

Creación de dietas para ancianos:
Sistemas Basados en el conocimiento
IA FIB @ UPC

Carlos Bergillos, Roger Vilaseca, Adrià Cabeza

May 9, 2019

Contents

1	Introducción	3
2	Descripción del problema	3
3	Identificación	3
4	Conceptualización	3
5	Conceptos del dominio	3
6	Formalización	3
6.1	Desarrollo de la ontología	3
6.1.1	Definición de clases y jerarquía	4
6.2	Método de resolución	7
7	Implementación	8
7.1	Creación de la ontología	8
8	Juego de Prueba	8
8.1	Ejemplos del conocimiento extraído del dominio	8

1 Introducción

Los **sistemas basados en el conocimiento (SBC)** son la parte de la Inteligencia Artificial donde se intenta hallar una solución utilizando el conocimiento. Hay ciertos problemas en la vida real que no se pueden resolver sin tener conocimiento específico sobre los elementos del dominio.

El motivo de esta práctica es comprender estos modelos de IA así que hemos atacado un problema con esta técnica: creación de menús para ancianos dependiendo de sus necesidades nutricionales y requisitos médicos.

Hemos desarrollado una ontología capaz de almacenar la información sobre los platos, los ingredientes, los nutrientes, las enfermedades y las limitaciones con **Protégé** y un sistema de reglas que describen el proceso de toma de decisiones usando **CLIPS**.

Para atacar el problema de una manera adecuada hemos utilizado las diferentes fases de la metodología de ingeniería del conocimiento:

- Identificación, donde identificaremos el problema y sus características y requisitos
- Conceptualización, donde
- Formalización, donde
- Implementación, donde
- Prueba, donde

2 Descripción del problema

La alimentación es una pieza clave de nuestras vidas: nuestra manera de alimentarnos define nuestra salud. Con nuestra alimentación podemos prevenir el riesgo y agravamiento de enfermedades y mantener un nivel óptimo de salud. Es por eso que cuantos más años tenemos, más tenemos que vigilar nuestra alimentación.

En esta práctica desarrollaremos una solución para este problema. Una solución basada en un sistema basado en el conocimiento que sea capaz de generar menús semanales personalizados adecuados a las características de una persona según su edad, sus restricciones de dieta considerando sus enfermedades y sus preferencias generales sobre la comida. Por ejemplo, si una persona es vegetariana y además diabética se le generaría un horario sin carne y con poca cantidad de azúcares.

3 Identificación

4 Conceptualización

En este apartado describiremos todos los conceptos del dominio.

5 Conceptos del dominio

6 Formalización

6.1 Desarrollo de la ontología

Para desarrollar correctamente la ontología hemos procurado que todos los conceptos del dominio sean representables. En este caso, los términos más importantes de la ontología son los **platos**, puesto que nuestra solución será un conjunto de ellos, los **ingredientes** y sus **características**

nutricionales y los conceptos relacionados con el usuario: **preferencias** y **enfermedades**. Además para poder representar bien los susodichos conceptos, hemos creído conveniente hacer uso de clases auxiliares para definir cantidades. Así pues, por ejemplo para definir que un plato contiene una cantidad de un ingrediente en concreto usaríamos la clase *IngredientQuantity*.

6.1.1 Definición de clases y jerarquía

La ontología que hemos realizado para nuestro SBC contiene diversas clases. Se puede observar en la figura ??

`class_menu`

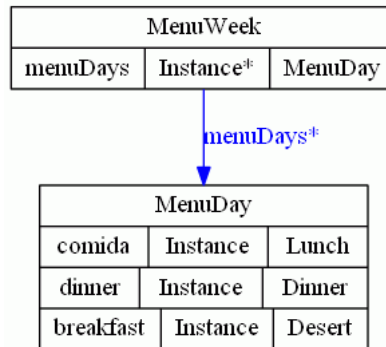


Figure 1: Clase Menu

Definimos toda la información de un menú en esta clase. Tenemos la cantidad de días que contiene nuestro plan alimenticio y distintas instancias de menús diarios los cuales contienen información acerca de sus comidas mediante a sus instancias desayuno, comida y cena.

