



**UNIVERSIDAD
DE GRANADA**

ETSIIT

Escuela Técnica Superior
de Ingenierías Informática
y de Telecomunicación



Práctica 6: Sistema Basado en Conocimiento

Ingeniería del Conocimiento

Autor:

Lugli, Valentino Glauco · YB0819879

Índice

1	Resumen de Funcionamiento del Sistema	2
2	Descripción del proceso seguido	2
2.1	Procedimiento seguido para el Desarrollo de la Base de Conocimiento	2
2.2	Procedimiento de Validación y Verificación del Sistema	3
3	Descripción del sistema	3
3.1	Variables de entrada del problema	4
3.1.1	Módulo α	4
3.1.2	Módulo δ	4
3.2	Variables de salida del problema	4
3.2.1	Módulo α	4
3.2.2	Módulo δ	5
3.3	Conocimiento global del sistema	5
3.4	Especificación de los módulos desarrollados	5
3.4.1	Módulo α	5
3.4.2	Módulo δ	5
4	Manual del Sistema	6
4.1	Utilización	6
4.2	Mantenimiento y Expansión	6

1. Resumen de Funcionamiento del Sistema

El SBC presentado a continuación está compuesto principalmente de dos módulos que están diseñados para ayudar al alumnado de la ETSIIT a seleccionar que rama o mención elegir y/o entre dos asignaturas del grado de Ingeniería Informática, en cual deberían matricularse en relación al perfil que tiene el estudiante que la realiza.

En esencia ambos módulos funcionan de manera similar, una vez activados piden al usuario una serie corta de preguntas que permite al sistema generar una especie de perfil de la persona quién lo está respondiendo, luego, esto permite dados los datos que se poseen en la Base de Conocimiento, realizar inferencias y deducciones de los datos y generarle una respuesta al usuario.

Internamente, el Módulo Recomendador de Menciones utiliza lógica clásica mientras que el Módulo de Recomendador de Matrículas utiliza lógica difusa, se decidió utilizar esta lógica ya que en este problema se manejan conceptos bastante subjetivos como lo que sería que una asignatura sea “difícil” o que sea más “teórica” o “práctica” que otra.

Los datos importantes se encuentran todos localizados de manera cómoda en el código, y es fácilmente extensible ya que las bases de conocimiento utilizadas hacen uso de registros por lo que añadir más información solo requiere de ello, porque el código en sí funciona con cualquier dato que se posea, a lo máximo se tendrían que añadir más reglas para información, pero las que realizan la deducción son completamente independientes del conocimiento que manejan.

De ahora en adelante, el módulo Recomendador de Menciones se llamará Módulo α , y el Recomendador de Asignaturas será el Módulo δ .

2. Descripción del proceso seguido

2.1. Procedimiento seguido para el Desarrollo de la Base de Conocimiento

Para desarrollar las Bases de Conocimiento del Módulo α , se hizo uso de un Experto –debido a que originalmente este módulo era un Sistema Experto por sí solo–, siendo el Experto el mismo autor de esta práctica. Se hizo uso de árboles de decisión con el conocimiento recabado para obtener las reglas, estas mismas fueron luego plasmadas en el código del SE añadiendo y mejorando las recomendaciones originales del Experto.

Para la BC del Módulo δ también se hizo uso del Experto pero también otras fuentes de información, más que nada, descripciones y anécdotas de otros estudiantes para las distintas asignaturas incluidas, ya que este problema está mejor situado para utilizar un conocimiento múltiple y una revisión de ciertas Guías Docentes para corroborar o ampliar como es cada asignatura actualmente en la Base de Conocimiento.

En todo momento se procuró que las Bases de Conocimiento fueran lo más fáciles de extender y mantener, por lo tanto, en el código se encuentran las secciones que mantienen la Base de Conocimiento de cada módulo claramente comentada y, como se ha mencionado, al utilizar registros, añadir conocimiento es muy fácil y rápido.

2.2. Procedimiento de Validación y Verificación del Sistema

Se siguió el procedimiento delineado en la última tarea de la asignatura una vez que la verificación fue finalizada, la cual consistió mayoritariamente de errores del Ingeniero de Conocimiento y de la Base de Conocimiento, más que nada se encontraron inconsistencias estructurales y lógicas. Una vez que se comprobó que el sistema estaba verificado se utilizaron tres metodologías para la validación:

- Casos de Prueba y Pruebas de Campo: Dado que el Ingeniero de Conocimiento y el Experto son ambas la misma persona en este caso, estas dos metodologías se hicieron en paralelo, se hicieron numerosas pruebas del sistema con diferentes asignaturas y diferentes parámetros de entrada, visualizando las salidas obtenidas y corroborando si estas tenían sentido con el conocimiento que se había ingresado en la Base de Conocimiento, y por lo tanto se tendría que modificar ciertos parámetros.
- Análisis de Sensibilidad para Sistemas con Incertidumbre: En los Casos de Prueba anteriores, se incluyó un grado de incertidumbre para verificar que con datos similares se obtenían valoraciones salientes similares del módulo.

