Multimed Tools Appl (2009) 41:407–436
DOI 10.1007/s11042-008-0239-7

Receiver-side semantic reasoning for digital TV personalization in the absence of return channels

Martín López-Nores · Yolanda Blanco-Fernández · José J. Pazos-Arias · Jorge García-Duque · Manuel Ramos-Cabrer · Alberto Gil-Solla · Rebeca P. Díaz-Redondo · Ana Fernández-Vilas

Published online: 21 October 2008

Abstract Experience has proved that interactive applications delivered through Digital TV must provide personalized information to the viewers in order to be perceived as a valuable service. Due to the limited computational power of DTV receivers (either domestic set-top boxes or mobile devices), most of the existing systems have opted to place the personalization engines in dedicated servers, assuming that a return channel is always available for bidirectional communication. However, in a domain where most of the information is transmitted through broadcast, there are still many cases of intermittent, sporadic or null access to a return channel. In such situations, it is impossible for the servers to learn who is watching TV at the moment, and so the personalization features become unavailable. To solve this problem without sacrificing much personalization quality, this paper introduces solutions to run a downsized semantic reasoning process in the DTV receivers, supported by a

M. López-Nores (✉) · Y. Blanco-Fernández · J. J. Pazos-Arias · J. García-Duque · M. Ramos-Cabrer · A. Gil-Solla · R. P. Díaz-Redondo · A. Fernández-Vilas
Department of Telematics Engineering, University of Vigo,
ETSE Telecomunicación, Campus Universitario s/n, Vigo 36310, Spain
e-mail: mlnores@det.uvigo.es

Y. Blanco-Fernández
e-mail: yolanda@det.uvigo.es

J. J. Pazos-Arias
e-mail: jose@det.uvigo.es

J. García-Duque
e-mail: jgd@det.uvigo.es

M. Ramos-Cabrer
e-mail: mramos@det.uvigo.es

A. Gil-Solla
e-mail: agil@det.uvigo.es

R. P. Díaz-Redondo
e-mail: rebeca@det.uvigo.es

A. Fernández-Vilas
e-mail: avilas@det.uvigo.es

pre-selection of material driven by audience stereotypes in the head-end. Evaluation results are presented to prove the feasibility of this approach, and also to assess the quality it achieves in comparison with previous ones.

Keywords Personalization · Digital TV · Semantic reasoning · Ontologies · Stereotypes

1 Introduction

Recent advances in Digital TV technologies for domestic set-top boxes and mobile receivers have made it possible to deliver interactive applications along with the TV programs. For some years, this possibility was exploited to provide simple services that displayed information in a *one-size-fits-all* manner (i.e. all the viewers would be always faced with the same information). This approach became a failure for providing very little added value to the TV viewers [34], to the point that the interactive offerings usually went unnoticed to a vast majority. In response to that, the research community has been working on mechanisms to tailor the information delivered to the preferences and needs of individual viewers, with applications in personalized programming guides [7, 16, 29, 51], personalized learning [14, 41, 53], personalized publicity and commerce [13, 22, 30, 32], personalized health care [11, 27, 33] and many others. Hereafter, we shall generically talk about *items* to refer to any pieces of information or services that may be linked to the TV programs (other TV programs, learning courses, advertisements, health care advice, hypermedia, etc).

In the aforementioned studies, the common approach has been to lodge the personalization engines in dedicated servers, powerful enough to apply complex reasoning processes over huge amounts of data. This works fine as long as the DTV receiver of any viewer can communicate with the personalization servers to learn what are the most suitable items to offer to him/her. During periods of disconnection, however, the personalization features become unavailable. The problem we face in this paper is precisely that, with many of the new DTV technologies, the TV programs are delivered through broadcast networks—a downstream flow of information into the receivers—and there are frequent cases of intermittent, sporadic or null access to a return channel for bidirectional communication. This happens, for instance, with the DVB-S and DVB-T standards for terrestrial and satellite in-home television,[1] as well as most of the solutions targeted to mobile receivers: DMB,[2] DVB-H,[3] MediaFLO,[4] etc. Typically, the reasons for being offline have to do with the cost of the connection (xDSL, 2G, 3G, etc) or lack of infrastructure [50]. Whichever the case, the problems of unavailability contribute to hampering the uptake of interactive DTV services by the viewers.

To ensure fully-available personalization even in the absence of return channels, it is necessary to broadcast more items than will be consumed by any individual,

[1] http://www.dvb.org/technology/standards.

[2] http://eng.t-dmb.org.

[3] http://www.dvb-h.org.

[4] http://www.qualcomm.com/mediaflo.

and then decide which ones to recommend to each viewer by local processing in the receiver he/she is using. There already exist a few approaches to this idea in literature (see [20, 35, 43, 56, 57]), but they rely on *syntactic matching* techniques that achieve low personalization quality. This limitation may be solved by borrowing techniques from the area of the *Semantic Web* [1]. However, contrary to what happens with Internet-enabled personal computers, there is no trace of *semantic reasoning* running on DTV receivers, because their memory and computing power are very limited. And, importantly, it is not a matter of time that we end up having more powerful terminals, because the market strives to keep these devices as consumer products affordable for the vast majority of the population [17].

In this paper, we introduce an architectural approach to build personalization engines that achieve full availability and good personalization quality by splitting the burden of a semantic reasoning process among dedicated servers (on the side of the DTV head-end) and DTV receivers. In brief, the idea is to achieve personalization in two steps, with the servers doing a broad pre-selection of items driven by audience clusters or stereotypes, and the receivers doing the final selection taking into account the details of each individual viewer. Prior to describing our proposal, Section 2 provides an overview of research in personalization, to clearly point out the weaknesses of the previous approaches to receiver-side reasoning in DTV. Then, we present our solution to those weaknesses in Section 3, along with a toy example for illustration purposes. In Section 4, we summarize the results of experiments carried out to assess the feasibility of the proposal over different DTV technologies, and also to evaluate the personalization quality achieved in comparison with previous systems. Finally, Section 5 provides a summary of conclusions and motivates our future work in this line of research.

2 Background in personalization

Personalization is achieved by matching the information in a user's profile against metadata descriptions of the items available. The first possibility explored to do so was *content-based filtering*, which consists in making recommendations by looking at contents that gained the user's interest in the past [3, 18, 57]. This strategy is easy to adopt, but bears a problem of *overspecialization*: the recommendations tend to be repetitive for considering that a user will always appreciate the same kind of stuff. To tackle this problem, the scientific community came up with *collaborative filtering*, which proceeds by evaluating not only the profile of the target user, but also those of users with similar interests (his/her *neighbors*) [31, 38, 46]. This approach can solve the lack of diversity in the recommendations, but faces problems like the *sparsity* when the number of items is high (which makes it hard to find users with similar evaluations for the same items) or the treatment given to users whose preferences are dissimilar to the majority (the *gray sheep*). Finally, there exist a number of *hybrid approaches* that attempt to neutralize the weaknesses and combine the strengths of content-based and collaborative filtering, e.g. recommending contents similar to the ones stored in the target user's profile, but considering two items similar if the users who show interest in the one tend to be interested in the other [5, 12, 48].

Regardless of the filtering strategy, the first recommender systems used *syntactic* matching techniques, which relate items by looking for common words in their attached metadata. Even though there exist plenty of different techniques—just to mention a few, we can cite *cosine similarity* [5] or *automatic classifiers* based on neural networks [24], decision trees [42], Bayesian networks [57] and association rules [6, 48],—they all miss a lot of knowledge during the personalization process, because they are unable to reason about the meaning of the metadata. This limitation is noticeable in the existing approaches to receiver-side reasoning in DTV, cited in Section 1. For instance, the personalized t-learning system of [20] could select courses for a given user by seeking specific topics appearing in his/her profile, but disregarding any interrelations with different ones (thus, it is not possible to link a course about "*electricity*" with another one about "*magnetism*", as the two words are dissimilar). By the way, a syntactic approach is also a source of overspecialization, because the recommendations so computed can only include items very similar to those the users already know.

To go one step beyond in personalization quality and diversity, research is now focused on applying techniques from the Semantic Web, because they enable reasoning processes that gain insight into the meaning of words (so that, for example, "*electricity*" and "*magnetism*" can be automatically recognized as two nearby topics within the broader area of "*physics*"). The key here lies within the use of *ontologies* to describe and interrelate items and their attributes by means of class hierarchies and properties [49]. Ontologies can be expressed in various different languages (OWL [36] being the most widely used nowadays) and queried in many different ways (e.g. considering hierarchical relationships as in RQL [26] or SquishQL [37], entering logical rules as in TRIPLE [47], or examining chains of properties as in SemDis [2]). The important point for this paper is that the ontologies can be enormous databases, totally unmanageable for limited devices such as DTV receivers. Accordingly, all applications of semantic reasoning in DTV personalization to date (see [4, 15, 40] for examples in different domains) have opted for server-based approaches following the design of Fig. 1, with ontologies and viewer profiles stored in dedicated servers and filtering algorithms running remotely, too.

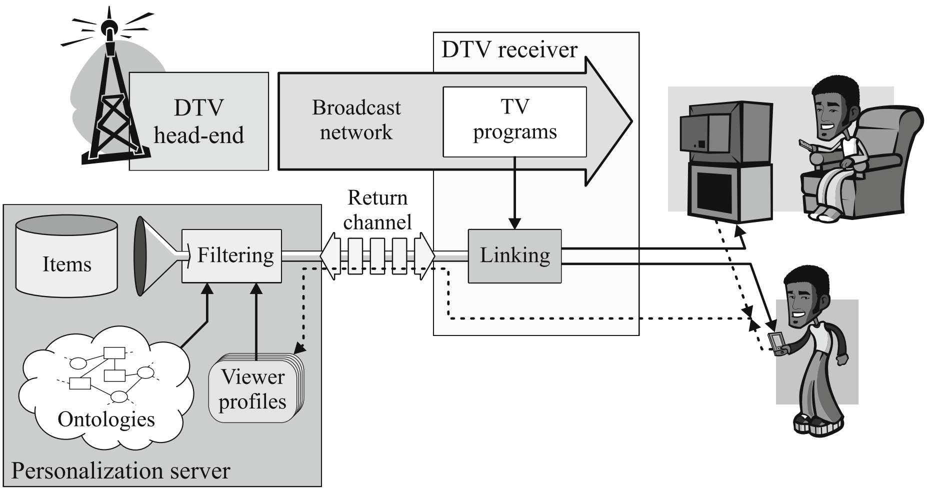


Fig. 1 The common design of server-based approaches to personalization in DTV

The reliance on permanently-connected return channels raises the problems of unavailability that we explained in Section 1—it follows from Fig. 1 that, during periods of disconnection, the viewers can only receive TV programs (no recommendations and no personalized items). In this paper, we solve those problems by introducing a generic approach to filtering in the DTV receivers, which only process a carefully-selected amount of information arriving through the broadcast networks. The key concept in our proposal is the use of *audience stereotypes*, previously explored for other purposes in DTV applications in [30, 54].

