AVATAR: Un marco formal para razonamiento e inferencia semántica

`TR1 (versión 1.1; última modificación: 6/4/2005)`

AVATAR es un sistema híbrido que combina dos técnicas descritas en la literatura, para personalizar los contenidos de TV mostrados a los usuarios. Los resultados de la investigación llevada a cabo por Robin Burke durante el año 2002, destacan la sinergia de las estrategias que vamos a analizar [4]. Los resultados obtenidos por este trabajo confirman las ventajas de una recomendación híbrida como la considerada en nuestra propuesta.

La primera de ellas se basa en una versión del filtrado colaborativo tradicional, mejorada gracias a la inferencia semántica, que será usada como métrica para detectar similaridad entre las preferencias de varios usuarios, tal como se describe en el documento Doc3 (versión 1.1), disponible en *http://avatar.det.uvigo.es*.

Por otra parte, la segunda de las estrategias —que abordamos en este documento— está basada en el empleo del conocimiento como fundamento del sistema recomendador. La principal contribución de la herramienta de personalización propuesta, es un proceso de razonamiento semántico, llevado a cabo sobre las descripciones de los contenidos de TV y las preferencias de los usuarios. En [4] se identifican en este tipo de sistemas tres elementos:

- El conocimiento sobre los contenidos que se van a recomendar. En AVATAR dicho conocimiento está representado formalmente en una ontología sobre el dominio de la TV, implementada mediante el lenguaje OWL [8].
- El conocimiento sobre los intereses de los usuarios del sistema. Las preferencias de los espectadores están modeladas en AVATAR mediante ontologías construídas incrementalmente, a partir de la base de conocimiento OWL mencionadas antes, a medida que el sistema dispone de información sobre su comportamiento de visionado. A tales estructuras las hemos denominado *perfiles-ontologías*.
- Una función de mapeo que permite establecer correspondencias entre los intereses de los usuarios y los objetos de recomendación. En nuestra propuesta, será un proceso de inferencia de relaciones semánticas entre las preferencias de los espectadores, y el resto de contenidos declarados en la ontología OWL, la función que permita asociar estos dos elementos, con el objetivo de elaborar la recomendación personalizada.

En este documento describimos el marco donde tiene lugar el proceso de razonamiento semántico llevado a cabo en AVATAR. Los elementos clave de este marco son: (i) la base de conocimiento del sistema, esto es, la ontología OWL implementada sobre el dominio de la TV, (ii) los *perfiles-ontología* del usuario, (iii) el lenguaje de consulta LIKO que permite acceder a la jerarquía de clases y propiedades establecida en la ontología y (iv) una clasificación de las relaciones semánticas que pueden ser descubiertas en nuestra base de conocimiento OWL. En relación a estas últimas, y con claro objetivo de garantizar la personalización en las sugerencias ofrecidas, describiremos el proceso que permite inferir de entre todas las posibles relaciones, aquellas que son especialmente significativas en el proceso de recomendación.

1. La ontología de TV

Para conseguir un sistema recomendador de TV personalizado basado en razonamiento semántico, necesitamos organizar el conocimiento de nuestro dominio de aplicación, en este caso, la TV. En el campo de la Web Semántica, las ontologías son una de las tecnologías de uso más extendido para este propósito [5,12]. Por esta razón, en nuestra propuesta hemos implementado una ontología sobre el dominio de la TV, empleando el lenguaje OWL.

Así, nosotros podemos representar las diferentes entidades relacionados con los contenidos audiovisuales mediante la inclusión de clases en nuestra ontología. Hemos definido una clase raíz en la jerarquía de conocimiento, llamada *TV Contents*, de la cual dependen el resto de clases incluidas en ésta (por ejemplo, *Movies*, *TV_Series*, *Informative_Programs*, *Sport_Programs*, etc.). También hemos declarado clases referidas a los créditos que están relacionados con los programas de TV, como presentadores, directores o actores, entre otros (clases *Presenters*, *Directors*, *Actors*, *Actresses*, *Producers*, *ScreenWriters*, etc.).

Los programas más especializados se definen como subclases de aquellos que están más próximos al elemento raíz. De esta forma, tenemos subclases como *Action_Movies*, *Terror_Series*, *News*, *Soccer_Broadcasting*, *Movie_Directors*, *Starring_Actors*, *Supporting_Actors*, etc.).

La ontología también contiene propiedades —organizadas jerárquicamente— para definir características distintivas de los programas de TV, y para establecer relaciones entre las instancias de la base del conocimiento. Dichas propiedades se corresponden con los diferentes tipos contemplados por OWL —propiedades funcionales, transitivas, simétricas, etc.—, favoreciendo mediante esta diversidad los procesos inferenciales llevados a cabo.

Además de la definición de las clases y propiedades, necesitamos popular la base de conocimiento, para disponer de información concreta sobre la que aplicar los preceptos de nuestro razonamiento semántico. Para ello, será necesario recurrir a la programación televisiva, extrayendo instancias concretas de las clases definidas.

2. Los perfiles de usuario

Los perfiles de usuario en AVATAR serán ontologías OWL, construidas incrementalmente a medida que el sistema infiere (o recibe) información sobre el comportamiento de visionado de los espectadores [3]. Conviene recordar que los perfiles almacenarán una parte estática —relativas a los datos personales del usuario facilitados cuando éste se da de alta en el sistema— y una parte dinámica, llamada PSC, referida a sus preferencias en relación a programas de TV.

Una descripción más detallada del proceso de formación y actualización de los *perfiles-ontología* está disponible en el documento Doc1 (versión 1.1) en *http://avatar.det.uvigo.es*.

3. LIKO: Language for Inferring Knowledge from Ontologies

Para poder llevar a cabo el proceso de razonamiento sobre la semántica de los programas declarados como instancias en la ontología del sistema, y sobre las preferencias de los usuarios, AVATAR requiere un lenguaje que permita acceder y consultar la base de conocimiento del sistema. A tal efecto, hemos desarrollado el lenguaje LIKO.

Las consultas típicas que permite éste incluyen (i) la obtención de superclases o subclases a partir de una clase dada, (ii) de superpropiedades y subpropiedades a partir de una propiedad, o (iii) la extracción de propiedades con determinadas restricciones sobre su rango o su dominio, tal como reflejan los operandos de consulta resumidos en la Tabla 1.

4. Clasificación de relaciones semánticas

En nuestra propuesta hemos dado cabida a tecnologías propias de la Web Semántica, en virtud de las semejanzas identificadas entre el contexto de esta última y nuestro entorno de TV personalizada. En la Web Semántica se maneja gran cantidad de información y se establecen multitud de relaciones entre las entidades identificadas, cuya inferencia ayuda a resolver de forma mejorada las consultas planteadas por los usuarios.

Del mismo modo, la TV proporciona una amplia oferta de programas, que obliga a someter a los usuarios a tediosos procesos de búsqueda, en los que es necesario discriminar los contenidos irrelevantes antes

Tabla 1. Operadores de consulta del lenguaje **LIKO**

Operators	Input	Output
$\triangleright$	Una clase de la ontología.	Subclases directas relativas a la clase de entrada.
$\triangleleft$	Una clase de la ontología.	Superclases directas relativas a la clase de entrada.
$>>$	Una propiedad de la ontología.	Subpropiedades directas relacionadas con la propiedad de entrada.
$<<$	Una propiedad de la ontología.	Superpropiedades directas referidas a la propiedad de entrada.
$>\Leftarrow$	Instancias de clases de la ontología.	Propiedades donde el dominio es una instancia de la clase de entrada.
$>\Rightarrow$	Instancias de class de la ontología.	Propiedades donde el rango es una instancia de la clase de entrada.
$>\odot$	Instancias de clases o valores concretos de propiedades *Datatype*.	Propiedades inferidas de las propiedades transitivas contenidas en la ontología.
$>\oplus$	Instancias de clases o valores concretos de propiedades *Datatype*.	Propiedades inferidas de las propiedades inversas contenidas en la ontología.
$>\hookleftarrow$	Instancias de clases o valores concretos de propiedades *Datatype*.	Propiedades inferidas de las propiedades simétricas contenidas en la ontología.
$>\ominus$	Una propiedad definida en la ontología.	Propiedad inversa de la propiedad de entrada.

de encontrar aquellos que realmente interesan. Así, nuestra propuesta contribuye a mejorar las estrategias convencionales empleadas en el contexto de la TV, mediante la inferencia de relaciones implícitas a partir de las propiedades declaradas en la ontología de TV. Dicha inferencia permite no sólo personalizar las sugerencias ofrecidas, sino también diversificarlas, ya que algunas de las relaciones que manejaremos, no serían descubiertas mediante los enfoques de personalización descritos en la literatura.

El creciente interés que está despertando actualmente este nuevo tipo de Web, ha provocado que la comunidad investigadora se centre en el desarrollo de capacidades de consulta semántica. Este tipo de técnicas usan descripciones (mediante metadatos) de los contenidos analizados para responder a las necesidades de búsqueda del usuario. Si bien existen diferentes lenguajes para consultar, por ejemplo, datos RDF —RDQL [10], SquishQL [9], TRIPLE [11], entre otros— éstos responden a las consultas planteadas empleando conceptos ontológicos o realizando inferencias muy simples, basadas en el principio de *subsumption*.

De lo que acabamos de decir se desprende que el descubrimiento de relaciones más complejas en una base de conocimiento aún no está soportado. De hecho, los lenguajes existentes se basan en un paradigma de consulta semántica de la forma: "*Recuperar todas las entidades que estén relacionadas mediante la relación R con la entidad A*". Es decir, es el usuario quien tiene que especificar qué tipo de relaciones le interesan, lo cual limita considerablemente los procesos de inferencia que pueden ser llevados a cabo.

En el descubrimiento de relaciones semánticas más complejas han centrado sus esfuerzos Kemafor Anyanwu y Amit Sheth, en el contexto del proyecto *SemDis*[1]. En este enfoque, se procesan consultas semánticas basadas en paradigmas de la forma: "*¿Cómo están relacionadas las entidades A y B?*". Para responder a este tipo de solicitudes por parte del usuario, en [1] los autores definen una clasificación de diferentes asociaciones semánticas. Nuestro enfoque tomará como punto de partida dichas asociaciones, con el claro objetivo de abordar el razonamiento semántico perseguido en **AVATAR**.