Los atributos son:

- **menuDays**: menú diario
- *comida*: comida de un menú diario
- *dinner*: cena de un menú diario
- *breakfast*: desayuno de un menú diario

`class_Course`

Course		
season	Symbol*	Summer
		Winter
		Autumn
		Spring
ingredients	Instance*	IngredientQuantity
name		String
cookingMethod	Instance*	CookingMethod
category	Symbol*	FirstCourse
		SecondCourse
		Desert
		Breakfast

Figure 2: Clase Platos

Definimos aquí toda la información que contiene un plato. Tenemos información acerca de qué estación pertenece el plato, los ingredientes que contiene y sus cantidades además de la categoría a la que pertenece que puede ser primer plato, segundo plato, postre o desayuno,

Los atributos son:

- **season:** estación del año; puede ser Otoño, Verano, Invierno o Primavera
- **ingredientes:** ingredientes que contiene un plato y su cantidad
- **name:** nombre del plato
- **category:** categoría del plato; puede ser primer plato, segundo plato, desayuno o postres.

class Ingredient

Ingredient		
season	Symbol*	Summer
		Winter
		Autumn
		Spring
name		String
nutrients	Instance*	NutrientQuantity
energy		Float
weight		String

Figure 3: Clase Ingredientes

Definimos aquí toda la información que contiene un ingrediente. Tenemos de manera parecida a los platos, una estación asociada al ingrediente. Por ejemplo: las castañas estarían asociadas al

otoño. Además también disponemos de los valores nutricionales que presenta el ingrediente.

Los atributos son:

- **season:** estación del año; puede ser Otoño, Verano, Invierno o Primavera
- **name:** nombre del ingrediente
- **nutrients:** nutrientes del ingrediente

class_IngredientQuantity

IngredientQuantity		
ingredient	Instance	Ingredient
quantity		Float

Figure 4: Clase Enfermedad

Para poder expresar la cantidad de ingredientes que contiene un plato hacemos uso de esta clase la cual contiene una referencia a una instancia ingrediente y la cantidad que hay en ese plato del ingrediente.

Los atributos son:

- **ingredient:** ingrediente
- **quantity:** cantidad del ingrediente

class_Nutrients

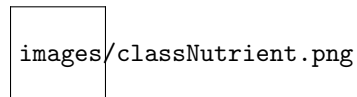


Figure 5: Clase Nutrientes

Definimos en esta clase toda la información que contiene un nutriente... TODO FINISH IT
TODO continuar amb els atributs

class_NutrientQuantity

NutrientQuantity		
nutrient	Instance	Nutrient
quantity		Float

Figure 6: Clase Cantidad de Nutrientes

Para poder expresar la cantidad de nutrientes que contiene un ingrediente en concreto hacemos uso de esta clase. Esta dispone de una referencia a una instancia de nutriente además de la cantidad que ese nutriente en ese ingrediente (las unidades han sido normalizadas en gramos).

Los atributos son:

- **nutrient**: nutriente
- **quantity**: cantidad del nutriente

class_Disease

Disease		
limitations	Instance*	Limitation
name		String

Figure 7: Clase Enfermedad

Definimos en esta clase toda la información que contiene una enfermedad. En nuestro caso, una enfermedad presenta una serie de limitaciones alimenticias que pueden estar referidas a un ingrediente o a un tipo de nutriente.

Los atributos son:

- **limitations**: limitaciones alimenticias que suponen la enfermedad
- **name**: nombre de la enfermedad

class_Limitation

NutrientQuantity		
nutrient	Instance	Nutrient
quantity		Float

Figure 8: Clase Limitación

Para ser capaces de expresar una limitación o incluso una preferencia alimenticia, hemos hecho uso de esta clase para poder relacionar un nutriente o un ingrediente con una limitación de cantidad *AtMost*(como máximo) o *atLeast*(como mínimo).

Los atributos son:

- **element**: elemento sobre el que se aplica la limitación, puede ser un nutriente o un ingrediente
- **atMost**: valor que representa el tope máximo necesario para cumplir la limitación
- **atLeast**: valor que representa el mínimo necesario para cumplir la limitación

6.2 Método de resolución

explicar problemas que intervienen en la resolución

7 Implementación

7.1 Creación de la ontología

Para generar la ontología hemos usado el programa Protégé añadiendo las diferentes características anteriormente mencionadas en los apartados (

8 Juego de Prueba

8.1 Ejemplos del conocimiento extraído del dominio