3 An architecture for receiver-side personalization in DTV

Figure 2 depicts our architecture for receiver-side personalization in DTV, which is an evolution of the server-based scheme we presented in [9]. In this case, since we are focusing on scenarios with only downstream communication from the DTV head-end, we have moved the viewer profiles and the filtering algorithms to the receivers. In contrast, for the reasons of scale explained above, the ontologies still have to be kept in dedicated servers. To bridge the gap between the two sides, we introduce server-side mechanisms for a twofold purpose:

– First, since it may not be possible to broadcast all the available items at the same time, we make a *pre-selection* to deliver only the ones which are potentially most

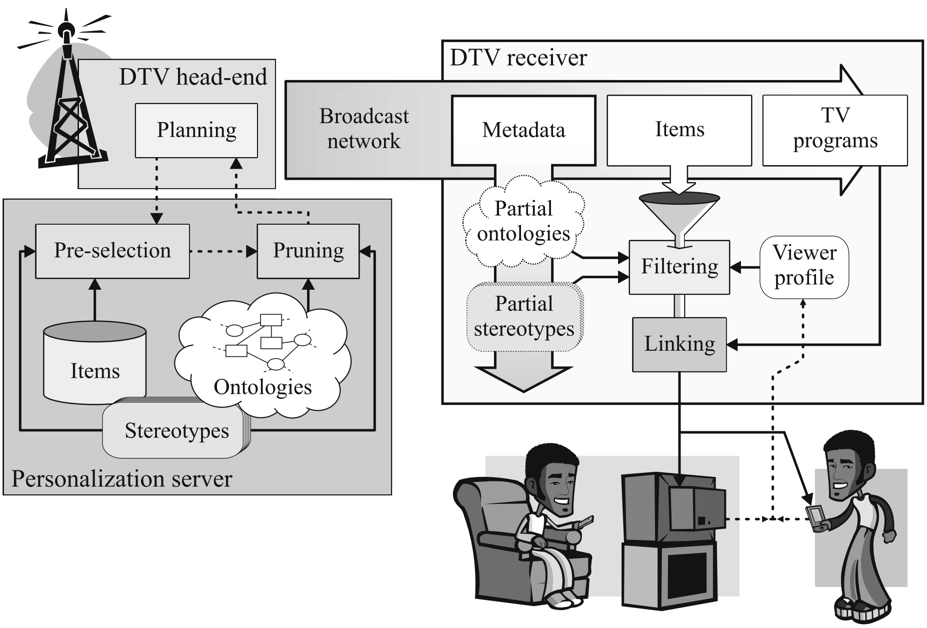


Fig. 2 The scheme of our architecture for receiver-side personalization

interesting for the audiences expected at any given moment. This is indeed a filtering process, though not driven by the profile of an individual viewer, but rather by a set of *stereotypes* that average the preferences and needs of different groups of viewers.
– Second, we have devised a *pruning* procedure to reduce the amount of information to be handled by the receivers. This procedure consists of cutting off metadata from the ontologies to leave only the most relevant concepts about the pre-selected items. As a result, we get *partial ontologies* of a manageable size for the receivers to work with, plus *partial stereotypes* to support the filtering.

Following the pre-selection and pruning processes, a *planning* module in the DTV head-end takes care to arrange items, partial ontologies and partial stereotypes in the broadcast emissions so as to facilitate access by the receivers and optimize some performance parameters. When those data have been loaded into the receivers, it is finally possible to run the filtering algorithms to decide what items will be offered to each individual viewer. All of these steps will be fully explained in the following subsections.

3.1 A few words about viewer profiles and stereotypes

In our approach, the viewer profiles consist of excerpts from the ontologies that contain the items a given individual has evaluated in the past, each one attached to a numerical index called DOI (*Degree of Interest*) that quantifies his/her liking of it.[5] The stereotypes have exactly the same structure, though with DOIs computed to reflect the average interest of the different items for the represented viewer groups.

DOIs take values in the range $[-1, 1]$, with -1 representing the greatest disliking and 1 representing the greatest acceptance. These values propagate through the hierarchy of classes and the attributes of the items as follows:

– The DOI of an attribute is taken as the average of the DOIs of the items it is linked to.
– Similarly, the DOIs of the most specific classes are computed as the average of the DOIs of the items classified under them. Upwards in the hierarchy, each class $\mathcal{C}$ contributes to the DOI of its immediate superclass with a value given by Eq. 1, where $\mathrm{sib}(\mathcal{C})$ is the number of sibling classes of $\mathcal{C}$.

$$\frac{\mathrm{DOI}(\mathcal{C})}{1 + \mathrm{sib}(\mathcal{C})} \tag{1}$$

[5] As explained for a server-based personalization engine in [10], the DOI for a given item can be explicitly entered by the viewer, or inferred from indirect measures such as the time he/she spends watching or executing it.

Example 1 To help understand the DOIs, consider the minimal ontology of Fig. 3, that classifies TV programs and educational courses as needed to deliver personalized t-learning services. This excerpt includes two TV programs ($\mathcal{P}_1$ and $\mathcal{P}_2$) and three learning courses ($\mathcal{L}_1$, $\mathcal{L}_2$ and $\mathcal{L}_3$), with attributes like topics and intended

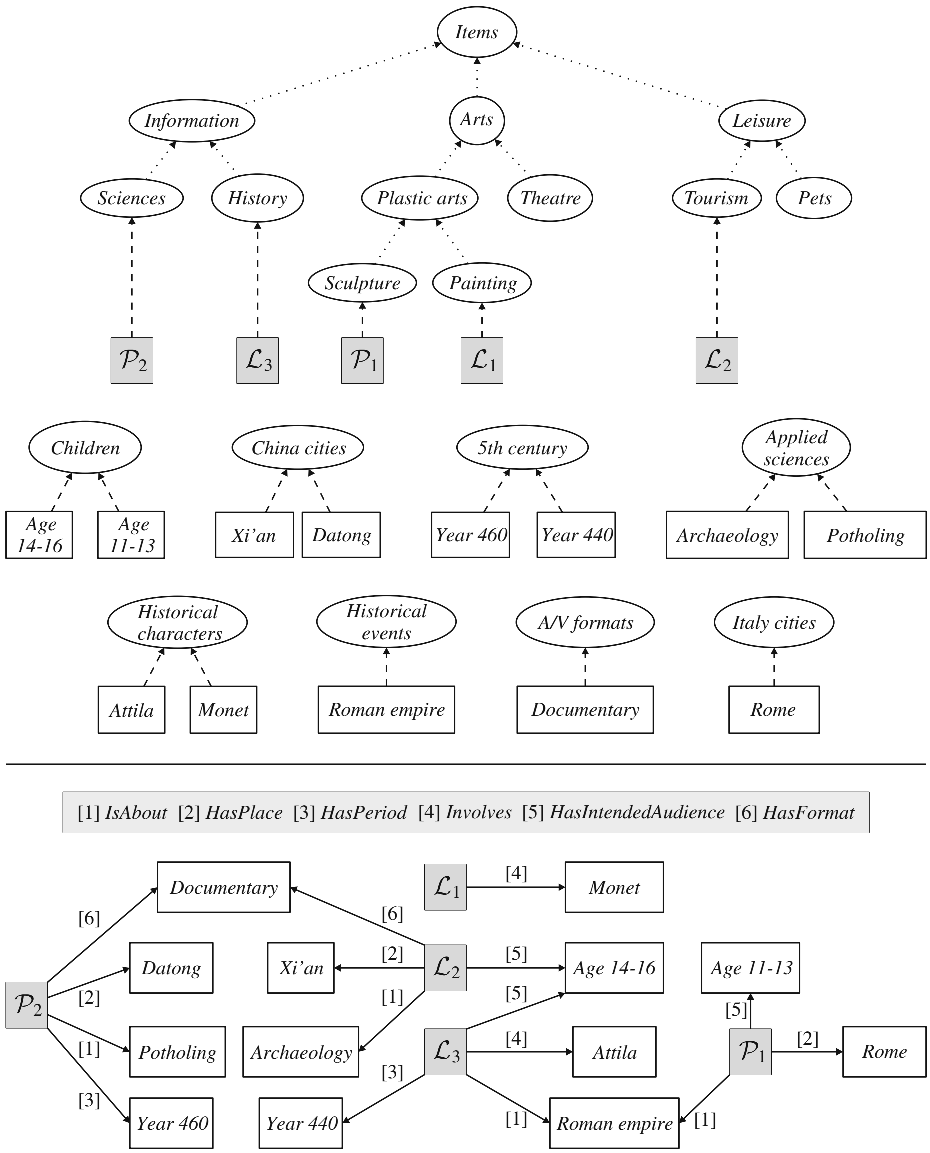


Fig. 3 Excerpt from an ontology of TV programs and learning courses

Table 1 The stereotype $\mathcal{S}_1$

Item	DOI	Attributes
Arts $\rightarrow$ *Plastic arts* $\rightarrow$ *Sculpture* $\rightarrow$ $\mathcal{P}_1$	1	*IsAbout* $\rightarrow$ *Roman empire* *HasPlace* $\rightarrow$ *Rome* *HasIntendedAudience* $\rightarrow$ *Age 11–13*
Information $\rightarrow$ *History* $\rightarrow$ $\mathcal{L}_3$	0.8	*IsAbout* $\rightarrow$ *Roman empire* *HasPeriod* $\rightarrow$ *Year 440* *Involves* $\rightarrow$ *Attila* *HasIntendedAudience* $\rightarrow$ *Age 14–16*

audiences indicated by the properties *isAbout* and *HasIntendedAudience*. According to this ontology, $\mathcal{P}_1$ is a program that explains Roman sculpture to children between 11 and 13 years old, while $\mathcal{P}_2$ is a science documentary about potholing in the Yungang grottoes, which are located in the Chinese city of Datong and date back to year 460 AD. As regards the learning resources, $\mathcal{L}_1$ is a painting course explaining Monet's style; $\mathcal{L}_2$ teaches Chinese archaeology to children between 14 and 16 years old, focusing on the terracotta army of Xi'an; and $\mathcal{L}_3$ is a history course also for children between 14 and 16 years old that describes the role of Attila the Hun in the fall of the Roman Empire, by year 440 AD.

