Enhancing content-based recommendation with the task model of classication

Wang, Y.; Wang, S.; Stash, N.; Aroyo, L.M.; Schreiber, A.Th.

published in
Proceedings of the 17th International Conference on Knowledge Engineering and Knowledge Management
2010

DOI (link to publisher)
10.1007/978-3-642-16438-5_33

document version
Peer reviewed version

Enhancing content-based recommendation with the task model of classication

Wang, Y., Wang, S., Stash, N., Aroyo, L.M., Schreiber, A.Th.

In: P. Cimiano, H.S. Pinto (Ed.), Proceedings of the 17th International Conference on Knowledge Engineering and Knowledge Management (pp. 431-440). Lisbon

Published version: http://dx.doi.org/10.1007/978-3-642-16438-5_33

Enhancing Content-based Recommendation with the Task Model of Classification

Yiwen Wang[1], Shenghui Wang[2], Natalia Stash[1], Lora Aroyo[1 2], and Guus Schreiber[2]

[1] Eindhoven University of Technology, Computer Science
{y.wang,n.v.stash}@tue.nl
[2] VU University Amsterdam, Computer Science
{l.m.aroyo,schreiber}@cs.vu.nl, swang@few.vu.nl

Abstract. In this paper, we define reusable inference steps (realize, classification by concepts, classification by instances and retrieve) for content-based recommender systems applied on semantically-enriched collections. In our case, we use the enriched museum collection. The core steps: (i) Classification by concepts brings explicitly related concepts via artwork features and semantic relations between artworks and concepts, e.g. "The Night Watch" has *creator* "Rembrandt van Rijn" and "Rembrandt van Rijn" is a *student of* "Pieter Lastman"; and (ii) Classification by instances brings implicitly related concepts using the method of instance-based ontology matching, e.g. "Cupid" is implicitly related to "Love and sex" because they describe sufficient artworks in common. To combine predictions from these two steps for each related concept, we set a parameter α to balance the strength of explicit and implicit recommendations. We test our strategy with the CHIP Art Recommender in terms of accuracy and discuss the added values of providing serendipitous recommendations and supporting more complete explanations for recommended items.

Key words: Content-based recommendation, reusability, ontology, inference, classification, semantic relation, cultural heritage

1 Introduction and Research Challenges

In recent years, the Semantic Web has put great effort on the reusability of knowledge. However, most work deals with reusable ontology and ontology patterns [1], there is hardly any work on reusable reasoning patterns, except the work from van Harmelen and ten Teije [9]. They made a first attempt at finding reusable task types and decomposing these tasks into a number of primitive reasoning patterns for Semantic Web applications. In CHIP, we collaborate with the Rijksmuseum Amsterdam[3] and built an art recommender system based on the semantically-enriched collection with the mappings to standard vocabularies [10]. Inspired by the work from van Harmelen and ten Teije, we pose the

[3] http://www.rijksmuseum.nl

question: can we identify reusable knowledge elements that can help designers of such recommender systems on the semantic web? In this paper, we address the following research challenges:

(i) ***Finding reusable inference steps for recommender systems based on rich semantic vocabularies***

As a first attempt, we analyze our demonstrator[4] (called the "CHIP Art Recommender") and identify several tasks, e.g. browse, search and content-based recommendation. In this paper, we focus on the task of content-based recommendation and decompose this task into a number of inference steps: realization, classification by concepts, classification by instances, and retrieval.

(ii) ***Bridging the vocabulary gap***

For the semantic enrichment of museum collections, most concepts of artworks have been mapped to common vocabularies for semantic-based knowledge representations [3][4]. However, because of the complexity of the museum collections, it still contains many concepts/terms that can not be mapped to common vocabularies. These unmapped concepts are often described in non-standard schemas or in different languages. In this context, how can we bridge the discrepancy between the semantically-structured data and the remaining unstructured/unmapped data? How can we combine data from these two parts for recommendations?

To address this issue, Isaac et al. [6] proposed a method of instance-based ontology matching. The basic idea is that the more significant the overlap of instances/artworks of two concepts is, the closer these two concepts are, and the level of significance is calculated by the corrected Jaccard measure [6]. We adopted their method in our system to build an implicit relation between two concepts even though there are no explicit semantic relations annotated between them. In such a way, most of the unmapped concepts are linked with the mapped concepts via implicit relations and this allows for further inference.

(iii) ***Improving accuracy, serendipity and explanation for recommendations***

It hardly needs arguing that the semantic enrichment of collections could retrieve more related items [11]. However, we still face the issue of how to maintain a relatively high accuracy for recommended items. This problem becomes even more complicated when there are multiple explicit and/or implicit relations involved for a recommended item, how can we still compute an accurate prediction for this item in a way suiting the user's art preference? Besides the accuracy, there are some other issues that also affect the user's satisfaction to a recommender system, e.g. recommending unexpected and new/unknown items (serendipity) and providing users an insight in the logic underlying the recommendations (explanation) [5].