Si establecemos una comparativa entre este enfoque y nuestro proceso de recomendación semántica personalizada, la entidad A se corresponde con cada uno de los programas definidos en el *perfil-ontología* del usuario, y B con los programas finalmente sugeridos al espectador.

[1] `http://lsdis.cs.uga.edu/Projects/SemDis/`

Por tanto, nuestra propuesta de razonamiento está basada en una metodología que permita resolver consultas del tipo "*¿Qué entidades están relacionadas con A y de qué tipo es dicha relación?*". En este sentido, es conveniente destacar que si bien dicha metodología se aplica en el sistema AVATAR, es lo suficiente flexible como para ser generalizable a todas aquellas aplicaciones de la Web Semántica, donde tengan interés este clase de consultas (recomendadores de libros, música, etc., a partir de los productos consumidos por el usuario). Para ello, bastaría con conceptualizar adecuadamente —en una ontología— el conocimiento relativo al nuevo dominio de aplicación (géneros literarios, musicales, etc.).

Las diferencias existentes entre los enfoques de consulta tradicionales, el abordado en **SemDis** y el propuesto en nuestro trabajo aparecen resumidos en la Fig. 1.

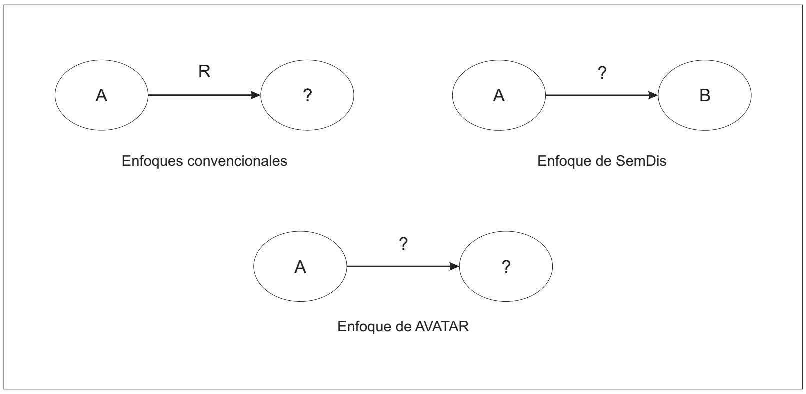

Figura 1. Diferentes paradigmas de consulta semántica

En la Secc. 4.1 describimos en detalle el marco formal descrito por Anyanwu y Sheth en [1] en relación al descubrimiento de asociaciones semánticas, para en la Secc. 4.2 centrarnos en los preceptos del razonamiento semántico propuesto.

4.1. El enfoque del proyecto SemDis

Para entender el formalismo propuesto en el marco del proyecto **SemDis**, resultan imprescindibles los siguientes conceptos:

- **Secuencias de propiedades** (*Property Sequences* ó PS): Una secuencia de propiedades PS es un conjunto finito de propiedades $[P_1, P_2, P_3, \ldots P_n]$, donde P_i es una propiedad definida en una esquema RDF (RDF(S)). La longitud de una secuencia de propiedades —*Length (PS)*— se define como el número total de propiedades que la integran.

Además de la secuencia PS, también se define en este trabajo la función *NodesOfPS*, que devuelve el conjunto de clases de la secuencia de propiedades PS, es decir:

$$\textit{PS.NodesOfPS ()} = \{C_1, C_2, C_3, \ldots C_k\}$$

donde se tiene que las clases C_j cumplen:

$$C_j = \begin{cases} domain(P_j) & \text{si} \quad \text{j = 1} \\ range(P_{j-1}) = domain(P_j) & \text{si} \quad \text{1 < j < k} \\ range(P_{j-1}) & \text{si} \quad \text{j = k} \end{cases}$$

Para recuperar las instancias de las clases C_j que figuran en la secuencia de propiedades PS, se define la función:

$$\textit{ps.PSNodesSequence ()} = [\ v_1, v_2, v_3, \ldots v_{n+1}\ ]$$

donde ps es una instancia de la secuencia PS, v_1 se denomina *origen* de la secuencia y v_{n+1}, *término* de la misma.

Entre las secuencias de propiedades es posible establecer relaciones binarias. En el trabajo de Anyanwu y Sheth se definen dos de éstas:

(i) **Secuencias de propiedades unidas** (*Joined Property Secuences*). Se dice que PS_1 y PS_2 son secuencias unidas — representado como $PS_1 \bowtie_\rho PS_2$— si se cumple que:

$$\textit{NodesOfPS}\ (PS_1) \bigcap \textit{NodesOfPS}\ (PS_2) \neq \emptyset$$

La clase de unión C ($C \in$ *NodesOfPS* $(PS_1) \bigcap$ *NodesOfPS* (PS_2)) entre PS_1 y PS_2 se denomina *nodo de unión* (*join node*). Por otra parte, cabe decir que la longitud de una secuencia de propiedades unidas, será igual a la suma de las propiedades de las dos secuencias por separado, esto es, la longitud del camino no dirigido que va desde el origen de una de las secuencias, hasta el origen de la otra.

(ii) **Secuencias de propiedades isomórficas** ($\cong_\rho$ ρ *- Isomorphic Property Sequences*): Partamos de las secuencias de propiedades PS_1 = [$P_1, P_2, P_3, \ldots P_m$] y PS_2 = [$Q_1, Q_2, Q_3, \ldots Q_m$].

Decimos que PS_1 y PS_2 son secuencias ρ - isomórficas —representado como $PS_1 \cong_\rho PS_2$— si se cumple que:

$$P_i = Q_i \quad \text{ó} \quad P_i \sqsubseteq Q_i \quad \text{ó} \quad Q_i \sqsubseteq P_i \quad (\sqsubseteq \text{ significa } \textit{subpropiedad}) \quad \forall\ \text{i}, 0 \leq \text{i} \leq \text{m}$$

A continuación, describimos las relaciones semánticas definidas en [1], basadas en las secuencias de propiedadesque acabamos de presentar. Estas relaciones reciben el nombre de *asociaciones semánticas*.

I **Asociación mediante camino semántico** (ρ *- pathAssociated*): ρ *- pathAssociated* (x,y) se cumple si existe una secuencia de propiedades PS y una instancia ps de dicha secuencia, donde x es el origen de ps e y, el término de la misma, o viceversa (y es el origen y x el término).

II **Asociación mediante unión** (ρ *- joinAssociated*): Sean PS_1 y PS_2 dos secuencias de propiedades con un nodo de unión C, tales que $PS_1 \bowtie_\rho PS_2$.
Además existen dos instancias de dichas secuencias, esto es, ps_1 y ps_2, y $n \in$ ps1.PSNodesSequence () $\bigcap$ ps2.PSNodesSequence ().

Diremos que ρ *- joinAssociated* (x,y) se cumple si se dan las siguientes condiciones:

- x es el origen de ps_1 e y es el origen de ps_2, o bien
- x es el término de ps_1 e y es el término de ps_2.

Este tipo de relación se daría entre las secuencias de propiedades indicadas a continuación:
ps_1.*PSNodesSequence* = [$\mathbf{x}, b, c \ldots \mathbf{n}, \ldots r$] y ps_2.*PSNodesSequence* = [$\mathbf{y}, \beta, \chi \ldots \mathbf{n}, \ldots \lambda$], o bien ps_1.*PSNodesSequence* = [$a, b, c \ldots \mathbf{n}, \ldots r, \mathbf{x}$] y ps_2.*PSNodesSequence* = [$\alpha, \beta, \chi \ldots \mathbf{n}, \ldots \lambda, \mathbf{y}$] .

III **Asociación mediante padre común** (ρ - *cpAssociated*): Este es un caso especial de la definición dada para *propiedades* ρ *- isomórficas*, y permite reflejar relaciones de hermandad entre los recursos analizados.

Así, ρ - *cpAssociated* (x,y) se cumple si existen dos secuencias de propiedades PS_1 y PS_2, tales que $PS_1 \bowtie_\rho PS_2$, para las cuales además se verifica *joinAssociated* (x,y). Ambas secuencias PS_1 y PS_2 son de la forma *rdf.typeOf. (rdfs:subClassOf)*. Es decir, esta relación establece que dos entidades están relacionadas cuando pertenecen a una misma clase, o bien, a clases hermanas (*common parent*).
De esta forma, estamos permitiendo que las secuencias ps_1 y ps_2 tengan en común no sólo instancias de la misma clase, sino también de clases hermanas, para establecer una relación semántica entre sus orígenes (o entre sus términos).

Sin embargo, para evitar recuperar relaciones poco significativas, en este enfoque se introducen dos parámetros, a modo de restricciones:

1. El nodo de unión (*join node*) j de la secuencia de propiedades unida (*Joined Property Sequences*) tendrá asociado un *techo*, es decir, se especifica la clase más general en la jerarquía de conocimiento, que puede ser *padre común* de las clases consideradas en la relación.
2. La longitud de la secuencia de propiedades unida debe ser mínima para que la relación inferida sea suficientemente relevante. Así, es posible especificar un valor que asumirá el papel de umbral en relación a la longitud de esta secuencia.

IV **Asociación mediante isomorfismo** (ρ - *IsoAssocited*): Se cumple ρ - *IsoAssociated* (x,y) si existen dos secuencias de propiedades PS_1 y PS_2, tales que $PS_1 \cong_\rho PS_2$, y además existen dos instancias ps_1 y ps_2 de las secuencias anteriores, tales que x es el origen de ps_1 e y es el origen de ps_2.

Finalmente, destacar que en el enfoque del proyecto **SemDis** se establece que dos nodos x e y están asociados semánticamente cuando están:

- asociados mediante camino (ρ - *pathAssociated* (x,y))
- asociados mediante padre común (ρ - *cpAssociated* (x,y))
- asociados mediante isomorfismo (ρ - *IsoAssociated* (x,y))   ó
- asociados mediante unión (ρ - *joinAssociated* (x,y))

4.2. El enfoque utilizado en AVATAR

En AVATAR emplearemos los conceptos y las *asociaciones semánticas* introducidas en la Secc. 4.1. Nuestro enfoque, incluirá nuevos criterios a la hora de relacionar tanto las clases como las propiedades de la base de conocimiento OWL, dando lugar a nuevas *asociaciones* basadas en la combinación de las manejadas en el proyecto **SemDis** [1].

