

Content-based Tag Propagation and Tensor Factorization for Personalized Item Recommendation based on Social Tagging

DIMITRIOS RAFAILIDIS, Information Technologies Institute
APOSTOLOS AXENOPOULOS, Information Technologies Institute
JONAS ETZOLD, Fulda University of Applied Sciences
STAVROULA MANOLOPOULOU, Information Technologies Institute
PETROS DARAS, Information Technologies Institute

In this paper, a novel method for personalized item recommendation based on social tagging is presented. The proposed approach comprises a content-based tag propagation method, to address the sparsity and “cold start” problems, which often occur in social tagging systems and decrease the quality of recommendations. The proposed method exploits (a) the content of items and (b) users’ tag assignments through a relevance feedback mechanism, in order to automatically identify the optimal number of content-based and conceptually similar items. The relevance degrees between users, tags, and conceptually similar items are calculated, in order to ensure accurate tag propagation and consequently to address the issue of “learning tag relevance”. Moreover, the ternary relation among users, tags and items is preserved by performing tag propagation in the form of triplets based on users’ personal preferences and “cold start” degree. The latent associations among users, tags and items are revealed based on a tensor factorization model, in order to build personalized item recommendations. In our experiments with real world social data, we show the superiority of the proposed approach over other state-of-the-art methods, since several problems in social tagging systems are successfully tackled. Finally, we present the recommendation methodology in the multimodal engine of I-SEARCH, where user’s interaction capabilities are demonstrated.

Categories and Subject Descriptors: H.3.3 [Information Search-Retrieval]: Information Filtering

General Terms: Algorithms, Performance

Additional Key Words and Phrases: Social tagging, recommender systems, tensor factorization, tag propagation, content-based information retrieval, relevance feedback

ACM Reference Format:

Rafailidis D., Axenopoulos A., Etzold J., Manolopoulou S., Daras P. 2012. Content-based Tag Propagation and Tensor Factorization for Personalized Item Recommendation based on Social Tagging. ACM Trans. Interact. Intell. Syst. V, N, Article A (January YYYY), 26 pages.
DOI = 10.1145/0000000.0000000 <http://doi.acm.org/10.1145/0000000.0000000>

This research was supported by the EC funded projects FP7/ICT-IP Grant agreement No. 287704 CUBRIK (www.cubrikproject.eu) and FP7/ICT-STREP Grant agreement No. 248296 I-SEARCH (www.isearch-project.eu).

Author’s addresses: D. Rafailidis, A. Axenopoulos S. Manolopoulou and P. Daras, Information Technologies Institute, Center for Research and Technology Hellas, 6th km Charilou-Thermi, Thessaloniki, Greece; email: drafail@iti.gr.

J. Etzold, Fulda University of Applied Sciences, Marquardstrasse 35, 36039 Fulda, Germany; email: jonas.etzold@informatik.hs-fulda.de

Permission to make digital or hard copies of part or all 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 show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or permissions@acm.org.

© YYYY ACM 2160-6455/YYYY/01-ARTA \$10.00

DOI 10.1145/0000000.0000000 <http://doi.acm.org/10.1145/0000000.0000000>

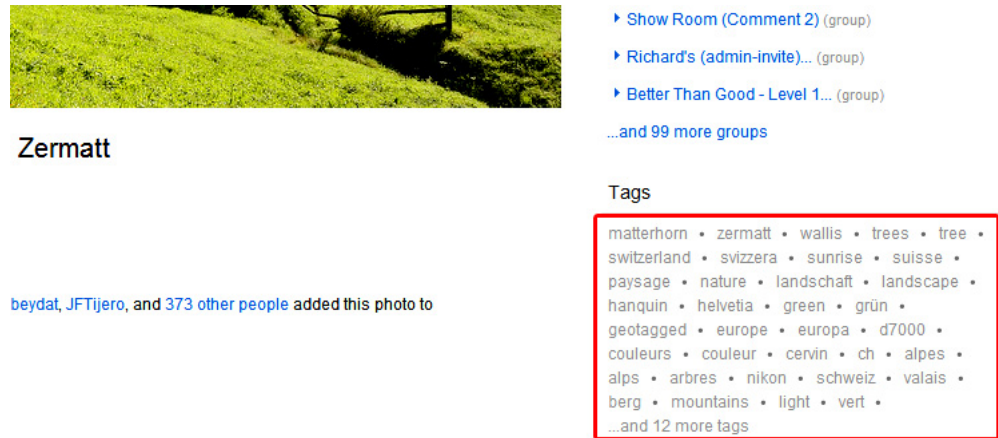


Fig. 1. Example of an item-based tag cloud in Flickr.

1. INTRODUCTION

Social Tagging Systems (STS) have attracted a lot of research attention recently due to their potential to improve search and personal organization of items. Users are able to annotate sets of items like photos (Flickr¹), songs (Last.fm²) or web sites (delicio.us³), in the form of free keywords, also known as social tags. Through social tags users can express their personal opinion to describe items. Therefore, users are able to browse and explore new interesting items, through item or user-based tag clouds. For example, an item-based tag cloud in Flickr (Figure 1) describes a certain image, where by clicking on a particular tag, all images that have been frequently annotated with that tag are displayed. Instead, a user-based tag cloud consists of the most frequent tags, which are applied by a certain user to annotate items.