To compute the prediction for related items, Ruotsalo and Hyvönen calculate the relevance of a concept with respect to an artwork using the TF-IDF metric

[4] http://www.chip-project.org/demo/

and give default weights for the general *braoder/narrower* relations [8]. Mobasher et al. propose the combination of user ratings of a particular artwork/concept and the semantic distance or similarity between two concepts by using latent semantic index (LSI) [7]. Our approach is to calculate the values from explicit and implicit relations separately, and then combine the values from these two parts by setting a parameter α. By tuning the value of α (between 0 and 1), we could change the strength of explicit (obvious) and implicit (serendipitous) recommendations. In addition, we develop a "Why recommend" function for each recommended item, explaining the various relations between the user's rated items and the recommended item.

The paper is organized as follows: In Section 2, we identify task types and corresponding inference steps. In Section 3, we explain the semantic-enhanced recommendation strategy. Further, in Section 4, we test our strategy with the CHIP Art Recommender in terms of accuracy, serendipity and explanation. We conclude and discuss the future work in Section 5.

2 Task Types and Inference Steps

The CHIP Art Recommender contains three different tasks: (i) browse, (ii) search, and (iii) content-based recommendation. For the first two tasks (browse and search), we adopted the definitions from van Harmelen and ten Teije [9]. In this paper, we focus on defining the third task (content-based recommendation) and analyzing the corresponding inference steps.

2.1 Defining Task of Content-based Recommendation

The standard content-based recommendation (CBR) usually takes the user profile plus the domain ontology and returns a set of instances, which might be of interest to the user [9]. In the case of CHIP, the system stores the user profile in the form of both a set of concepts and a set of instances. Based on the user profile and the domain ontology, it recommends both related concepts and instances via various relations from the collection.

As described in Table 1, we use formal preliminaries: a terminology T is a set of concepts c organized in a hierarchy. Instance i is a member of such concepts c and this is described as *$(i, \in, c)$* where $\in$ refers to the membership relation. An ontology O consists of a terminology T and a set of instances I. Sometimes we write *(T, I)* instead of O if we want to refer separately to the terminology and the instance set of the ontology. In the case of CHIP, instances refer to artworks and each artwork is described with a number of concepts. Based on the semantically-enriched Rijksmuseum collection [11], we specify three different kinds of relations between artworks and concepts: (i) artwork feature, (ii) semantic relation, and (iii) implicit relation.

(i) Explicit relation (or called "artwork feature") between an artwork/instance and a concept, denoted as $(i, \in, c)$. For example, the artwork "The Night Watch" is related to the concept "Rembrandt van Rijn" via the artwork feature

Table 1. The task of content-based recommendation

Input:	a user profile characterized as both a set of instance $I_{profile}$ and a set of concepts $C_{profile}$
Knowledge:	an ontology O = (T, I) consisting of a terminology T and an instance set I
Output:	a set of related concepts ($C^i \cup C^j \cup C^k$) with C^i: Recommend($I_{profile}$, O) = $\{$(i, $\in$, c^i) | $\exists$i: i $\in I_{profile} \wedge$ i $\in c^i\}$ C^j: Recommend($C_{profile}$, T) = $\{$($c^j \sim$ c) | $\exists$c: c $\in C_{profile} \wedge c^j \sim$ c$\}$ C^k: Recommend($C_{profile}$, O) = $\{$($c^k \simeq$ c) | $\exists$c: c $\in C_{profile} \wedge c^k \simeq$ c $\wedge$ i $\in$ c $\wedge$ i $\in c^k\}$ and a set of related instances I' with I': Recommend($C_{profile}$, C^i, C^j, C^k, O) = $\{$(i', $\in$, c')| c' $\in$ ($C_{profile} \cup C^i \cup C^j \cup C^k$) $\wedge$ i' $\in$ c'$\}$

"*creator*". In CHIP, we apply three artwork features for recommendations: *creator, creationSite* and *subject*. Each of them has a reverse relation, e.g. *creatorOf, creationSiteOf* and *subjectOf*.

(ii) Explicit relation between two concepts with a direct link (or called "semantic relation"), denoted as (c_i, $\sim$, c_j). In CHIP, most art concepts from the collection are mapped to the standard Getty vocabularies[5] (ULAN, AAT and TGN) and the Iconclass thesaurus[6] [10], which provides a rich semantic structure for further inference. Among various semantic relations between concepts, there are domain-specific relations within one vocabulary (e.g. *teacherOf*) and across two different vocabularies (e.g. *style*). Besides, there are also general relations within one vocabulary (e.g. *borader/narrower*).

(iii) Implicit relation between two concepts without a direct link, denoted as (c_i, $\simeq$, c_j). This relation is built based on common artworks/instances these two concepts both describe, although there are no explicit/direct links between them. For example, concepts "Rembrandt van Rijn" and "Chiaroscuro" are not directly connected but they describe 8 artworks in common out of 34 artworks that are described by either one of these two concepts. Thus we could assume that these two concepts are in a way extensionally related. Surprisingly, this implicit relation is confirmed by domain experts, since: Chiaroscuro in Italian means strong contrast of light and dark shading. The Italian painter Caravaggio originally made chiaroscuro his trademark and this effect is widely used in late 16th century by many Dutch painters, such as Rembrandt van Rijn. In such a way, "Rembrandt van Rijn" and "Chiaroscuro" are implicitly related. Another example is concepts "Venus" and "Aphrodite", which share 4 artworks out of 6 artworks. Aphrodite means the goddess of love and fertility in the Greek mythology and the goddness is called "Venus" in Roman.