Es conveniente destacar que las asociaciones/relaciones semánticas inferidas, deberán ser sometidas a un proceso de filtrado/*ranking*, en el que se extraigan aquellas que sean más significativas para el proceso de recomendación. Dado que las sugerencias mostradas a los usuarios deben ser personalizadas, en este proceso de *ranking* el elemento clave será la información personal que figure en el *perfil-ontología* del espectador, así como el conocimiento representado en la ontología de TV.

En esta exposición, iremos analizando los fundamentos del proceso de inferencia descrito en la sección previa, destacando tanto las partes comunes como las diferencias reseñables entre este trabajo y el enfoque propuesto en AVATAR.

Empecemos centrándonos en las secuencias de propiedades. Nosotros adoptamos la definición dada en [1] para este tipo de estructuras, si bien en lugar de trabajar sobre bases de conocimiento RDF(S), lo haremos sobre la ontología OWL descrita en la Secc. 1.

Aunque las secuencias pueden estar formadas por propiedades *Object* y *Datatype* (como ya hemos mencionado en los documentos Doc1 y Doc3), en esta propuesta emplearemos únicamente las integradas por propiedades *Object*, ya que nuestra inferencia se centra en el razonamiento sobre instancias concretas[2].

Así, nuestras secuencias de propiedades serán conjuntos de propiedades *Object* cumpliendo que el dominio de una de ellas es el rango de la siguiente en la secuencia, como ya adelantamos en la Secc. 4.1.

Por otra parte, definimos un subconjunto de dichas secuencias, incluido en nuestra propuesta bajo el nombre de *subsecuencias de propiedades*.

Definición 0: **Subsecuencia de propiedades**

Sea $PS = [P_1, P_2, P_3, \ldots P_n]$ una secuencia de propiedades. A partir de ésta, podemos definir la subsecuencia de propiedades PSS como una conjunto de propiedades de la forma:

$$\mathbf{PSS} = [P_1, \ldots, P_j], \text{ con } j < n$$

De forma análoga a las secuencias de propiedades, la función *PSS.NodesOfPSS ()* extrae las clases que integran la subsecuencia PSS. Por otra parte, una instancia de la subsecuencia PSS —por ejemplo, pss— puede ser definida como un conjunto de propiedades que unen diferentes instancias entre sí. Éstas últimas se obtienen mediante la función *pss.PSNodesSequence ()*.

Además, siendo *pss.PSNodesSequence ()* = $\{w_0, \ldots, w_{j+1}\}$, emplearemos la notación

$$\mathbf{w_0[P_1]w_1 \ldots w_{j-1}[P_j]w_j}$$

para referirnos a las relaciones establecidas entre los nodos incluidos en la subsecuencia, mediante las propiedades P_l $(0 \leq l \leq j)$.

En lo que resta de documento, asumimos que todo lo que sea válido para una secuencia de propiedades, también lo será para las subsecuencias correspondientes.

Antes de abordar las relaciones establecidas entre las secuencias de propiedades, introducimos un concepto útil a la hora de presentar nuestro enfoque.

Definición 1: **Relación de subsumption directa** ($\sqsubseteq_D$ *Direct subsumption relation*)

Esta relación se establecerá tanto entre las clases como entre las propiedades de la ontología OWL de TV.

Sean P_1 y P_2 dos propiedades definidas en la base de conocimiento OWL. Diremos que entre ambas hay relación de *subsumption* directa —representado como $P_1 \sqsubseteq_D P_2$— cuando P_2 es superpropiedad directa de P_1.

De forma análoga, siendo C_1 y C_2 dos clases incluidas en la ontología, diremos que entre ambas hay relación de *subsumption* directa —representado como $C_1 \sqsubseteq_D C_2$— cuando C_2 es superclase directa de C_1.

Al igual que [1], nuestro enfoque también considera diferentes relaciones establecidas entre las secuencias de propiedades definidas en la base de conocimiento del sistema. Éstas son:

- **Secuencias de propiedades unidas** ($\bowtie_\rho$ *Joined Property Secuences*)

---
[2] Las propiedades *Datatype* tenían interés para el enfoque de filtrado colaborativo semántico descrito en el Doc3, ya que en este caso sí analizábamos sus valores concretos.

- *Definición 2*: **Nexo semántico entre propiedades** ($\leftrightarrow_\rho$ *Semantic Nexus between Properties*)

  La inclusión de esta relación en nuestro proceso de inferencia, supone extender la definición de *secuencias de propiedades isomórficas* dada en [1].

  Partamos de las propiedades P_i y Q_j definidas en la ontología OWL de TV.

  Decimos que P_i y Q_j son propiedades relacionadas mediante *nexo semántico* —representado mediante la notación $P_i \leftrightarrow_\rho Q_j$— si se cumple(n) alguna(s) de las siguientes condiciones:

  - $P_i = Q_j$
  - $P_i \sqsubseteq Q_j$
  - $Q_j \sqsubseteq P_i$
  - Existen r y l (propiedades de la ontología OWL), tales que $P_i \sqsubseteq_D r \sqsubseteq_D l$ y $Q_j \sqsubseteq_D l$. Es decir, en este caso P_i es subpropiedad *directa* de una propiedad hermana de Q_j.
  - Existen r y l (propiedades en la ontología OWL), tales que $Q_j \sqsubseteq_D r \sqsubseteq_D l$ y $P_i \sqsubseteq_D l$. En este caso P_i es propiedad hermana de una superpropiedad *directa* de Q_j.
  - P_i es propiedad inversa de Q_j ó viceversa.

Una vez establecidos los criterios que usaremos para considerar que las secuencias de propiedades definidas en nuestra ontología de conocimiento están relacionadas, pasamos a describir la clasificación de las asociaciones semánticas que se desprende de las mismas.

- **Asociación mediante camino semántico** (ρ - *pathAssociated*), tal como se describió en la Secc. 4.1. Para referirnos a esta relación utilizaremos indistintamente *asociación mediante camino semántico* ó *camino semántico*.
- **Asociación mediante unión** (ρ - *joinAssociated*).
- **Asociación mediante padre común** (ρ - *cpAssociated*).
- *Definición 3*: **Asociación semántica indirecta mediante isomorfismo** (ρ - *Indirect IsoAssociated*)

  Decimos que se cumple ρ *-Indirect IsoAssociated* (x,y) si existen dos secuencias de propiedades PS_1 y PS_2, tales que sus propiedades están relacionadas mediante *nexo semántico*, y además existen dos instancias ps_1 y ps_2 de las secuencias anteriores, en las cuales x es el origen de ps_1 e y es el origen de ps_2.

Partiendo de la premisa de que cuanto más directas sean las relaciones inferidas, más significativas serán a la hora de descubrir nuevo conocimiento, nuestro enfoque combina las asociaciones semánticas que hemos presentado hasta este punto. Dado que de las relaciones descritas, aquellas que se establecen mediante camino semántico son las más directas, serán precisamente éstas las que combinemos —utilizando el principio de transitividad— con las restantes, permitiendo así al sistema enriquecer los procesos inferenciales llevados a cabo.

Para abordar la descripción de este tipo de asociaciones semánticas (que hemos denominado *asociaciones semánticas mixtas tipo I, II y III*), partamos de la siguiente información. Sean PS_1, PS_2 y PS_3 tres secuencias de propiedades, y ps_1, ps_2 y ps_3 instancias de las mismas. Supongamos además que se cumple que x es el origen de ps_1, y el origen de ps_2 y z, el correspondiente a ps_3.

- *Definición 4*: **Asociación semántica mixta tipo I** (*ρ - join-pathAssociated*)

  Esta relación será una combinación de las *asociaciones mediante camino semántico* y las *asociaciones mediante unión*.

  Diremos que *ρ - join-pathAssociated* (x,z) es cierto si se cumplen cualquiera de las dos condiciones expuestas a continuación:

  ρ - pathAssociated (x,y) y *ρ - joinAssociated* (y,z) o bien
  ρ - joinAssociated (x,y) y *ρ - pathAssociated* (y,z)

- *Definición 5*: **Asociación semántica mixta tipo II** (*ρ - cp-pathAssociated*)

  Esta relación es análoga a la presentada antes, con la salvedad de que en esta ocasión se combinan *asociaciones mediante camino semántico* y las *asociaciones mediante padre común*.

  Diremos que *ρ - cp-pathAssociated* (x,z) es cierto si se cumplen cualquiera de las dos condiciones expuestas a continuación:

  ρ - pathAssociated (x,y) y *ρ - cpAssociated* (y,z) o bien
  ρ - cpAssociated (x,y) y *ρ - pathAssociated* (y,z)

- *Definición 6*: **Asociación semántica mixta tipo III** (*ρ - iso-pathAssociated*)

  Esta relación es la combinación de las *asociaciones mediante camino semántico* y las *asociaciones indirectas mediante isomorfismo*.

  Diremos que *ρ - iso-pathAssociated* (x,z) es cierto si se cumplen cualquiera de las dos condiciones expuestas a continuación:

  ρ - pathAssociated (x,y) y *ρ - Indirect Iso-Associated* (y,z) o bien
  ρ - Indirect Iso-Associated (x,y) y *ρ - pathAssociated* (y,z)

En la siguiente sección describiremos cómo se lleva a cabo la inferencia de las asociaciones semánticas que acabamos de presentar durante el proceso de razonamiento perseguido en **AVATAR**. De esta tarea se encargan los agentes semánticos incluidos en la arquitectura del sistema [2].

5. Razonamiento e Inferencia de Asociaciones Semánticas en AVATAR

5.1. Ideas generales

A la hora de descubrir conocimiento mediante asociaciones semánticas complejas podemos definir dos enfoques claramente diferenciados.

- El primero de ellos consistiría en inferir todas las relaciones semánticas que hemos definido en la Secc. 4.2, para después filtrar aquellas que sean significativas para la elaboración de recomendaciones personalizadas para el usuario. Esta es la dirección que han seguido en el proyecto **SemDis** [6,7].
- La segunda posibilidad es simultanear los procesos de inferencia y filtrado de asociaciones semánticas (en adelante *inferencia-filtrado*). Es decir, en lugar de obtener todas las posibles relaciones, para posteriormente extraer las más relevantes, se infieren únicamente aquellas que son significativas en el análisis.