Consequently, the complex multifaced information of items can be exploited and thus, STS are able to generate personalized recommendations, allowing users to pose tags as queries. STS recommenders consider the following collaborative-based strategy: users having the same tagging behavior tend to get similar recommendations. Therefore, such recommender systems are often characterized as item “Collaborative Filtering” (CF) recommenders in STS. An example of the CF paradigm in STS is depicted in Figure 2. The tagging process creates triplets in the form user-tag-item (U, T, I) , where user U_1 has annotated item I_1 with tag T_1 “fish”, and item I_2 with tag T_2 “car”, denoted by triplets (U_1, T_1, I_1) and (U_1, T_2, I_2) in Table I, respectively. User U_2 has also annotated item I_2 with tag T_2 “car”, corresponding to triplet (U_2, T_2, I_2) . Since U_1 and U_2 have shown similar behavior, by assigning the same tag to the same item, the CF mechanism associates user U_2 with tag T_1 and item I_1 , denoted by the sixth latent relation in Figure 2 and the respective triplet (U_2, T_1, I_1) in Table I. Note that each triplet (U, T, I) is associated with a weight w , corresponding to the likelihood that user U will annotate item I with tag T . These triplets constitute the underlying structure of STS and are also known as folksonomies. According to Marinho et al. [2011], a folksonomy is defined as a relational structure $\mathbb{F} := (\mathbf{U}, \mathbf{T}, \mathbf{I}, \mathbf{Y})$ in which \mathbf{U} , \mathbf{T} and \mathbf{I} are disjoint non-empty finite sets, whose elements are users, tags and items, respectively, and \mathbf{Y} is the set of observed ternary relations between them, i. e., $\mathbf{Y} \subseteq \mathbf{U} \times \mathbf{T} \times \mathbf{I}$, where for each triplet holds $(U, T, I) \in \mathbf{Y}$, with $U \in \mathbf{U}$, $T \in \mathbf{T}$ and $I \in \mathbf{I}$. Therefore, tags introduction

¹<http://www.flickr.com/>

²<http://www.lastfm.com/>

³<http://www.delicious.com/>

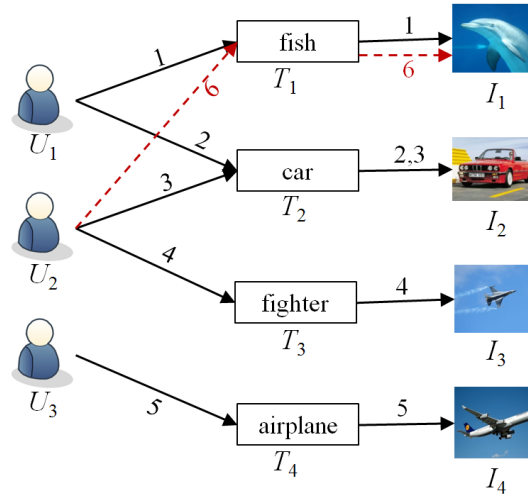


Fig. 2. Example of Collaborative Filtering mechanism in STS.

Table I. Generated triplets derived from the STS in Figure 2.

ID	User	Tag	Item	Weight(w)
1	U_1	T_1	I_1	$w(U_1, T_1, I_1)$
2	U_1	T_2	I_2	$w(U_1, T_2, I_2)$
3	U_2	T_2	I_2	$w(U_2, T_2, I_2)$
4	U_2	T_3	I_3	$w(U_2, T_3, I_3)$
5	U_3	T_4	I_4	$w(U_3, T_4, I_4)$
6	U_2	T_1	I_1	$w(U_2, T_1, I_1)$

in recommender systems has turned the usual binary relation between users and items into a ternary relation \mathbf{Y} between users, items and tags. This ternary relation is mapped to a tripartite network [Halpin et al. 2007; Lambiotte and Ausloos 2005; Cattuto et al. 2007; Wu et al. 2006], which should be considered by STS recommenders, in order to capture the latent associations among users, tags and items.

Nevertheless, several important problems affect the accuracy of STS recommenders [Tang et al. 2009]. For example, polysemy and synonymy (also known as the “vocabulary problem”) [Furnas et al. 1987; Golder and Huberman 2005] are very common problems in STS recommenders, since tags are characterized by their free nature and thus, they are subjected to multiple interpretations. Therefore, in order to handle the aforementioned problems and to capture the ternary relation in folksonomies, works of Xu et al. [2006], Rendle et al. [2009] and Symeonidis et al. [2010], follow tensor factorization techniques. Such methods are called tensor factorization models [Marinho et al. 2011], consisting of two basic steps. Firstly, the ternary relation \mathbf{Y} in folksonomies is modeled into a third-order tensor \mathcal{A} , where a tensor is a multi-dimensional matrix, corresponding to the three dimensions of users, tags and items. The stored values within tensor \mathcal{A} are the weights w , shown in Table I. Before applying tensor factorization techniques, all weights are set initially either to 0 or to 1, so as to denote the absence or existence of a triplet-association, respectively. Next, in order to exploit the underlying latent semantic structure in \mathcal{A} and to generate personalized recommendations, tensor factorization models compute the low rank tensor approximations [Lathauwer et al. 2000] of tensor \mathcal{A} and transform the recommendation problem into a third-order tensor completion problem, where the non-observed entries in \mathcal{A} have to be completed.

Therefore, tensor factorization models are able to (a) solve problems like polysemy and synonymy, (b) preserve the ternary relation, (c) reveal the latent associations among users, tags and items, (d) reduce the noise in STS, and (e) provide more accurate recommendation compared to methods that suppress the 3-way relationship to 2-way like the one presented by Tso-Sutter et al. [2008], where the personalized opinion of users is omitted.

1.1. Cold Start, Sparsity and Learning Tag Relevance: Three Major Factors Affecting the Quality of Personalized Recommendation based on Social Tagging

Despite the fact that tensor factorization models seem to be suitable for the case of STS recommenders, they do not handle the real world problems of (a) “cold start”, (b) sparsity and (c) “learning tag relevance”. In particular, a very common problem in STS recommenders is the “cold start” problem, which according to Herlocker et al. [2004] refers to the fact that users participate rarely in the tagging process and therefore, there are only few tags on which to base the recommendation. The “cold start” problem is presented in Figure 2, where U_3 has only annotated item I_4 with tag T_4 , denoted by the triplet (U_3, T_4, I_4) . Consequently, an STS recommender will fail to provide recommendations for user U_3 , since the CF mechanism will not be able to identify similar tagging behavior of user U_3 with the rest of users (U_1 and U_2).

