

Extending a Tag-based Collaborative Recommender with Co-occurring Information Interests

Noemi Mauro

Computer Science Department, University of Torino
Torino, Italy
noemi.mauro@unito.it

Liliana Ardissono

Computer Science Department, University of Torino
Torino, Italy
liliana.ardissono@unito.it

ABSTRACT

Collaborative Filtering is largely applied to personalize item recommendation but its performance is affected by the sparsity of rating data. In order to address this issue, recent systems have been developed to improve recommendation by extracting latent factors from the rating matrices, or by exploiting trust relations established among users in social networks.

In this work, we are interested in evaluating whether other sources of preference information than ratings and social ties can be used to improve recommendation performance. Specifically, we aim at testing whether the integration of frequently co-occurring interests in information search logs can improve recommendation performance in User-to-User Collaborative Filtering (U2UCF). For this purpose, we propose the *Extended Category-based Collaborative Filtering (ECCF)* recommender, which enriches category-based user profiles derived from the analysis of rating behavior with data categories that are frequently searched together by people in search sessions. We test our model using a big rating dataset and a log of a largely used search engine to extract the co-occurrence of interests. The experiments show that ECCF outperforms U2UCF and category-based collaborative recommendation in accuracy, MRR, diversity of recommendations and user coverage. Moreover, it outperforms the SVD++ Matrix Factorization algorithm in accuracy and diversity of recommendation lists.

CCS CONCEPTS

• **Information systems** → **Recommender systems**; • **Human-centered computing** → **Collaborative Filtering**;

KEYWORDS

Tag-based recommender systems; Collaborative Filtering; Category-based user profiles; Preference co-occurrence in information search.

ACM Reference format:

Noemi Mauro and Liliana Ardissono. 2019. Extending a Tag-based Collaborative Recommender with Co-occurring Information Interests. In *Proceedings of 27th Conference on User Modeling, Adaptation and Personalization, Larnaca, Cyprus, June 9–12, 2019 (UMAP '19)*, 10 pages.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

UMAP '19, June 9–12, 2019, Larnaca, Cyprus

© 2019 Copyright held by the owner/author(s). Publication rights licensed to Association for Computing Machinery.

ACM ISBN 978-1-4503-6021-0/19/06...\$15.00

<https://doi.org/10.1145/3320435.3320458>

<https://doi.org/10.1145/3320435.3320458>

1 INTRODUCTION

Recommender systems research has employed item ratings, bookmarking actions and other user activities as primary sources of information to generate personalized suggestions because they provide evidence about user preferences. In particular, User-to-User Collaborative Filtering [9] (henceforth, denoted as U2UCF) analyzes the ratings of items provided by users in order to identify “like-minded” people for preference prediction. However, the sparsity of the rating matrices affects recommendation performance. Thus, recent algorithms have been proposed to improve the recognition of preference similarity from rating data (e.g., Matrix Factorization algorithms [22] such as SVD++ [21]), possibly combined with trust information derived from the establishment of social links among users; e.g., [48, 55]. While these algorithms achieve good accuracy and coverage, they challenge the explanation of recommendation results because the policies applied to rank items can hardly be described in an intuitive way.

In the present work, we are interested in assessing whether U2UCF, which has nice explanation properties, can be improved by using other types of information that are complementary to rating data. Specifically, we investigate whether the identification of frequently co-occurring interests in information search can be used to improve recommendation performance. We start from the observation that, if the people who search for items tagged with a certain information category typically also search for items tagged with another category, the two categories might represent related interests. Therefore, even though we ignore the reasons behind this relatedness, we might leverage the strength of the association in preference estimation. In this perspective, we propose to build rich user profiles by extending the preferences for categories of items identified from rating behavior with frequently co-occurring interests for item categories, extracted from the logs of search engines. It can be noticed that interest co-occurrence can be learned by analyzing anonymous interaction sessions because it is aimed at describing general user behavior. Therefore, it can be applied to anonymized search logs, as long as search sessions can be identified.

Starting from a category-based representation of user preferences, based on the analysis of ratings and on items categorization, we propose the following research question:

RQ: How does the integration of data about interest co-occurrence in information search influence the performance of a collaborative recommender system that manages category-based user profiles?

In order to answer this question we start from a *Simple Category-based Collaborative Filtering (SCCF)* algorithm which infers a user's preferences on the basis of the distribution of her/his ratings on

item categories: a category-based user profile provides a conceptual view on preferences, so that user similarity can be computed by abstracting from item ratings, thus contrasting data sparsity; see [46, 47]. Then, we propose the *Extended Category-based Collaborative Filtering (ECCF)* algorithm that enriches category-based user profiles with evidence about interests that frequently co-occur in information search. ECCF employs the extended user profiles for rating estimation.

In order to evaluate the recommendation performance of ECCF, we extract information about co-occurring interests by analyzing the query log of a largely used search engine. Then, we test our algorithm by applying it to the Yelp Dataset [56], which stores user ratings of various types of businesses.

We analyze a few settings of ECCF in order to integrate different amounts of information about co-occurring preferences with rating data. In our experiments, we evaluate performance by taking U2UCF and SCCF as baselines: these algorithms differ in neighbor identification but are based on the same rating estimation approach. Therefore, they are a good basis to assess the impact of extended category-based user profiles on preference prediction. We also compare these algorithms with SVD++ to evaluate whether preference extension challenges the capability of recommending relevant items. The results of our experiments show that ECCF outperforms U2UCF and SCCF in accuracy, MRR, diversity of recommendations and user coverage; moreover it outperforms SVD++ in accuracy and diversity of the generated suggestion lists. We thus conclude that preference co-occurrence information can positively contribute to the identification of good neighbors for rating estimation.

In summary, the main contributions of this work are:

- The integration of data about frequently co-occurring information interests (inferred by observing general search behavior) with category-based user preferences, in order to acquire rich individual user profiles.
- The ECCF category-based recommendation algorithm, which extends User-to-User Collaborative Filtering to take both frequently co-occurring information interests and preference similarity into account in neighbor identification.
- Evaluation results aimed at proving the benefits of frequently co-occurring interests to Collaborative Filtering.

In the following, Section 2 positions our work in the related one. Section 3 presents ECCF. Section 4 describes the experiments we carried out to validate ECCF and discusses the evaluation results. Section 5 concludes the paper and outlines our future work.

2 RELATED WORK

2.1 Recommender Systems

Cross-domain recommendation has received the researchers' attention as a way to employ multiple information sources to contrast data sparsity; e.g., [12]. Moreover, holistic user models have been developed that jointly analyze different types of user behavior to enhance the recognition of the user's needs; e.g., [36, 49]. However, the fusion of personal information from different applications is problematic, unless it is done within a tightly integrated software environment. For instance, most people operate anonymously [16] or have multiple identities [10]; moreover, most user activity logs

are anonymized for privacy preservation purposes. It is thus interesting to consider other types of knowledge integration that do not require user identification across applications. Our work investigates this path of research.