[5] http://www.getty.edu/research/
[6] http://www.iconclass.nl/

2.2 Decomposing the Task into Inference Steps

To decompose the task of content-based recommendation, we identified four basic inference steps (see Fig. 1): (i) Realization, (ii) Classification by concepts, (iii) Classification by instances, and (iv) Retrieval. For each of them, we give a description, a signature (input and output datatypes), and a definition of the functionality (relation between input and output).

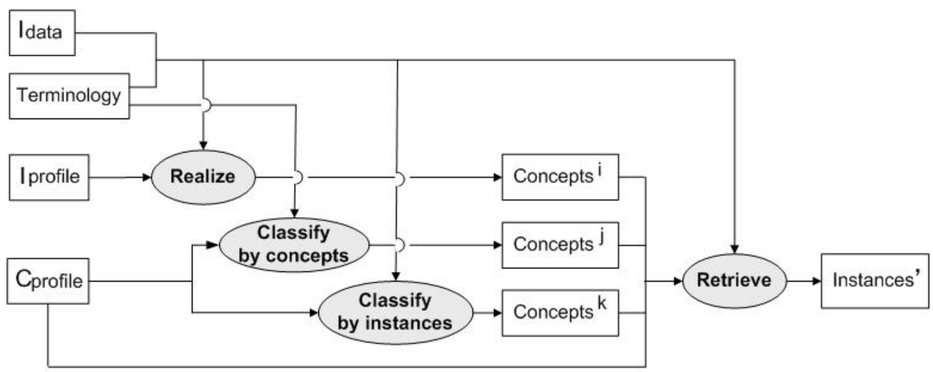

Fig. 1. Inference steps for the task of content-based recommendation

Realization is the task of finding a concept c that describe the given instances i.

• Definition: Find a concept c^i such that $O \vdash i \in c^i$

• Signature: $i \times O \mapsto c^i$

Classification by concepts is the task of finding a concept c^j which is directly linked to the given concept c through a semantic relation $\sim$ in the hierarchy of terminology T.

• Definition: Find a related concept c^j through various semantic relations $\sim$ (e.g. *broader, narrower, teacherOf, birthPlace*, etc.) in the terminology such that $T \vdash c \sim c^j$

• Signature: $c \times T \mapsto c^j$

Classification by instances is the task of finding a concept c^k which shares sufficient common instances with the given concept c using the instance-based ontology matching $\simeq$.

• Definition: Find a concept c^k through the instance-based ontology matching $\simeq$ such that $O \vdash c \simeq c^k \wedge \mathrm{i} \in \mathrm{c} \wedge \mathrm{i} \in c^k$

• Signature: $c \times O \mapsto c^k$

Retrieval is the inverse of realization: determining which instance *i'* belong to the related concept *c'*, where *c'* is a element of the unification of $C_{profile}$, C^i (Realization), c^j (Classification by concepts) and c^k (Classification by instances).

• Definition: Find an instance *i'* such that *i'* ∈ *c'* where *c'* ∈ $(C_{profile} \cup C^i \cup C^j \cup C^k)$

• Signature: *c'* × *O* ↦ *i'*

Compared with the original definition of recommendation and its corresponding inference steps from van Harmelen and ten Teije [9], we mainly extended the inference step of classification, which now consists of two components: classification by concepts and classification by instances. The original classification only determines where a given class should be placed in a subsumption hierarchy. It refers to the classification by concepts in our extended version, but we applied more semantic relations, e.g. the domain-specific relations (*teacherOf, style*) and the general relations (*broader/narrower*). In addition, we proposed a new component "classification by instances", which explores the implicit relations between concepts in the ontology.

3 Semantic-Enhanced Recommendation Strategy

Following the inference steps, in this section we will explain how the system computes the prediction for related concepts and artworks based on the user's profile. As a general strategy, we apply the content-based recommendation (CBR) in CHIP, which analyzes item features/descriptions in order to identify items that are likely of interest to the user [2]. Compared with other recommendation strategies (e.g. collaborative filtering), CBR performs well when there are sufficient features for items, even when there are only few user ratings [11]. Therefore it suits very well in the context of CHIP because the semantically-enriched collection could indeed provide us with rich metadata vocabularies, where artworks are connected with concepts via artworks features and concepts are linked with each other via various relations [10].

Suppose the user likes the artwork "The Little Street", concepts "Rembrandt van Rijn" and "Venus", Fig. 2 shows how the CHIP system recommends related concepts and artworks based on the user profile by taking all four inference steps.

• **Realization**: Based on the artwork "The Little Street", it recommends the concept "Johannes Vermeer" via the artwork feature *creator* and the concept "Townscape" via the artwork feature *subject*.