Hereafter, we will assume there are two stereotypes in the system, $\mathcal{S}_1$ and $\mathcal{S}_2$, that give DOIs to items as represented in Tables 1 and 2, respectively. By assigning high DOIs to the TV program $\mathcal{P}_1$ and the learning course $\mathcal{L}_3$, the stereotype $\mathcal{S}_1$ roughly models an audience of young people who enjoy contents related to European history. On the other hand, the DOIs assigned to $\mathcal{P}_2$ and $\mathcal{L}_2$ make the stereotype $\mathcal{S}_2$ represent an audience of people who like documentaries and Asian culture.

Next, we compute the DOIs of the different attributes as per the stereotype $\mathcal{S}_1$.

– The attributes *Age 11–13* and *Rome* inherit directly the DOI of the TV program $\mathcal{P}_1$:

$$\mathrm{DOI}_{\mathcal{S}_1}(\mathit{Age\ 11\text{–}13}) = \mathrm{DOI}_{\mathcal{S}_1}(\mathit{Rome}) = 1$$

Table 2 The stereotype $\mathcal{S}_2$

Item	DOI	Attributes
Information $\rightarrow$ *Sciences* $\rightarrow$ $\mathcal{P}_2$	1	*IsAbout* $\rightarrow$ *Potholing* *HasPlace* $\rightarrow$ *Datong* *HasPeriod* $\rightarrow$ *Year 460* *HasFormat* $\rightarrow$ *Documentary*
Leisure $\rightarrow$ *Tourism* $\rightarrow$ $\mathcal{L}_2$	0.6	*IsAbout* $\rightarrow$ *Archaeology* *HasPlace* $\rightarrow$ *Xi'an* *HasFormat* $\rightarrow$ *Documentary*

- Analogously, attributes *Age 14–16*, *Attila* and *Year 440* inherit the DOI of the learning course $\mathcal{L}_3$:[6]

$$\mathrm{DOI}_{\mathcal{S}_1}(\textit{Age 14–16}) = \mathrm{DOI}_{\mathcal{S}_1}(\textit{Attila}) = \mathrm{DOI}_{\mathcal{S}_1}(\textit{Year 440}) = 0.8$$

- The attribute *Roman empire* is linked to both $\mathcal{P}_1$ and $\mathcal{L}_3$, so its DOI results from averaging those of the two items:

$$\mathrm{DOI}_{\mathcal{S}_1}(\textit{Roman empire}) = \frac{\mathrm{DOI}_{\mathcal{S}_1}(\mathcal{P}_1) + \mathrm{DOI}_{\mathcal{S}_1}(\mathcal{L}_3)}{2} = 0.9$$

- The rest of the attributes are not linked to either $\mathcal{P}_1$ or $\mathcal{L}_3$, so their DOI is implicitly taken as 0.
  In what concerns the class hierarchy, we get the following:
- As in stereotype $\mathcal{S}_1$ there is only one program of class *Sculpture*, it follows that

$$\mathrm{DOI}_{\mathcal{S}_1}(\textit{Sculpture}) = \mathrm{DOI}_{\mathcal{S}_1}(\mathcal{P}_1) = 1.$$

- Since *Sculpture* has one only sibling class (*Painting*), it contributes to the DOI of its superclass *Plastic arts* with a value $\frac{\mathrm{DOI}_{\mathcal{S}_1}(\textit{Sculpture})}{1+1} = 0.5$. The contribution of class *Painting* is 0 because there are no DOIs in $\mathcal{S}_1$ for items of that class. So,

$$\mathrm{DOI}_{\mathcal{S}_1}(\textit{Plastic arts}) = 0.5.$$

- Analogously, we compute the following DOIs (the remaining ones are implicitly taken as 0):

$$\mathrm{DOI}_{\mathcal{S}_1}(\textit{Arts}) = \frac{0.5}{1+1} = 0.25$$
$$\mathrm{DOI}_{\mathcal{S}_1}(\textit{History}) = \mathrm{DOI}_{\mathcal{S}_1}(\mathcal{L}_3) = 0.8$$
$$\mathrm{DOI}_{\mathcal{S}_1}(\textit{Information}) = \frac{\mathrm{DOI}_{\mathcal{S}_1}(\textit{History})}{1+1} = 0.4$$

3.2 Sorting out available items by stereotypes

Our approach to personalization starts out from the assumption that any TV program watched by a viewer is related to his/her interests at the moment (otherwise, he/she would not be watching it). Therefore, to pre-select the items that will be delivered through broadcast, we look for the items which are most related to the TV programs scheduled to be transmitted. This way, for example, when a nature documentary is broadcast, it will more likely go along with hypermedia about animals or climate change than with pop music compilations or do-it-yourself home improvements.

[6]The DOI of the attribute *Age 14-16* is not influenced by $\mathcal{P}_2$, because this program is not rated in $\mathcal{S}_1$.

According to these observations, the *pre-selection* module of our architecture (Fig. 2) is driven by a *semantic similarity* metrics that measures the strength of the relations that can be inferred—from the knowledge captured in the ontologies—between the TV programs and any items that could be linked to them. To this aim, we have adapted a metrics presented in [7], that considers not only the explicit relations defined by the hierarchy of classes, but also others which are hidden behind the attributes of the items. Therefore, we talk about two similarity criteria, that we refer to as *hierarchical similarity* and *inferential similarity*, respectively.

– The notion of hierarchical similarity has appeared in many previous approaches (see for example [19, 28, 44, 45]), and consists of valuing the relation between two nodes of the ontologies by the existence and specificity of a common ancestor in a hierarchy of classes. The expression we use in this regard is given in Eq. 2:

$$\mathrm{SemSim}_{Hie}(\mathcal{I}, \mathcal{P}) = \frac{\mathrm{depth}(\mathrm{LCA}(\mathcal{I}, \mathcal{P}))}{\max(\mathrm{depth}(\mathcal{I}), \mathrm{depth}(\mathcal{P}))} \tag{2}$$

According to this expression, the value of hierarchical similarity between an item $\mathcal{I}$ and a program $\mathcal{P}$ grows with the depth of their *lowest common ancestor* (LCA) and also with its proximity to $\mathcal{I}$ and $\mathcal{P}$ in the hierarchy. The depth of a node is given by the number of hierarchical links traversed to reach the node from the root of the hierarchy; thereby, the hierarchical similarity between two nodes is 0 if they do not have other common ancestor than the root class.

Example 2

- With the hierarchy of Fig. 3, the value of hierarchical similarity between courses $\mathcal{L}_2$ and $\mathcal{L}_3$ and program $\mathcal{P}_1$ is $\mathrm{SemSim}_{Hie}(\mathcal{L}_2, \mathcal{P}_1) = \mathrm{SemSim}_{Hie}(\mathcal{L}_3, \mathcal{P}_1) = 0$, because their only common ancestor is the root class *Items*.
- In contrast, the existence of a very specific LCA between $\mathcal{L}_1$ and $\mathcal{P}_1$ (*Plastic Arts*, of depth 3) yields a hierarchical similarity of $\mathrm{SemSim}_{Hie}(\mathcal{L}_1, \mathcal{P}_1) = \frac{3}{4}$.

– Following the ideas of [21, 38], the notion of inferential similarity consists of measuring similarity by looking at relationships between the semantic attributes of the items compared. In this regard, two items are considered similar if they share some attributes (*common attributes*), or if they have attributes belonging to the same class in some hierarchy (*sibling attributes*).

Example 3

- In the ontology of Fig. 3, it can be seen than $\mathcal{L}_3$ and $\mathcal{P}_1$ share the topic *Roman empire*, whereas $\mathcal{L}_2$ and $\mathcal{P}_2$ share the *Documentary* format.
- As examples of sibling attributes, we see that $\mathcal{L}_2$ and $\mathcal{P}_1$ have intended audiences classified under *Children* (the same happens with $\mathcal{L}_3$ and $\mathcal{P}_1$); similarly, $\mathcal{L}_2$ and $\mathcal{P}_2$ have locations classified as *China cities* and deal with *Applied sciences*, while $\mathcal{L}_3$ and $\mathcal{P}_2$ are both settled in years of the *5th century*.

To calculate the value of inferential similarity between an item $\mathcal{I}$ and a program $\mathcal{P}$, in addition to the existence of common or sibling attributes, we consider the level of interest of those attributes in the system stereotypes. The formula we use in this regard is that of Eq. 3, which yields the value of inferential similarity between $\mathcal{I}$ and $\mathcal{P}$ as per the stereotype $\mathcal{S}$:

$$\mathrm{SemSim}_{Inf}(\mathcal{I},\mathcal{P},\mathcal{S}) = \frac{1}{\mathrm{CSA}_{max}(\mathcal{I},\mathcal{P})} \sum_{k=1}^{\mathrm{CSA}(\mathcal{I},\mathcal{P})} \delta(a_k)\cdot \mathrm{DOI}_{\mathcal{S}}(a_k) \tag{3}$$

In this formula, $\mathrm{CSA}(\mathcal{I},\mathcal{P})$ is the number of common or sibling attributes between $\mathcal{I}$ and $\mathcal{P}$, a_k is the k-th of those attributes, $\mathrm{DOI}_{\mathcal{S}}(a_k)$ is the degree of interest of a_k in the stereotype $\mathcal{S}$, and $\mathrm{CSA}_{max}(\mathcal{I},\mathcal{P})$ is the maximum number of attributes that $\mathcal{I}$ and $\mathcal{P}$ could share (i.e. the minimum between the number of attributes of $\mathcal{I}$ and the number of attributes of $\mathcal{P}$). Finally, $\delta(a_k)$ is a factor that gives some more weight to common attributes (1) than sibling ones (0.85), because the former indicate a closer relationship.