Teniendo en mente nuestro propósito de personalización, es más adecuado abordar un enfoque como el descrito en segundo lugar, donde el proceso de filtrado emplea la información personal definida en el *perfil-ontología* del usuario, complementándola con el conocimiento representado en la ontología OWL. Esta propuesta tiene la ventaja de que la salida del proceso de inferencia-filtrado, será directamente una lista de los programas relacionados semánticamente con los contenidos de interés del usuario, de la cual se extraen fácilmente las sugerencias personalizadas.

En la siguiente sección describimos detalladamente nuestra propuesta de *filtrado-inferencia*, mostrando el proceso mediante el cual se descubren todas las relaciones semánticas que hemos presentado en la Secc. 4.2.

5.2. Nuestro enfoque: Inferencia y Filtrado en un solo paso

Como ya adelantamos en la sección anterior, el punto de partida de los procesos inferenciales llevados a cabo en AVATAR serán las instancias definidas en el *perfil-ontología* del usuario, preservando así la personalización de las recomendaciones. Más concretamente, nos centraremos en analizar las secuencias de propiedades definidas en la ontología de TV, cuyo inicio sean dichas instancias.

En relación a los perfiles de usuario, es conveniente empezar recordando algunos aspectos relativos a las secuencias almacenados en éstos. En el Doc1 de la intranet decíamos que de los *perfiles-ontología* en AVATAR almacenan secuencias de longitud uno o de longitud dos. Esto se debe a que junto a cada programa se almacenan propiedades *Datatype* para definir características concretas de los mismos (longitud uno), y también se almacenan aquellas instancias que estén directamente relacionadas con el programa mediante una propiedad *Object*, junto a las *Datatype* asociadas a dichas instancias (longitud dos). De esta forma, cada secuencia almacenará una única instancia (el propio programa en las secuencias de longitud uno), o bien dos instancias (el programa y la relacionada con él mediante la propiedad *Object*, en las secuencias de longitud dos). Para no sobrecargar el perfil, hemos optado por incluir este subconjunto de información en éste.

Dicho en otras palabras, el perfil únicamente almacena subsecuencias construidas a partir de las sucuencias definidas en la ontología OWL. Es decir, de toda la secuencia de la ontología en el perfil se registran únicamente una instancia (para las de longitud uno) o dos (para las de longitud dos), siendo necesario recurrir a la base de conocimiento OWL para acceder al resto de nodos de la secuencia. Tal decisión de diseño, evita una sobrecarga de información en los perfiles sin necesidad de perder conocimiento de cara al proceso de razonamiento. Así, para llevar a cabo las inferencias bastará con localizar los programas del perfil, entre el conjunto de instancias definido en la ontología OWL. De esta forma, será posible analizar todas las instancias que integran las secuencias que parten de dichos programas, en lugar de analizar únicamente una o dos instancias que son las que se almacenan en el perfil para cada secuencia.

Para poder describir nuestro enfoque, asumimos un *perfil-ontología* concreto, que hemos denominado P_X. Supongamos que éste contiene M instancias entre las preferencias positivas del espectador y G instancias entre las preferencias negativas. Todas ellas están incluidas en el conjunto In_{P_X}.

$$\mathbf{In_{P_X}} = \{x_0, x_1, \ldots, x_{M-1}, y_0, y_1, \ldots y_{G-1}\}$$

Por otra parte, asumimos que cada una de las instancias x_i ($0 \leq i < M$) tiene asociadas R_i secuencias de propiedades en nuestra ontología OWL, incluidas en el conjunto $PropSeq_{x_i}$. Esto significa que en la ontología OWL de TV hay R_i secuencias cuyo inicio es x_i. Análogamente también las instancias y_m ($0 \leq m < G$) también tendrán Q_m secuencias. Centremos nuestro análisis en las primeras.

$$\mathbf{PropSeq_{x_i}} = \{ps_0^{x_i}, \ldots, ps_{R_i-1}^{x_i}\}$$

Sean además ps_k^i.*PSNodesSequence ()*, los nodos (instancias) que integran cada una de las $R_i \cdot M$ secuencias que acabamos de presentar. Suponemos que cada uno de estos conjuntos almacena N^{k,x_i} instancias. De ello se deduce fácilmente la expresión mostrada a continuación:

$$\mathbf{ps_k^{x_i}}.\textit{PSNodesSequence}() = \{x_i, v_0^{k,x_i}, v_1^{k,x_i}, \ldots, v_{N^{k,x_i}-2}^{k,x_i}\}, \;\; 0 \leq i < M, \;\; 0 \leq k < R_i \tag{1}$$

El proceso de de razonamiento propuesto consiste en analizar las secuencias mostradas en la Ec. 1 (a las que denominaremos en adelantes *secuencias originales*), verificando si todos o sólo algunos de sus nodos deben formar parte de las relaciones inferidas (de ahí el nombre de *filtrado-inferencia*).

Así, a partir de dichas secuencias se obtiene un nuevo conjunto llamado *secuencias objetivo*. Estas últimas pueden ser idénticas a las definidas en las *secuencias originales* (se incluyen todos los nodos de la secuencia), o bien subsecuencias construídas a partir de las mismas (únicamente figuran parte de los nodos de las *secuencias originales*). Las secuencias objetivo serán el elemento clave a partir de las cuales se infieren todas las asociaciones semánticas presentadas en la Secc. 4.2.

Empecemos centrándonos en la obtención de los *caminos semánticos*, para en la Secc. 5.2.2 abordar el proceso de descubrimiento de las restantes. Esta relación se obtienen fácilmente desde las secuencias de propiedades exploradas, sin más que extraer el inicio y el término de las mismas. Así, si tenemos la *secuencia objetivo* $[a, b, \ldots, z]$, la asociación *ρ-pathAssociated* (a,z) es fácilmente deducible.

Tal como dijimos antes, los inicios de las *secuencias objetivo* serán siempre instancias definidas en el perfil del usuario, ahora bien, ¿cuál es el término de las mismas: la última instancia de las *originales* o un nodo intermedio de las mismas?

Dicho de otra forma, si establecemos una comparativa con la Fig. 1, tenemos que A se corresponde con cada una de las instancias definidas en el *perfil-ontología* del usuario y la relación establecida es, en este caso concreto, una *asociación mediante camino semántico*. Sin embargo, falta por identificar la entidad que está relacionada con A. Así, según la clasificación de relaciones definida en la Secc. 4.2, nuestro objetivo es detectar asociaciones de la forma *ρ - pathAssociated* $(x_i,?)$.

Para inferir este tipo de relación, en primer lugar, establecemos un parámetro (cuyos valores están comprendidos entre 0 y 1), que denominamos *umbral semántico*. A continuación, el sistema explora uno a uno los nodos de cada *secuencia original* iniciada en x_i. Para cada uno de ellos, se calcula un factor — llamado en adelante *intensidad semántica* (λ_{Sem})— , cuyo valor (también entre 0 y 1) es comparado con dicho umbral. De forma intuitiva, podemos decir que la intensidad semántica será el índice que decida si una determinada instancia está suficientemente relacionada con cada instancia almacenada en el perfil personal. Cuanto más elevado sea el valor de dicha intensidad semántica, más directa será la relación existente entre las preferencias de un usuario, y la instancia a la que dicha intensidad pertenece.

Como hemos mencionado, el proceso que proponemos debe ser aplicado a cada una de las secuencias de propiedades cuyo inicio sean las instancias del perfil del usuario, tal como indica el *pseudocódigo* mostrado en el Alg. 1.

Algoritmo 1

Recorrer las M instancias del perfil-ontología P_x (i = 0:M)
  Recorrer las R_i secuencias asociadas a la cada x_i en la ontología (k = 0:R_i)
    for each secuencia $ps_k^{x_i}$ referida a x_i
      Recorrer los N^{k,x_i} nodos del conjunto $ps_k^{x_i}$.PSNodesSequence () (j = 0:N^{k,x_i})
        if $\lambda_{Sem}(v_j^k) \geq \gamma_{Sem}$ then *ρ - pathAssociated* (x_i, v_j^k)
        else *ρ - pathAssociated* (x_i, v_{j-1}^k)

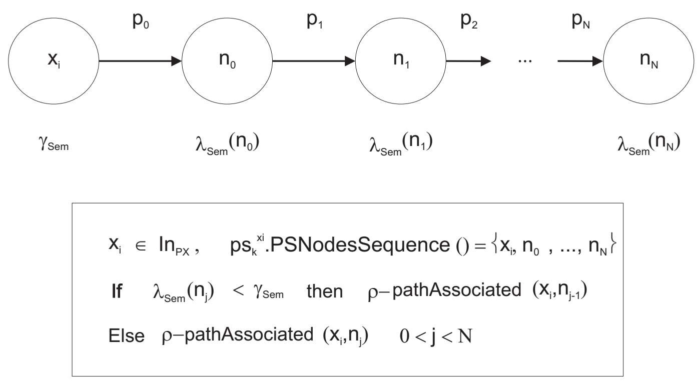

Figura 2. El enfoque de *inferencia-filtrado* propuesto en AVATAR

La inferencia para el caso de una secuencia concreta referida a una de las instancias x_i, se representa también en la Fig. 2. Destacar que para mayor claridad en la figura, hemos renombrado los nodos de la secuencia mostrada, denominándolos n_0 , . . ., n_N en lugar de recurrir al uso de superíndices como los indicados en la Ec. 1.

En esta figura se ve que para cada nodo n_j dado, se comprueba si su intensidad es mayor o igual que el umbral fijado. En caso afirmativo, n_j forma parte de la relación ρ - *pathAssociated* (x_i,?) que intentamos inferir. En este punto, la inferencia semántica nos habría conducido a un camino de la forma ρ - *pathAssociated* (x_i,n_j). Esta misma comprobación deberá trasladarse a los restantes nodos de la secuencia en cuestión, verificando así si el camino semántico termina en n_j, o si por el contrario, éste forma parte del mismo pero no lo finaliza. Por otra parte, en caso de que la intensidad asociada a n_j sea inferior al *umbral semántico*, el camino termina en la instancia n_{j-1} por ser ésta la última que ha verificado la condición mencionada antes.