Additionally, a very challenging task for recommenders is to handle the sparsity that occurs in STS, which further affects recommenders’ accuracy. More specifically, since item recommendation in social tagging is based on the CF mechanism, high accuracy is achieved only if users annotate the same items with similar tags, corresponding to users with the same tagging behavior. However, this is extremely difficult for the case of real world applications, since the number of online users, content and tags are grown exponentially. Consequently, the “cold start” and sparsity problems in tensor factorization models, affect the quality of item recommendation in STS.

In order to tackle both problems, several works in the literature apply tag propagation methods, where tags are assigned to items which have not yet been annotated. For example, the works of Papadopoulos et al. [2010] and Sevil et al. [2010] apply clustering methods to achieve tag propagation. Other approaches exploit the content of items (e.g. content of images, songs etc.), in order to perform tag propagation between content-similar items. Such an approach of content-based tag propagation is followed by Li et al. [2008], where a neighbor-voting algorithm is proposed. In particular, common tags are propagated between items, according to their content-based similarity. Each tag accumulates its relevance credit by receiving neighbors’ votes and thus, the tag is propagated to the proper item, according to the computed relevance credit. Moreover, a very important challenge for content-based tag propagation is handled, called “learning tag relevance” [Smeulders et al. 2000], based on which the semantic connections between the assigned tag and the content it represents must be revealed, so as to perform accurate tag propagation. However, the aforementioned tag propagation algorithms discard user’s information and thus, they are not suitable for the case of STS recommenders, where the ternary relation of users, tags and items should be preserved.

A very interesting study on the combination of content-based and tag information is the work of Qi et al. [2012], which handles the sparsity problem. The algorithm builds a latent semantic space, where each item is mapped into a latent feature vector, by encoding the content-based and tag information. Then, the latent feature vectors describing the items can be indexed, classified and retrieved with vector-based methods. However, in this work the user dimension is omitted and therefore personalization is not achieved. Tang et al. [2012] introduced a cross-space affinity (similarity) learning algorithm over the heterogeneous feature spaces of users and items, in order to achieve high recommendation accuracy. A Cross-Space Tensor (CST) model was constructed to learn the affinity measures across the heterogeneous spaces and to capture the correlations between users and items, by ex-

About 7,670,000 results (0.44 seconds)

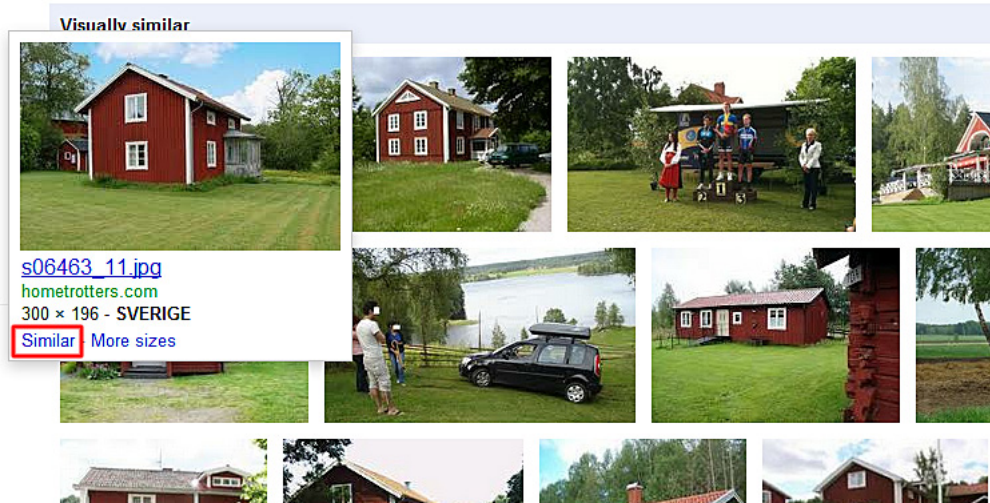


Fig. 3. In Google Image Search, the “Similar” item button is attached to each image-result.

plotting a set of order constraints on the affinity from a training pool. Authors further enhanced their model with a factorization form which reduces the number of parameters of a model with a controlled complexity. However, in this work the “cold start” problem is not handled.

A first attempt to solve the “cold start” and sparsity problems, by exploiting the content information in tensor factorization models, is the MusicBox (MB) method presented in [Nanopoulos et al. 2010]. However, the main disadvantage of the MB method is that the proposed content-based tag propagation cannot be uncontrolled. More precisely, by allowing extensive tag propagation, the noise that incurs may affect the quality of recommendations, due to the irrelevant tag assignments to items. Thus, in order to constraint the amount of propagated tags, authors control a threshold parameter based on the items similarity. However, the fundamental problem in this approach is strongly correlated to “learning tag relevance”, since tag propagation between content-based similar items should be performed only if they are also conceptually similar.