Example 4

- Since there are no common or sibling attributes between the TV program $\mathcal{P}_1$ and the learning courses $\mathcal{L}_1$ and $\mathcal{L}_2$, the values of inferential similarity are 0 for all the stereotypes:

$$\mathrm{SemSim}_{Inf}(\mathcal{L}_1,\mathcal{P}_1,\mathcal{S}_1) = \mathrm{SemSim}_{Inf}(\mathcal{L}_1,\mathcal{P}_1,\mathcal{S}_2) = 0$$
$$\mathrm{SemSim}_{Inf}(\mathcal{L}_2,\mathcal{P}_1,\mathcal{S}_1) = \mathrm{SemSim}_{Inf}(\mathcal{L}_2,\mathcal{P}_1,\mathcal{S}_2) = 0$$

- As explained above, $\mathcal{L}_3$ and $\mathcal{P}_1$ have *Roman empire* as a common attribute and *Children* as a sibling one. Since they could share at most three attributes (because $\mathcal{P}_1$ has three properties), it follows that:

$$\mathrm{SemSim}_{Inf}(\mathcal{L}_3,\mathcal{P}_1,\mathcal{S}_1) = \frac{1\cdot \mathrm{DOI}_{\mathcal{S}_1}(\textit{Roman empire}) + 0.85\cdot \mathrm{DOI}_{\mathcal{S}_1}(\textit{Age 14-16})}{3} =$$
$$= \frac{0.9+0.85\cdot 0.8}{3} = 0.527$$
$$\mathrm{SemSim}_{Inf}(\mathcal{L}_3,\mathcal{P}_1,\mathcal{S}_2) = \frac{1\cdot \mathrm{DOI}_{\mathcal{S}_2}(\textit{Roman empire}) + 0.85\cdot \mathrm{DOI}_{\mathcal{S}_2}(\textit{Age 14-16})}{3} =$$
$$= \frac{0+0.85\cdot 0.6}{3} = 0.17$$

The two notions of semantic similarity are finally combined by means of a factor $\alpha \in [0, 1]$, as shown in Eq. 4:

$$\mathrm{SemSim}(\mathcal{I},\mathcal{P},\mathcal{S}) = (1-\alpha)\cdot \mathrm{SemSim}_{Hie}(\mathcal{I},\mathcal{P}) + \alpha\cdot \mathrm{SemSim}_{Inf}(\mathcal{I},\mathcal{P},\mathcal{S}) \tag{4}$$

We use this formula to decide what items will be broadcast along with each TV program. To this aim, we compute the values of semantic similarity between all the items and the program with regard to all the stereotypes. Then, we sort out the maximum values computed for each item, from highest to lowest. As we shall explain in Section 3.4, the closer an item is to the beginning of the ordered list, the greater the possibility that it will be broadcast. Likewise, the entries of the stereotypes are annotated with the maximum values of semantic similarity computed for the corresponding items, to ensure that the partial stereotypes will include the entries corresponding to the items available through broadcast.

Example 5 If we were to transmit program $\mathcal{P}_1$, a factor $\alpha = 0.6$ would yield the following values of semantic similarity:

$$\begin{aligned}
\text{SemSim}(\mathcal{L}_1, \mathcal{P}_1, \mathcal{S}_1) &= 0.4 \cdot \frac{3}{4} + 0.6 \cdot 0 = 0.3 \\
\text{SemSim}(\mathcal{L}_2, \mathcal{P}_1, \mathcal{S}_1) &= 0.4 \cdot 0 + 0.6 \cdot 0 = 0 \\
\text{SemSim}(\mathcal{L}_3, \mathcal{P}_1, \mathcal{S}_1) &= 0.4 \cdot 0 + 0.6 \cdot 0.527 = 0.3162 \\
\text{SemSim}(\mathcal{L}_1, \mathcal{P}_1, \mathcal{S}_2) &= 0.4 \cdot \frac{3}{4} + 0.6 \cdot 0 = 0.3 \\
\text{SemSim}(\mathcal{L}_2, \mathcal{P}_1, \mathcal{S}_2) &= 0.4 \cdot 0 + 0.6 \cdot 0 = 0 \\
\text{SemSim}(\mathcal{L}_3, \mathcal{P}_1, \mathcal{S}_2) &= 0.4 \cdot 0 + 0.6 \cdot 0.17 = 0.102
\end{aligned}$$

From to these results, we find that the learning course $\mathcal{L}_3$ is the most relevant item to deliver along with program $\mathcal{P}_1$ (following the preferences expressed by stereotype $\mathcal{S}_1$). The next item is $\mathcal{L}_1$ (equally relevant to both stereotypes), and $\mathcal{L}_2$ goes the last.

3.3 Identifying the most relevant metadata in the ontologies

The goal of the *pruning* module of our architecture (Fig. 2) is to cut off metadata from the ontologies, to prevent DTV receivers from dealing with an overwhelming amount of information. Again, what we do is select the information that is semantically most related to the TV programs transmitted at any time. The procedure to measure relevance is different for classes and attributes:

- On the one hand, to ensure that the receivers will be able to classify the broadcast items, the classes are assigned a relevance value equal to the maximum of the semantic similarities computed for the items to which it is ancestor.

  Example 6 With the values of semantic similarity computed above with regard to the TV program $\mathcal{P}_1$, $\mathcal{L}_3$ propagates the value 0.3162 upwards to the classes *History*, *Information* and *Items*, while $\mathcal{L}_2$ propagates 0.3 to *Painting*, *Plastic arts* and *Arts*. Since there are no other values of semantic similarity different from 0, the relevance of the rest of the classes with regard to $\mathcal{P}_1$ is also 0.

- On the other hand, the relevance of an attribute a with regard to a program $\mathcal{P}$ is computed as the average of the maximum values of semantic similarity computed for the items that have a as an attribute. Let $NI_{\to a}$ be the number of such items, and $\mathcal{I}_i$ the i-th of them. Also, let $N_\mathcal{S}$ be the number of stereotypes in the system, with $\mathcal{S}_j$ the j-th one. Then, the relevance of an attribute is given by Eq. 5:

$$\text{Rel}(a, \mathcal{P}) = \frac{1}{NI_{\to a}} \cdot \sum_{i=1}^{NI_{\to a}} \max_{1 \le j \le N_\mathcal{S}} \left(\text{SemSim}(\mathcal{I}_i, \mathcal{P}, \mathcal{S}_j) \right) \tag{5}$$

  The relevance values computed with this formula propagate upwards in the hierarchies of attributes, so each class is given a relevance equal to the maximum among its descendants.

Example 7

– In Fig. 3, for instance, the attributes *Attila*, *Roman empire* and *Year 440* are linked only to the learning course $\mathcal{L}_3$, whose semantic similarity with regard to program $\mathcal{P}_1$ is maximum with stereotype $\mathcal{S}_1$. Therefore, we get the following relevance values for these attributes:

$$\text{Rel}(\textit{Attila}, \mathcal{P}_1) = \text{SemSim}(\mathcal{L}_3, \mathcal{P}_1, \mathcal{S}_1) = 0.3162$$
$$\text{Rel}(\textit{Roman empire}, \mathcal{P}_1) = \text{SemSim}(\mathcal{L}_3, \mathcal{P}_1, \mathcal{S}_1) = 0.3162$$
$$\text{Rel}(\textit{Year 440}, \mathcal{P}_1) = \text{SemSim}(\mathcal{L}_3, \mathcal{P}_1, \mathcal{S}_1) = 0.3162$$

These values propagate to the attribute classes *Historical character*, *Historical event* and *5th century*, respectively.

– Attribute *Age 14–16* is linked to two items, $\mathcal{L}_2$ and $\mathcal{L}_3$, so its relevance with regard to $\mathcal{P}_1$ is computed as:

$$\begin{aligned}\text{Rel}(\textit{Age 14-16}, \mathcal{P}_1) &= \frac{\max_{1\le j\le 2}\big(\text{SemSim}(\mathcal{L}_2, \mathcal{P}_1, \mathcal{S}_j)\big) + \max_{1\le j\le 2}\big(\text{SemSim}(\mathcal{L}_3, \mathcal{P}_1, \mathcal{S}_j)\big)}{2} \\ &= \frac{0 + 0.3162}{2} = 0.1581\end{aligned}$$

This value propagates to the attribute class *Children*.

Once we have calculated relevance values for all classes and attributes, we can sort them out in decreasing order, so that the ones which are closer to the beginning of the list are more likely to be included in the partial ontologies delivered to the DTV receivers. The decision of how many nodes to include is up to the *planning* module of the DTV head-end, which is explained in the next section.

3.4 The actual pre-selection and pruning

As explained in [39], broadcast emissions in DTV are arranged in data structures called *transport streams*, that may multiplex live audiovisual contents (i.e. TV programs), application data, signaling information and various other types of traffic. The point of interest for our architecture is that the application data (including the items, partial ontologies and partial stereotypes that will be handled by the DTV receivers) are mounted on *carousels* that enable the view of a filesystem by transmitting their contents repeatedly and periodically, just like the pages of Teletext. Knowing this, once we have sorted out the items and the metadata with regard to the TV programs that will be broadcast, the *planning* module of our architecture (Fig. 2) decides on what material will be sent to the receivers by considering two limiting conditions:

– On the one hand, there is a question of loading times due to the fact that the information mounted on a carousel may be accessed with noticeable *latencies*, which grow directly with the amount of information transmitted.[7]

[7] For example, a carousel containing 4 MBytes of data, transmitted at a rate of 256 Kbps, takes 128 s to complete a cycle; so, it may take more than 2 min to load a particular piece of information.

– On the other hand, there is a question of computational cost related to the size of the partial ontologies that can be handled by the DTV receivers to produce timely personalized recommendations.

The most suitable values for the size of the carousels and the partial ontologies depend on multiple parameters, like the bandwidth available to carousels in the broadcast networks, the type of receivers considered, the duration of the TV programs, etc. The *planning* module takes those factors into account to produce two thresholds, β_1 and β_2, that set the minimum value of semantic similarity for an item to the transmitted, and the minimum relevance of a class or an attribute to be included in a partial ontology, respectively. The stereotypes are pruned by removing any entries related to discarded items, attributes and classes.

Example 8 In the running example, when it is time to transmit the TV program $\mathcal{P}_1$, we assume for simplicity that the thresholds provided by the *planning* module are $\beta_1 = \beta_2 = 0$.

– The value $\beta_1 = 0$ leads to pre-selecting the items $\mathcal{L}_1$ and $\mathcal{L}_3$ to be included in the broadcast emissions; $\mathcal{L}_2$ is discarded because the maximum value of semantic similarity computed for it with regard to $\mathcal{P}_1$ is 0.
– On its part, the value $\beta_2 = 0$ turns the ontology of Fig. 3 into the partial one of Fig. 4, which does not include any classes or attributes that are not relevant with regard to $\mathcal{P}_1$.
– Finally, the value $\beta_1 = 0$ does not affect the stereotype $\mathcal{S}_1$ (Table 1), because all of its entries relate to items that have relevance values greater than 0. In contrast, the stereotype $\mathcal{S}_2$ (Table 2) is emptied out, because it refers to items and attributes that are irrelevant with regard to $\mathcal{P}_1$.[8] As a result, only $\mathcal{S}_1$ is inserted in the broadcast emissions to support the filtering in the receivers.

