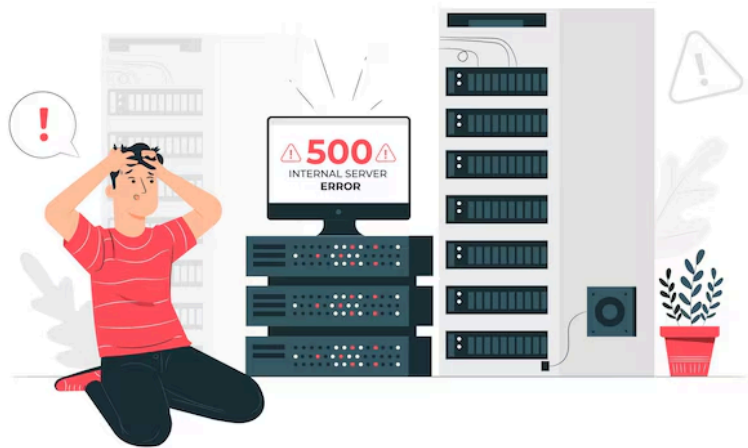


Resumen del Sistema Experto Bayesiano para Diagnóstico de Fallas en Servidores de un Centro de Procesamiento de Datos



Pablo Cabeza Lantigua

Universidad de Las Palmas de Gran Canaria

Inteligencia Artificial para Ciencia de Datos

2nd Año en Grado en Ciencia e Ingeniería de Datos

2024-2025

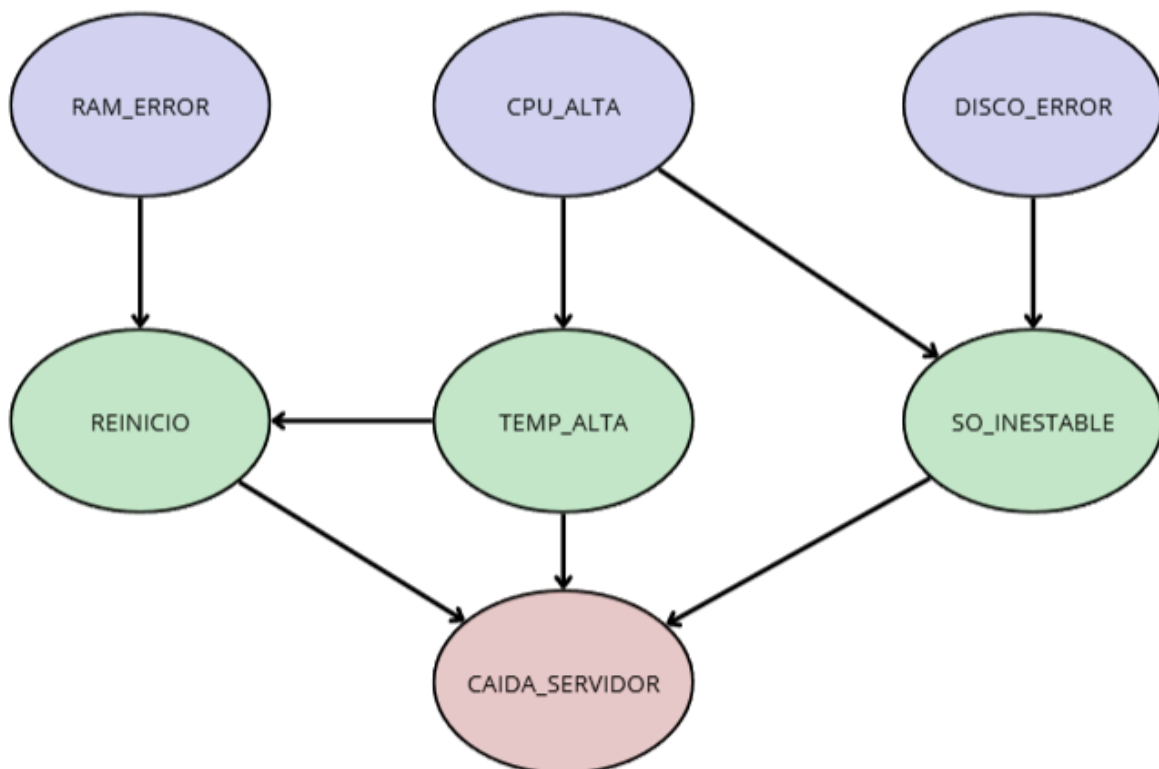
Índice

Introducción	2
Estructura de la Red Bayesiana	2
Descripción de los Nodos	2
Relaciones Causales	3
Implementación en CLIPS	3
Proceso de Ejecución del Código	3
Preguntas	5
Utilidad del Modelo	6
Conclusiones	7

Introducción

La Red Bayesiana diseñada tiene como propósito diagnosticar fallas en servidores de un Centro de Procesamiento de Datos (CPD). Este modelo probabilístico permite identificar y analizar las causas de problemas en los servidores, facilitando un mantenimiento más eficiente y una resolución más rápida de incidencias.

Estructura de la Red Bayesiana



Descripción de los Nodos

- **CPU_Alta:** Representa una situación de alta carga en la unidad central de procesamiento del servidor.
- **RAM_Error:** Indica la presencia de errores en la memoria de acceso aleatorio del servidor.
- **Disco_Error:** Señala fallos en el sistema de almacenamiento del servidor.
- **Temp_Alta:** Representa una temperatura elevada en el servidor, lo cual puede ser indicativo de problemas de refrigeración o sobrecarga.
- **Reinicio:** Indica la ocurrencia de reinicios inesperados en el servidor.
- **SO_Inestable:** Representa inestabilidad en el sistema operativo del servidor.
- **Caída_Servidor:** Indica una caída completa del servidor, lo cual puede ser resultado de varios factores.

Relaciones Causales

Las conexiones entre los nodos representan las relaciones de causa-efecto en el diagnóstico de fallas:

- **CPU_Alta** puede provocar **Temp_Alta** y contribuir a **SO_Inestable**.