Collaborative Filtering generates suggestions by analyzing item ratings to identify similar users or similar items. Several algorithms have been developed, from K-Nearest Neighbors (KNN) to more recent ones such as Matrix Factorization [9, 22]. In our work we adopt KNN because it has nice explanation capabilities and has proved to achieve good performance in a comparison with other approaches [19, 26].

Ontological user profiles model preferences at the semantic level. In [46, 47], Sieg et al. propose to exploit a taxonomy whose concepts represent item types, and to infer user interests on the basis of the observed ratings to the instances of such concepts. The neighborhood for rating estimation is then identified by measuring the semantic similarity between ontological user profiles. The category-based user similarity we propose is close to this approach. However, we go one step forward in the identification of preferences by extending the user profiles with frequently co-occurring information interests. This type of extension also differentiates our work from that of Ronen et al., who propose to extend the preferences of the individual user by analyzing her/his behavior in search logs [44]: that work assumes that the user's activities can be tracked across applications and extends the user profile by analyzing her/his overall behavior. In contrast, we extend user preferences by analyzing anonymous data about general search behavior.

Sen et al. define tag-aware recommender systems as "recommender algorithms that predict user's preferences for tags". In [45] they describe different signs of interest; e.g., searching or applying a tag, and so forth. Our work relates to tag-aware recommender systems because we analyze rating behavior on items associated to categories expressed as tags. However, we do not consider any other types of interaction with tags for estimating user preferences.

In [14], Gemmel et al. present a linear-weighted hybrid framework for resource recommendation that models different scenarios, among which tag-specific item recommendation. They propose to match users and items on the basis of their tag profiles. Differently, we match users on the basis of category-based profiles learned from rating behavior. The same kind of difference holds between our work and the one of Nakamoto [37].

While TagiCoFi [57] employs user similarities defined from tagging information to regularize Matrix Factorization, we use tags in a KNN algorithm. In [50] Tso and Sutter extend the ratings matrix using tagging information. They reduce the three-dimensional correlations $\langle user, tag, item \rangle$ to two-dimensional correlations $\langle user, tag \rangle$, $\langle item, tag \rangle$ and $\langle user, item \rangle$. Then, they apply a fusion method to combine the correlations for rating prediction. Differently, we extend the rating matrix with the categories (tags) associated to the items rated by users and with further categories identified from general search behavior.

Recently, rating information has been combined with other types of data to improve recommendation. For instance, item reviews are used, possibly in combination with ratings, in [7, 25, 31, 32]. Moreover, trust relations and reputation are used to steer recommendation on the basis of the feedback on items provided by trusted

parties; e.g., [1, 11, 23, 24, 30, 48, 55]. In [29], we investigate multifaceted trust for personalized recommendation. However, in the present work we focus on rating information to assess the potential improvement of Collaborative Filtering, when combined with general preference co-occurrence.

2.2 Analysis of Interaction Sessions

The identification of interest co-occurrence we propose is related to a few works supporting query expansion, query reformulation and term suggestion in Information Retrieval. Some researchers propose to analyze session-based user behavior in order to detect co-occurrence relations useful to improve search queries, taking the search context into account. For instance, in [6] Cao et al. suggest queries on the basis of the context provided by the user's recent search history, by clustering queries on the basis of the search results visited by users. Moreover, Huang et al. [17] and Chen et al. [8] detect term co-occurrence in search sessions to group sets of relevant words that can be mutually suggested. Our work is different because we adopt a linguistic interpretation approach (based on lemmatization and Word Sense Disambiguation) to find the concepts referenced in the queries; see [27]. Therefore, we extract information about *concept co-occurrence*, which is more general than *term co-occurrence*.

It is worth mentioning that our analysis of interaction sessions differs from session-based recommendation, which analyzes the user's behavior during an interaction session to identify relevant item(s) to suggest; e.g., see [13, 16, 19, 20]. In fact, we mine interest co-occurrence by abstracting from the particular sequence of queries performed by the users. Moreover, as previously discussed, we mine concept associations.

2.3 Graph-based Information Filtering

Knowledge graphs describe item features and relations among entities, supporting the analysis of item relatedness, as well as similarity for information filtering and top-N recommendation. In several works these graphs are extracted from document pools and/or from the Linked Data Cloud. For instance, CoSeNa [5] employs keyword co-occurrence in the corpus of documents to be retrieved, and ontological knowledge about the domain concepts, to support the exploration of text collections using a keywords-by-concepts graph. Moreover, in [38], Di Noia et al. create a relatedness graph by analyzing external data sources such as DBpedia in order to support the evaluation of semantic similarity between items. Analogously, item features have been extracted from the Linked Data Cloud to improve recommendation performance in [33–35, 41].

Some works attempt to extend the relations among information items by integrating data derived from the observation of different types of user behavior. E.g., Google search engine manages the Knowledge Graph [15] to relate facts, concepts and entities depending on their co-occurrence in queries. Moreover, entity2rec learns user-item relatedness from knowledge graphs by analyzing data about users' feedback and item information from Linked Open Data [40]. Furthermore, in [39] Oramas et al. propose a hybrid recommender that integrates users implicit feedback into a knowledge graph describing item information, enriched with semantic data extracted from external sources. Finally, in [51], Vahedian et al.

generalize graph-based approaches by simultaneously taking into account multiple types of relations among entities: they introduce meta-paths to represent patterns of relations and apply random-walk along such paths to identify relevant entities to suggest.

Our work has analogies to the above listed ones because we employ a graph-based type of knowledge representation. However, we work at the conceptual level: our knowledge graph relates item categories instead of individual users and/or items. Moreover, we do not compute similarity or relatedness by means of the knowledge graph: we use the graph to extend category-based user profiles. In turn, those profiles are employed in neighborhood identification. The separation between how preferences are inferred and how they are used for recommendation makes it possible to extend both types of activities in a modular way.

3 EXTENDED CATEGORY-BASED COLLABORATIVE FILTERING

We describe ECCF incrementally, starting from U2UCF that provides the basic match-making approach for rating estimation.

3.1 User-to-User Collaborative Filtering

In [42], Ricci et al. define U2UCF as follows: "the simplest and original implementation of this approach recommends to the active user the items that other users with similar tastes liked in the past. The similarity in taste of two users is calculated based on the similarity in the rating history of the users". Given:

- U as the set of users and I as the set of items;
- $r : UXI \Rightarrow \mathbb{R}$ as a map of ratings;
- $R \in \mathbb{R}^{UXI}$ as the users-items rating matrix, where each value is a rating $r_{ui} = R[u, i]$ given by a user $u \in U$ to an item $i \in I$.

The recommender system estimates u 's rating of i (\hat{r}_{ui}) as follows:

$$\hat{r}_{ui} = \bar{r}_u + \frac{\sum_{v \in N_i(u)} \sigma(u, v)(r_{vi} - \bar{r}_v)}{\sum_{v \in N_i(u)} |\sigma(u, v)|} \quad (1)$$

where $N_i(u)$ is the set of neighbors of u that rated item i and $\sigma(u, v)$ is the similarity between user u and user v ($v \in N_i(u)$). The similarity among users is computed by applying a distance metric, e.g., Cosine or Pearson similarity, to their rating vectors.

