Semantic Pattern-based Recommender

Valentina Maccatrozzo, Davide Ceolin, Lora Aroyo and Paul Groth

The Network Institute
Department of Computer Science
VU University Amsterdam, The Netherlands
v.maccatrozzo@vu.nl, d.ceolin@vu.nl,l.m.aroyo@vu.nl, p.t.groth@vu.nl

Abstract. This paper presents a novel approach for Linked Data-based recommender systems by means of semantic patterns. We associate to each pattern the rating of the arrival book (0 or 1) and compute user profiles by aggregating, for each book in the user training set, the ratings of all the patterns pointing to that book. Ratings are aggregated by estimating the expected value of a Beta distribution describing the rating given to the book. Our approach allows the determination of a rating for a book, even if the book is poorly connected with user profile. It allows for a "prudent" estimation thanks to smoothing, obtained by using the Beta distribution. If many patterns are available, it considers all the contributions. Nevertheless, it allows for a lightweight computation of ratings as it exploits the knowledge encoded in the patterns. Without any setup of the system, this approach allowed us to reach a precision of 0.60 and an overall F-measure of about 0.52.

1 Introduction

Content-based recommender systems ground their approach on the characteristics of the items to be recommended. Items are more similar to each other the more characteristics they have in common. These similarity values are then used to recommend items to the users given their profiles. Our approach extends classical content-based approaches by using semantic patterns extracted from Linked Data sources. The central hypothesis is that there are patterns in the vast amount of structured (linked) data on the Web, that could be useful to link items and discover interesting paths for recommendation [5]. For instance we could link a book written by Ernest Hemingway with books written by Jack Kerouac, since the latter influenced the first.

We use semantic patterns for both building users profiles and for recommending items. We first extract all the semantic patterns which connect all pairs of books that the user rated, and then, for each pattern, we aggregate the ratings of the end book. For each book to be recommended, we consider all the patterns in the user profile that point at it, and we compute a smoothed average of the corresponding aggregated ratios (the smoothed average is, in fact, the expected value of a Beta probability distribution [9]).

Our approach allows easily the determination of a rating for a book, even if the book is poorly connected with the user profile. It allows for a "prudent" rating

estimation thanks to the smoothing performed by using the Beta probability distribution. Finally, it allows a lightweight computation of ratings as it exploits the knowledge encoded in the patterns. The lightweight computation is favored by the fact that the semantic patterns are extracted beforehand.

The novelty of our approach resides in the fact that we build user profiles in such a way that they guide the algorithm during the recommendation process. Basically, having patterns associated with ratings in the user profile allows the system to know which patterns to follow to provide relevant recommendations for that specific user.

The paper develops as follows. In Section 2, we provide an overview of the related work. In Sections 3 and 4, we present our method. Finally, in Section 5 and, 6 we provide the results of our approach and conclusions, respectively.

2 Related Work

The link between recommender systems and Linked Data has been explored by many researchers. For instance, Di Noia et al. [1], present a content-based recommender system based only on linked datasets. They propose a vector space model approach to compute similarities between RDF resources but do not make use of content patterns for the recommendation task. Fossati et al. [2] propose a news recommender systems based on entity linking techniques in unstructured text and knowledge extraction from structured knowledge bases. Their results show that using the entity relation approach provides the user with unexpected results and more specific explanations which attract the users' attention. Our work shares with these the type of sources used in the recommendation process. However, these works do not explore patterns as we do.

Our pattern-based approach is a follow-ups on the work of [11], and is inspired by [4]. It aligns most with the work of Passant who proposes *dbrec* [6], a recommender system built on top of DBpedia, which also introduces the notion of linked data semantic distance. Our approach also shares some similarities with the approach proposed by Sun et al. in [10] to define a path-based semantic similarity. The work by Peska and Vojtas [7] has interesting commonalities with ours, as they use the Czech DBpedia in order to retrieve non-trivial connections between items, although they do not explore longer paths.