Precisamente será la condición de inclusión que acabamos de mencionar en nuestro enfoque, la que permite simultanear los procesos de inferencia y filtrado que adelantábamos en la sección anterior. De esta forma, no sólo inferimos conocimiento, sino que extraemos aquel que sea relevante para nuestro proceso de recomendación personalizada. Para este propósito será necesario formalizar una expresión adecuada para la *intensidad semántica*, a cuyo cálculo dedicaremos la Secc. 5.2.1. Antes hagamos algunas reflexiones sobre los valores del *umbral semántico* en nuestra propuesta.

Como ya adelantamos al comienzo de la Secc. 4, nuestro proceso de razonamiento contribuye a diversificar la oferta de programas ofrecidos al usuario, sin perder el carácter personalizado de los mismos, gracias a los procesos inferenciales llevados a cabo sobre la ontología OWL. Sin embargo, en dichos procesos es necesario llegar a un compromiso entre la diversidad y la personalización. La primera se ve favorecida por la inferencia de asociaciones indirectas, difíciles de detectar a simple vista, mientras la personalización se corresponde con relaciones directas, fácilmente deducibles, o incluso explícitamente declaradas entre las instancias de la ontología. En otras palabras, si inferimos relaciones demasiado indirectas, corremos el riesgo de despersonalizar las recomendaciones elaboradas. Esta es la premisa en la que basamos nuestro proceso de *inferencia-filtrado*.

En esta dirección, uno de los elementos decisivos será el valor establecido para el *umbral semántico*, en cuyo análisis nos centramos a continuación. Para ello, conviene recordar que cuanto mayor sea la intensidad de una instancia, mayor será su relación con las preferencias recogidas en el perfil del usuario.

Empecemos asumiendo un valor muy bajo para el umbral. En este escenario, se favorecen enormemente los procesos inferenciales, detectándose relaciones muy indirectas, y por ende poco significativas, entre las preferencias del espectador y los nodos explorados en las *secuencias originales* analizadas. Esto último se debe a que la *intensidad semántica* exigida a dichos nodos no tiene que ser muy elevada, para superar el umbral establecido y, por tanto integrar las *secuencias objetivo*.

Considerando que en el supuesto que acabamos de describir, obtenemos gran cantidad de relaciones poco significativas para el proceso de personalización, podemos pensar en filtrar únicamente las más relevantes. Si para ello establecemos un umbral demasiado elevado, por ejemplo máximo, estaríamos exigiendo a las instancias analizadas una *intensidad semántica* máxima (para así igualar dicho umbral). Teniendo en cuenta que cuanto mayor es la intensidad de un nodo, mayor es su relación con las preferencias del usuario, es obvio que en este caso las recomendaciones ofrecidas serían fruto de las propiedades declaradas explícitamente en la ontología OWL, eliminando la posibilidad de inferir nuevo conocimiento. En esta clasificación podríamos identificar las recomendaciones ofrecidas mediante los enfoques existentes en la TV personalizada, donde las capacidades de razonamiento son especialmente limitadas.

Sin embargo, nuestro objetivo no es sugerir contenidos al espectador desde las propiedades explícitas de la ontología, sino inferir a partir de ellas relaciones implícitas que sean suficiente relevantes para asegurar un alto grado de personalización. Así, parece claro que el umbral debería tomar un valor intermedio. De esta forma, habremos alcanzado el compromiso al que hacíamos alusión antes, entre la diversificación de las recomendaciones —inferencia de relaciones indirectas (umbral reducido) — y la personalización de las mismas —relaciones significativas para la recomendación (umbral elevado).

A la vista de las reflexiones que hemos hecho hasta este punto, es obvio que el elemento clave de nuestra propuesta será la expresión de la *intensidad semántica*, ligada íntimamente a los procesos inferenciales y, por ende, a las recomendaciones ofrecidas. Esto se debe a que el valor de este parámetro es el que decide si una instancia es o no relevante para la inferencia de conocimiento, en relación al umbral establecido.

Será precisamente en el cálculo de la *intensidad semántica* donde centremos nuestra atención en la siguiente sección.

5.2.1 Cálculo de la *Intensidad Semántica*: Infiriendo *asociaciones de tipo camino semántico*

En este análisis partimos del *perfil-ontología* P_X definido en la Secc. 5.2 , que contiene M instancias (x_i, con $0 \leq i < M$). Supongamos además que estamos analizando una secuencia concreta iniciada en una de las x_i, por ejemplo, la representada en la Fig. 2 para x_i = x_0.

$$\mathbf{ps_k^{x_0}.PSNodesSequence} = \{x_0, n_0, n_1, \ldots, n_N\}$$

El objetivo es obtener una expresión que permita calcular la *intensidad semántica* referida a cada nodo n_j ($0 \leq j \leq N$), en la secuencia iniciada en x_0 descrita arriba. Una vez obtenido el valor de $\lambda_{Sem}(n_j)$, será necesario aplicar la condición reflejada en la Fig. 2, obteniendo finalmente las *secuencias objetivo* que pretendemos inferir.

En el cómputo de la *intensidad semántica* se utilizará el conocimiento explícito representado en la ontología de TV. Para contemplar esta información en nuestro enfoque nos apoyamos en algunos de los conceptos que ya introdujimos en el documento Doc3 (versión 1.1) —dedicado a describir un enfoque de filtrado colaborativo en el que se comparaban las preferencias de diferentes usuarios —, junto a otros que presentamos ahora. Antes de continuar se recomienda una consulta al Apéndice A, donde se recogen las definiciones de todos ellos.

Para obtener una expresión para la *intensidad semántica* asociada a una instancia n_j —que recordemos será mayor cuanto más relacionada esté esta instancia con las preferencias del usuario— será necesario evaluar la influencia de diferentes factores. La contribución de cada uno de ellos se analiza a continuación.

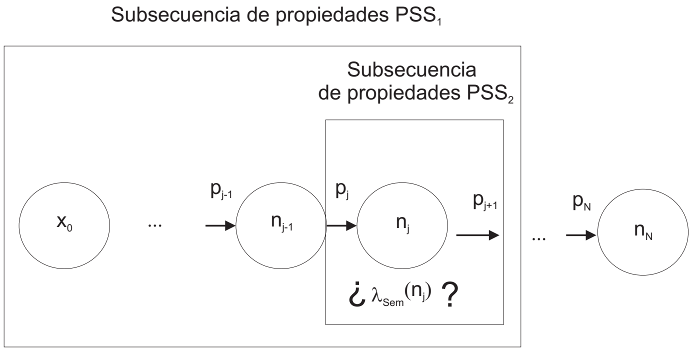

Figura 3. Secuencia de propiedades iniciada en la instancia x_0

I Diferencial semántico entre las instancias x_i del *perfil-ontología* y los nodos de la subsecuencia comprendida entre x_i y el nodo n_j analizado

En nuestro enfoque calcularemos el *diferencial semántico* entre la instancia n_j analizada y las instancias x_i definidas en el *perfil-ontología* del usuario ($\varDelta_{Sem}(x_i, n_j)$).

A la vista de los valores tomados por este parámetro (Ec. 12 en Apéndice A), es obvio que cuanto menores sean los $\varDelta_{Sem}(x_i, n_j)$ registrados, mayor será la relación entre la instancia n_j y las preferencias del espectador. Por esta razón, la *intensidad* de una instancia será inversamente proporcional a los valores del diferencial semántico entre ésta y las x_i del perfil, tal como indica la contribución señalada en la Ec. 2.

$$\lambda_{\mathbf{Sem}}(\mathbf{n_j}) \;\; \alpha \;\; \frac{1}{\sum_i \varDelta_{Sem}(x_i, n_j) + 1} \tag{2}$$

II Interés asociado en el *perfil-ontología* del usuario a la instancia x_0

El nivel de interés del usuario (representado como DOI, degree of interest) relativo a una instancia de su perfil, en este caso x_0, debe ser tenida en cuenta durante el cálculo de la intensidad de aquellas otras instancias relacionadas con ella, ya que es un parámetro relevante si tenemos en cuenta nuestro objetivo de personalización.

$$\lambda_{\mathbf{Sem}}(\mathbf{n_j}) \;\; \alpha \;\; DOI(x_0) \tag{3}$$

III Longitud de la subsecuencia iniciada en x_0 y finalizada en n_{j+1}

Para el n_j analizado, será necesario explorar la longitud de la subsecuencia $PSS_1 = [p_0, \ldots, p_{j+1}]$ representada en la Fig. 3. En este sentido cabe destacar que cuanto mayor sea ésta, menor será la intensidad semántica de n_j ($\lambda_{Sem}(n_j)$), tal como indica la Ec. 4. Esto se debe a la relación entre n_j y la instancia del perfil del usuario es más indirecta y menos significativa, por involucrar a diferentes instancias intermedias.

$$\lambda_{\mathbf{Sem}}(\mathbf{n_j}) \;\; \alpha \;\; \frac{\gamma_{Sem}}{\mathit{Length}(\rho - pathAssociated(x_0, n_j)) + 1} \tag{4}$$

En la Ec. 4 hemos utilizado el producto del umbral por el inverso de la longitud, de forma que si el umbral es muy bajo, esta contribución pesará muy poco en la expresión final de la intensidad. Esto respalda nuestra teoría de que si el umbral es reducido, nuestro sistema infiere gran cantidad de relaciones indirectas, poco significativas, de ahí que no se tenga en cuenta la longitud de las asociaciones descubiertas.

Por el contrario, si el umbral es muy alto, tal como dijimos antes, se infieren relaciones muy directas, de ahí que en este caso el producto del umbral y la longitud del camino sea una contribución fuerte en el cálculo de la intensidad. Por esta razón, se primarán aquellos caminos cuya longitud fuerse menor, y por tanto más significativos para el proceso de recomendación.

IV Procesos inferenciales aplicados sobre subsecuencias

A partir de la *secuencia original* analizada, definimos la subsecuencia $PSS_2 = [p_j, p_{j+1}]$, representada en la Fig. 3.