3.2 Simple Category-based Collaborative Filtering (SCCF)

SCCF manages user profiles in which the user's interest in each item category is represented as a positive number; the higher is the value, the stronger is the interest. We define:

- U, I, r and R as above;
- C as the set of item categories;
- $f : UXC \Rightarrow \mathbb{N}$ as a map between users and categories;
- $UC \in \mathbb{N}^{UXC}$ as the Users-Categories matrix. For each $u \in U$ and $c \in C$, $UC[u, c]$ represents the interest of u in c . We take as evidence of interest the *frequency of exploration* of a category, i.e., the frequency of interaction of the user with items associated with the category.

Category exploration can be mapped to different types of user behavior; e.g., tagging items and searching for items by tag. We map exploration to rating behavior and we define $UC[u, c]$ as the number of ratings that u has given to the items associated with c .

SCCF computes user similarity on the basis of the estimated user preferences for item categories. Specifically, $\sigma(u, v)$ is defined as the Cosine similarity of the users vectors in the UC matrix and it is used in Equation (1) to estimate ratings. Thus, \hat{r}_{ui} is computed on the basis of the ratings r_{vi} provided by the users $v \in U$ whose preferences for categories are similar to those of u .

3.3 Acquisition of Preferences Co-occurrence

In order to learn the strength of the associations between item categories in search behavior, we analyze their co-occurrence in the search sessions of a query log. By co-occurrence we mean the fact that two or more categories are referred by the queries belonging to the same session. In the following we summarize the analysis of category co-occurrence; see [28] for details.

The Category Co-occurrence Graph (CCG) represents category co-occurrence: in the CCG , nodes represent the data categories referenced in the analyzed queries and the weight of edges represents the co-occurrence frequency of the connected categories; i.e., how many times the categories have been identified within the same search sessions.

We retrieve the categories occurring in the queries by applying a Natural Language approach that identifies the referred concepts in a flexible way, by considering synonyms and by applying Word Sense Disambiguation to resolve the meaning of words; see [2, 27]. For Word Sense Disambiguation we use the Babelfy tool [3].

The CCG is built as follows: given two categories x and y , the weight of the edge that connects them is defined as:

$$w_{xy} = \sum_{S \in |Sessions|} Freq_{S_{xy}} \quad (2)$$

where $Freq_{S_{xy}}$ represents the evidence provided by session S to the co-occurrence frequency of x and y . Given $S = \{Q_1, \dots, Q_n\}$, $Freq_{S_{xy}}$ is computed as the maximum evidence of co-occurrence of x and y in S :

$$Freq_{S_{xy}} = \text{Max}_{k=1}^n (Freq_{xy_{Q_k}}, ev_{xy_{Q_{k-1}}}) \quad (3)$$

where $Freq_{xy_{Q_k}}$ is the co-occurrence evidence of x and y provided by query Q_k , and $ev_{xy_{Q_{k-1}}}$ is the one provided by Q_1, \dots, Q_{k-1} . Similar to [28], we take the maximum, and not the sum of evidence because co-occurrence could derive either from query reformulation [43], or from the repetition of queries in click-through events of the log; see Section 4.2 that describes the query log we used.

A query Q contributes to the estimation of co-occurrence as follows:

- If Q contains k terms ($k \geq 0$), each one identifying a non-ambiguous category: $T_1 \Rightarrow c_1, \dots, T_k \Rightarrow c_k$, then, for each category c of Q :
 - The co-occurrence evidence between c and every other category d of Q is $Freq_{cd_Q} = 1$.
 - The co-occurrence evidence between c and every other category e identified in a non-ambiguous way in the other queries of S is $Freq_{ce_Q} = 1$.

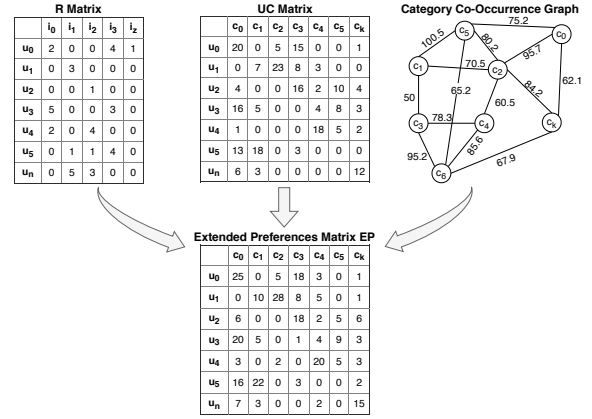


Figure 1: Extension of Category-based User Profiles.

- The co-occurrence evidence between any other categories w and z identified in S is $Freq_{wz_Q} = 0$.
- If Q contains an ambiguous term t that refers to m categories, the particular category the user is focusing on cannot be identified. Therefore, the co-occurrence evidence brought by t is computed as above, but the assigned evidence is $\frac{1}{m}$ in order to consider the possible interpretations of Q , and divide evidence among ambiguous categories.

3.4 Extended Category-based Collaborative Filtering (ECCF)

In this recommendation model we employ frequent co-occurring information interests to extend category-based user profiles. We reinforce the preferences for item categories learned by analyzing rating behavior (stored in the Users-Categories matrix UC) with interest co-occurrence associations (stored in the CCG graph) in order to acquire an extended set of user preferences for neighbor identification.

The idea behind preference extension is that, the more the user has appreciated the items of a category, the more interest co-occurrence makes sense. Therefore, starting from the category-based user profiles stored in the UC matrix, we increment user preferences with the contribution of the strongest co-occurrence relations of the CCG graph, depending on the number of positive ratings available in the users-items matrix R . The output of this process is stored in the Extended Preferences matrix EP , which is used to compute $\sigma(u, v)$ in Equation 1.

Figure 1 provides a graphical view of the computation of EP : the information stored in UC is combined with that stored in the CCG to set the values of this matrix. In this process, the users-ratings matrix R is used to limit the reinforcement of preferences to the categories of the positively rated items. Moreover, the CCG is used to propagate preference information according to the strongest co-occurrence of interests. In detail, we compute the values of EP as follows:

- let Cat_i be the set of categories associated to item i ;
- let $CatSet_i$ be the set of categories directly connected to any category $c \in Cat_i$ in the CCG through the heaviest outbound

Table 1: Statistics about the Filtered Datasets

Yelp	Number of users	26,600
	Number of businesses	76,317
	Number of ratings	1,326,409
AOL	Number of sessions	1,248,803
	Number of queries	2,136,029

arcs. These are the categories which most frequently co-occur with some categories of Cat_i in search sessions.

Then:

$$EP[u, c] = UC[u, c] + \sum_{i \in |I|} f(u, i, c) \quad (4)$$

where

$$f(u, i, c) = \begin{cases} 1 & \text{if } R[u, i] \in \text{PositiveRatings} \wedge c \in \text{CatSet}_i \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

In Equation 5 *PositiveRatings* denotes the set of ratings that are considered as positive in the dataset; e.g., {5}, or {4, 5} in a [1, 5] Likert scale.