Satisfactoriamente, el sistema lograba dar respuestas a todos los casos que se habían planteado, con un nivel de aceptación por el Experto alto respecto a su propio conocimiento; de lo que él mismo recomendaría.

También se verificó que el utilizar la Lógica Difusa era una manera correcta de dar resultados, se adaptó para que la salida fuese también un valor abstracto entre 0 y 10 que indica, casi como si fuese la reseña de una película o videojuego, una puntuación de qué tanto puede que le guste una asignatura al usuario; se imprime siempre la asignatura con mejor valoración. En el caso de que haya un empate, para evitar incertidumbre por parte del usuario, el propio recomendador escoge una de las dos para mostrar, debido a que, se entiende que si ambas han obtenido la misma puntuación es porque son similares desde la perspectiva del usuario.

Si existieron casos en el que se daba una respuesta errónea, muchos errores por omisión más que por comisión, y tal como se especificó en el documento de tarea, se lograron solventar modificando los parámetros de los conjuntos difusos del problema y también de las reglas que generaban las reglas que el módulo de conjuntos difuso necesitaba.

3. Descripción del sistema

El sistema está estructurado en 3 módulos, dos módulos para los Recomendadores (α y δ) y un módulo para la Lógica Difusa (γ). Cada uno de ellos está claramente comentado dónde comienza y donde finaliza, y además, lo necesario del conocimiento se encuentra todo relativamente en un mismo lugar por lo que, añadir o revisar el conocimiento de los módulos es una tarea muy fácil, además de que, las reglas que realizan la inferencia son en gran parte independientes del conocimiento que se tenga por lo tanto habría que modificar ligeramente o en todo caso generar pocas reglas adicionales si hubiera la necesidad de ello.

El sistema inicia con un pequeño menú del cual se puede seleccionar el módulo de Menciones (α) o el de Matrículas (δ), una vez dentro de cada módulo se tienen diferentes variables de entradas

del sistema.

3.1. Variables de entrada del problema

3.1.1. Módulo α

Este módulo realiza preguntas de cómo es el estudiante, que cosas de la carrera disfruta más o menos, por ejemplo si le gusta más el hardware o si le gustan las matemáticas; también se le pregunta si es una persona más orientada a trabajar o más ociosa.

Para realizar inferencias se concretamente se necesitan los siguientes hechos:

```
(answer hardware      <si|no|me da igual>) ;¿Te gusta el hardware?  
(answer programar    <si|no|me da igual>) ;¿Te gusta programar?  
(answer promedio     <alto|medio|bajo>)    ;¿Qué promedio tienes?  
(answer matematicas   <si|no|me da igual>) ;¿Te gusta la matemática?  
(answer trabajo      <mucho|medio|poco>)    ;¿Qué tanto trabajas?
```

3.1.2. Módulo δ

Este módulo realiza también preguntas sobre el tipo de persona que es el estudiante pero esta vez centrado específicamente en la manera en que una asignatura es, es decir, si es una asignatura mayoritariamente teórica o práctica; o bien si es muy trabajosa o poco. Además que también se pregunta por un par de las mismas del Grado de Ingeniería Informática desde el 1º curso hasta el 3º curso 1º cuatrimestre, puesto que el resto de asignaturas ya debería depender del otro recomendador al ser de rama.

Para realizar inferencias se concretamente se necesitan los siguientes hechos:

```
(delta course <asignatura1> firstCourse)      ; Primera asignatura  
(delta course <asignatura2> secondCourse)     ; Segunda asignatura  
(dato work      <valor entre 0 y 8>)           ; ¿Qué tantas horas al día en  
promedio estudias?  
(dato practice  <valor entre 0 y 10>)          ; De 0 a 10, con 10 siendo que  
te gusta mucho y 0 nada, ¿que tanto te gustan las asignaturas prácticas?  
(dato theory    <valor entre 0 y 10>)          ; De 0 a 10, con 10 siendo que  
te gusta mucho, y 0 nada ¿que tanto te gustan las asignaturas teóricas?  
(dato typeEval  <valor entre 0 y 10>)          ; De 0 siendo que te gusta más  
la evaluación al final a 10 siendo que te gusta evaluación continua, ¿qu  
é te gusta más?
```

Estos hechos luego son convertidos por medio de reglas a un formato que es aceptado por el Módulo γ el cual es el de Lógica Difusa.