- **RAM_Error** puede causar **Reinicio**.
- **Disco_Error** puede llevar a **SO_Inestable**.
- **Temp_Alta** puede ocasionar **Reinicio** y **Caida_Servidor**.
- **SO_Inestable** y **Reinicio** pueden contribuir a **Caida_Servidor**.

Estas relaciones permiten inferir las causas probables de una falla observada en el servidor.

Implementación en CLIPS

La implementación de la red bayesiana en CLIPS se realiza mediante una combinación eficiente de elementos clave. Se utilizan plantillas para representar los estados de cada nodo (como p-CPU, p-RAM, p-Disco) y funciones específicas para calcular probabilidades condicionales (calc-prob-cpu, calc-prob-ram, etc.). El sistema incluye reglas para realizar inferencias sobre el estado del sistema, apoyadas por funciones auxiliares para cálculos y manipulación de datos. Además, se definen hechos iniciales que establecen las probabilidades a priori y condicionales de la red. Esta estructura permite una implementación eficiente y flexible del modelo bayesiano en CLIPS, facilitando tanto la representación del conocimiento como el razonamiento probabilístico.

Proceso de Ejecución del Código

Para ejecutar el sistema experto bayesiano en CLIPS, sigue estos pasos:

1. **Carga del Programa:** Primero, carga el archivo CLIPS que contiene el sistema experto utilizando el comando:

(load expertSystemForCPDHandlingUncertainty.clp)
2. **Reiniciar el Entorno:** Después de cargar el programa, haz un **(reset)** para cargar los hechos iniciales, sino, **el programa no funcionará (se carga sin consultas ni evidencias, solo probabilidades y hechos que nos servirán para controlar las funciones).**

3. **Definir Consultas y Evidencias:** A continuación, debes definir las consultas y las evidencias que se van a utilizar. **Es importante hacerlo en este orden para evitar problemas en la ejecución del programa.**