4 VALIDATION OF ECCF

4.1 Dataset of Item Ratings

As a source of rating data we exploit the Yelp Dataset [56], which contains information about a set of businesses, users and reviews and is available for academic purposes. In the dataset, item ratings take values in a [1, 5] Likert scale where 1 is the worst value and 5 is the best one. Moreover, each item is associated with a list of categories describing the kind of service it offers.

The full list of Yelp categories is available at www.yelp.com/developers/documentation/v3/category_list and is organized in a taxonomy to specify businesses at different levels of detail. The taxonomy includes a large set of first-level categories, representing broad types of businesses; e.g., “Active life”, “Arts & entertainment”, “Automotive”, ..., “Food”, “Restaurants”, and many others. In turn, the first-level categories are specialized into sub-categories; e.g., “Restaurants” includes many types of restaurants such as “Indian”, “Chinese” and the like. We apply two filters to the dataset:

- (1) We select all the Yelp categories that are subclasses of “Restaurants” or “Food”: e.g., “Indian”, “Chinese”, “Cafes”, “Kebab”, “Pizza”, and so forth; the total number of categories is 254. Then, we project the Yelp dataset on the set of items associated with at least one of these categories. In the rest of this paper we refer to this set of categories as *CATS*.
- (2) We further filter the dataset on the users who rated at least 20 items.

The higher portion of Table 1 summarizes the number of users, businesses and ratings of the filtered Yelp dataset.

4.2 Dataset of Search Sessions

For the generation of the Category Co-occurrence Graph we use the AOL query log.¹ Each line of the log represents either a query

or a click-through event on one of the search results of a query. The line contains various fields, among which the submitted query and the submission date and hour.

In order to build a graph that is thematically related to the items of the filtered Yelp dataset, we select from the log the search sessions relevant to the categories $c \in \text{CATS}$ enriched with the following two types of external knowledge. The enrichment is useful to abstract from the specific category names used in Yelp and to take into account semantically related information:

- (1) *Lemmatized knowledge*: we enrich each element $c \in \text{CATS}$ with a set of keywords and synonyms from WordNet [53] lexical database.
- (2) *Relevant terms from the Probase [54] taxonomy*:
 - For each element $c \in \text{CATS}$, we enrich c with the $\langle \text{concept}, \text{instance} \rangle$ pairs of ProBase such that *concept* has at least 85% WordNet similarity with any term of the lemmatized knowledge of c , and the WordNet similarity between the two components of the pair is 85%.
 - ProBase, recently called Microsoft Concept Graph, is a large concept network harnessed from web pages and search logs. It is organized as a list of $\langle \text{instance}, \text{concept} \rangle$ pairs related by a subclass relation and it contains 5,376,526 classes and 12,501,527 instances.

For the selection of relevant search queries in the AOL log we match the lemmatized words occurring in the queries to the enriched categories of *CATS*. If there is at least one match between a term and a query, we consider the query as relevant and we include its parent session in the filtered log.

We identify the search sessions by aggregating the queries performed by the same user according to their temporal proximity, following the widely applied rule that two consecutive queries belong to different sessions if the time interval between them exceeds half an hour; see [52].

The lower portion of Table 1 shows the number of sessions and queries of the filtered AOL dataset.

It is worth noting that the AOL log was involved in an information leak issue but we decided to use it for two reasons. Firstly, our analysis is ethically correct because we study general search behavior to acquire aggregate data abstracting from the search histories of individual users. Secondly, to the best of our knowledge, the AOL log is the only available large dataset that reports textual search queries, and which can therefore be used for linguistic interpretation. We analyzed some public datasets but they did not meet our requirements. For instance, the Excite query dataset² contains about 1M queries while AOL log contains 20M queries. Moreover, in the Yahoo dataset³ the queries are coded; thus, it is not possible to extract any linguistic information to learn category co-occurrence.

4.3 Category Co-occurrence Graph

We instantiate the *CCG* with the interests that co-occur in the sessions of the filtered AOL dataset by applying the procedure described in Section 3.3. The resulting graph is strongly connected: almost all of the categories are linked to each other by an edge having weight > 0 . However, the distribution of weights in the

¹The log is available at <http://www.cim.mcgill.ca/~dudek/206/Logs/AOL-user-ct-collection/>.

²<https://svn.apache.org/repos/asf/pig/trunk/tutorial/data/>

³<https://webscope.sandbox.yahoo.com/catalog.php?datatype=l&did=50>

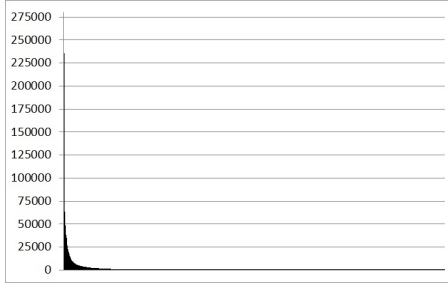


Figure 2: Distribution of the Weight of Edges in the CCG.

graph shows that there is a large number of weakly connected categories and a very small number of strongly associated ones. The “heavy” edges identify the interests that co-occur very frequently in search sessions and suggest to select the arcs having maximum weight in the CCG for the extension of the user profiles, as done in Section 3.4. Figure 2 shows this distribution; the x-axis represents the edges of the graph, and the y-axis represents their weights, which take values in $[1, 272224]$.

4.4 Test Methodology

We evaluate the recommendation performance of ECCF by comparing it to U2UCF and SCCF, which we consider as baselines. Moreover, we compare these algorithms with SVD++ in order to assess the improvement in the suggestion of relevant items given by frequently co-occurring interests.

The SCCF and ECCF recommendation algorithms are developed by extending the Surprise library [18], while we use the default Surprise implementations of U2UCF and SVD++.

We test the algorithms by applying a 10-fold cross-validation on the filtered Yelp dataset, after having randomly distributed ratings on folds: we use 90% of the ratings as training set and 10% as test set. In all the tests, we configure the KNN algorithms to work with 50 neighbors.

In order to analyze the impact on recommendation performance of a looser, or stricter extension of user preferences with category co-occurrence, we validate ECCF on different settings of *PositiveRatings* in Equation 5, i.e., on different interpretations of what is a good rating. For each fold we generate three versions of

the Extended Preferences matrix EP having set *PositiveRatings* to $\{3, 4, 5\}$, $\{4, 5\}$, and $\{5\}$ respectively.

We evaluate Top-k recommendation performance with $k=10$ and $k=20$ by taking the ratings observed in the Yelp dataset as ground truth. For the evaluation we consider the following metrics: Precision, Recall, F1, RMSE, MRR, Diversity and User Coverage.

Diversity describes the mean intra-list diversity of items in the suggestion lists @k; see [4]. In this work, we interpret diversity from the viewpoint of item classification. Therefore, we measure the diversity of a recommendation list as follows:

$$\text{intra-list diversity}@k = \frac{\sum_{i=1}^k \sum_{j=i}^k (1 - \text{sim}(i, j))}{\frac{k*(k+1)}{2}} \quad (6)$$

where $\text{sim}(i, j)$ is the cosine similarity between the lists of categories associated to items i and j in the ratings dataset.