3.2. Variables de salida del problema

3.2.1. Módulo α

La salida que se da por pantalla posee esta única regla, esto se debe a que la explicación se encuentra contenida en un registro, y una vez, por medio de las reglas y las respuestas se obtiene una mención, ya esta posee la explicación del porqué fue elegida.

```
(ansAlpha <nombre de asignatura> <explicación>)
```

3.2.2. Módulo δ

La salida se imprime por pantalla, y el valor que espera es esto, que es a su vez un registro que se obtiene de la salida del Módulo γ de Lógica Difusa; dentro de este módulo se imprimen por pantalla los razonamientos que realiza el módulo y finalmente el veredicto es recibido y convertido a lo siguiente:

```
(endValue (ID <nombre corto de asignatura>) (val <valor de recomendación>))
```

3.3. Conocimiento global del sistema

El sistema mantiene en la Agenda varios hechos que son utilizados por todos los módulos, aunque esto no presenta ningún problema debido a que se ha tenido la precaución de evitar repetir los nombres utilizados, y además, las no se pueden disparar a menos que el módulo al que pertenecen esté activo en ese momento, este conocimiento serían las Bases de Conocimiento de los dos módulos, específicamente el Conocimiento que ya se conoce: las menciones del grado, que condiciones necesitan para que sean seleccionadas, se tienen las asignaturas y su descripción de como son a nivel de ser teóricas, prácticas, su método de evaluación y si son trabajosas.

3.4. Especificación de los módulos desarrollados

3.4.1. Módulo α

El módulo α contiene la Base de Conocimiento `branchRecc` que usa el registro `recommendationAlpha` la cual contiene las ramas a recomendar junto con las condiciones que se tienen que cumplir, al menos parcialmente, para que se pueda recomendar esa rama a un usuario; se recogen los casos más generales, por lo tanto existe una regla que se dispara en caso de que no se haya podido inferir una buena rama para el estudiante.

En general el módulo se compone de ese registro junto a las reglas que preguntan y 3 reglas que deducen dadas las respuestas y la BC, los hechos específicamente son los que aparecen en la sección 3.1 de este manual, con estas se realiza el razonamiento.

Se tienen adicionalmente unas reglas para verificar que las respuestas que se introducen son válidas, así asegurando el funcionamiento correcto del SE.

3.4.2. Módulo δ

Este módulo hace uso también del módulo γ , pero inicialmente se tiene la base de conocimiento `courseDataBase` que hace uso del registro `course`, que contiene las asignaturas junto a una descripción de la misma, es decir, describe si una asignatura es teórica, práctica, trabajosa y su tipo de evaluación. Se tienen, posteriormente, los hechos que definen los conjuntos difusos de esas variables, notar que las asignaturas aún no poseen un conjunto difuso: este es generado dependiendo de las asignaturas que sean elegidas, de esta manera no se llena la Agenda de tantos hechos que no serán utilizados.

Una vez el usuario ingresa el código de las asignaturas que desea comparar, se generan las reglas para tener el conjunto difuso de salida y también genera las reglas que utilizará el módulo γ para razonar, se tienen cuatro reglas para ello, dependiendo de lo que se tenga en la BC de la asignatura se generan las reglas correspondientes para que dada una respuesta del usuario, se dé una valoración de la asignatura.

Una vez se tiene esto, y se terminan de hacer las preguntas, los hechos específicos que se utilizan son los que aparecen descritos en la sección 3.1 de esta memoria, se activa propiamente el módulo γ creado por el Profesor Juan Luis Castro, este realiza las deducciones dadas las reglas generadas y el resultado se captura, se reestructura y se utiliza una última regla para mostrar por pantalla la asignatura que se cree le gustará más al usuario, esta valoración es entre 0 y 10, con 10 siendo el una predicción de que la asignatura le gustará muchísimo.

4. Manual del Sistema

4.1. Utilización

Para utilizar el sistema, en CLIPS cargar el archivo y realizar (reset) y (run).

Una vez realizado esto, aparece el menú principal. Hay dos opciones: 1 para el Recomendador de Menciones y 2 para el Recomendador de Asignaturas.

Una vez seleccionado pulsando Enter, dependiendo del Recomendador elegido, se piden por pantalla texto o números el cual el usuario inserta normalmente y una vez se realiza una deducción el programa finaliza, se muestra por pantalla el resultado de la ejecución con el razonamiento generado.

4.2. Mantenimiento y Expansión

Internamente, se encuentra el código comentado en donde comienza y donde termina cada módulo. Se posee una cabecera de la siguiente manera:

```
; #####
; # INCIO DELTA
; #####

... Código ...

; #####
; # FIN DELTA
; #####
```

y se encuentra el código dispuesto de manera que es fácil ubicar lo que se necesite cambiar, ya sean reglas o hechos. Como se ha mencionado múltiples veces, se hace uso de muchos registros para facilitar la adición o modificación de conocimiento adicional, ya que esto no es más que añadir otro registro con los datos que se quieren añadir o bien, modificar los que ya están en la Base de Conocimiento.

Encima de esto, el código en general está comentado de manera que alguien con conocimientos básicos de CLIPS pueda entender que se está realizando en cada regla y en cada hecho.