Our definition of semantic patterns is inspired by Gangemi and Presutti [3], who introduced knowledge patterns to deal with semantic heterogeneity in the Semantic Web and to identify units of meaning. Presutti et al. use knowledge patterns to analyze Linked Data as a new level of abstraction that can be used for multiple purposes [8].

3 Semantic Pattern Extraction

A semantic pattern is a path that connects a source type T_1 to a target type T_{l+1} through pairs *property-type*. This can be defined as a set:

$$\{T_1, P_1, T_2, P_2, \ldots, T_l, P_l, T_{l+1}\}.$$

The length of a pattern is given by l. A pattern can be of a specific type, determined by the types involved in it, e.g. a people path. We can define both homogenous and heterogenous paths.

In this experiment, we extracted DBpedia patterns of length 1 and 2 between all the books in the training set. The patterns extraction was performed by means of SPARQL queries, like the one in Listing 1.1.

```
PREFIX db:<http://dbpedia.org/ontology/>.
    rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>.

SELECT DISTINCT ?prop1 ?v1 ?t2 ?prop2 WHERE
        {<Book1> ?prop1 ?v1 .
        ?v1 ?prop2 <Book2> .
        ?v1 rdf:type ?t2 .}
```

Listing 1.1: Query to retrieve patterns of length 2.

When building user profiles we store both the general pattern, e.g. "`http://dbpedia.org/ontology/country`, `http://dbpedia.org/ontology/Country`, `http://dbpedia.org/ontology/country`", and also the instantiation of the type of the entity involved in the pattern, e.g. "`http://dbpedia.org/resource/United_States`". In this way, we collect also a list of DBpedia resources which describe in more detail the user profile.

4 User Profiling and Rating Estimation

For each user, we consider all the books he rated, and we extract all the patterns that link them each other. Different instances of the same pattern may link elements with different ratings (in our case, ratings are boolean values). We focus on the rating of the end element of the pattern, because that is what we want to predict. In other words, given a known starting point (a book in the user profile), and using one of the patterns in the user profile, we want to be able to predict the rating of the item at the end of the pattern. For instance, in the pattern `http://dbpedia.org/ontology/country`, `http://dbpedia.org/ontology/Country`, `http://dbpedia.org/ontology/country` links the book `http://dbpedia.org/resource/Dragonfly_in_Amber` with `http://dbpedia.org/resource/The_Pelican_Brief`}. We associate the rating of the latter to the pattern, as we could recommend `http://dbpedia.org/resource/The_Pelican_Brief`} using it. We assume that the ratings of the recommendations made using that pattern can be inferred by the observations at our disposal. Thus we count the positive and the negative pieces of evidence (that is, all the observations, 1 and 0 rating respectively), and we associate the evidence counts to each pattern, to obtain the following mathematical function f that describes the evidence that is associated to each pair (*user*,*pattern*):

$$f : user \times pattern \rightarrow \{\#p, \#n\} \tag{1}$$

To estimate the rating of a newly proposed book, we proceed as follows. First, we identify all the patterns that link any of the user profile to that book. Then, we aggregate all the positive and negative evidence related to those patterns.

$$evidence(user_k, book_i) = \left\{ \sum_{j \in profile(user_k)} f(user_k, pattern_{j,book_i}) \right\} \quad (2)$$

where $pattern_{j,book_i}$ is the set of patterns which starting item j is in the profile of $user_k$ and which ending element is $book_i$. The function f defined above returns a couple of positive and negative counts ($\{\#p, \#n\}$). The sum $\sum$ is applied pairwise.

$evidence(user_k, book_i)$ returns a pair of aggregated positive and negative evidence counts. Based on these, we obtain a rating for $book_i$ by computing the expected value of a Beta probability distribution based on the evidence observed. Such a value is computed as follows:

$$rating(book_i) = \frac{\#p_i + 1}{\#p_i + \#n_i + 2} \quad (3)$$

where $\#p$ is the count of positive pieces of evidence, and $\#n$ is the count of negative ones. So, the rating is equal to a smoothed average, and the values 1 and 2 are due to the fact that the prior of the probability distribution is "neutral", that is, when no evidence is available, a pattern has 50% probability to end in a book rated 1 or 0.