4.5 Results

Table 2 shows the performance results of the KNN recommenders we compared, by taking into account a maximum of 10 suggested items (performance@10).

- **Precision:** similar to previous results described in [46], all of the category-based recommenders outperform U2UCF. This can be explained by the fact that the matrices describing preferences for item categories are denser than the ratings one. Thus, they improve recommendation by supporting a better identification of neighbors for Equation 1. However, SCCF outperforms all of the ECCF variants. The second best recommender is ECCF{5} that extends user profiles in the strictest way: it only considers as pivots for extension the categories associated to the items that the user has rated 5 stars. Notice also that the precision of ECCF decreases when *PositiveRatings* is lax. The reason is that the extension of user profiles with frequently co-occurring interests can increase the estimated interest in some noisy categories with respect to the pure observation of ratings distribution on categories. In particular, noise grows when the policy applied to extend preferences is less restrictive.
- **Recall:** ECCF outperforms the baselines in all the settings of *PositiveRatings*. Specifically, ECCF{4,5} achieves the best result, while recall is lower in ECCF{3,4,5} and further decreases in ECCF{5}. We explain this finding as follows: an extension of user profiles based on the categories of highly rated items supports the identification of a richer set of user preferences, and a more efficacious identification of neighbors, than only considering rating distribution on categories. However, if we restrict *PositiveRatings* too much, the user profiles are not extended enough to sensibly improve Recall. Moreover, as noticed for Precision, if *PositiveRatings* is lax, noise in the estimation of user preferences challenges neighbor selection.
- **F1:** ECCF outperforms the baselines. In detail, ECCF{4,5} achieves the best F1 = 0.7691; moreover, F1 varies consistently with Recall, depending on *PositiveRatings*.
- **RMSE:** SCCF reduces the mean error between estimated and observed ratings with respect to the baseline, showing the benefits of category-based user profiles. Moreover,

Table 2: Performance Evaluation @10; the Best Values Are in Boldface, the Worst Ones Are Strikethrough

Metrics	U2UCF	SCCF	ECCF {3,4,5}	ECCF {4,5}	ECCF {5}
Precision	0.7823	0.786	0.7857	0.7855	0.7859
Recall	0.7473	0.7526	0.7536	0.755	0.7529
F1	0.7644	0.7689	0.7693	0.7699	0.769
RMSE	1.0001	0.9899	0.9897	0.9893	0.9892
MRR	0.733	0.7367	0.737	0.7391	0.7384
Diversity	0.3042	0.3053	0.3056	0.3053	0.3049
User cov.	0.8497	0.8521	0.8526	0.8542	0.8534

Table 3: Performance Evaluation @20

Metrics	U2UCF	SCCF	ECCF {3,4,5}	ECCF {4,5}	ECCF {5}
Precision	0.7806	0.7842	0.7839	0.7838	0.7842
Recall	0.757	0.7624	0.7634	0.7649	0.7626
F1	0.7686	0.7731	0.7735	0.7742	0.7732
RMSE	0.9935	0.9838	0.9835	0.9832	0.9832
MRR	0.733	0.7369	0.7372	0.7391	0.7384
Diversity	0.3059	0.307	0.3073	0.307	0.3067
User cov.	0.8497	0.8521	0.8526	0.8542	0.8534

consistently with the variation of Precision, the best results are obtained by ECCF{5}, i.e., with a strict extension of user profiles. RMSE progressively increases (i.e., gets worse) for $PositiveRatings = \{4, 5\}$ and $\{3, 4, 5\}$.

- **MRR:** ECCF outperforms the baselines. Specifically, ECCF{4,5} obtains the best MRR = 0.7391. The second best value corresponds to a more selective extension of user profiles in ECCF{5}; moreover, if $PositiveItems = \{3, 4, 5\}$ results get worse.
- **Diversity:** both SCCF and ECCF outperform U2UCF. In this case, the best results are obtained with a lax extension of user preferences (ECCF{3,4,5}) and Diversity decreases while the preference extension policy becomes stricter. We explain these findings with the fact that category-based user profiles improve the estimation of user preferences concerning a variegated set of item categories, with respect to a flat recommendation based on ratings. However, the stricter is the extension of user preferences, the less item categories are used in neighbor identification.
- **User coverage:** ECCF outperforms the baselines, confirming the usefulness of preference extension. However, the selection of the ratings for the extension influences coverage: ECCF{4,5} achieves the best results by suggesting at least one relevant item to 85.42% of the users, against 84.97% of U2UCF. The second best is ECCF{5} and ECCF{3,4,5} has the worst results.

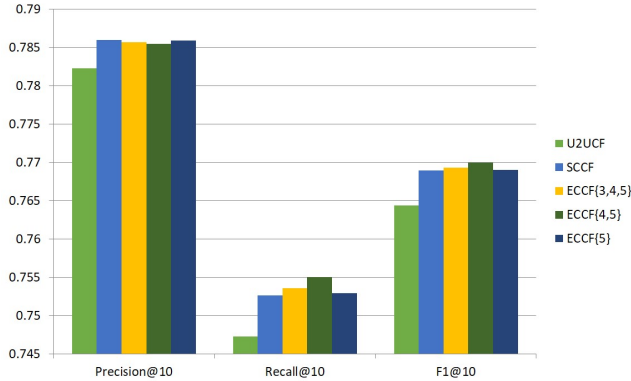


Figure 3: Graphical Representation of Accuracy@10.

In the described experiments the *EP* Matrix is defined by only taking into account positive ratings. In order to get a broader view on the performance of ECCF, we also consider its application to all the user ratings; i.e., we set $PositiveRatings$ to $\{1, \dots, 5\}$. With respect to the previous results, in this case the algorithm achieves similar Precision but lower Recall (0.7524), MRR (0.7369) and User coverage (0.8155).

Table 3 shows the results obtained by comparing performance@20. These results confirm the usefulness of category-based user profiles and of their extension with frequently co-occurring information interests:

- Also in this case, ECCF{4,5} is the best recommendation algorithm. It outperforms the others in Recall, F1, MRR and User coverage. Moreover both ECCF{5} and ECCF{4,5} achieve the best RMSE in comparison with the other recommenders.
- However, while SCCF has the best Precision@10, both SCCF and ECCF{5} achieve the best Precision@20.

With respect to $k=10$, Precision@20 is lower while Recall@20 and F1@20 take higher values; this makes sense because we are considering longer suggestion lists. Moreover, RMSE@20 is lower, which tells us that the longer lists contain proportionally less errors in the estimation of ratings. Differently, most algorithms obtain the same MRR for $k=10$ and $k=20$ (except for SCCF and ECCF{3,4,5}): this shows that the first relevant item is almost always placed in the first 10 positions of the suggestion lists. Furthermore, the Diversity@20 has the highest values for all the recommenders: this might be due to the fact that the longer suggestion lists have more chances to include items belonging to different categories. Finally, User coverage@10 = User coverage@20 because we interpret coverage as the percentage of users who receive at least one suggestion.