• **Classification by concepts**: Based on the concept "Rembrandt van Rijn", it recommends the concept "Pieter Lastman" via the semantic relation *studentOf* and the concept "Baroque" via the semantic relation *style*.

• **Classification by instances**: Based on the concept "Rembrandt van Rijn", it recommends the concept "Chiaroscuro" because they share sufficient (by setting the threshold) common artworks. Based on the concept "Venus", it recommends concepts "Francois van Bossuit" and "Aphrodite" also because of the sufficient common artworks they describe.

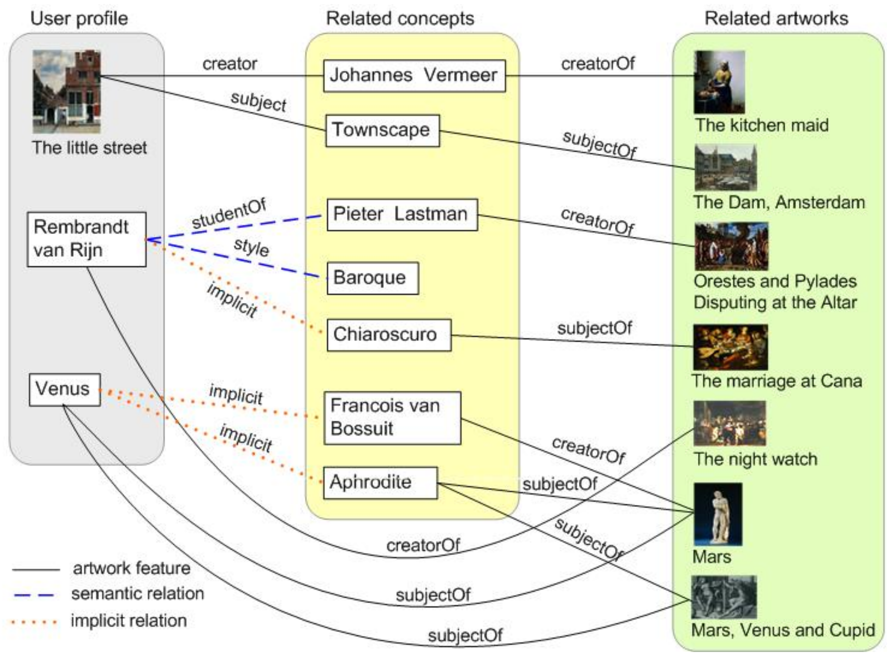

Fig. 2. Example of semantically-enhanced recommendations

• **Retrieval**: Based on three sets of concepts: (i) rated concepts ("Rembrandt van Rijn" and "Venus"); (ii) explicitly related concepts via artwork features and semantic relations ("Johannes Vermeer", "Townscape", "Pieter Lastman" and "Baroque"); and (iii) implicitly related concepts ("Chiaroscuro", "Francois van Bossuit" and "Aphrodite"), it recommends artworks "The Kitchen Maid", "The Dam, Amsterdam", "Orestes and Pylades Disputing at the Altar", "The Marriage at Cana", "The Night Watch", "Mars" and "Mars, Venus and Cupid" via artwork features *creatorOf* and *subjectOf*.

3.1 Computing the Explicit Value for the Steps of Realization and Classification by Concepts

In a previous user study [11], we explored the use of various explicit relations between artworks and concepts for recommendations. These relations include: (i) artwork features between an artwork and concepts (e.g. *creator*); and (ii) semantic relations between two concepts within one vocabulary (e.g. *broader*) and across two different vocabularies (e.g. *style*).

Using the existing user ratings collected from this study, we investigated the preliminary weights $W_{(r)}$ (see Table 2) for each explicit relation $R_{(i,j)}$, which is either an artwork feature between an artwork i and a concept j or a semantic relation between two concepts (i and j). For example, the relation between artwork "The Little Street" and concept "Johannes Vermeer" is *creator*, denoted

Table 2. Weights of explicit relations

Relation	creator	creation Site	subject	style	birth Place	death Place	teacher Of	aat Broader	tgn Broader	ic Broader
Weight	0.67	0.35	0.50	0.63	0.32	0.26	0.43	0.53	0.22	0.50
Inverse Relation	creator Of	creation SiteOf	subject Of	style Of	birth PlaceOf	death PlaceOf	student Of	aat Narrower	tgn Narrower	ic Narrower
Weight	0.68	0.31	0.54	0.61	0.28	0.21	0.44	0.55	0.16	0.52

as $R_{(TheLittleStreet,JohannesVermeer)} = creator$. From Table 2, we know that the weight of this relation $W_{(creator)}$ is 0.67. In the formulas below we write $W_{(i,j)}$ instead of $R_{(i,j)}$ and $W_{(r)}$.