En este sentido, es conveniente destacar que nuestra propuesta incluye, en el cálculo de $\lambda_{Sem}(n_j)$, la búsqueda de relaciones de *nexo semántico* entre las propiedades p_j y p_{j+1} y el resto de propiedades *Object* almacenadas en el perfil del usuario. Esta decisión nos parece acertada, ya que permite considerar procesos inferenciales sencillos entre las propiedades consideradas.

Si asumimos la notación presentada en la Secc. 5.2, $ps_k^{x_i}$ identifica la secuencia k-ésima asociada a la instancia x_i en la ontología OWL. Conviene recordar que las secuencias del perfil consisten en una propiedad *Datatype* (longitud uno) o bien una propiedad *Object* y una propiedad *Datatype* (longitud dos). Así, el perfil almacena de toda la secuencia $ps_k^{x_i}$ representada en la ontología OWL, únicamente las propiedades *Datatype* asociadas a x_i en la ontología, y también las propiedades *Object* que establecen relaciones entre x_i y otras instancias, junto con las *Datatype* asociadas a estas últimas. Nuestro proceso de inferencia en este caso se centra en las propiedades *Object* del perfil, de ahí que empleemos únicamente las secuencias de longitud dos. Por tanto, nuestras propiedades de interés serán las $ps_k^{x_i}[0]$, tal que *Length* *(*ps_k*)* = 2.

Tal como indica la Ec. 5, para cada una de las M instancias x_i definidas en el perfil P_X, será necesario recorrer las R_i secuencias iniciadas en cada una de ellas en la ontología, comprobando si la primera propiedad de cada secuencia (la que está almacenada en el perfil para las secuencias de longitud dos), está relacionada mediante *nexo semántico* con p_j o con p_{j+1}.

$$\lambda_{\mathbf{Sem}}(\mathbf{n_j}) \;\; \alpha \;\; \sum_{i=0}^{M-1} \sum_{k=0}^{R_{i-1}} V_{ik} \tag{5}$$

donde

$$V_{ik} = \begin{cases} 1 \;\; \text{si} \quad p_j \leftrightarrow_\rho ps_k^{x_i}[0] \quad \text{o bien} \quad p_{j+1} \leftrightarrow_\rho ps_k^{x_i}[0] \\ 0 \;\; \text{si no existe dicho nexo semántico} \end{cases}$$

Siguiendo con el análisis de las propiedades de la subsecuencia PSS_1, centremos ahora nuestra atención en aquellas que sean *transitivas*, viendo cómo contribuyen al cómputo de $\lambda_{Sem}(n_j)$.

Empecemos recordando los preceptos del *principio de transitividad*. Sean *a*, *b* y *c* instancias de la ontología y *p* una propiedad transitiva. Si se cumple que $a[p]b$ y $b[p]c$, también se verifica —en virtud de la transitividad— que $a[p]c$.

Para describir esta nueva contribución, retomemos la subsecuencia $PSS_1 = [p_0, \ldots, p_{j-2}, p_{j-1}, p_j, p_{j+1}]$ de la Fig. 3.

Asumamos que existe en PSS_1 uno (o varios) pares de propiedades (p_{z-1}, p_z) cumpliendo que éstas son iguales y transitivas, con $1 \leq z \leq j$. En este caso, teniendo en cuenta la Fig. 3 y la restricción que acabamos de imponer, es obvio que se cumple $n_{z-2}[p_z]n_{z-1}$ y $n_{z-1}[p_z]n_z$, y por tanto también se verifica

$n_{z-2}[p_z]n_z$. Cabe destacar en este sentido que si z = 1, el nodo n_{z-2} sería la propia instancia x_0 del perfil, mientras que el papel de la propiedad p_{z-2} lo jugaría p_0.

De esta forma, es posible construir una nueva subsecuencia pss^* de longitud menor a la de PSS_1, en la que no estarían incluidos aquellos nodos que sea posible suprimir tras aplicar el principio de transitividad. Por ejemplo, si se da $x_0[p_0]n_0$ y $n_0[p_0]n_1$, entonces suprimiríamos el nodo n_1 en la subsecuencia inferida.

Tal como indica la Ec. 6, la expresión de la intensidad será inversamente proporcional a la longitud de la subsecuencia inferida pss^*. De esta forma, cuantas más propiedades transitivas haya en el camino desde x_0 hasta n_j (esto es, cuantos más pares (p_{z-1}, p_z) con las restricciones impuestas antes podamos definir), menor será dicha longitud y, por tanto mayor la intensidad semántica de esta instancia, dado que su relación con x_0 es muy directa.

$$\lambda_{\mathbf{Sem}}(\mathbf{n_j}) \quad \alpha \quad \frac{1}{\mathit{Length}(pss^*)} \tag{6}$$

V Centralidad Semántica

Por último, en esta sección presentamos un concepto que influirá considerablemente en el cálculo de la *intensidad* de la instancia n_j, la *centralidad semántica*.

En nuestro enfoque, la instancia n_j será más céntrica, cuanto mayor sea el número de propiedades asociadas a ella, y cuanto más relevantes sean las instancias con las que está relacionada mediante dichas propiedades. Esta relevancia se acentúa cuanto más directa sea la relación entre estas instancias y las preferencias declaradas en el perfil del usuario. Así, la centralidad semántica y la intensidad serán directamente proporcionales.

Para su cuantificación, emplearemos dos de los conceptos definidos en el Apéndice A: el *grado* y la *intermediación*.

Habida cuenta que el grado de una instancia es el número de propiedades *Object* relativas a ésta (apareciendo la instancia como dominio), podemos decir que cuanto más elevado sea el valor de este parámetro, más relaciones podrán ser descubiertas. Esto último se debe a que esta instancia está muy relacionada con otras definidas en la ontología OWL, de lo que se desprende que serán muchos los procesos inferenciales favorecidos en este escenario.

Como ya dijimos al comienzo de esta sección, nuestra propuesta no sólo valorará que n_j esté relacionada con muchas instancias, sino también la relevancia de estas últimas. Conviene recordar que dichas instancias están definidas en el conjunto de grado de n_j (Apéndice A).

Teniendo en cuenta que la intermediación (*betweeness*) mide con qué frecuencia aparece un nodo en el camino que conecta dos instancias declaradas en el perfil del usuario, parece claro que este parámetro es una métrica adecuada a la hora de cuantificar semejante relevancia. A la vista de lo explicado, es obvio que la intermediación y la intensidad semántica de una instancia son directamente proporcionales, ya que cuanto mayor sea la primera, más directa será su relación con las preferencias del espectador, incleméntandose así la segunda.

Por otra parte, el interés de los usuarios es también un elemento clave en este análisis. De forma intuitiva si una instancia n_j conecta entre sí aquellos nodos que se corresponden con las preferencias positivas del espectador, su intensidad será muy significativa. Por el contrario, si dicha instancia permite relacionar entre sí las preferencias negativas, su intensidad deberá ser decrementada. Esto último es debido a que nuestro proceso de personalización no debería dar cabida a un nodo que conecta contenidos que no interesan al usuario.

De esta manera, para obtener una expresión analítica de la centralidad de n_j —que será usada en el cálculo de su intensidad— consideraremos dos contribuciones: una en relación a las preferencias positivas y otra referida a aquellos contenidos que no agrandan al espectador ($Centralidad_P$ (n_j) y $Centralidad_N$ (n_j), respectivamente). Para calcular la componente $Centralidad_P$ procedemos de la siguiente forma:

- En primer lugar, se calcula la proporción de secuencias que parten de n_j y que conectan dos instancias del perfil. Así, siendo estas últimas x_i, x_l, será necesario recorrer las M instancias disponibles ($0 \leq i, l < M, i \neq l$), dividiendo la intermediación $b_{n_j}(x_i, x_l)$, entre el grado de n_j (degree (n_j)).
- Con el objeto de considerar la relevancia de las instancias relacionadas con n_j en el proceso de inferencia, a la componente anterior se le añade la misma proporción que hemos descrito, calculada en este ocasión a partir de las instancias del conjunto de grado de n_j.
- Finalmente, se multiplica este resultado por el promedio de los intereses asociados a las instancias x_i, x_l consideradas, tal como refleja la Ec. 7.

$$\mathit{Centralidad}_{\mathbf{P}}(\mathbf{n_j}) = \sum_{i,l=0;i\neq l}^{M-1} \left(\frac{b_{n_j}(x_i,x_l)}{degree(n_j)} + \sum_{s=0}^{degree(n_j)-1} \frac{b_{w_s}(x_i,x_l)}{degree(w_s)} \right) \cdot \frac{DOI(x_i)+DOI(x_l)}{2} \tag{7}$$

donde w_s es la instancia s-ésima en el conjunto de grado de n_j, esto es $w_s = D(n_j)[s]$.

Análogamente, el cómputo de la componente $Centralidad_N$ (n_j) se lleva a cabo sin más que sustituir las preferencias positivas por las negativas (instancias y_m, $0 \leq m < G$) en el procedimiento descrito antes (Ec. 8).

Finalmente, la combinación de ambas componentes se recoge en la Ec. 9.

$$\mathit{Centralidad}_{\mathbf{N}}(\mathbf{n_j}) = \sum_{m,n=0;m\neq n}^{G-1} \left(\frac{b_{n_j}(y_m,y_n)}{degree(n_j)} + \sum_{s=0}^{degree(n_j)-1} \frac{b_{w_s}(y_m,y_n)}{degree(w_s)} \right) \cdot \frac{DOI(y_m)+DOI(y_n)}{2} \tag{8}$$

$$\lambda_{\mathbf{Sem}}(\mathbf{n_j}) \quad \alpha \quad (\ \mathit{Centralidad}_P(n_j) + \mathit{Centralidad}_N(n_j)\ ) \tag{9}$$

En este punto, bastaría con reunir las ecuaciones Ec. 2 a Ec. 9 para obtener la expresión final que permita el cómputo de la *intensidad semántica* de una instancia concreta, n_j en este caso.

Una vez calculada la intensidad de las diferentes instancias que forman parte de las *secuencias originales* analizadas, y comparado su valor con el umbral semántico establecido, disponemos ya en este punto del conjunto definitivo de *secuencias objetivo*, a partir de las cuales se infieren todas las relaciones descritas en la Secc. 4.2.