Figures 3 and 4 depict the accuracy @10 and @20:

- All of the category-based recommenders outperform U2UCF, confirming the benefits of the introduction of category-based preferences in KNN Collaborative Filtering. The conceptual representation of user preferences generally improves performance because the matrices describing user preferences (*UC* and *EP*) are denser than the users-items matrix storing ratings (*R*). Therefore, better neighbors can be identified for the computation of Equation 1.

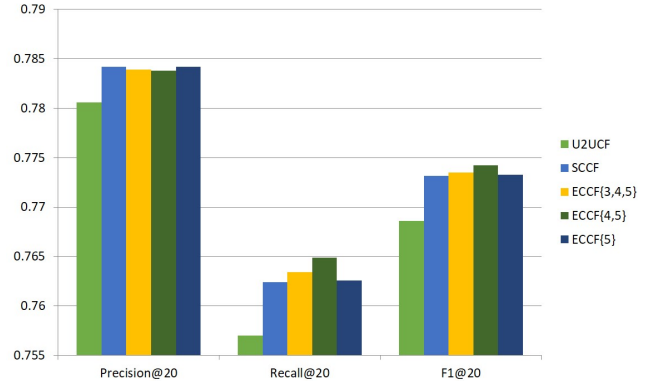


Figure 4: Graphical Representation of Accuracy@20.

- A comparison between category-based algorithms shows that the best performance results are obtained by extending user profiles on the basis of the items that users have rated very well, i.e., with 4 or 5 stars in a [1, 5] Likert scale. If the items that received middle ratings are considered as well, accuracy decreases.
- The category-based representation of user profiles has positive impact on the Diversity of recommendation lists. Conversely, the extension of user profiles does not further help this aspect, unless user profiles are extended in a lax way. However, a lax extension is not convenient because it decreases other measures.

In order to assess the usefulness of preference extension in Top-k recommendation, we also compare the previously described algorithms with SVD++ [21], which adopts Matrix Factorization to learn latent user and item factors, basing rating prediction on the sole analysis of user ratings. The comparison results show that:

- SVD++ is more accurate than U2UCF and SCC. On the filtered Yelp dataset, SVD++ obtains $F1@10 = 0.7696$. This finding shows that the management of category-based user profiles helps recommendation but it can be outperformed by a deeper understanding of the features of items and users.
- SVD++ achieves similar accuracy results with respect to ECCF but it is outperformed by ECCF{4, 5}. Therefore, the extension of user profiles with frequently co-occurring information interests, integrated into a KNN recommender, improves accuracy and makes it comparable or higher than that of Matrix Factorization algorithms.
- ECCF outperforms SVD++ as far as the diversity of the recommendation lists is concerned: SVD++ has $Diversity@10 = 0.3041$; this is comparable to the diversity achieved by U2UCF and lower than that of all the category-based recommenders we presented.
- In contrast, SVD++ has the highest User coverage of all the algorithms (0.8709), showing its superior capability to contrast data sparsity.

4.6 Discussion

In summary, the evaluation results show that ECCF outperforms U2UCF, SCCF and SVD++ in accuracy and intra-list diversity. Moreover, it outperforms U2UCF and SCCF in MRR and user coverage, while SVD++ excels in the latter metric. The results also show that ECCF achieves the best results when applied to positive ratings, while its performance slightly decreases when the user profiles are extended by taking both positive and negative ratings.

These results support the hypothesis that preference extension, based on frequently co-occurring information interests, improves the accuracy of the suggestions generated by a KNN recommender system. However, research has to be carried out to improve other performance metrics, possibly also investigating the integration of preference co-occurrence in Matrix Factorization algorithms.

It might be questioned whether extending user profiles with general interest co-occurrence data might provide less personalized recommendations than, e.g., focusing the extensions on the user's neighborhood. In this respect, we point out that we aim at developing a model that does not depend on cross-domain user

identification. However, an investigation of this issue can be interesting to deal with the cases in which user information can be shared among the applications, or public information about the users can be connected to the local profiles; e.g., public data on social networks.

Before closing this discussion, it is worth noting that, even though the AOL query log dates back to 2006, it can be considered as a good information source as long as it is analyzed from the viewpoint of the concepts expressed by the users. In other words, while the specific information items mentioned in the log might not exist any more, the topics referred in the queries are general and long-lasting. Of course, some new topics (e.g., new types of restaurants) might have emerged since 2006, and maybe new concept associations could exist now. However, the described performance results show that the co-occurring interests we identified are useful to improve recommendation performance; moreover, the methodology described in this paper can be applied to other more recent datasets, if available.

5 CONCLUSIONS

We investigated whether the identification of frequently co-occurring interests in information search can be used to improve the performance of KNN collaborative recommender systems. For this purpose, we defined a preference extension model that, applied to a category-based representation of user profiles, infers user preferences by exploiting frequently co-occurring information interests. Then, we implemented the model in the Extended Category-based Collaborative Filtering algorithm (ECCF). This is a variant of User-to-User Collaborative Filtering that works on category-based user profiles, enriched with preferences inferred from general search behavior. For the analysis of user interests, we analyzed the query log of a largely used search engine.

We evaluated ECCF on a large dataset of item ratings, by applying different levels of strictness in the extension of user profiles. The evaluation showed that ECCF outperforms User-to-User Collaborative Filtering in accuracy, MRR, intra-list diversity and user coverage. Interestingly, ECCF also obtains higher accuracy and diversity than the SVD++ recommender system, based on Matrix Factorization; however, ECCF has lower user coverage than SVD++.

In our future work we will focus on the coverage aspect in order to improve the performance of KNN Collaborative Filtering. Moreover, we will carry out further experiments, considering (i) a broader domain than Restaurants and Food, on which we have focused our current work, and (ii) users who have provided few or zero ratings. We will also analyze other datasets to check whether the performance results described in this article can be generalized. Finally, we will compare the performance of ECCF with a larger set of recommendation approaches based on preference extension.

ACKNOWLEDGMENTS

This work was supported by the University of Torino through projects "Ricerca Locale", MIMOSA (MultiModal Ontology-driven query system for the heterogeneous data of a Smartcity, "Progetto di Ateneo Torino_call2014_L2_157", 2015-17) and the Computer Science PhD program. We are grateful to Zhongli Filippo Hu, who helped us filter the Yelp dataset.