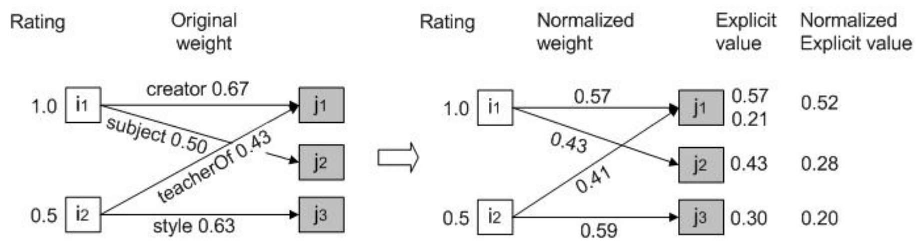

Fig. 3. Example of calculating the normalized explicit value

Considering that a rated item (either an artwork or a concept) could be linked to multiple items via various explicit relations, we need to normalize the weight(s) for each related item. As shown in Fig. 3, the rated item i_1 is linked to items j_1 and j_2. The relation between i_1 and j_1 is *creator* and the corresponding weight of *creator* is denoted as $W_{(i_1,j_1)}$. From Table. 2, we know that $W_{(i_1,j_1)}$ (creator) is 0.67, $W_{(i_1,j_2)}$ (subject) is 0.50, $W_{(i_2,j_1)}$ (teacherOf) is 0.43, and $W_{(i_2,j_3)}$ (style) is 0.63.

$$NW_{(i,j)} = \frac{W_{(i,j)}}{\sum_{j=1}^{J} W_{(i,j)}} \qquad \text{(Formula 1: Normalized weight)}$$

To normalize the weights, Formula 1 is applied. For example, based on i_1, the normalized weight of j_1: $NW_{(i_1,j_1)} = \frac{0.67}{0.67+0.50} = 0.57$ and the the normalized weight of j_2: $NW_{(i_1,j_2)} = \frac{0.50}{0.67+0.50} = 0.43$. In this way, we could calculate that based on i_2, normalized weight of j_1: $NW_{(i_2,j_1)} = \frac{0.43}{0.43+0.63} = 0.41$ and the normalized weight of j_3: $NW_{(i_2,j_3)} = \frac{0.63}{0.43+0.63} = 0.59$.

$$Exp_{(i,j)} = NW_{(i,j)} \times R_{(i)} \qquad \text{(Formula 2: Explicit value)}$$

Based on the normalized weights and user ratings, the next step is to compute the semantic value, see Formula 2. Based on i_1, the semantic values of j_1 and j_2

are: $Exp_{(i_1,j_1)} = 0.57 * 1.0 = 0.57$, and $Exp_{(i_1,j_2)} = 0.43 * 1.0 = 0.43$. Based on i_2, $Exp_{(i_2,j_1)} = 0.41 * 0.5 = 0.21$, and $Exp_{(i_2,j_3)} = 0.59 * 0.5 = 0.30$.

$$NExp_{(j)} = \frac{\sum_{i=1}^{I} Exp_{(i,j)}}{\sum_{i=1}^{I}\sum_{j=1}^{J} Exp_{(i,j)}} \qquad \text{(Formula 3: Normalized explicit value)}$$

Finally, we also need to normalize these semantic values for each related item, see Formula 3. $NExp_{j_1} = \frac{0.57+0.21}{0.57+0.21+0.43+0.30} = 0.52$; $NExp_{j_2} = \frac{0.43}{0.57+0.21+0.43+0.30} = 0.28$; and $NExp_{j_3} = \frac{0.30}{0.57+0.21+0.43+0.30} = 0.20$.

3.2 Computing the Implicit Value for the Step of Classification by Instances

Sometimes there is no explicit relations between two concepts, however, they could be actually very similar or close to each other via some implicit relations. For example (see Fig. 2), "Rembrandt van Rijn" is famous for his technique using strong contrast of light and dark shading, which in Italian corresponds to "Chiaroscuro"; "Francois van Bossuit" often took "Venus" as a subject to paint; and "Venus" in Roman refers to "Aphrodite" in Greek. Compared with the "obvious recommendations" via explicit relations, these implicitly related concepts might be surprisingly new/unknown to users. The main challenge is to define how close these two concepts are in the collection.

To address this issue, Issaac et al. [6] propose a method of instance-based ontology matching. The basic idea is that the more significant the overlap of artworks of two concepts is, the closer these two concepts are, and the level of significance is calculated by the corrected Jaccard measure, see Formula 4. In the formula, the set of instances described by a concept S is called the extension of S and abbreviate by S^i. The *JCcorr(S, T)* measures the fraction of the refinement (by choosing the factor of 0.8) of instances described by both concepts relative to the set of instances described by either one of the concepts [6].

$$JCcorr(S,\ T) = \frac{\sqrt{|S^i \bigcap T^i| \times (|S^i \bigcap T^i| - 0.8)}}{|S^i \bigcup T^i|} \qquad \text{(Formula 4: Corrected Jaccard measure)}$$

Adopting this method, we calculated the Corrected Jaccard values for all pairs of concepts in the collection. In general, the higher the Corrected Jaccard value is, the more common artworks these two concepts described. Below we give a brief look at the Corrected Jaccard values for some pairs of concepts:

0.96 (Sculptural studies – Terracotta models)
0.91 (unknown lacquerer – Lacquerware)
0.85 (Hermes – Mercury)
0.75 (Food and other objects – Still lifes with food)
0.63 (Militias – Militia paintings)