It is worth noting that the *planning* module introduces the nodes of the partial ontologies in the carousels sorted by decreasing relevance. Thus, when a receiver is not powerful enough to cope with all the metadata arriving on the broadcast networks, at least we ensure that its filtering mechanisms will work with the most relevant information. This feature makes the architecture adaptable to the capabilities of a wide range of receivers.

3.5 Filtering for individual viewers

As explained in Section 2, there are two main strategies to identify relevant items according to a given viewer's profile: content-based filtering, to recommend items similar to others that the viewer liked in the past, and collaborative filtering, to recommend items that have interested other viewers with similar profiles (the so-

[8] In a real scenario, the disappearance of a stereotype would mean that the group of viewers that it aims to represent should be a residual audience for the TV program in question.

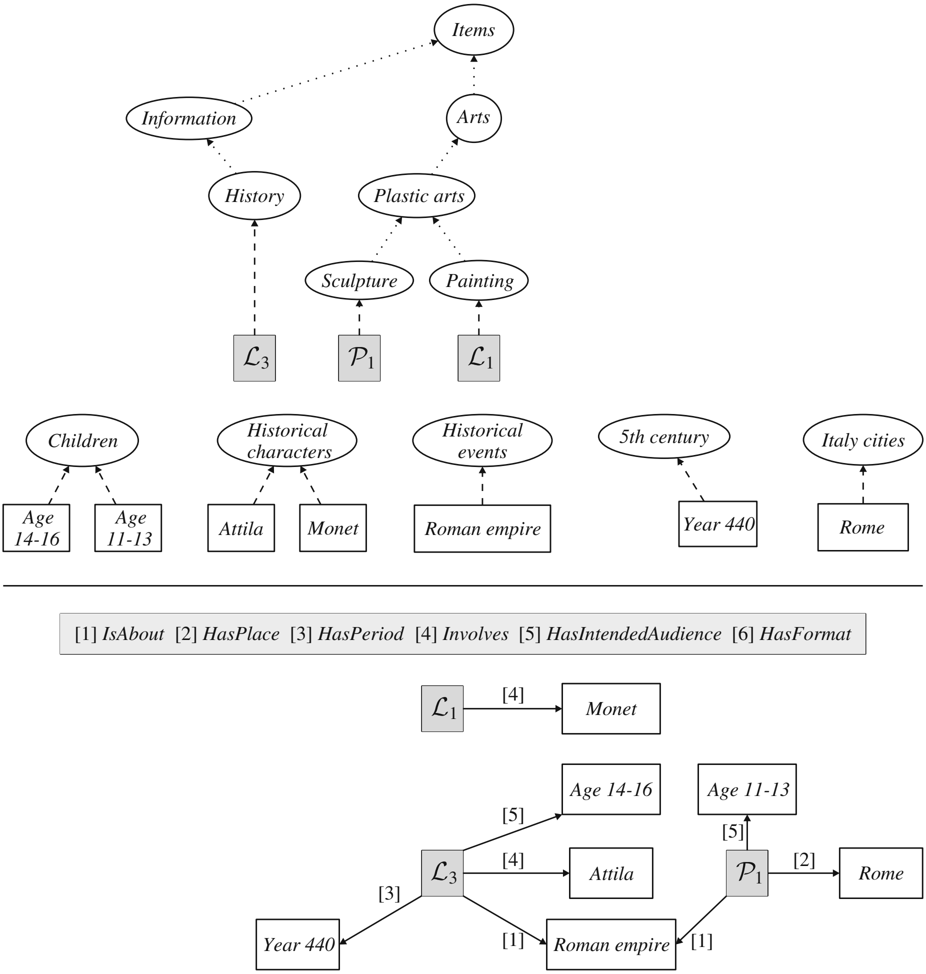


Fig. 4 The ontology of Fig. 3 after pruning out nodes with zero relevance with regard to $\mathcal{P}_1$

called neighbors). The *filtering* module of our architecture (Fig. 2) incorporates algorithms we had presented in [7, 8] to support both strategies. However, since we are now focusing on scenarios with only downstream communication from servers to receivers, it is important to note that the collaborative approach cannot do with the profiles of real viewers, but rather with the partial stereotypes delivered through broadcast. We keep this *pseudo-collaborative* approach because it helps overcome the problem of overspecialization that is typical of content-based strategies.

In brief, our two filtering strategies proceed as follows:

– The content-based strategy consists of averaging the levels of semantic similarity between every item available through broadcast and the items stored in the profile of the target viewer, weighed by their respective DOIs. This is expressed in Eq. 6, which returns the level of *matching* between an item $\mathcal{I}$ and a viewer profile $\mathcal{V}$ ($N_{\mathcal{V}}$ is the number of items rated in $\mathcal{V}$, and $\mathcal{I}_j$ refers to the j-th of those items):

$$\text{Match}(\mathcal{I}, \mathcal{V}) = \frac{1}{N_{\mathcal{V}}} \sum_{j=1}^{N_{\mathcal{V}}} \text{SemSim}(\mathcal{I}, \mathcal{I}_j, \mathcal{V}) \cdot \text{DOI}_{\mathcal{V}}(\mathcal{I}_j) \tag{6}$$

Example 9 In the running example, assume that we are computing recommendations for a viewer whose profile $\mathcal{V}$ (shown in Table 3) indicates that he/she has greatly enjoyed the TV program $\mathcal{P}_x$, which deals with the Roman empire, and the learning course $\mathcal{L}_y$, which explains the sculpture of the 5th century to children between 11 and 13 years old.
The filtering in the viewer's receiver would consist of making computations for each one of the items available through broadcast (the learning courses $\mathcal{L}_1$ and $\mathcal{L}_3$ from previous examples) with regard to the profile $\mathcal{V}$, relying on the knowledge captured in the partial ontology of Fig. 4. To begin with, the content-based strategy would compute the level of matching between $\mathcal{L}_1$ and $\mathcal{V}$ as follows:

$$\begin{aligned}
\text{Match}(\mathcal{L}_1, \mathcal{V}) &= \frac{\text{SemSim}(\mathcal{L}_1, \mathcal{P}_x, \mathcal{V}) \cdot \text{DOI}_{\mathcal{V}}(\mathcal{P}_x) + \text{SemSim}(\mathcal{L}_1, \mathcal{L}_y, \mathcal{V}) \cdot \text{DOI}_{\mathcal{V}}(\mathcal{L}_y)}{2} = \\
&= \frac{\left(0.4 \cdot \text{SemSim}_{Hie}(\mathcal{L}_1, \mathcal{P}_x) + 0.6 \cdot \text{SemSim}_{Inf}(\mathcal{L}_1, \mathcal{P}_x, \mathcal{V})\right) \cdot \text{DOI}_{\mathcal{V}}(\mathcal{P}_x)}{2} + \\
&\quad + \frac{\left(0.4 \cdot \text{SemSim}_{Hie}(\mathcal{L}_1, \mathcal{L}_y) + 0.6 \cdot \text{SemSim}_{Inf}(\mathcal{L}_1, \mathcal{L}_y, \mathcal{V})\right) \cdot \text{DOI}_{\mathcal{V}}(\mathcal{L}_y)}{2} = \\
&= \frac{\left(0.4 \cdot 0 + 0.6 \cdot 0\right) \cdot 0.9}{2} + \frac{\left(0.4 \cdot \frac{2}{3} + 0.6 \cdot 0\right) \cdot 1}{2} = 0.1333
\end{aligned}$$

Analogously, the receiver would obtain $\text{Match}(\mathcal{L}_3, \mathcal{V}) = 0.5183$, indicating that $\mathcal{L}_3$ matches the preferences captured in the profile $\mathcal{V}$ nearly four times as much as $\mathcal{L}_1$. Therefore, the content-based strategy would greatly prefer recommending $\mathcal{L}_3$ to the viewer.

– The pseudo-collaborative strategy starts out by delimiting the neighborhood of the target viewer, which in our case is the same as identifying the partial stereotypes in which he/she fits best. To this aim, we first create a *rating vector* containing the DOIs of the item classes most appealing or most unappealing to the viewer (identified by DOIs close to 1 and -1, respectively). Next, we look for the partial stereotypes that contain DOIs for at least 70% of those classes, and create their respective rating vectors. Finally, we compute the Pearson-r

Table 3 The profile of a sample viewer, $\mathcal{V}$

Item	DOI	Attributes
Information → *History* → $\mathcal{P}_x$	0.9	*IsAbout* → *Roman empire*
Arts → *Plastic arts* → *Sculpture* → $\mathcal{L}_y$	1	*HasPeriod* → *Year 440*
		HasIntendedAudience → *Age 11–13*

correlation between the rating vector of each partial stereotype $\mathcal{S}$ ($v_{\mathcal{S}}$) and the vector corresponding to the viewer ($v_{\mathcal{V}}$) by Eq. 7, where N denotes the length of both vectors:

$$\mathrm{corr}(\mathcal{S},\mathcal{V}) = \frac{\sum_{r=1}^{N} v_{\mathcal{S}}[r] \cdot v_{\mathcal{V}}[r]}{\sqrt{\sum_{r=1}^{N}(v_{\mathcal{S}}[r])^2 \cdot \sum_{r=1}^{N}(v_{\mathcal{V}}[r])^2}} \tag{7}$$

The viewer's neighborhood is formed by the M partial stereotypes that yield correlation values greater than a given threshold γ. Once the neighbors have been identified, we predict the level of interest of the viewer in an item $\mathcal{I}$ using Eq. 8, where $\mathcal{S}_k$ denotes the k-th stereotype neighbor and $\delta(\mathcal{S}_k)$ is a factor whose value depends on whether $\mathcal{I}$ appears rated in $\mathcal{S}_k$ or not (if it does, we use the corresponding DOI, else we resort to the level of matching between the item and the stereotype as per Eq. 6):

$$\mathrm{Pred}(\mathcal{I},\mathcal{V}) = \frac{1}{M}\sum_{k=1}^{M} \delta(\mathcal{S}_k) \cdot \mathrm{corr}(\mathcal{S}_k,\mathcal{V}) \tag{8}$$

$$\delta(\mathcal{S}_k) = \begin{cases} \mathrm{DOI}_{\mathcal{S}_k}(\mathcal{I}) & \text{if } \mathcal{I} \text{ appears rated in } \mathcal{S}_k \\ \mathrm{Match}(\mathcal{I},\mathcal{S}_k) & \text{otherwise} \end{cases}$$

Intuitively, the interest value predicted for $\mathcal{I}$ is greater when this item is very appealing to the selected neighbors and these are strongly correlated with $\mathcal{V}$.