- **Consultas:** Utiliza la plantilla "consulta" para definir qué probabilidad se quiere calcular. El parámetro "valor" por defecto es TRUE, lo que significa que se consulta la probabilidad de que el evento ocurra (en lugar de que no ocurra, que sería usando FALSE, esto es lo mismo para las evidencias). Por ejemplo:

(assert (consulta (nombre Caída_Servidor) (valor TRUE)))

- **Evidencias:** Utiliza la plantilla evidencia para introducir los datos observados, en este no habrá valor por defecto, por lo que lo tendrás que especificar. Por ejemplo:

(assert (evidencia (nombre CPU_Alta) (valor TRUE)))

Los nombres de las consultas y evidencias deben coincidir con los nombres de los nodos de la red bayesiana, descritos anteriormente(CPU_Alta, RAM_Error, Disco_Error...).

4. **Ejecutar el Programa:** Una vez que hayas introducido todas las consultas y evidencias, ejecuta el motor de inferencia con el comando:

(run)

Esto hará que el sistema experto realice los cálculos necesarios y muestre las probabilidades solicitadas.

5. **Reiniciar para Nuevas Consultas:** Si deseas realizar nuevas consultas o introducir diferentes evidencias, debes reiniciar el sistema con **(reset)** y repetir el proceso, sino las probabilidades pueden salir erróneas.

Este proceso te permite interactuar con el sistema experto de manera flexible y obtener resultados basados en diferentes escenarios de evidencia.

Preguntas

1. *El sistema debe calcular la probabilidad de que ocurra una causa dado un conjunto de posibles síntomas. Por ejemplo, calcular la probabilidad de que el servidor se caiga dado que se ha detectado un reinicio inesperado y alta temperatura.*

Como se puede observar, y siguiendo el proceso de ejecución anterior, se puede calcular con facilidad esta cuestión. En este caso la probabilidad es de 0.94813 (aproximado a 5 decimales):

```
CLIPS (6.4.2 1/14/25)
CLIPS> (load C:\UNIVERSIDAD\SEGUNDO_ANYO\SEGUNDO_SEMESTRE\IACD\Practica\Entregas\Trabajos\expertSystemForCPDHandlingUncertainty.clp)
*****$*!!!!!!!!!!!!!!*
TRUE
CLIPS> (reset)
CLIPS> (assert (consulta (nombre Caída_Servidor)))
<Fact-49>
CLIPS> (assert (evidencia (nombre Reinicio)(valor TRUE)))
<Fact-50>
CLIPS> (assert (evidencia (nombre Temp_Alta)(valor TRUE)))
<Fact-51>
CLIPS> (run)
La probabilidad de que el servidor caiga (TRUE) dadas las evidencias observadas es: 0.94813
```

2. *¿Cómo cambia la probabilidad anterior si además sabemos que hay errores en la memoria RAM?*

Si añadimos a la probabilidad anterior la evidencia de RAM, podemos observar en la imagen, que sigue siendo 0.94813.

```
CLIPS> (reset)
CLIPS> (assert (consulta (nombre Caída_Servidor)))
<Fact-49>
CLIPS> (assert (evidencia (nombre Reinicio)(valor TRUE)))
<Fact-50>
CLIPS> (assert (evidencia (nombre Temp_Alta)(valor TRUE)))
<Fact-51>
CLIPS> (assert (evidencia (nombre RAM_Error)(valor TRUE)))
<Fact-52>
CLIPS> (run)
La probabilidad de que el servidor caiga (TRUE) dadas las evidencias observadas es: 0.94813
```

Esto se debe porque al añadir la evidencia de RAM_Error porque, según la estructura de la red bayesiana, los errores de RAM influyen en la caída del servidor únicamente a través del nodo intermedio "Reinicio". Como ya teníamos evidencia de un reinicio inesperado en el cálculo original, la información adicional sobre errores de RAM no aporta nueva evidencia que afecte directamente a la probabilidad de caída. Esto demuestra que una vez conocido el estado del nodo "Reinicio", conocer la causa de dicho reinicio (en este caso, RAM_Error) no modifica nuestra certeza sobre la caída del servidor.