Meanwhile, a plethora of research works has been proposed concerning relevance feedback methods [Hoi et al. 2006; Jing et al. 2001; Liu and Chen 2005], which aim at retrieving the optimal number of conceptually similar items, by refining the results of content-based retrieval methods. An example of a relevance feedback procedure can be found in Google Image Search, which supports content-based retrieval, where by clicking on the “Similar” item button (Figure 3), the relevance feedback mechanism is activated. By clicking on this button, results-filtering is enabled and the images which are visually similar to the chosen one are retrieved. Since the quality of content-based retrieval of multimedia search engines is increased by identifying visually and conceptually similar results, the lack of relevance feedback methods in STS is of great importance. However, two factors should be clearly mentioned: (a) relevance feedback methods are not able to capture the person-to-person correlation, as the CF methods do [Scafer et al. 2007]; and (b) in relevance feedback methods, users’ input is required, in order to identify the conceptually similar results and to activate the mechanism of results-filtering. Thus, conceptually similar results cannot be identified automatically by the relevance feedback methods. This happens because in

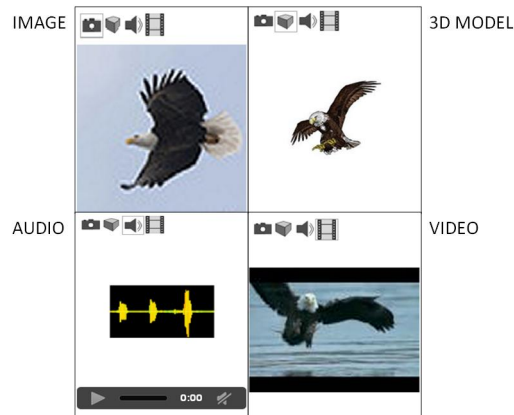


Fig. 4. A Multimedia Document (MMD) example in I-SEARCH, containing four modalities: image, 3D, audio and video.

contrast to STS, in the relevance feedback methods the tag information is ignored, able to identify conceptually similar results.

1.2. The Multimodal Engine of I-SEARCH

In traditional multimedia search engines, users can either enter a keyword as query or a media so as to perform either text-based or content-based retrieval. Recently, multimedia search engines have evolved, allowing combinations of queries of different media types. For example in Google Image Search, users search for images by combing a keyword with a similar image. Multimodal search allows users to enter multiple query types and retrieve multiple types of media simultaneously. This is a significant step towards content-based multimedia retrieval, since users can search and retrieve media of any type using a single unified retrieval framework and not a specialized system for each separate media type. Moreover, through multimodal retrieval, users can enter multiple queries simultaneously and thus, retrieve more relevant results. However, this is a highly complicated process, since the successful modeling of the low-level feature associations among the different media types is required.

An approach for multimodal search has been introduced by the EU-funded project I-SEARCH, available at <http://vcl.iti.gr/is/>. The search engine of I-SEARCH enables retrieval of several types of media (3D objects, 2D images, sound, video and text) using as query any of the above types or their combinations. In I-SEARCH, multiple media types, which share the same semantics, are enclosed into a media container, which is called Multimedia Document (MMD). The concept of MMD was introduced in [Yang et al. 2009], where a method for connecting various semantically similar media of different types has been proposed. An example of a MMD is presented in Figure 4, which describes a physical entity of an “eagle” and consists of its 3D representation, real image, sound and video.

The creation of MMDs is not a trivial task, since it involves merging of different types of media with the same semantics. In order to facilitate the creation of a multimodal dataset of MMDs in I-SEARCH, the CoFetch tool⁴ tool has been implemented. The CoFetch tool performs search on public media sources and creates corresponding MMDs. More specifically, users enter a keyword to the search interface and CoFetch performs multiple search tasks simultaneously: search for text in Wikipedia, images in Flickr, audio files in freesound.org, 3D objects in Google 3D Warehouse and videos in youTube. Then, the tool returns multi-

⁴<http://youtu.be/PwzhJ-3-zgY>

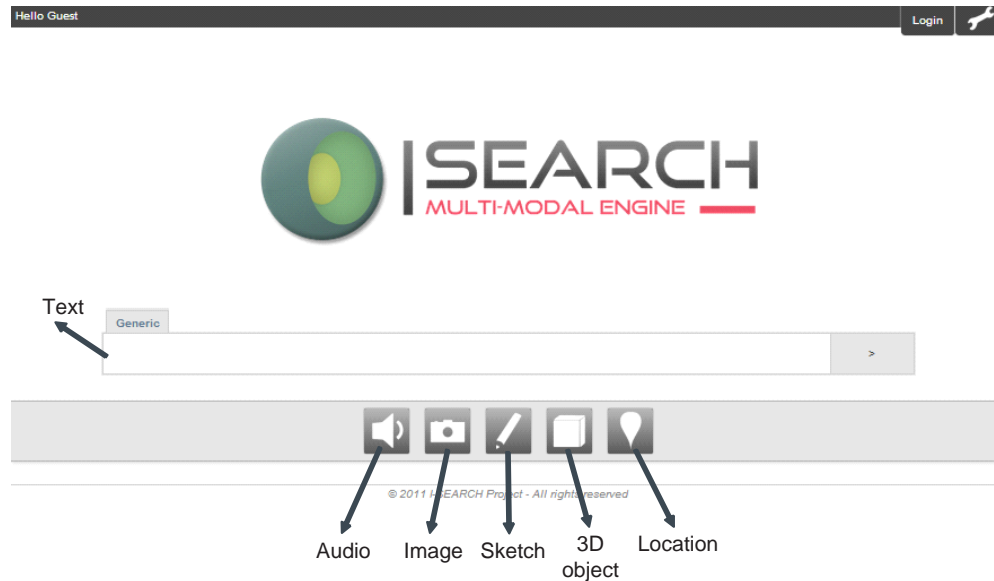


Fig. 5. The query compilation interface in I-SEARCH.

ple ranked lists of relevant results of different types of media. Then, users can combine the retrieved media of different types so as to create MMDs.

Multimodal retrieval is based on identifying correlations among media of different types and mapping their descriptor vectors on a common feature space. Manifold learning has been extensively used for this purpose. A promising multimodal retrieval method has been recently proposed in [Daras et al. 2012]. The method creates a low-dimensional feature space, using the Manifold Learning method of Laplacian Eigenmaps [Belkin and Niyogi 2003], where all MMDs can be mapped irrespective of their constituting media types. In order to preserve the local neighborhood of each media into the low-dimensional MMD space, a multimodal adjacency matrix is constructed, where items that correspond to pairs of neighboring MMDs are denoted by ones, whereas the remaining items are denoted by zeros. Consequently, multimodal descriptor vectors are generated from the constructed MMD space. Then, multimodal retrieval of MMDs is achieved by computing the pairwise distances among their low-dimensional multimodal descriptor vectors. In [Daras et al. 2012], it is experimentally shown that search using multimodal queries achieves higher retrieval accuracy than using monomodal queries. This method has been developed in the context of I-SEARCH to address the problem of multimodal search and retrieval.