Example 10 Applying the pseudo-collaborative strategy in the same settings of Example 9, the receiver would start out by computing the correlation between the preferences captured in $\mathcal{V}$ and those defined by the partial stereotypes delivered through broadcast (in this case, only $\mathcal{S}_1$ from Table 1). Assuming that we build the rating vector of $\mathcal{V}$ to contain only the classes that have DOIs greater than 0.6 or lower than −0.6, we would get the following vectors to correlate:

$$v_{\mathcal{V}} = \begin{pmatrix} \mathrm{DOI}_{\mathcal{V}}(\textit{Sculpture}) \\ \mathrm{DOI}_{\mathcal{V}}(\textit{History}) \\ \mathrm{DOI}_{\mathcal{V}}(\textit{Information}) \end{pmatrix} = \begin{pmatrix} 1 \\ 0.9 \\ 0.9 \end{pmatrix}$$

$$v_{\mathcal{S}_1} = \begin{pmatrix} \mathrm{DOI}_{\mathcal{S}_1}(\textit{Sculpture}) \\ \mathrm{DOI}_{\mathcal{S}_1}(\textit{History}) \\ \mathrm{DOI}_{\mathcal{S}_1}(\textit{Information}) \end{pmatrix} = \begin{pmatrix} 1 \\ 0.8 \\ 0.8 \end{pmatrix}$$

The correlation between $v_{\mathcal{S}_1}$ and $v_{\mathcal{V}}$ turns out to be very high, 0.998 (the maximum would be 1), because the two vectors contain very similar ratings for the classes considered. Knowing this value, the pseudo-collaborative strategy can predict the levels of interest corresponding to the learning courses $\mathcal{L}_1$ and $\mathcal{L}_3$, applying Eq. 8 as follows:

$$\begin{aligned}
\text{Pred}(\mathcal{L}_1, \mathcal{V}) &= \text{corr}(\mathcal{S}_1, \mathcal{V}) \cdot \text{Match}(\mathcal{L}_1, \mathcal{S}_1) = \\
&= 0.998 \cdot \frac{\text{SemSim}(\mathcal{L}_1, \mathcal{P}_1, \mathcal{S}_1) \cdot \text{DOI}_{\mathcal{S}_1}(\mathcal{P}_1) + \text{SemSim}(\mathcal{L}_1, \mathcal{L}_3, \mathcal{S}_1) \cdot \text{DOI}_{\mathcal{S}_1}(\mathcal{L}_3)}{2} = \\
&= 0.998 \cdot \frac{0.3 \cdot 1 + 0 \cdot 0.8}{2} = 0.1497 \\
\text{Pred}(\mathcal{L}_3, \mathcal{V}) &= \text{corr}(\mathcal{S}_1, \mathcal{V}) \cdot \text{DOI}_{\mathcal{S}_1}(\mathcal{L}_1) = 0.998 \cdot 0.8 = 0.7984
\end{aligned}$$

Again, $\mathcal{L}_3$ is found much more relevant than $\mathcal{L}_1$, so it would also be the first item to be recommended to the viewer by the pseudo-collaborative strategy.

The way to combine the outcomes of the two filtering strategies is domain-dependent. In some cases (e.g. in personalized advertising to fill one ad slot), it is necessary to decide on only one item for each viewer; in others (e.g. in a personalized programming guide), the common approach is to offer a menu of various items. Furthermore, one could choose to run only the content-based approach if it returns high matching values, to recommend only items that exceed certain thresholds in either Eq. 6 or Eq. 8, to combine the two measures, etc. Given the wide range of alternatives, our architecture does not enforce any particular solution in this regard.

4 Evaluation

In order to validate the proposed architecture in practice, we have applied it to modify a system that used to deliver personalized services through server-based semantic reasoning. Specifically, we have made experiments to assess the personalization quality achieved before and after moving part of the personalization logic to the DTV receivers. The following subsection explains the relevant technical details of the experiments, and Section 4.2 describes the evaluation methodology and results.

4.1 Technical settings

The system we took as a reference for our experiments was the ATLAS[9] t-learning platform presented in [41], which provides solutions to build and deliver informal learning services linked to TV programs, within the technological framework defined by the MHP[10] standard. On the one hand, ATLAS includes a development tool called ATTOS[11] to create courses from reusable parts, and to annotate them with metadata from the LOM (*Learning Object Metadata*) specification [23]. On the other, it provides mechanisms to deliver courses through IP networks or MPEG-2 broadcast

[9] ATLAS is an acronym for "*Architecture for T-Learning interActive Services*".

[10] http://www.mhp.org.

[11] ATTOS is an acronym for "*ATlas, TOol Support*".

streams, and to ensure the coherent presentation of multiple pieces of information, among others.

In the aspect most related to this paper, the original version of ATLAS featured a server-based recommender system that could identify potentially interesting services for a viewer, dealing with two main sources of information:

– On the one hand, there was an OWL ontology describing and interrelating TV programs and learning courses, using metadata fields and properties from the TV-Anytime [52], LOM and LIP (*Learner Information Package*) [25] specifications.
– On the other hand, for every viewer, there was one profile (excerpted from the ontology just as explained in Section 3.1) that quantified his/her satisfaction with TV programs and learning courses watched or executed in the past by means of DOIs. The profiles also contained some demographical data standardized by LIP, such as age, gender or occupation.

The recommender system ran continually behind the scenes, matching the descriptions of the TV programs being watched by connected viewers and the learning courses available. Whenever suitable courses were found for a viewer, a blinking button would appear at the top-right corner of his/her screen, warning about the existence of interactive applications that he/she might find interesting at the moment. Pressing the button would display a list of the recommended courses, each one accompanied by a brief sentence explaining the main reasons why it is suggested (i.e. indicating the attributes that bear the greatest contributions to Eq. 6 or Eq. 8). This sequence is illustrated in Fig. 5 for a viewer who is watching a documentary about the Grand Canyon: first, the warning button appears on screen; second, the viewers browses the list of recommended courses; and, finally, a course about the natural wonders of the world appears on screen. Upon closing any application, as shown in Fig. 5(d), the viewers would be asked to rate it with a number between 0 and 9, which we would map internally into a DOI.

In [41], we tried the server-based recommender system with viewers who owned MHP-compliant set-top boxes connected to a DVB-C[12] cable broadcast network and with return channels permanently enabled (through the same cable). Using the Protégé tool,[13] the ontology handled by the recommender system was first populated with metadata about 170 learning courses that we created ourselves (reusing many pieces of content around different topics and with different storylines); then, it was enriched with information from repositories like the BBC Backstage[14] and Internet databases like the IMDB[15] to reach a size of more than 25, 000 nodes.

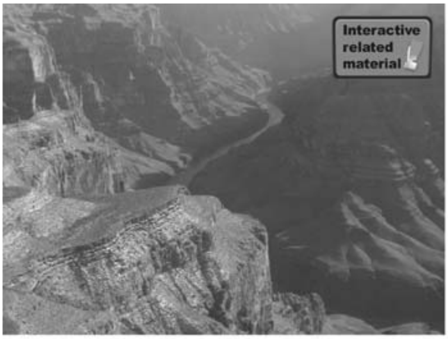
To test the new architecture, we prepared a new deployment also for MHP set-top boxes, but this time considering terrestrial broadcast through DVB-T networks—with a conservative bit rate of 256 Kbps for carousels—and no return channels at all.

[12] http://www.dvb.org/technology/standards/.

[13] http://protege.stanford.edu/.

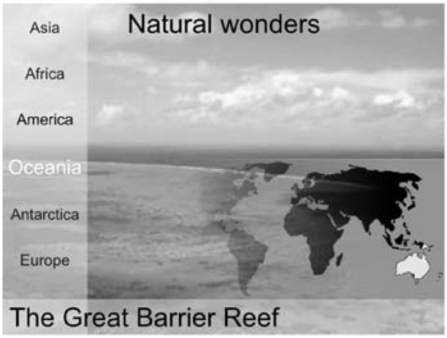
[14] http://backstage.bbc.co.uk.

[15] http://www.imdb.com.



(a) Warning about potentially interesting courses

(b) Browsing the list of recommended courses



(c) An interactive course on screen

(d) Asking the viewer to rate the course

Fig. 5 Snapshots of the recommender system in action (**a**–**d**)

We worked with the same ontology and the same set of courses as above, fixing a maximum size of 800 nodes for the partial ontologies and a limit of 1.5 MBytes for the total size of the items delivered through broadcast at any time.[16] The individual viewer profiles maintained the same structure as before, but they were now stored in the set-top boxes. For the head-end, we defined a set of 15 stereotypes as follows:

- First, we clustered the viewer profiles that had built up during the experiments reported in [41]. Specifically, 14 clusters contained the profiles that had comparatively high (close to 1) or comparatively low (close to -1) DOIs for items classified under *Sports*, *Nature*, *Politics*, *History*, *Science*, *Art* or *Economy*. One final cluster gathered the profiles that did not meet any of those conditions.
- From the profiles in each cluster, one stereotype was computed by averaging the DOIs they contained for any TV programs and learning courses. The DOIs so computed for the stereotypes propagated upwards in the ontology as explained in Section 3.1.

[16]Thanks to the reuse capabilities of ATLAS (see [41]), 1.5 MBytes sufficed to deliver up to 20 courses at the same time.

To complete the settings for the new experiments, we adapted the semantic similarity metrics and the filtering algorithms of [7, 8] as explained in Section 3, measuring an average time of 7 s for a set-top box with 166 MHz processor speed and 64 MBytes of RAM memory to compute recommendations.

4.2 Evaluation methodology and results

Within the aforementioned settings, we have conducted experiments to assess the personalization quality achieved by receiver-side semantic reasoning, mainly in terms of *precision* (percentage of recommended courses that the viewers rate positively), *recall* (percentage of interesting courses recommended) and viewers' overall perception of the personalization service. The performance of the previous server-based approach had been evaluated in [41], obtaining the charts of Fig. 6.

The new experiments involved nearly 230 viewers recruited among our graduate/undergraduate students and their relatives or friends, incentivized by the possibility of winning coupons for pay-per-view services, recharge vouchers for mobile phones or cash prizes. These people formed a diverse audience, with disparate demographical data and educational backgrounds. There were nearly as many men as women (52% vs 48%), their ages ranging from 9 to 62 years old.