REFERENCES

- [1] S. Alotaibi and J. Vassileva. 2016. Personalized Recommendation of Research Papers by Fusing Recommendations from Explicit and Implicit Social Networks. In *Proc. of IFUP 2016: Workshop on Multi-dimensional Information Fusion for User Modeling and Personalization*, Vol. 1618. CEUR, Halifax, Canada, paper 2.
- [2] L. Ardissono, M. Lucenteforte, N. Mauro, A. Savoca, A. Voghera, and L. La Riccia. 2016. Exploration of Cultural Heritage Information via Textual Search Queries. In *MobileHCI '16 Proceedings of the 18th Int. Conf. on Human-Computer Interaction with Mobile Devices and Services Adjunct*. ACM, 992–1001.
- [3] Babelify. Multilingual Word-sense disambiguation and entity linking together!
- [4] K. Bradley and B. Smyth. 2001. Improving Recommendation Diversity. In *Proc. of the 12th National Conference in Artificial Intelligence and Cognitive Science*, Diarmuid O'Donoghue (Ed.). Maynooth, Ireland, 75–84.
- [5] K.S. Candan, M. Cataldi, L. Di Caro, M.L. Sapino, and C. Schifanella. 2009. CoSeNa: a context-based search and navigation system. In *MEDES '09 Int. Conf. on Management of Emergent Digital EcoSystems*. Chia Laguna, Italy, Art. 33.
- [6] H. Cao, D. Jiang, J. Pei, Q. He, Z. Liao, E. Chen, and H. Li. 2008. Context-aware query suggestion by mining click-through and session data. In *Proc. of the 14th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD '08)*. ACM, New York, NY, USA, 875–883.
- [7] L. Chen, G. Chen, and F. Wang. 2015. Recommender systems based on user reviews: the state of the art. *USER MODELING AND USER-ADAPTED INTERACTION - The Journal of Personalization Research* 25, 2 (2015), 99–154.
- [8] Y. Chen, G-R Xue, and Y. Yu. 2008. Advertising Keyword Suggestion Based on Concept Hierarchy. In *Proc. of the 2008 Int. Conf. on Web Search and Data Mining (WSDM '08)*. ACM, New York, NY, USA, 251–260.
- [9] C. Desrosiers and G. Karypis. 2011. A Comprehensive Survey of Neighborhood-based Recommendation Methods. In *Recommender systems handbook*, F. Ricci, L. Rokach, B. Shapira, and P.B. Kantor (Eds.). Springer, 107–144.
- [10] D. Doychev, A. Lawlor, R. Rafter, and B. Smyth. 2014. An analysis of recommender algorithms for online news. In *Proc. of CLEF 2014 Conference and Labs of the Evaluation Forum*. Sheffield, UK.
- [11] Xixi Du, Huafeng Liu, and Liping Jing. 2017. Additive Co-Clustering with Social Influence for Recommendation. In *Proceedings of the Eleventh ACM Conference on Recommender Systems (RecSys '17)*. ACM, New York, NY, USA, 193–200. DOI: <http://dx.doi.org/10.1145/3109859.3109883>
- [12] I. Fernández-Tobías, P. Tomeo, I. Cantador, T. Di Noia, and E. Di Sciascio. 2016. Accuracy and Diversity in Cross-domain Recommendations for Cold-start Users with Positive-only Feedback. In *Proc. of the 10th ACM Conference on Recommender Systems (RecSys '16)*. ACM, New York, NY, USA, 119–122.
- [13] F. Garcin, C. Dimitrakakis, and B. Faltings. 2013. Personalized news recommendation with context trees. In *Proc. of 7th ACM Conf. on Recommender Systems (RecSys 2013)*. Honk Kong, China, 105–112.
- [14] J. Gemmel, T. Schimoler, B. Mobasher, and R. Burke. 2012. Resource recommendation in social annotation systems: a linear-weighted hybrid approach. *Journal of computer and system sciences* 78 (2012), 1160–1174.
- [15] Google. 2017. Knowledge Graph. https://www.google.com/intl/it_it/insidesearch/features/search/knowledge.html.
- [16] A. Greenstein-Messica, L. Rokach, and M. Friedman. 2017. Session-based recommendations using item embedding. In *Proc. of the 22nd Int. Conf. on Intelligent User Interfaces (IUI '17)*. ACM, New York, NY, USA, 629–633.
- [17] C.-K. Huang, L.-F. Chien, and Y.-J. Oyang. 2003. Relevant term suggestion in interactive web search based on contextual information in query session logs. *Journal of the American Society for Information Science and Technology* 54, 7 (2003), 638–649.
- [18] Nicolas Hug. 2018. Surprise, a Python library for recommender systems. <http://surpriselib.com>.
- [19] D. Jannach and M. Ludewig. 2017. When Recurrent Neural Networks Meet the Neighborhood for Session-Based Recommendation. In *Proc. of the Eleventh ACM Conference on Recommender Systems (RecSys '17)*. New York, NY, USA, 306–310.
- [20] D. Jannach, M. Ludewig, and L. Lerche. 2017. Session-based Item Recommendation in e-Commerce: On Short-term Intents, Reminders, Trends and Discounts. *User Modeling and User-Adapted Interaction* 27, 3-5 (2017), 351–392.
- [21] Yehuda Koren. 2008. Factorization Meets the Neighborhood: A Multifaceted Collaborative Filtering Model. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '08)*. ACM, New York, NY, USA, 426–434. DOI: <http://dx.doi.org/10.1145/1401890.1401944>
- [22] Y. Koren and R. Bell. 2011. Advances in collaborative filtering. In *Recommender systems handbook*, F. Ricci, L. Rokach, B. Shapira, and P.B. Kantor (Eds.). Springer, 145–186.
- [23] U. Kuter and J. Golbeck. 2007. SUNNY: A New Algorithm for Trust Inference in Social Networks Using Probabilistic Confidence Models. In *Proc. of the 22nd National Conference on Artificial Intelligence - Volume 2 (AAAI'07)*. AAAI Press, 1377–1382.
- [24] Fengkun Liu and Hong Joo Lee. 2010. Use of Social Network Information to Enhance Collaborative Filtering Performance. *Expert Syst. Appl.* 37, 7 (July 2010), 4772–4778. DOI: <http://dx.doi.org/10.1016/j.eswa.2009.12.061>
- [25] Y. Lu, R. Dong, and B. Smyth. 2018. Coevolutionary Recommendation Model: Mutual Learning Between Ratings and Reviews. In *Proc. of the 2018 World Wide Web Conference (WWW '18)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 773–782.
- [26] M. Ludewig and D. Jannach. 2018. Evaluation of Session-based Recommendation Algorithms. *User-Modeling and User-Adapted Interaction* (2018).
- [27] N. Mauro and L. Ardissono. 2017. Concept-aware Geographic Information Retrieval. In *Proc. of 2017 IEEE/WIC/ACM Int. Conf. on Web Intelligence (WI)*. ACM, Leipzig, Germany, 34–41.
- [28] N. Mauro and L. Ardissono. 2018. Session-based Suggestion of Topics for Exploratory Search. In *Proc. of ACM IUI 2018*. ACM, Tokyo, Japan, 341–352.
- [29] N. Mauro, L. Ardissono, and Z.F. Hu. 2019. Multi-faceted Trust-based Collaborative Filtering. In *Proc. of ACM UMAP 2019*. ACM, Larnaca, Cyprus, to appear.
- [30] Kevin McNally, Michael P. O'Mahony, and Barry Smyth. 2014. A Comparative Study of Collaboration-based Reputation Models for Social Recommender Systems. *User Modeling and User-Adapted Interaction* 24, 3 (Aug. 2014), 219–260. DOI: <http://dx.doi.org/10.1007/s11257-013-9143-6>
- [31] K. Muhammad, A. Lawlor, E. Rafter, and B. Smyth. 2015. Great Explanations: Opinionated Explanations for Recommendations. In *Case-Based Reasoning Research and Development*, Eyke Hüllermeier and Mirjam Minor (Eds.). Springer International Publishing, Cham, 244–258.
- [32] C.-C. Musat and B. Faltings. 2015. Personalizing product rankings using collaborative filtering on opinion-derived topic profiles. In *Proc. 24th IJCAI Buenos Aires, Argentina*, 830–836.
- [33] C. Musto, P. Basile, P. Lops, M. de Gemmis, and G. Semeraro. 2017. Introducing Linked Open Data in Graph-based Recommender Systems. *Inf. Process. Manage.* 53, 2 (2017), 405–435.
- [34] C. Musto, T. Franza, G. Semeraro, M. de Gemmis, and P. Lops. 2018. Deep Content-based Recommender Systems Exploiting Recurrent Neural Networks and Linked Open Data. In *Adjunct Publication of the 26th Conference on User Modeling, Adaptation and Personalization (UMAP '18)*. ACM, New York, NY, USA, 239–244.
- [35] C. Musto, P. Lops, M. de Gemmis, and G. Semeraro. 2017. Semantics-aware Recommender Systems exploiting Linked Open Data and graph-based features. *Knowledge-Based Systems* 136 (2017), 1 – 14.
- [36] Cataldo Musto, Giovanni Semeraro, Cosimo Lovascio, Marco de Gemmis, and Pasquale Lops. 2018. A Framework for Holistic User Modeling Merging Heterogeneous Digital Footprints. In *Adj. Publ. of UMAP '18*. ACM, New York, USA, 97–101.
- [37] Reyn Nakamoto, Shinsuke Nakajima, Jun Miyazaki, and Shunsuke Uemura. 2007. Tag-based contextual collaborative filtering. *IAENG Int. Journal of Computer Science* 34, 2 (2007).
- [38] T. Di Noia, V.C. Ostuni, J. Rosati, P. Tomeo, E. Di Sciascio, R. Mirizzi, and C. Bartolini. 2016. Building a Relatedness Graph from Linked Open Data. *Expert Syst. Appl.* 44, C (2016), 354–366.
- [39] S. Oramas, V.C. Ostuni, T. Di Noia, X. Serra, and E. Di Sciascio. 2015. Sound and music recommendation with knowledge graphs. *ACM Transactions on Intelligent Systems and Technology* 8, 2 (2015), Art. 21.
- [40] E. Palumbo, G. Rizzo, and R. Troncy. 2017. Entity2Rec: Learning User-Item Relatedness from Knowledge Graphs for Top-N Item Recommendation. In *Proc. of the Eleventh ACM Conference on Recommender Systems (RecSys '17)*. ACM, New York, NY, USA, 32–36.
- [41] A. Ragone, P. Tomeo, C. Magarelli, T. Di Noia, M. Palmonari, A. Maurino, and E. Di Sciascio. 2017. Schema-summarization in Linked-data-based Feature Selection for Recommender Systems. In *Proc. of the Symposium on Applied Computing (SAC '17)*. ACM, New York, NY, USA, 330–335.
- [42] F. Ricci, L. Rokach, and Bracha B. Shapira. 2011. *Introduction to Recommender Systems Handbook*. Springer US, Boston, MA, 1–35.
- [43] S.Y. Rieh and H. I Xie. 2006. Analysis of multiple query reformulations on the web: the interactive information retrieval context. *Information processing and management* 42 (2006), 751–768.
- [44] R. Ronen, E. Yom-Tov, and G. Lavee. 2016. Recommendations meet web browsing: enhancing collaborative filtering using internet browsing logs. In *Proc. of 2016 IEEE 32nd International Conference on Data Engineering (ICDE)*. Kelsinki, Finland, 1230–1238.
- [45] S. Sen, J. Vig, and J. Riedl. 2009. Tagommenders: Connecting Users to Items Through Tags. In *Proc. of the 18th International Conference on World Wide Web (WWW '09)*. ACM, New York, NY, USA, 671–680.
- [46] A. Sieg, B. Mobasher, and R. Burke. 2007. Web Search Personalization with Ontological User Profiles. In *Proc. of the 16th ACM Conf. on Information and Knowledge Management (CIKM '07)*. ACM, New York, NY, USA, 525–534.
- [47] A. Sieg, B. Mobasher, and R. Burke. 2010. Ontology-based collaborative recommendation. *Computing* (2010).
- [48] Jiliang Tang, Xia Hu, Huiji Gao, and Huan Liu. 2013. Exploiting Local and Global Social Context for Recommendation. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence (IJCAI '13)*. AAAI Press, 2712–2718. <http://dl.acm.org/citation.cfm?id=2540128.2540519>
- [49] J. Teevan, S.T. Dumais, and E. Horvitz. 2005. Personalizing Search via Automated Analysis of Interests and Activities. In *Proc. of the 28th Annual Int. ACM SIGIR*

