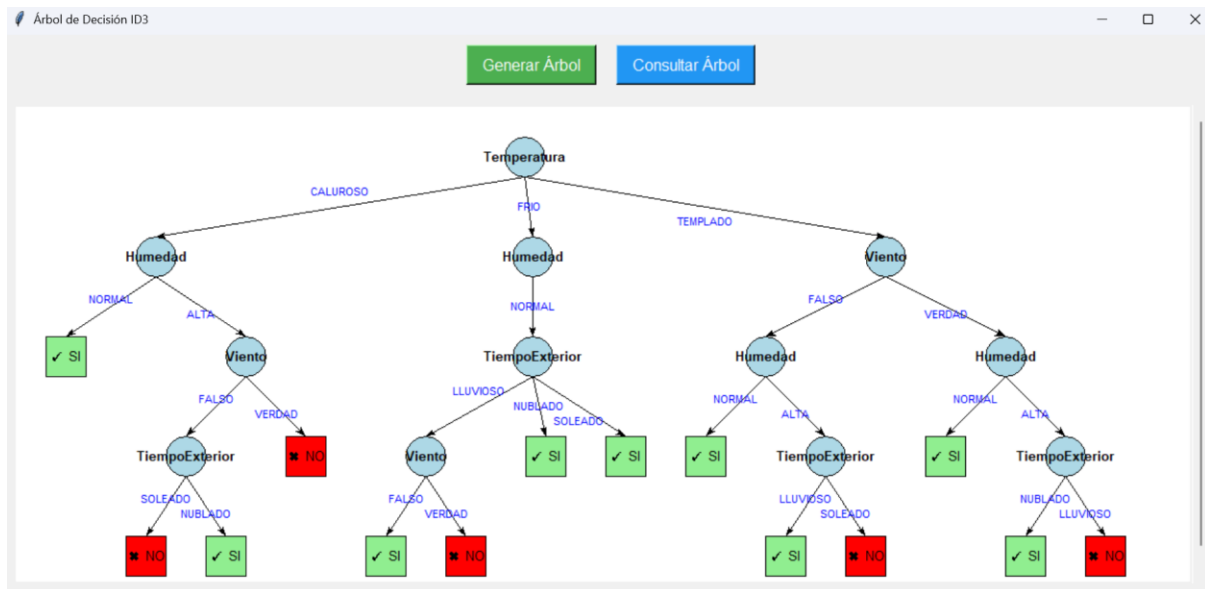
### **Controles e Interacción**

Para ejecutar el código tan solo basta con usar el ejecutable que se adjunta en la carpeta junto con el resto de ficheros. Sin necesidad de instalar Python ni bibliotecas adicionales.

### Uso de la aplicación.

Para generar el árbol deberá:

1. Haz clic en "Generar Árbol".
2. La aplicación leerá los datos y mostrará el árbol gráficamente.



A continuación para realizar consultas del árbol:

1. Una vez generado el árbol, haz clic en "Consultar Árbol".
2. Selecciona los valores de cada atributo desde los menús desplegables.
3. Presiona "Consultar" y se mostrará la clasificación (✓ SI o ✗ NO).

Por ejemplo en las siguientes imágenes se representa una consulta cuyo resultado es "No".

The screenshot shows two windows. The left window, titled "Consult...", has four dropdown menus: "TiempoExterior" (soleado), "Temperatura" (caluroso), "Humedad" (alta), and "Viento" (falso). Below these is a green "Consultar" button. The right window, titled "Resultado", shows an information icon and the text "Clasificación: NO", with an "Aceptar" button below it.

Por el contrario en la siguiente configuración el resultado es que "Sí".

Consult...


TiempoExterior:

Temperatura:

Humedad:

Viento:

Resultado

 Clasificación: SI