An example of the query compilation interface of I-SEARCH is presented in Figure 5, where users are able to enter multiple types of queries simultaneously, such as text, images, audio, sketch, 3D objects and location. By doing so, users can create MMDs on-the-fly and pose them as queries, so as to exploit the full potential of the multimodal search engine and thus to retrieve more accurate results. It should be clarified that the location is used as a filtering step, i.e. after the ranked list is retrieved, the location filters out the MMDs that are lying far from the predefined location. In Figure 6, the results interface is presented, where the most relevant MMDs to the query (combination of 3D object, image and text) are retrieved. By clicking on an MMD, a pop-up window appears, where users can visualize all the constituting modalities. Although multimodal search can produce more accurate results

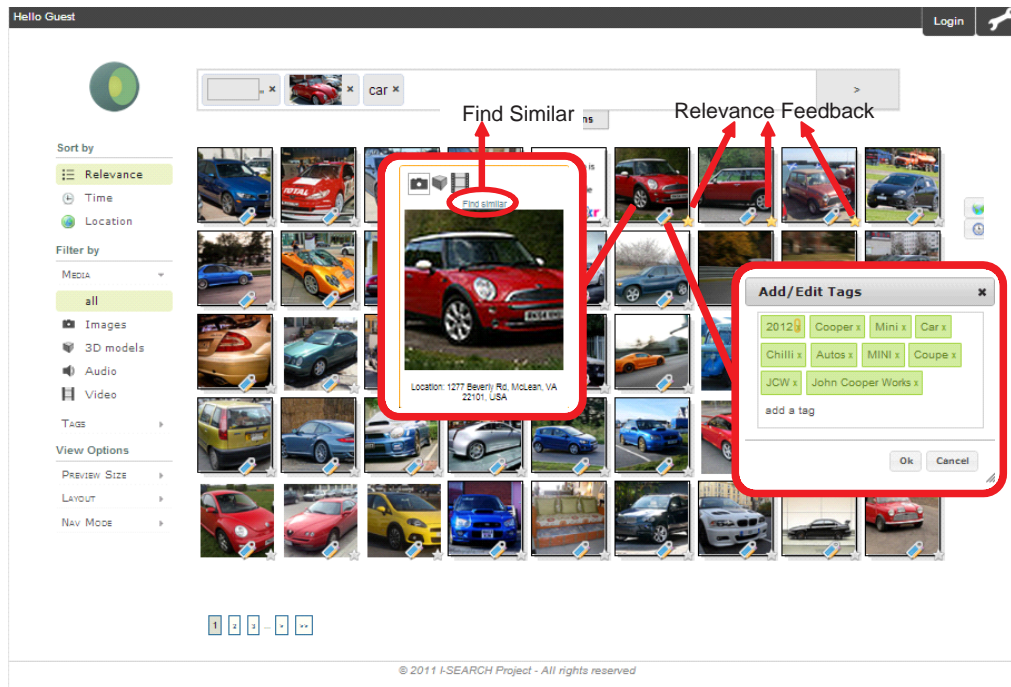


Fig. 6. In I-SEARCH the results are represented in the form of MMDs, by a representative screenshot. Moreover, users can tag each result separately, by selecting the result and then assign a tag using the “add a tag button”.

than monomodal search, in a real-life scenario building a multimodal query is not always straightforward. This is due to the fact that users may not always have all media types available to form a MMD query (e.g. most users rarely have 3D objects at their local disk).

Additionally, users can log into I-SEARCH, where the following personalized functionalities are supported: (a) relevance feedback and (b) recommendation based on social tagging. Through the personalized relevance feedback, the retrieval accuracy is improved, making users an active part of the system. Two different relevance feedback functionalities are supported. In the first one, users select only one of the retrieved MMDs and start a new search using the selected MMD as query. This is achieved by clicking on the “Find Similar” button of the selected MMD, as presented in Figure 6. The second relevance feedback functionality is a bit more complex, where users mark many MMDs as relevant by pressing the “star” button on the bottom right corner of the results’ thumbnails and then press the “search” button again. In this case, a Query Expansion strategy is performed where all relevant MMDs are used as queries simultaneously. However, user’s input is required, in order to activate both relevance feedback mechanisms and thus, the conceptually similar MMDs cannot be identified automatically.

Following the STS paradigm, users can assign tags to each result-MMD separately, by using the “Add/Edit tag” button, as presented in Figure 6. Moreover, each time users are logged in I-SEARCH, their personal (user-based) tag cloud is displayed, as presented in the bottom of Figure 7. Users can select a tag from their personal tag cloud, to generate a list of recommended MMDs. By selecting one of the recommended MMDs, a multimodal search is initiated using as query the selected MMD. In the example of Figure 7, user “user@example.com” logs in to I-SEARCH and selects the tag “Clownfish” from his personal