- Conf. on Research and Development in Information Retrieval (SIGIR '05)*. ACM, New York, NY, USA, 449–456.
- [50] Karen H. L. Tso-Sutter, Leandro Balby Marinho, and Lars Schmidt-Thieme. 2008. Tag-aware Recommender Systems by Fusion of Collaborative Filtering Algorithms. In *Proc. of SAC '08*. ACM, New York, NY, USA, 1995–1999.
 - [51] Fatemeh Vahedian, Robin Burke, and Bamshad Mobasher. 2017. Multirelational Recommendation in Heterogeneous Networks. *ACM Trans. Web* 11, 3, Article 15 (2017), 34 pages.
 - [52] R.W. White, M. Bilenko, and S. Cucerzan. 2007. Studying the use of popular destinations to enhance web search interaction. In *Proc. of the 30th Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval (SIGIR '07)*. ACM, New York, NY, USA, 159–166.
 - [53] WordNet. 2017. WordNet - a lexical database for English. <https://wordnet.princeton.edu/>.
 - [54] W. Wu, H. Li, H. Wang, and K.Q. Zhu. 2012. Probase: A Probabilistic Taxonomy for Text Understanding. In *Proc. of the 2012 ACM SIGMOD Int. Conf. on Management of Data*. ACM, New York, NY, USA, 481–492.
 - [55] B. Yang, Y. Lei, J. Liu, and W. Li. 2017. Social collaborative filtering by trust. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39, 8 (2017), 1633–1647.
 - [56] Yelp. Yelp Dataset Challenge. https://www.yelp.com/dataset_challenge.
 - [57] Yi Zhen, Wu-Jun Li, and Dit-Yan Yeung. 2009. TagiCoFi: Tag Informed Collaborative Filtering. In *Proc. of RecSys '09*. ACM, 69–76.