0.50 (Hinduism – Hindu deities)
0.40 (Still-life painting – Food and other objects)
0.30 (Drinking games – Sport and Games)
0.20 (Cupid – Love and Sex)
0.15 (Polychromy – Golden Legend)

0.10 (Rendering of texture – Woman)

There are in total 24249 pairs of concepts and the range of the Corrected Jaccard value is between 0 and 1. Looking at these values and checking the corresponding number of artworks the pair of concepts describe in common, we set 0.20 as a preliminary threshold, which might needs more refinement in the future. An example for the threshold 0.20 is "Cupid" and "Love and sex", which describe 8 artworks in common out of 40 artworks that are described by either one of these two concepts. In comparison, the Corrected Jaccard value between "Rendering of texture" and "Woman" is 0.10 and they describe 4 artworks in common out of 41 artworks.

After getting the Corrected Jaccard values for all concept pairs, we follow the same steps (Formula 1, 2 and 3) as the calculation of the explicit semantic value in Section 3.1. The only difference is that we use the Corrected Jaccard value to replace the original weight between two concepts and then normalize the Corrected Jaccard value in Formula 1. In the end, we will get a normalized implicit value $NImp_{(j)}$ for each implicitly related concept j.

3.3 Combining the Explicit and Implicit Values for the Step of Retrieval

Considering a related concept j could be linked to rated items via not only explicit relations but also implicit relations, we need to combine values from these two parts in order to get a final prediction $PreC_{(j)}$ for recommendation. Inspired by the work from Mobasher et. al [7], we set a parameter α to combine these two parts, see Formula 5. This combination parameter α measures the strength of the explicit and implicit components with respect to the current context. Taking two extreme examples: When α is 1, the system recommends items purely based on explicit relations and this will work well if the collection is well structured with rich semantic relations. When α is 0, it recommends items purely based on implicit relations which is suitable for recommender systems working on databases without semantic structures between concepts. Ideally, the parameter α could be manually set by the user, or dynamically adapted by the system, which enables the flexibility of the recommendation algorithm.

$$PreC_{(j)} = \alpha \times NExp_{(j)} + (1 - \alpha) \times NImp_{(j)} \qquad \alpha \in [0, 1]$$

(Formula 5: Final prediction for related concepts)

After collecting related concepts via both explicit and implicit relations, the system retrieves related artworks based on these related concepts. Since there are only explicit relations, which are artwork features between concepts and artworks, we only need to compute the normalized semantic value for related artworks, which is explained in details in Formula 3.

4 Evaluation and Discussion

In the evaluation, we use the existing user ratings collected from the previous user study [11]. There were 48 users that participated in this study. They used the CHIP Art Recommender to browse the semantically-enriched digital Rijksmuseum collection, which contains 729 artworks and 4320 art concepts. Each user rated 53 items (artworks and concepts) on average.

In the following sub-sections, we discuss how our approach behaviors in terms of (i) accuracy, (ii) serendipity, and (iii) explanation for recommendations, and we compare the results with those of the standard content-based recommendation strategy.

4.1 Influencing the Recommendation Accuracy

To measure the accuracy, we compute the standard Mean Absolute Error (MAE) by Leave-one-out cross validation [5]. It measures the average absolute deviation between ratings and predictions, using a single observation from the original sample as the validation data, and the remaining observations as the training data. This is repeated such that each observation in the sample is used once as validation data, see Fig. 4. Note that ratings in the CHIP Art Recommender are based on a 5-star scale, which refers to -1, -0.5, 0, 0.5 and 1. Thus, the maximum possible value for MAE is 2 and the minimum value is 0. The lower MAE values represent the higher recommendation accuracy.

$$\mathrm{MAE} = \frac{\sum_{i=1}^{N} |r_i - p_i|}{n}$$

leave-one-out cross validation

```
for each user
   for each item i
      withhold rating r
      compute prediction p from other ratings of user
      compute  r - p, and remember
   end for
end for

add up all | r - p |
divide this sum by the number of | r - p | values
```

Fig. 4. Compute the MAE by Leave-one-out cross validation

In order to see whether the semantic-enhanced content-based recommendation (SE-CBR) strategy in general improves or hamper the accuracy, we also measure the MAE for the standard content-based recommendation (CBR) strategy [10], which was applied in the previous version of our CHIP Art Recommender. The standard CBR takes the inference steps of realization and retrieval, but no classification by concepts and instances, which means that based on user rated items, standard CBR only recommends items via artwork features.