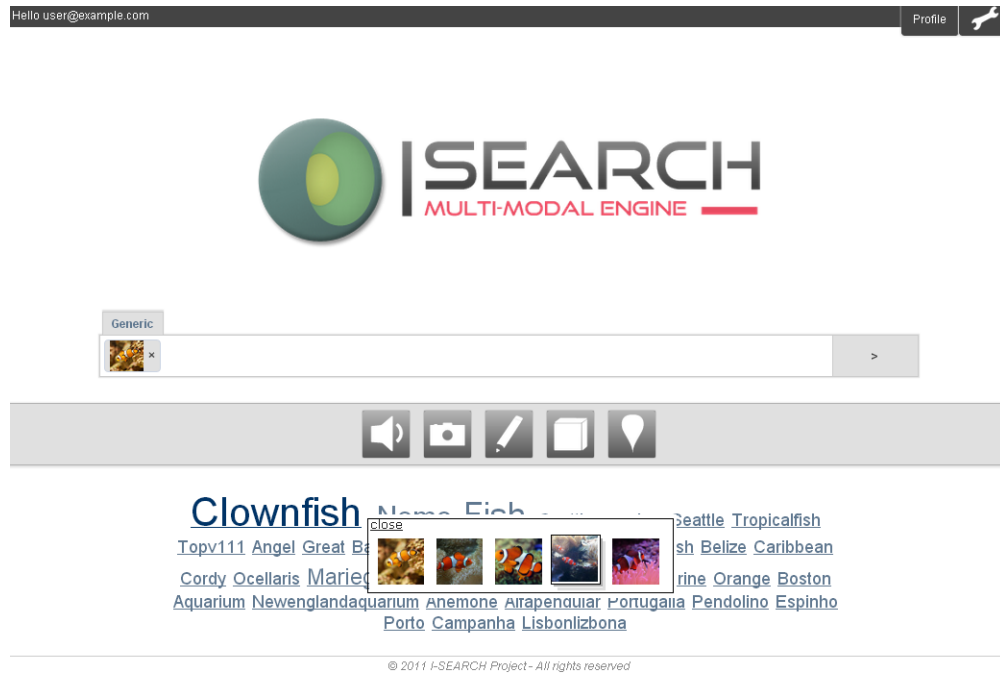


Fig. 7. In I-SEARCH, users can select a tag from their personal tag cloud, to generate a list of recommended MMDs. Then one of the recommended MMDs can be selected and added to query compilation interface, in order to start a multimodal search task.

tag cloud. Then, the recommended MMDs are presented in the pop-up window, where the user can select one of them to generate a multimodal search task. The recommendation functionality can assist users in the multimodal environment of I-SEARCH, since uploading multiple media of different modalities to construct a MMD query is avoided and thus, users' interactions can be simplified with the system. However, since the personalized MMD recommendation is based on social tagging, the STS' inherited problems of "cold start", sparsity and "learning tag relevance" should be handled, in order to increase the quality of recommendations.

1.3. Contribution and Layout

In the context of I-SEARCH, we propose a personalized item recommendation methodology based on social tagging, which is of great importance for users in I-SEARCH, since users' interaction is simplified by recommending items in the form of MMDs. Our contribution is summarized as follows:

- Exploit users' tag assignments through a relevance feedback mechanism to automatically identify content-based and conceptually similar items.
- Handle the sparsity and "cold start" problems, by propagating tags between the identified conceptually similar items, where (a) the issue of "learning tag relevance" is handled, by calculating the relevance degrees between users, tags and items and (b) the ternary relation among users, tags and items is preserved, by performing tag propagation in the form of triplets based on users' personalized preferences and "cold start" degree.
- Present the recommendation methodology in the multimodal engine of I-SEARCH at <http://vcl.iti.gr/is/>, where the recommended items are unified sets of media of different modalities in the form of MMDs.

In our experiments with two real world social datasets, the first one crawled from FLICKR and the second one created within I-SEARCH with the recommended items be images and MMDs, respectively, we show the superiority of the proposed methodology over the methods of Symeonidis et al. [2010], Nanopoulos et al. [2010] and an ad-hoc retrieval method of a pseudo-relevance feedback scheme, by adequately handling the aforementioned problems in STS while performing the same tensor factorization technique of HOSVD.

The remainder of this paper is organized as follows: after describing the proposed method in Section 2, we present the experimental results on real-world social tagging data in Section 3 and the basic conclusions of our study are furnished in Section 4.

2. THE PROPOSED METHOD FOR PERSONALIZED ITEM RECOMMENDATION BASED ON SOCIAL TAGGING

2.1. Method Overview

Following the case scenario of I-SEARCH, a recommended item I is in the form of a MMD and the social tagging data are in the form of triplets (U, T, I) , associated with a weight w , corresponding to the likelihood that user U will annotate MMD I with tag T . For each triplet holds: $(U, T, I) \in \mathbf{Y}$, with $U \in \mathbf{U}$, $T \in \mathbf{T}$ and $I \in \mathbf{I}$, where \mathbf{U} , \mathbf{T} , \mathbf{I} , \mathbf{Y} are the sets of users, tags, MMDs and triplets, respectively. The input of the proposed method are (a) the set of triplets \mathbf{Y} , with all weights w initially set to 1; (b) the set of multimodal descriptor vectors, where from each MMD $I \in \mathbf{I}$ the multimodal descriptor \overrightarrow{DV}_I is extracted according to the extraction process of Daras et al. [2012]; and (c) the $|\mathbf{R}_Q|$ ⁵ length of the results list. The task of item recommendation is for each pair (U, T) to compute the ranking of the $|\mathbf{I}|$ MMDs, with the top- N ranked items be the N recommended MMDs. The personalized item recommendation methodology consists of the following three steps:

- Firstly, \forall triplet $(U, T, I) \in \mathbf{Y}$, I is posed as query Q to generate the \mathbf{R}_Q result list of MMDs, according to *dis*, i.e. the content-based distance of the multimodal descriptors \overrightarrow{DV}_Q and \overrightarrow{DV}_R , with $R \in \mathbf{I}$ and $R \neq Q$. Then, based on the relevance feedback mechanism, the \mathbf{I}_Q^+ list of conceptually similar MMDs (positives) is generated.
- Next, the tag propagation method is performed \forall triplet $(U, T, I) \in \mathbf{Y}$ and the respective \mathbf{I}_Q^+ list, where the relevance degrees between user U , tag T and MMD I are computed and stored into the respective weight w . To preserve the ternary relation in \mathbf{Y} , the tag propagation method is performed in the form of triplets, associated with the computed weight w , and thus, new propagating triplets are generated. Therefore, the outcome of this step is a superset \mathbf{Y}^+ , with $\mathbf{Y} \subseteq \mathbf{Y}^+$.
- Finally, the generated data set \mathbf{Y}^+ is modeled into a tensor, and the tensor factorization technique of HOSVD is followed, in order to reveal the latent associations among users, tags and MMDs and to generate the final personalized recommendations.