Este comportamiento demuestra el concepto de independencia condicional en redes bayesianas, cuando conocemos el estado de un nodo intermedio ("Reinicio"), las variables padre (RAM_Error) se vuelven independientes de las variables hijo (Caída_Servidor). Esta propiedad es clave para la eficiencia de las redes bayesianas.

3. ¿Cuál es la causa más influyente en la caída del servidor?

Haciendo los cálculos en CLIPS:

```
CLIPS> (assert (consulta (nombre Caída_Servidor)))
<Fact-49>
CLIPS> (assert (evidencia (nombre Reinicio)(valor TRUE)))
<Fact-50>
CLIPS> (run)
La probabilidad de que el servidor caiga (TRUE) dadas las evidencias observadas es: 0.89605
```

```
CLIPS> (assert (consulta (nombre Caída_Servidor)))
<Fact-49>
CLIPS> (assert (evidencia (nombre Temp_Alta)(valor TRUE)))
<Fact-50>
CLIPS> (run)
La probabilidad de que el servidor caiga (TRUE) dadas las evidencias observadas es: 0.85466
```

```
CLIPS> (assert (consulta (nombre Caída_Servidor)))
<Fact-49>
CLIPS> (assert (evidencia (nombre CPU_Alta)(valor TRUE)))
<Fact-50>
CLIPS> (run)
La probabilidad de que el servidor caiga (TRUE) dadas las evidencias observadas es: 0.81765
```

```
CLIPS> (assert (consulta (nombre Caída_Servidor)))
<Fact-49>
CLIPS> (assert (evidencia (nombre SO_Inestable)(valor TRUE)))
<Fact-50>
CLIPS> (run)
La probabilidad de que el servidor caiga (TRUE) dadas las evidencias observadas es: 0.75687
```

```
CLIPS> (assert (consulta (nombre Caída_Servidor)))
<Fact-49>
CLIPS> (assert (evidencia (nombre RAM_Error)(valor TRUE)))
<Fact-50>
CLIPS> (run)
La probabilidad de que el servidor caiga (TRUE) dadas las evidencias observadas es: 0.72904
```

```
CLIPS> (assert (consulta (nombre Caída_Servidor)))
<Fact-49>
CLIPS> (assert (evidencia (nombre Disco_Error)(valor TRUE)))
<Fact-50>
CLIPS> (run)
La probabilidad de que el servidor caiga (TRUE) dadas las evidencias observadas es: 0.56823
```

Como se puede observar, la causa que más afecta a la caída del servidor es **Reinicio Inesperado**.

Utilidad del Modelo

El modelo se utiliza para diagnosticar las causas más probables de fallos en el servidor basándose en los síntomas observados. Permite estimar la probabilidad de caída del servidor dadas ciertas condiciones, lo que es crucial para la prevención de fallos críticos. Además, ayuda a identificar los componentes más críticos que influyen en el fallo del sistema, facilitando la priorización de mantenimiento y mejoras. Una de sus características más útiles es la capacidad de realizar inferencias sobre el estado del servidor a partir de evidencias parciales, lo que permite una evaluación rápida y precisa incluso con información limitada.

Conclusiones

Esta Red Bayesiana ofrece varias ventajas significativas en el contexto de mantenimiento de servidores. Proporciona un enfoque probabilístico para el diagnóstico de fallas, lo que permite manejar la incertidumbre inherente en sistemas complejos, mejorando la precisión de las predicciones y diagnósticos. Facilita la toma de decisiones informadas para el mantenimiento preventivo y la resolución rápida de problemas, lo que puede reducir significativamente el tiempo de inactividad del servidor y los costos asociados.

La capacidad de actualización continua del modelo a medida que se recopilan más datos es otra ventaja clave, ya que permite que el sistema mejore su precisión con el tiempo, adaptándose a las características específicas del entorno en el que opera. Finalmente, la red ofrece una representación visual intuitiva de las relaciones entre diferentes componentes y fallas del sistema, lo que facilita la comprensión y comunicación de problemas complejos entre los equipos técnicos y de gestión.