Although there are a number of variables influencing the MAE (e.g. the parameter α, the weights for explicit relations and the threshold for the Corrected Jaccard value), in this evaluation, we only look at the impact of α on MAE in order to get a first insight and we leave the experiment with other variables to future work. As we see, Fig. 5 shows the impact of Alpha (α) on MAE for SE-CBR and the MAE for the standard CBR. From these preliminary results, we observe that:

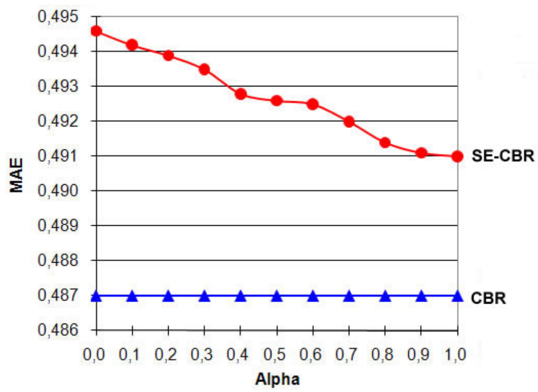

Fig. 5. MAE for Semantic-enhanced CBR (SE-CBR) and CBR

(i) Compared with the standard CBR (MAE is 0.487), the SE-CBR reaches a close value of MAE (between 0.491 and 0.495) with a slight average increase of 0.006. Results of our previous study [11] shows that by allowing semantic relations, the system bring two times more number of related items. Combining these findings, we see an indication that SE-CBR could find more related items for recommendations via various relations, without jeopardizing the recommendation accuracy.

(ii) The impact of α on MAE is not significant, with a small change of 0.004 from 0.495 (α is 0) to 0.491 (α is 1). This means that the accuracy of implicit recommendations is very close to the accuracy of explicit recommendations, which is surprisingly good.

The reason could be that we set a very high threshold (0.20) for the Corrected Jaccard value when selecting implicitly related items. Among all 24249 pairs of concepts in the collection, only 4% (1175 pairs) has the Corrected Jaccard value above 0.20 and most of these pairs are either synonyms or very similar to each other, e.g. "Unknown lacquerer"-"Lacquerware" and "Food and other objects"-"Still lifes with food". The high similarity ensures a high accuracy for implicit recommendations, which is close to the accuracy for explicit recommendations. In the future work, we would like to explore how the threshold of the Corrected Jaccard value influence the MAE. Considering the majority (75%: 18186 concept pairs) has the Corrected Jaccard values between 0.01 and 0.10, if we set a threshold in this low range, it will bring a lot of noisy recommendations, which might significantly decrease the recommendation accuracy.

(iii) The higher α is, the lower the MAE will be. It means that explicit recommendations improve the accuracy and implicit recommendations hamper the accuracy. This could be interpreted that implicit relations bring more unexpected/new recommended items but these items do not always fit the user's art preferences. In comparison, explicit semantic relations bring more reliable recommended items.

4.2 Providing serendipitous recommendations

As many researchers have argued [2][5], accuracy alone is not sufficient for selecting a good recommendation algorithm. A serendipitous recommendation helps a user find a surprising and new/unknown item that he/she might not have otherwise discovered. For example, if a user likes the famous Dutch painter "Rembrandt van Rijn", the standard CBR could only recommend the artwork "The Night Watch" via the artwork feature *creatorOf*. In comparison, the SE-CBR could recommend more items besides "The Night Watch". As illustrated in Fig. 6.(a), following the semantic relations between concepts, it finds two additional concepts "Baroque" (*style*) and "Pieter Lastman" (*studentOf*); and based on instance ontology matching, it finds an implicitly related concept "Chiaroscuro". Based on these concepts, it recommends more remotely-related artworks "The Marriage at Cana" and "Orestes and Pylades Disputing at the Altar".

Our previous study shows that compared with artwork features (e.g. *subject*), some specific semantic relations (e.g. *style*) offers surprisingly interesting and new recommendations for users [11]. To follow-up in this, it is indeed valuable to see, whether the implicitly related items are found by users also surprisingly interesting and new.

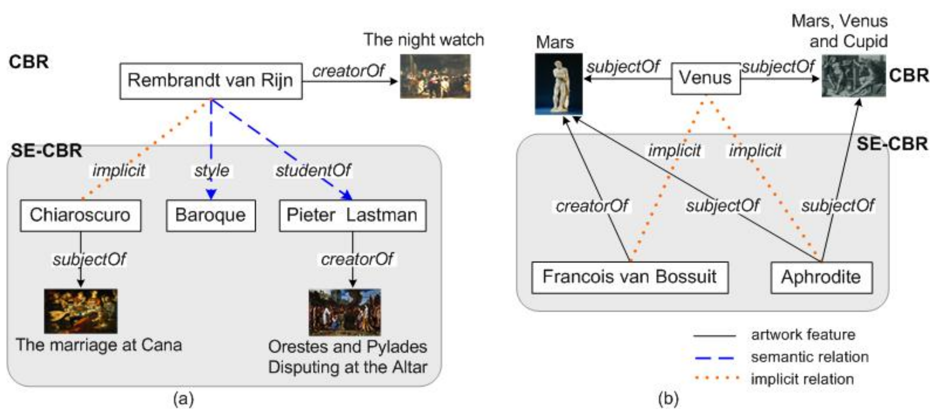

Fig. 6. Semantic-enhanced recommendations and explanations

4.3 Supporting more complete explanations

Besides the accuracy that affects user satisfaction, explanations of why an item was recommended also helps users gain confidence in the system's recommendations [5]. As shown in Fig. 6.(b), if a user likes "Venus", the standard CBR recommends two artworks "Mars" and "Mars, Venus and Cupid" via the artwork feature *subjectOf*. In the SE-CBR, it could find two implicitly related concepts "Francois van Bossuit" and "Aphrodite" based on instance ontology matching. And these two additional concepts are also linked to the recommended artworks: "Francois van Bossuit" is the *creatorOf* of "Mars", and "Aphrodite" is the *subjectOf* of "Mars" and "Mars, Venus and Cupid".