An overview of the proposed method is presented in Figure 8, based on the STS example of Figure 2, where the initial social tagging data \mathbf{Y} is the set of triplets shown in Table I. For presentation purposes, in this example only the image-modality is considered. As aforementioned, the “cold start” problem is faced for generating recommendations to user U_3 . Thus, we consider the corresponding triplet (U_3, T_4, I_4) , based on which we initiate the procedure of the proposed methodology, as depicted in Figure 8.

In the first step, I_4 is posed as query Q to retrieve content-based similar results in \mathbf{R}_Q , with length $|\mathbf{R}_Q| = 5$. Then, based on the relevance feedback mechanism, MMDs I_5 and I_3 are identified as positives and stored in the \mathbf{I}_Q^+ list. Additionally, I_5 and I_3 are denoted by pos_1 and pos_2 , respectively, based on their rank in \mathbf{I}_Q^+ .

⁵In this paper, lists are denoted by capital letters X and sets are denoted by capital bold letters \mathbf{X} .

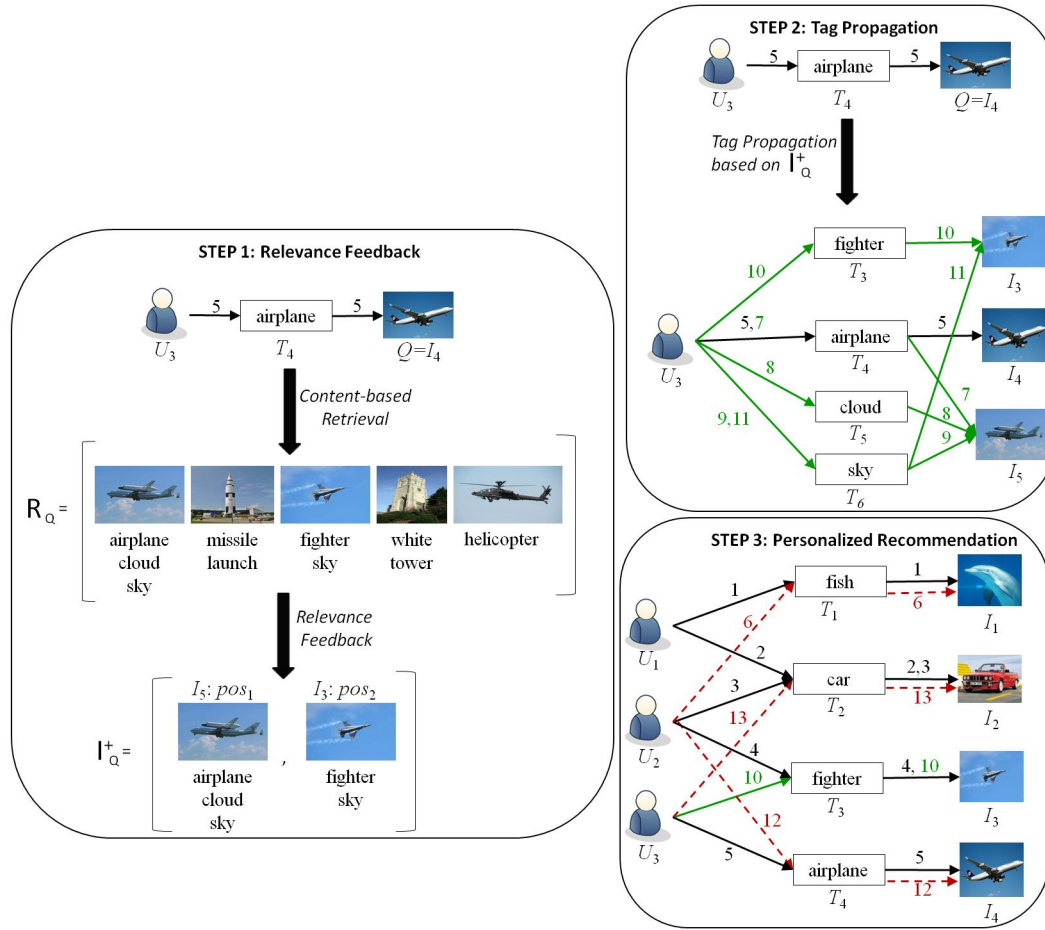


Fig. 8. Overview of the proposed method for personalized item recommendation.

In the second step, tag propagation is performed, by considering: (a) the triplet (U_3, T_4, I_4) , and (b) the $I_5=pos_1$ and $I_3=pos_2$ positives in the I_Q^+ list. In particular, five additional triplets are generated with IDs from 7 to 11, as shown in Table II. Consequently, the propagated triplets are added to the initial data set \mathbf{Y} , and thus the superset \mathbf{Y}^+ is generated, with $\mathbf{Y} \subseteq \mathbf{Y}^+$.

In the third step, the generated set of triplets \mathbf{Y}^+ is modeled into the tensor \mathcal{A} . Next, in order to reveal the latent associations among users, tags and MMDs, the low rank approximation $\hat{\mathcal{A}}$ of \mathcal{A} is computed, based on the HOSVD tensor factorization technique. Therefore, except for the previously revealed latent association 6 from Figure 2, the new latent associations 12 and 13 are revealed. Consequently, the “cold start” problem is addressed, by performing content-based tag propagation in the form of triplets. In the following Sections, each step of the proposed methodology is described in further details.