Dado que el proceso inmediato de extracción de los *caminos semánticos* a partir de dichas secuencias ya ha sido mencionado en esta sección, en la siguiente describimos el procedimiento llevado a cabo para descubrir el resto de asociaciones incluidas en nuestro enfoque.

5.2.2 Inferencia de las restantes asociaciones semánticas

Para inferir las relaciones semánticas contempladas en esta sección, será necesario explorar de dos en dos, las *secuencias objetivo* que han sido extraídas mediante el proceso propuesto a tal efecto. Para describir nuestro enfoque recogemos a continuación dos de éstas:

$$PS_1 = [P_0, P_1, \ldots P_N]$$
$$PS_2 = [Q_0, Q_1, \ldots Q_M]$$

Supongamos dos instancias de las secuencias indicadas arriba, ps_1 y ps_2, iniciadas en las instancias x_i definidas en el *perfil-ontología* del usuario. Las distintas instancias que integran ambas instancias se muestran a continuación:

$$ps_1.PSNodesSequence = [x_i, n_0, \ldots, n_N]$$
$$ps_2.PSNodesSequence = [x_j, m_0, \ldots, m_P]$$

A partir de esta información, describamos el procedimiento llevado a cabo para cada tipo de relación:

- **Asociaciones mediante unión** (*ρ-joinAssociated*): A la vista de las definición dada en la Secc. 4.2, es fácil ver que basta con examinar las dos secuencias estudiadas ps_1 y ps_2, buscando instancias en común entre los conjuntos $ps_1.PSNodesSequence$ y $ps_2.PSNodesSequence$. En caso de que estas existan, habremos descubierto la nueva relación *ρ - joinAssociated* (x_i, x_j).

- **Asociaciones mediante padre común** (*ρ-cpAssociated*): Conviene recordar que esta asociación permite relacionar dos secuencias de propiedades, siempre que éstas estén integradas por instancias de una misma clase o bien, por instancias de clases subsumidas por un *techo*. Además de dicho techo, otro de los parámetros a considerar en esta relación era la longitud de las secuencias analizadas, que no debía superar un umbral establecido.
  Considerando esta información, es fácil ver que el proceso de inferencia de *asociaciones mediante padre común* incluye dos pasos. En primer lugar, es necesario seleccionar únicamente aquellas secuencias de longitud menor o igual al umbral al que nos referíamos antes. A continuación, se verifica que las clases a la que pertenecen los nodos de los conjuntos $ps_1.PSNodesSequence$ y $ps_2.PSNodesSequence$ sean iguales o estén subsumidas por el techo establecido.

- **Asociaciones semánticas indirectas mediante isomorfismo** (*ρ-IndirectIsoAssociated*): En este caso, es necesario recorrer las propiedades de las secuencias ps_1 y ps_2, comprobando que todas ellas estén relacionadas mediante *nexo semántico*, tal como comentábamos en la Secc. 4.2. En este caso, la nueva relación inferida sería *ρ-IndirectIsoAssociated* (x_i, x_j).
  En este sentido, conviene hacer una aclaración. Dado que las longitudes de las secuencias ps_1 y ps_2 pueden ser diferentes, será necesario comprobar la condición de *nexo semántico* hasta alcanzar el final de la más corta de ambas. Esto es:

  $$\rho\text{-}IndirectIsoAssociated\,(x_i, x_j) \quad \textbf{\textit{si y sólo si}} \quad P_k \leftrightarrow_\rho Q_k, \forall\, 0 \leq k < \min\,(\mathrm{N,P})$$

- **Asociaciones semánticas mixtas**: A la vista de la definición dada para las tres relaciones mixtas que contemplamos en la Secc. 4.2, es fácil ver que su inferencia es inmediata sin más que aplicar el principio de transitividad sobre las relaciones consideradas en cada caso.
  Así, por ejemplo, en relación a las *asociaciones semánticas mixtas tipo I* (*ρ-join-pathAssociated*), basta con explorar los caminos semánticos y las relaciones mediante unión inferidas anteriormente, verificando si es posible combinarlas en virtud de la transitividad.
  El procedimiento para las *asociaciones semánticas mixtas tipo II* (*ρ-cp-pathAssociated*) *tipo III* (*ρ-iso-pathAssociated*) es completamente análogo.

5.2.3 Extracción de los programas recomendados

Llegados a este punto, nuestro recomendador —más concretamente los agentes semánticos descritos en la arquitectura de AVATAR [2]— deben extraer de las asociaciones semánticas inferidas, la lista final de programas que sugerirán al usuario.

Conviene recordar que el perfil almacena junto a un programa concreto, las propiedades *Object* que permite relacionar ésta con otras instancias de la base de conocimiento OWL (secuencias de longitud dos). Dado que estas últimas están asociadas a programas que han interesado al usuario (en el caso de preferencias positivas), también es conveniente almacenarlas en el perfil. De esta forma, entre las instancias declaradas en el perfil de un usuario, podemos diferenciar tanto categorías de programas (comedias, documentales, informativos, etc.), como instancias de clases de otra naturaleza (actores, género, directores, etc.). En cualquier caso, es importante destacar que toda instancia del perfil que no se corresponda con un contenido de TV (por ejemplo, una instancia de la clase *Actors*), siempre está relacionada con un programa concreto definido en el mismo, ya que de lo contrario no figuraría entre las preferencias del usuario.

Por ejemplo, si en un perfil aparecen una película de acción y su director, consideramos que todos aquellos programas que estén relacionados con este director, lo están también con la mencionada película. Por esta razón, siempre es posible identificar en cada asociación inferida, una relación con los programas que interesan al usuario, ya sea porque dicha asociación involucra directamente tales contenidos, o por el contrario, porque participan en ella instancias relacionadas con éstos mediante propiedades *Object*. Esto significa que en cada asociación se define una relación entre un programa del perfil del usuario y otras instancias de la ontología OWL. Para obtener la recomendación final, basta con extraer de entre las citadas instancias, aquellas que se correspondan con programas concretos de TV.

En este sentido conviene hacer una aclaración. En el supuesto que la inferencia de conocimiento haya sido nula, es decir, que entre las relaciones descubiertas no se detecte ningún contenido de TV (por ejemplo porque hayamos fijado un umbral demasiado elevado y no hayamos encontrado instancias que integren las *secuencias objetivo*), se incluirán en la recomendación final instancias de las clases definidas en el perfil del usuario. Esto se debe a que dichas clases serían el único conocimiento disponible para el sistema a la hora de sugerir programas concretos. Este tipo de sugerencias —limitadas en lo que a capacidades de inferencia se refiere— serían las ofrecidas mediante los enfoques existentes en el estado del arte.

Por otra parte, el tipo de relación establecida entre los programas del perfil del usuario, y los descubiertos en nuestro proceso de inferencia, también debe ser incorporado en la ordenación de los contenidos sugeridos. Así, teniendo en cuenta que las relaciones más directas son las más significativas de cara a la personalización, en nuestra propuesta consideramos que los programas relacionados con uno del perfil mediante asociaciones directas, son mostrados antes que otros que hayan sido inferidos desde relaciones más indirectas. Si recordamos las asociaciones semánticas que hemos manejado durante toda la exposición, es fácil ver que —ordenadas de más a menos directas— éstas son: *caminos semánticos*, *asociaciones mediante unión*, *asociación mediante padre común*, *asociación indirecta mediante isomorfismo* y las *asociaciones mixtas I, II* y *III*.

Una vez identificados los contenidos específicos que se ofrecerán al espectador —extraídos de las asociaciones inferidas— será necesario ordenarlos. Para este propósito, recurrimos a los índices almacenados en su *perfil-ontología*. Así, aquellos contenidos seleccionados que estén relacionados con programas que tengan un índice de relevancia (*relevance* en [3]) elevado en el perfil, ocuparán los primeros lugares de la clasificación elaborada por el sistema. De los tres índices que figuran en el perfil para cada instancia y para cada clase, esto es, *DOI*, *confianza* y *relevancia*, hemos usado en la ordenación el último, debido a que combina la contribución de los dos anteriores. Basta recordar que DOI mide el nivel de interés asociado a una instancia (o clase), mientras que la *confianza* permite cuantificar el éxito o fracaso obtenido por el sistema en las recomendaciones hechas en el pasado.

6. Algunas conclusiones

En este documento hemos presentado AVATAR, un sistema recomendador de contenidos de TV, mejorado gracias a la sinergia de dos técnicas de personalización conocidas en la literatura —métodos basados en conocimiento y filtrado colaborativo— en las que hemos incorporado capacidades de inferencia y razonamiento semántico.

Concretamente nos hemos centrado en la primera de las técnicas mencionadas, ya que la segunda de ellas ha sido abordada en otro documento (Doc2 disponible en la intranet de AVATAR). La inclusión de estrategias basadas en conocimiento proporciona importantes ventajas a la herramienta de personalización propuesta.

Por una parte, este tipo de métodos impiden un problema muy extendido en el contexto de los recomendadores descritos en la literatura, el *ramp-up*. Éste consiste en que el sistema debe disponer de un número suficientemente alto de clasificaciones proporcionadas por los usuarios, antes de que la calidad de las sugerencias presentadas se considere aceptable. Por el contrario, los sistemas basados en conocimiento superan tal limitación, ya que se basan en el empleo de funciones de mapeo entre los intereses de los usuarios y los contenidos a recomendar a los mismos. De esta forma, aún cuando el sistema disponga de poca información sobre los intereses de un usuario, será suficiente con aplicar el mapeo correspondiente sobre, para obtener una recomendación atractiva para éste.

Otra de las ventajas reseñables de los sistemas basados en conocimiento, frente al empleo de estrategias centradas en las clasificaciones hechas por otros usuarios —como el filtrado colaborativo o los métodos basados en contenido— es que un contenido puede ser mostrado a un usuario, aún cuando ningún otro espectador lo haya clasificado en su perfil. Es decir, en el filtrado colaborativo no se examinan todos los contenidos disponibles en la base de datos usada por el recomendador, sino únicamente aquellos que estén definidos en los perfiles de los usuarios que se usan en la comparación de preferencias. Por el contrario, en los sistemas de personalización basados en conocimiento, basta con que el mapeo utilizado detecte una correspondencia entre los intereses del usuario y la descripción de un contenido concreto, para que éste sea sugerido, sin necesidad de que otro usuario haya declarado dicho contenido entre sus preferencias.