The rating we estimate is a real number between zero and one. Although the initial ratings were boolean ones, and we could still discretize our results by rounding them, the challenge did not require us to do so.

The reasons why we choose this probability distribution are the following:

- the Beta probability distribution ranges between zero and one, that is, it models the probability for each value in the $[0, 1]$ interval to be the right value for the user rating associated to that pattern.
- the expected value of the Beta represents a smoothed ratio between positive and negative observations. Smoothing is important, because it allows us to avoid relying too heavily upon small evidence sets.

5 Results

We implemented our approach[1] to address Task 2 in the ESWC-14 Challenge: Linked Open Data-enabled Recommender Systems. We participated as "VUA group". Task 2 requested to calculate top-N recommendation from binary user feedback. We were asked to complete the user-item pairs in the evaluation data by providing the correspondent relevance score. These scores have been used to build a Top-5 recommendation list for each user and have been evaluated with the

[1] Code available online at `http://tinyurl.com/kslwyou`

F-measure@5. Our system reached Precision@5 equal to 0.6059, Recall@5 equal to 0.4497 and F-measure@5 equal to 0.5162 (values obtained by the evaluation system of the challenge).

6 Conclusions and Future Work

We presented an innovative approach to recommendation based on Linked Data that allowed us to achieve promising results. The plasticity of the method opens up for further exploration, and the lightness of the recommendation effort leaves us room for further computational extensions. For instance, we intend to add pattern selection based on user preferences. Moreover, we aim at extending the length of the patterns adopted in the recommendations, as to both increase the availability of evidence and improve the performance.

Acknowledgments. This research is supported by the EU FP7 STREP "ViSTA-TV" project and by the Dutch COMMIT Data2Semantics project.

References

1. T. Di Noia, R. Mirizzi, V. C. Ostuni, D. Romito, and M. Zanker. Linked open data to support content-based recommender systems. In *I-SEMANTICS '12*, pages 1–8. ACM, 2012.
2. M. Fossati, C. Giuliano, and G. Tummarello. Semantic Network-driven News Recommender Systems: a Celebrity Gossip Use Case. In *SeRSy*, pages 25–36. CEUR-WS.org, 2012.
3. A. Gangemi and V. Presutti. Towards a pattern science for the Semantic Web. *Semantic Web - Interoperability Usability Applicability*, 1:61–68, 2010.
4. L. Hollink, G. Schreiber, and B. Wielinga. Patterns of semantic relations to improve image content search. *J. Web Sem.*, 5(3):195–203, 2007.
5. V. Maccatrozzo, L. Aroyo, and W. R. van Hage. Crowdsourced Evaluation of Semantic Patterns for Recommendation. In *UMAP Workshops*. CEUR-WS.org, 2013.
6. A. Passant. Dbrec: Music Recommendations Using DBpedia. In *ISWC'10*, pages 209–224. Springer-Verlag, 2010.
7. L. Peska and P. Vojtás. Enhancing Recommender System with Linked Open Data. In *FQAS*, volume 8132, pages 483–494. Springer, 2013.
8. V. Presutti, L. Aroyo, A. Adamou, B. Schopman, A. Gangemi, and G. Schreiber. Extracting Core Knowledge from Linked Data. In *COLD*. CEUR-WS.org, 2011.
9. A. Prokhorov. Beta-distribution. In M. Hazewinkel, editor, *Encyclopedia of Mathematics*. Springer, 2001.
10. Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. *PVLDB*, 4(11):992–1003, 2011.
11. Y. Wang, N. Stash, L. Aroyo, L. Hollink, and G. Schreiber. Semantic relations for content-based recommendations. In *K-Cap 2009*, pages 209–210. ACM, 2009.