Prior to receiving any recommendations, each viewer was asked to rate his/her interest in topics related to *Sports*, *Nature*, *Politics*, *History*, *Science*, *Art* and *Economy* with a number between 0 and 9. Thereupon, their individual profiles were initialized by weighing the DOIs of the corresponding stereotypes. Having done this, the personalization system ran for 4 weeks, during which the set-top boxes recorded the recommendations made and the ratings provided by each viewer. At the end, we collected the log files and ran a poll offline to ask the viewers about their perceptions of the personalization service. We analyzed the data following the same procedures as in [41], getting to the charts of Fig. 7:

– For the estimation of precision, we divided the number of courses that the viewers had liked (i.e. rated greater than 5) by the number of courses they had launched. As a result, we got a value nearing 69%, which is about 7% lower than with the original, server-based approach. This reduction is partly due to the fact

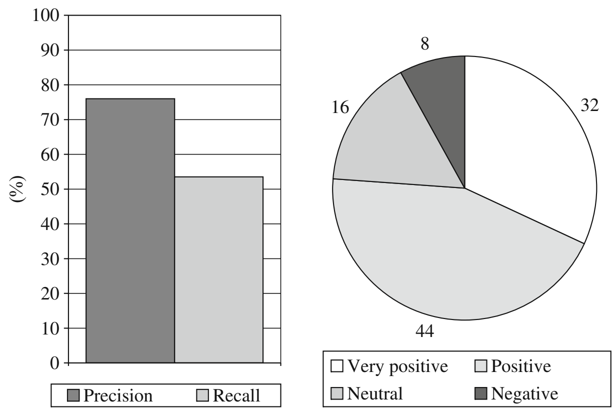


Fig. 6 Precision, recall and viewer perceptions of the original server-based recommender

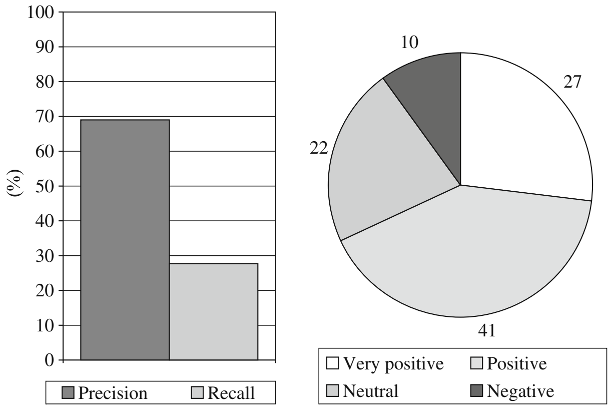


Fig. 7 Precision, recall and viewer perceptions of the new receiver-based recommender

that the pruned ontologies do not always include all of the attributes that may relate different items by means of inferential similarity, so this measure becomes somewhat less effective. Also, there is an issue with viewers who happen not to be adequately represented by any of the stereotypes, implying that the services most suited to their preferences are not included in the broadcast emissions. Nevertheless, our 69% remains much greater than the precision achieved by syntactic approaches to receiver-side reasoning in DTV (e.g. in [41] we had measured the syntactic approach of [20] to reach barely above 20%), and is indeed competitive with regard to server-side semantic engines.

– For the estimation of recall, we examined only the logs of the viewers who had previously agreed to classify the courses as *potentially appealing* or *potentially unappealing*. The value represented in Fig. 7 (about 28%) corresponds to the number of *potentially appealing* courses that ever appeared in a list of recommendations for each one of those viewers.[17] This is nearly 25% lower than with the server-based approach. In this case, the reduction is due to the fact that, in the absence of return channels, the viewers can only be faced with some of the courses delivered through broadcast, which most of the times do not include all the material of interest for any given individual.
– Finally, in what concerns the viewers' perceptions, we got nearly 120 responses to the poll. Despite the reductions in precision and recall, the good news is that the viewers' satisfaction with the personalized offerings remained more or less the same as with the server-based approach, with 68% of the viewers rating the experience positively or very positively. More than half of those viewers recognized that they had enjoyed courses they would not have found by themselves (in many cases, admittedly, because they would not even have bothered to look for learning material). On the negative side, around 10%

[17] Interestingly, we could check that the ratings given by the viewers to courses they had classified as *potentially appealing* were lower than 5 only in 8% of the cases. This fact undoubtedly supports the validity of our estimation of recall.

of the viewers considered the recommender system a nuisance; among them, nearly 80% said that they would rather search for interactive applications by themselves, whereas the rest showed no interest at all in the interactive offerings.

It is worth noting that the reductions in precision and recall can be tuned by modifying the values of parameters like the bandwidth available to carousels and the limit size of the partial ontologies, which directly affect the service loading times and the computational cost of the recommendations, respectively. With these trade-offs in mind, we have been recently working with an in-lab testbed to apply our new architecture in DVB-H networks for mobile DTV receivers. In this context, we believe to have found a reasonable balance with 64 Kbps for carousels and up to 250 nodes in the partial ontologies, which yields an average time of 10 s to compute recommendations in devices with 66 MHz processor speed and 2 Mbytes of RAM memory. Practical experiments will be initiated soon to assess the personalization quality achieved and the viewers' perceptions, with a new version of ATLAS intended to deliver learning services more suited to mobile settings.

5 Conclusions and future work

The possibility of transmitting interactive applications along with the TV programs was conceived as one of the most promising innovations of the Digital TV technologies. Years of experience, however, have shown that those applications are likely to go unnoticed unless they provide information that matches the preferences and needs of the viewers. Bearing this in mind, we have presented a new architecture for DTV personalization around two major ideas: (i) to run the personalization engines in the viewers' receivers, ensuring availability of the personalized offerings even in the absence of return channels, and (ii) to apply semantic reasoning techniques as a means to achieve good personalization quality. The major features of the proposal have been devised to reduce the amount of information to be handled by the receivers, assuring that they always deal with the most relevant data.

Our experiments with the ATLAS t-learning platform have proved the feasibility of the proposed architecture in domestic and mobile DTV settings, since we have been able to deliver ontologies big enough to make recommendations driven by subtle semantic relationships between TV programs and educational services. This rich reasoning process has made it possible to outperform previous (syntactic) approaches to receiver-side personalization in terms of viewers' satisfaction with the recommendations. Nonetheless, our proposal achieves slightly lower personalization quality than existing approaches to server-side semantic reasoning, but we could confirm that this reduction does not affect significantly the viewers' positive perception of the services delivered, so the loss may well be worth in exchange for uninterrupted availability.

It is important to note that our two-layered approach to personalization can also be applied with DTV technologies for which return channels are readily available, such as the different flavors of IPTV[18] or the DVB-C standard for cable television. In these settings, running the filtering algorithms in the receivers would allow to reduce

[18] http://www.itu.int/ITU-T/IPTV/.

the burden on the personalization servers, harnessing the computational power of thousands or millions of consumer devices. Alternatively, it would be possible to do the filtering in the personalization servers, because they can easily know what viewers are on the other side. In the latter case, the servers would be in charge of all the personalization tasks, as it happens in most of the previous systems; however, since the pre-selection and pruning processes can be performed in advance (just knowing the TV schedule of programs that will be broadcast in the future), the work to do at viewing time is significantly reduced, so increasing the responsiveness of the system when it comes to serve huge numbers of viewers at the same time. By the way, this optimization could leave place to enhance semantic recommenders with *context awareness* features, which have been typically considered (see [55]) too demanding for the responsiveness required.

Probably, the greatest concern that the reader may have now about our proposal relates to the stereotypes used in the pre-selection and pruning processes. Clearly, the personalization quality we can achieve depends on having a number of stereotypes that represent accurately the preferences and needs of the potential audiences of the TV programs delivered at any time. The problem with the stereotypes is that they are difficult to obtain, and also difficult to keep up-to-date with regard to the new audiovisual contents, new pieces of learning, new commercial products, ... that appear every day. Nowadays, the management of stereotypes is primarily driven by audience studies performed offline, or with the participation of a reduced number of viewers who have installed special equipment. While this approach has been working reasonably well for decades, it misses a lot of information that could lead to even better quality in the new wave of personalized services. In response to that, we are currently doing research on mechanisms to gather feedback about TV programs and interactive services from unrestricted audiences, taking advantage of any periods of connectivity. Therefrom, our goal is to develop analysis and visualization tools to help process that feedback, together with indirect measures of viewers who watch the TV programs from beginning to end, viewers who do zapping, viewers who launch interactive services, etc. This approach will help improve the current audience measurement methodologies, aiding analysts to manage a dynamic base of stereotypes more efficiently and effectively than today.

Acknowledgements This work has been partially funded by the Ministerio de Educación y Ciencia (Gobierno de España) project TSI2007-61599, by the Consellería de Educación e Ordenación Universitaria (Xunta de Galicia) incentives file 2007/000016-0, and by the Consellería de Innovación, Industria e Comercio (Xunta de Galicia) project PGIDIT05PXIC32204PN.

References

1. Antoniou G, van Harmelen F (2004) A semantic web primer. MIT, Cambridge
2. Anyanwu K, Sheth A (2003) ρ-Queries: enabling querying for semantic associations on the semantic web. In: Proceedings of 12th international world wide web conference, Budapest, 20–24 May 2003
3. Ardissono L, Gena C, Torasso P, Bellifemine F, Chiarotto A, Difino A, Negro B (2004) User modeling and recommendation techniques for personalized electronic program guides. In: Personalized digital television. Targeting programs to individual users. Kluwer, Dordrecht
4. Ardissono L, Kobsa A, Maybury M (eds) (2004) Personalized digital television. Targeting programs to individual users. Kluwer, Norwell