En otro orden de cosas, la principal contribución de este trabajo —referida al razonamiento semántico y la inferencia de relaciones no explícitas en una base de conocimiento— permite mejorar considerablemente los enfoques de recomendación existentes en la literatura, en los que no se contemplan capacidades de razonamiento como las incluidas en nuestro enfoque. De esta forma, mediante un sondeo de diferentes valores del *umbral semántico* debería ser posible detectar los diferentes tipos de recomendaciones ofrecidas por nuestra herramienta, en función de lo intensa que sea la inferencia llevada a cabo[3].

Por otra parte, la capacidad de relacionar las preferencias de los usuarios con otros contenidos de TV mediante inferencia semántica, contribuye a superar una limitación identificada en estrategias tradicionalmente utilizadas en las herramientas de personalización, referida al excesivo parecido existente entre los contenidos conocidos por los usuarios y los sugeridos mediante dichos métodos. Nuestra propuesta permite diversificar las recomendaciones ofrecidas al espectador, preservando en todo momento la personalización, debido a que la información empleada en este proceso se extrae de su *perfil-ontología*, permanentemente actualizado.

Finalmente, es interesante destacar que aún cuando nuestro enfoque se aplica en AVATAR es lo suficientemente flexible como para ser extendido a aplicaciones de la Web Semántica, donde sea útil personalizar la información presentada a un usuario, con arreglo a sus intereses. Si bien la inclusión de sistemas de personalización en la Web Semántica constituye una contribución sin duda relevante, no hay muchos enfoques en esta dirección. De hecho, el único recomendador en este contexto que conocemos se describe en [13] y no contempla los procesos inferenciales incluidos en nuestra herramienta. En esta dirección, basta con examinar la gran cantidad de información manejada en la Web y las relaciones que a partir de la misma

[3] Recordar que valores muy bajos del umbral se corresponden con una inferencia prácticamente nula, basada en las propiedades de la ontología. Por el contrario, valores mayores permitirían considerar relaciones implícitas, inferidas desde dichas propiedades. Esto último se debe a que en este supuesto, las instancias deberían acumular una intensidad semántica suficiente para superar dicho umbral. Esto se traduce en una inferencia de conocimiento más intensa, ya que sería necesario que las diferentes contribuciones de la intensidad identificadas en la Secc. 5.2.1, tomasen valores elevados, es decir, sería necesario inferir relaciones entre las propiedades y clases del perfil y las declaradas en la ontología de TV, detectar centralidad semántica...

se definen, para comprobar la utilidad de un enfoque como el propuesto, que permite detectar desdes estas últimas, asociaciones complejas y significativas para el usuario.

Apéndice

A. Conceptos de Interés

A.1. Nivel jerárquico

Identifica el lugar ocupado por cada una de las clases, y por tanto por sus instancias, en la jerarquía de clases definida en la ontología del sistema. El nivel jerárquico asignado a una clase es menor cuanto más alejada esté del elemento raíz de la ontología, al que le corresponde un nivel 0.

A.2. Distancia semántica (*semantic closeness*)

La distancia semántica entre las instancias $prog_j$ y $prog_k$, que representaremos como $dist$ ($prog_j$,$prog_k$), se define como la diferencia entre los niveles jerárquicos ocupados por las clases C_j y C_k, a las que dichas instancias pertenecen en la base de conocimiento del sistema. La expresión de este índice se muestra en la Ec. 10.

$$\mathbf{dist}(\mathbf{prog_j}, \mathbf{prog_k}) = |Nivel(prog_j) - Nivel(prog_k)| \tag{10}$$

A.3. Diferencial semántico

Este parámetro permite incluir en nuestro análisis procesos inferenciales sencillos, fácilmente deducibles desde la estructura de la base de conocimiento utilizada. Para analizar esa estructura emplearemos los dos conceptos presentados a continuación.

- Por una parte, diremos que las instancias $prog_j$ y $prog_k$ están **conectadas semánticamente** —representado como $prog_j \bowtie_{Sem} prog_k$— cuando se cumple alguna de las condiciones recogidas a continuación:
  - $prog_j$ y $prog_k$ están relacionadas mediante el principio de *subsumption* ($C_j \sqsubseteq C_k$, o bien $C_k \sqsubseteq C_j$),
  - ambas instancias son hermanas "directas" ($\exists C \in Cl_O$, tal que C_j, $C_k \sqsubseteq_D C$).
- Por otra parte, definimos el **pc-diámetro** de las instancias $prog_j$ y $prog_k$, como la distancia semántica que hay entre cada una de ellas y el **p**adre **c**omún más cercano, tal como se muestra en la Ec. 11.

$$\mathbf{pc - diam}(\mathbf{prog_j}, \mathbf{prog_k}) = dist(prog_j, pc) + dist(prog_k, pc), \text{ donde } \; pc \in Cl_O \;\text{ y }\; C_j, C_k \sqsubseteq pc \tag{11}$$

Una vez introducidos ambos conceptos, podemos presentar los valores asociados al *diferencial semántico* asociado a las instancias $prog_j$ y $prog_k$ ($\Delta_{Sem}(prog_j, prog_k)$), los cuales serán menores cuanto más relacionadas estén ambas instancias.

$$\Delta_{Sem}(prog_j, prog_k) = \begin{cases} 0 & \text{si} \quad C_j = C_k \\ 1 & \text{si} \quad prog_j \bowtie_{Sem} prog_k \\ \textit{pc-Diam}(prog_j, prog_k) & \text{en otro caso} \end{cases} \tag{12}$$

A.4. Grado o valencia de una instancia

Se define el grado de $prog_j$ —representado como *degree* ($prog_j$)— como el número de propiedades *Object* en las que la instancias $prog_j$ figura como dominio en la ontología de conocimiento OWL, esto es, el número de secuencias de propiedades relativas a $prog_j$.

A.5. Conjunto de grado de una instancia

El conjunto de grado de $prog_j$ —representado como *D* ($prog_j$)— incluye las instancias relacionadas con $prog_j$ mediante propiedades *Object*. Es decir, cada componente de *D* ($prog_j$), será la instancia que figura como rango en la primera propiedad de cada secuencia asociada a $prog_j$, tal como se indica en la Ec. 13. Es claro que la cardinalidad de este conjunto coincide con el grado de la instancia $prog_j$.

$$D(\mathbf{prog_j})[\mathbf{i}] = ps_i^{prog_j}.PS_i^{P_j}\ [0][1], \quad 0 \leq i < \mathit{degree}(prog_j) \tag{13}$$

A.6. Intermediación (*betweeness*)

En teoría de grafos la intermediación es una medida de centralidad de un determinado nodo en el conjunto del grafo. Se define como el número de veces que hay que atravesar un determinado nodo para poder conectar otros dos en la red.

De forma análoga, la intermediación de una instancia $prog_j$ en relación a las instancias a y b (en adelante $b_{prog_j}(a, b)$) es definida en nuestra propuesta como la proporción de secuencias de propiedades cuyo inicio es a y cuyo fin es b, en las que está incluido el nodo $prog_j$. Analíticamente podemos expresar tal situación mediante la Ec. 14.

$$\mathbf{b_{prog_j}(a, b)} = \frac{\mathit{seq}(a, prog_j, b)}{\mathit{seq}(a, b)} \tag{14}$$

donde es claro que *seq* (a,b) representa el número de secuencias desde a a b, y *seq* (a,$prog_j$,b), el subconjunto de las mismas en las que la instancia $prog_j$ está incluida.

Referencias

1. Anyanwu K and Sheth A. ρ-Queries: Enabling Querying for Semantic Associations on the Semantic Web. In *Proc. of the Twelfth International World Wide Web Conference (WWW-03)*, pages 690–699. ACM Press, 1-58113-680-3/03/0005, 2003.
2. Blanco Y., Pazos J., Gil A., Ramos M., Barragáns B., López M., Fernández A. and Díaz R. In *Proc. of 5th International Conference on Web Information Systems Engineering (WISE-04)*, volume 3306 of *Lectures Notes in Computer Science*, pages 415–421. Springer-Verlag, 2004.
3. Blanco Y., Pazos J. J., Gil A., Ramos M., López M. and Barragáns B. AVATAR: Modeling Users by Dynamic Ontologies in a TV Recommender System based on Semantic Reasoning. In *Proc. of the 3rd European Conference on Interactive Television: User Centred ITV Systems, Programmes and Applications (EuroITV-05)*, 2005.
4. Burke R. Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction*, 12(4):331–370, 2002.
5. Davies J. et al. *Towards the Semantic Web: Ontology-driven Knowledge Management*. John Wiley & Sons, 2003.
6. Halascheck C. *A Flexible Approach for Ranking Complex Relationships on the Semantic Web*. PhD thesis, Department of Computer Science. University of Georgia, 2004.
7. Halaschek C., Aleman-Meza B., Budak Arpinar I., Ramakrishnan C. and Sheth A. A Flexible Approach for Analyzing and Ranking Complex Relationships on the Semantic Web. In *Technical Report, LSDIS Lab, Computer Science Department (University of Georgia)*, 2004.
8. McGuinness D. and Harmelen F. OWL Web Ontology Language Overview. W3C Recommendation. 2004.

9. Miller L., Seaborne A. and Reggiori A. Three Implementations of SquishQL, a Simple RDF Query Language. In *Proc. of the first International Semantic Web Conference (ISWC-02)*, 2002.
10. Seaborne A. RDQL: A Data Oriented Query Language for RDF Models. In *http://www.hpl.hp.com/semweb/rdql-grammar.html*, 2001.
11. Sheth A., Thacker S. and Patel S. Semantic Content Management for Enterprises and the Web. *IEEE Internet Computing*, pages 80–87, 2002.
12. Staab S. and Studer R., editor. *Handbook on Ontologies*. International Handbooks on Information Systems. Springer, 2004.
13. Ziegler C.N., Schmidt-Thieme L. and Lausen G. Exploting Semantic Product Descriptions for Recommender Systems. In *Proc. of the 2nd ACM SIGIR Semantic Web and Information Retrieval Workshop (SWIR-04)*, 2004.