In the explanation of "Why recommend", these relations between the user's rated items and recommended items are automatically derived from the ontology. We also found previously that explanations for "Why recommend" are useful, especially for indirectly related or serendipitous recommendations [11]. In general, for content-based recommender systems, this type of explanation proved to be preferred by most users [5]. In such a way, the user could receive not only more recommended items, but also more complete explanations, which help them better understand the recommendations.

5 Conclusion

The main contribution of this paper is to provide reusable inference steps and components for content-based recommender systems, which are based on semantically-enriched collections. Using classification by concepts and instances, our approach brings about three improvement: (i) retrieving more explicitly and implicitly related items without jeopardizing the recommendation accuracy; (ii) providing serendipitous recommendations, which users find new and interesting; and (iii) supporting more complete explanations for recommended items, which users consider useful.

For classification by concepts, we applied various explicit relations (artwork features and semantic relations) from the semantically-enriched museum collection in order to find explicitly-related concepts and artworks. We derived a preliminary weight for each relation from previous study to compute a explicit value for each related concept.

For classification by instances, we adopted the method of instance-based ontology matching in order to find implicitly related concepts. Based on common instances, it builds an implicit relation between semantically-structured concepts and unstructured concepts. In this way, it bridges the vocabulary gap and provides serendipitous recommendations. We used the Corrected Jaccard value to compute an implicit value for each related concept.

To combine the explicit and implicit values for each related concept, we set a parameter α, which allows for the flexibility of the recommendation algorithm. In different domains or with different collections, the combination parameter α can be adjusted according to factors, such as how strong the semantic structure in the collection is, whether the user prefers more serendipitous recommendations than obvious recommendations, etc.

We regard our work as a first step towards a methodology for building recommender systems on the semantic web out of reusable knowledge elements. In the future work, we would like to further investigate the impact of different variables (e.g. weights for different semantic relations, the threshold for the Corrected Jaccard value) on the outcome of recommendations in the evaluation.

6 Acknowledgements

We greatly appreciate the contribution of Annette ten Teije from VU University Amsterdam. She provided the inspiration to our work and also actively involved in the discussion.

The CHIP (Cultural Heritage Information Personalization) project is funded by the Dutch Science Foundation funded program CATCH[7] (Continuous Access to Cultural Heritage).

References

1. Eva Blomqvist and Kurt Sandkuhl. Patterns in ontology engineering: Classification of ontology patterns. *ICEIS*.
2. Peter Brusilovsky, Alfred Kobsa, and editors Wolfgang Nejdl. *The Adaptive Web, Methods and Strategies of Web Personalization*. Springer, 2007.
3. E. Hyvönen, E. Mäkelä, M. Salminen, A. Valo, K. Viljanen, S. Saarela, M. Junnila and S. Kettula. MUSEUMFINLAND - Finnish Museums on the Semantic Web. *Journal of Web Semantics*, 3(2), 2005.
4. G. Schreiber, et al. MultimediaN E-Culture Demonstrator. In *Proc. of the International Semantic Web Conference (ISWC'06)*, 2006.
5. Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, John, and T. Riedl. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22:5–53, 2004.
6. Antoine Isaac, Lourens Van der Meij, Stefan Schlobach, and Shenghui Wang. An empirical study of instance-based ontology matching. In *6th International and 2nd Asian Semantic Web Conference (ISWC '07 + ASWC '07)*, pages 252–266, 2007.
7. Bamshad Mobasher, Xin Jin, and Yanzan Zhou. Semantically enhanced collaborative filtering. *In Web Mining: From Web to Semantic Web*, 3209, 2004.
8. Tuukka Ruotsalo and Eero Hyvönen. A method for determining ontology-based semantic relevance. In *DEXA*, 2007.
9. Frank van Harmelen, Annette ten Teije, and Holger Wache. Knowledge engineering rediscovered: towards reasoning patterns for the semantic web. In *5th International Conference on Knowledge Capture (K-CAP '09)*, pages 81–88, 2009.
10. Yiwen Wang, Natalia Stash, Lora Aroyo, Peter Gorgels, Lloyd Rutledge, and Guus Schreiber. Recommendations based on semantically-enriched museum collections. *Journal of Web Semantics*, 6(4):43–58, 2008.
11. Yiwen Wang, Natalia Stash, Lora Aroyo, Laura Hollink, and Guus Schreiber. Using semantic relations for content-based recommender systems in cultural heritage. In *Workshop on Ontology Patterns (WOP) at ISWC*, 2009.

[7] http://www.nwo.nl/catch