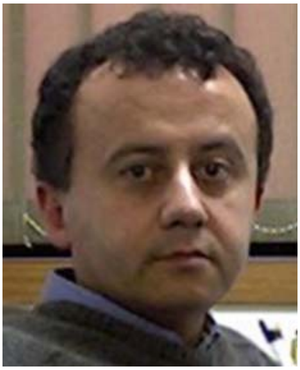
5. Balabanović M, Shoham Y (1997) Fab: content-based collaborative recommender. Commun ACM 40:66–72
6. Basu C, Hirsh H, Cohen W (1998) Recommendation as classification: using social and content-based information in recommendation. In: Proceedings of the 15th national conference on artificial intelligence, Madison, 26–30 July 1998
7. Blanco-Fernández Y, Pazos-Arias JJ, Gil-Solla A, Ramos-Cabrer M, López-Nores M, García-Duque J, Fernández-Vilas A, Díaz-Redondo RP, Bermejo-Muñoz J (2007) AVATAR: enhancing the personalized television by semantic inference. Int J Pattern Recogn Artif Intell 21: 397–422
8. Blanco-Fernández Y, Pazos-Arias JJ, Gil-Solla A, Ramos-Cabrer M, López-Nores M, García-Duque J, Díaz-Redondo RP, Fernández-Vilas A, Bermejo-Muñoz J (2008) A flexible semantic inference methodology to reason about user preferences in knowledge-based recommender systems. Knowl Based Syst 21:305–320
9. Blanco-Fernández Y, Pazos-Arias JJ, Gil-Solla A, Ramos-Cabrer M, López-Nores M, García-Duque J, Fernández-Vias A, Díaz-Redondo RP, Bermejo-Muñoz J (2008) An MHP framework to provide intelligent personalized recommendations about Digital TV contents. Softw Pract Exp. doi:10.1002/spe.855
10. Blanco-Fernández Y, Pazos-Arias JJ, López-Nores M, Gil-Solla A, Ramos-Cabrer M (2006) AVATAR: an improved solution for personalized TV based on semantic inference. IEEE Trans Consum Electron 52:223–231
11. Bobillo F, Delgado M, Gómez-Romero J (2008) Representation of context-dependant knowledge in ontologies: a model and an application. Expert Syst Appl. doi:10.1016/j.eswa.2007.08.090
12. Burke R (2002) Hybrid recommender systems: survey and experiments. User Model User-Adapt Interact 12:331–370
13. Cao Y, Li Y (2007) An intelligent fuzzy-based recommendation system for consumer electronic products. Expert Syst Appl 33:230–240
14. Cheung B, Hui L, Zhang J, Yiu SM (2003) SmartTutor: an intelligent tutoring system in web-based adult education. J Syst Softw 68:11–25
15. Chorianopoulos K (2008) Personalized and mobile digital TV applications. Multimed Tools Appl 36:1–10
16. Difino A, Negro B, Chiarotto A (2002) A multi-agent system for a personalized electronic program guide. In: Proceedings of the 2nd workshop on personalization in future TV, Malaga, 29–31 May 2002
17. Digital Tech Consulting Inc (2006) Digital TV receivers: worldwide history and forecasts (2000–2010). http://www.dtcreports.com/report_dtvr.aspx
18. Foltz PW, Dumais ST (1992) Personalized information delivery: an analysis of information filtering methods. Commun ACM 35:51–60
19. Ganesan P, Garcia-Molina H, Widom J (2003) Exploiting hierarchical domain structure to compute similarity. ACM Trans Inf Syst 21:64–93
20. Ghaneh M (2004) System model for t-learning application based on home servers (PDR). Broadcast Technol. 19. http://www.nhk.or.jp/strl/publica/bt/en/rep0019.pdf
21. Haase P, Ehrig M, Hotho A, Schnizler B (2004) Personalized information access in a bibliographic peer-to-peer system. In: Proceedings of the AAAI workshop on semantic web personalization, San Jose, 25–26 July 2004
22. Hung L (2005) A personalized recommendation system based on product taxonomy for one-to-one marketing online. Expert Syst Appl 29:383–392
23. IEEE Learning Technology Standards Committee (2002) IEEE standard for learning object metadata. IEEE standard 1484.12.1
24. Im KH, Park SC (2007) Case-based reasoning and neural network based expert system for personalization. Expert Syst Appl 32:77–85
25. IMS Global Learning (2005) IMS learner information packaging information model v1.0.1. http://www.imsglobal.org
26. Karvounarakis G, Alexaki S, Christophides V, Plexousakis D, Scholl M (2002) RQL: a declarative query language for RDF. In: Proceedings of the 11th world wide web conference, Honolulu, 7–11 May 2002
27. Lacal J (ed) (2007) Proceedings of the 1st workshop of the IEEEP2407 personalized health informatics working group. http://www.IEEE2407.org
28. Lee J, Kim M, Lee Y (1993) Information retrieval based on conceptual distance in IS-A hierarchies. J Doc 49:188–207

29. Lee WP, Yang TH (2003) Personalizing information appliances: a multi-agent framework for TV programme recommendations. Expert Syst Appl 25:331–341
30. Lekakos G, Giaglis G (2004) A lifestyle-based approach for delivering personalised advertisements in digital interactive television. J Comput Mediat Commun 9
31. Leung CW, Chan SC, Chung FL (2006) A collaborative filtering framework based on fuzzy association rules and multiple-level similarity. Knowl Inf Syst 10:357–381
32. Lim J, Kim M, Lee B, Kim M, Lee H, Lee HK (2008) A target advertisement system based on TV viewer's profile reasoning. Multimed Tools Appl 36:11–35
33. López-Nores M, Pazos-Arias JJ, García-Duque J, Blanco-Fernández Y (2008) A smart medicine manager delivering health care to the networked home and beyond. In: Proceedings of the international conference on health informatics, Funchal, 28–31 January 2008
34. Lougos C, Vassilopoulou C, Vrechopoulos A (2002) Interactive digital TV services - viewers' perceptions. In: Proceedings of the eBusiness and eWork conference, Prague, 16–18 October 2002
35. Maybury M, Greiff W, Boykin S, Ponte J, McHenry C, Ferro L (2004) Personalcasting: tailored broadcast news. In: Personalized digital television. Targeting programs to individual users. Kluwer, Norwell
36. McGuinness D, van Harmelen F (2004) OWL Web Ontology Language overview. W3C Recommendation
37. Miller L, Seaborne A, Reggiori A (2002) Three implementations of SquishQL, a simple RDF query language. In: Proceedings of the 1st international semantic web conference, Sardinia, 10–12 June 2002
38. Mobasher B, Jin X, Zhou Y (2004) Semantically-enhanced collaborative filtering on the web. In: Web mining: applications and techniques. Idea Group. Hershey (PA), USA
39. Morris S, Smith-Chaigneau A (2005) Interactive TV standards. Focal Press, St. Louis
40. Pazos-Arias JJ, Delgado Kloos C, López-Nores M (eds) (2008) Personalization of interactive multimedia services: a research and development perspective. Nova Science, Hauppauge
41. Pazos-Arias JJ, López-Nores M, García-Duque J, Díaz-Redondo RP, Blanco-Fernández Y, Ramos-Cabrer M, Gil-Solla A, Fernández-Vilas A (2008) Provision of distance learning services over interactive digital TV with MHP. Comput Educ 50:927–949
42. Pazzani M (1999) A framework for collaborative, content-based and demographic filtering. Artif Intell Rev 13:393–408
43. Potonnieé O (2004) A decentralized privacy-enabling TV personalization framework. In: Proceedings of the 2nd European conference on interactive television, Brighton, 31 March–2 April 2004
44. Rada R, Mili H, Bicknell E, Blettnet M (1989) Development and application of a metric on semantic nets. IEEE Trans Syst Man Cybern 19:17–30
45. Resnik P (1999) Semantic similarity in a taxonomy: an information-based measure and its application to problems of ambiguity in natural language. J Artif Intell Res 11:95–130
46. Resnick P, Iacovou N, Suchak M, Bergstrom P, Riedl J (1994) Grouplens: an open architecture for collaborative filtering of netnews. In: Proceedings of the ACM conference on computer-supported cooperative work, Chapel Hill, 22-26 October 1994
47. Sintek M, Decker S (2002) TRIPLE: a query, inference and transformation language for the Semantic Web. In: Proceedings of the 1st international semantic web conference, Sardinia, 10–12 June 2002
48. Smyth B, Cotter P (1999) Surfing the digital wave: generating personalized TV listings using collaborative, case-based recommendation. In: Proceedings of the 3rd international conference on case-based reasoning, Munich, 27–30 July 1999
49. Staab S, Studer R (eds) (2003) Handbook on ontologies. Springer, New York
50. Trifonova A, Ronchetti M (2005) Hoarding content in m-learning context. In: Proceedings of the 3rd IEEE conference on pervasive computing and communications workshops, Kauai Island, 8–12 March 2005
51. Tsunoda T, Hoshino M (2008) Automatic metadata expansion and indirect collaborative filtering for TV program recommendation system. Multimed Tools Appl 36:37–54
52. TV-Anytime forum (2003) TV-anytime specification series. ETSI standard TS 102 822
53. Tzouveli P, Mylonas P, Kollias S (2008) An intelligent e-learning system based on learner profiling and learning resources adaptation. Comput Educ 51:224–238
54. Velusamy S, Gopal L, Varatharajan S, Bhatnagar S (2007) Fuzzy clustering based ad recommendation for TV programs. In: Proceedings of the 5th European conference on interactive TV, Amsterdam, 24–25 May 2007

55. Woerndl W, Schlichter J (2007) Introducing context into recommender systems. In: Proceedings of the AAAI workshop on recommender systems in e-commerce, Vancouver, 22–23 July 2007
56. Zhang H, Zheng S, Yuan J (2005) A personalized TV guide system compliant with MHP. IEEE Trans Consum Electron 51:731–737
57. Zimmerman J, Kurapati K, Buczak AL, Schaffer D, Gutta S, Martino J (2004) TV personalization system. Design of a TV show recommender engine and interface. In: Personalized digital television. Targeting programs to individual users. Kluwer, Norwell

Martín López-Nores received the Ph.D. degree in Computer Science from the University of Vigo in 2006. His research deals primarily with the design of personalization architectures for a range of DTV applications, considering both fixed and mobile receivers.

Yolanda Blanco-Fernández received the Ph.D. degree in Computer Science from the University of Vigo in 2007. Her research is focused on knowledge representation, semantic reasoning technologies and recommender systems.



José J. Pazos-Arias received the Ph.D. degree in Computer Science from the Madrid University of Technology (UPM) in 1995, and worked with Alcatel Laboratories in Madrid prior to joining the University of Vigo. He is the founder and director of the Networking & Software Engineering Group, which is currently involved with several projects related to DTV middleware and applications.

Jorge García-Duque received the Ph.D. degree in Computer Science from the University of Vigo in 2000. His research is focused on the deployment of information services over heterogeneous networks of consumer devices.

Manuel Ramos-Cabrer received the Ph.D. degree in Computer Science from the University of Vigo in 2000. His research interests include the application of artificial intelligence techniques to personalization systems.

Alberto Gil-Solla received the Ph.D. degree in Computer Science from the University of Vigo in 2000. His research is currently involved with different aspects of middleware design and interactive multimedia services.

Rebeca P. Díaz-Redondo received the Ph.D. degree in Computer Science from the University of Vigo in 2002. Her research is now focused on interactive DTV applications playing a central role in the control of smart home environments.

Ana Fernández-Vilas received the Ph.D. degree in Computer Science from the University of Vigo in 2002. Her research interests deal with Web Services technologies and ubiquitous computing environments.