2.2. The Relevance Feedback Mechanism

The first step of the proposed method is the relevance feedback mechanism to generate for each triplet (U, T, I) the I_Q^+ list of positives. The input of Algorithm 1 of the relevance feedback mechanism is (a) the input triplet $(U_{in}, T_{in}, I_{in}) \in \mathbf{Y}$; (b) the set of multimodal

Table II. Propagated triplets derived from the second step in Figure 8, based on triplet (U_3, T_4, I_4) . Weights w are calculated according to Equations (5), (6), (7), (8)

ID	User	Tag	Item	Weight(w)
7	U_3	T_4	I_5	$w(U_3, T_4, I_5) = 0.2$
8	U_3	T_5	I_5	$w(U_3, T_5, I_5) = 0.2$
9	U_3	T_6	I_5	$w(U_3, T_6, I_5) = 0.4$
10	U_3	T_3	I_3	$w(U_3, T_3, I_3) = 0.1$
11	U_3	T_6	I_3	$w(U_3, T_6, I_3) = 0.1$

descriptor vectors, where from each MMD $I \in \mathbf{I}$ the multimodal descriptor \overrightarrow{DV}_I is extracted according to the extraction process of Daras et al. [2012]; and (c) the $|\mathbf{R}_Q|$ length of the results list. The algorithm consists of (a) the initialization process and (b) the iterative process of Query Expansion.

Initialization process

Given the input triplet (U_{in}, T_{in}, I_{in}) , I_{in} is posed as query Q to generate the \mathbf{R}_Q ranking list based on the distance dis , i.e. the L_2 distance of the extracted multimodal descriptors \overrightarrow{DV}_Q and \overrightarrow{DV}_R , with $R \in \mathbf{I}$ and $R \neq Q$. Then, the contextual information of users' tag assignments to MMDs is considered, in order to identify the positives and to generate the \mathbf{I}_Q^+ list. More precisely, given tag T_{in} of user U_{in} , let \mathbf{T}_Q be the set of tags assigned to Q by all users, and \mathbf{T}_{R_j} be the respective tag set of each result $R_j \in \mathbf{R}_Q$, with $j = 1 \dots |\mathbf{R}_Q|$. The Sum Squared Error (SSE) is calculated between tag T_{in} and tag set \mathbf{T}_Q according to the following equation:

$$SSE(T_{in}, \mathbf{T}_Q) = \frac{1}{|\mathbf{T}_Q|} \cdot \sum_{T_k \in \mathbf{T}_Q} TagDis(T_{in}, T_k)^2 \quad (1)$$

where $TagDis$ is the distance between tags T_1 and T_2 according to the Dice ⁶ similarity measure:

$$TagDis(T_1, T_2) = 1 - \frac{2 \cdot (\mathbf{I}_{T_1} \cap \mathbf{I}_{T_2})}{|\mathbf{I}_{T_1}| + |\mathbf{I}_{T_2}|} \quad (2)$$

where \mathbf{I}_{T_1} and \mathbf{I}_{T_2} are the sets of all MMDs that tags T_1 and T_2 have been assigned to. The value of $SSE(T_{in}, \mathbf{T}_Q)$ contains valuable information, since it can express the association degree of tag T_{in} of user U_{in} with query Q , by considering all users' tag assignments to Q . In the initialization process, the SSE_{thres} threshold is set equal to $SSE(T_{in}, \mathbf{T}_Q)$. Each result R_j , with $j = 1 \dots |\mathbf{R}_Q|$, is examined based on the condition $SSE(T_{in}, \mathbf{T}_{R_j}) \leq SSE_{thres}$ and in case that the condition is satisfied, then R_j is identified as positive and is appended to the end of list \mathbf{I}_Q^+ . The main idea is that if the condition is satisfied, it means that according to users' tag assignments the association degree of tag T_{in} with result R_j is equal or higher than the respective association degree of tag T_{in} with query Q and thus, result R_j is considered as conceptually similar (positive) to Q . By examining all results R_j in \mathbf{R}_Q and identifying the positive results, threshold SSE_{thres} is updated and calculated based on the minimum $SSE(T_{in}, \mathbf{T}_{R_j})$, with $R_j \in \mathbf{I}_Q^+$, i.e. the minimum SSE of the identified positive results of the \mathbf{I}_Q^+ list. Additionally, for the next

⁶In our method, we consider the Dice similarity measure for calculating the tags distances due to its low complexity. However, there are different types of tag similarity measures, thoroughly examined by Markines et al. [2009].

ALGORITHM 1: The Relevance Feedback Mechanism

Input: Triplet (U_{in}, T_{in}, I_{in}) , the set of multimodal descriptors $\overrightarrow{DV}_I, \forall I \in \mathbf{I}$ and length $|\mathbf{R}_Q|$.

Output: List \mathbf{I}_Q^+ .

INITIALIZATION PROCESS

Set $SSE_{thres} = SSE(T_{in}, \mathbf{T}_Q)$, with \mathbf{T}_Q be the set of tags assigned to $Q = I_{in}$ by all users in \mathbf{U} and T_{in} be the tag of user U_{in} ;

Calculate $\mathbf{R}_Q = \{R_1, R_2, \dots, R_{|\mathbf{R}_Q|}\}$ according to $dis(\overrightarrow{DV}_Q, \overrightarrow{DV}_R)$, with $R \in \mathbf{I}$ and $R \neq Q$;

for $(j = 1 : |\mathbf{R}_Q|)$;

do

if $(SSE(T_{in}, \mathbf{T}_{R_j}) \leq SSE_{thres})$;

then

 Update $\mathbf{I}_Q^+ \leftarrow \{\mathbf{I}_Q^+, R_j\}$;

end

end

Update $SSE_{thres} = \min_{\forall R_j \in \mathbf{I}_Q^+} SSE(T_{in}, \mathbf{T}_{R_j})$;

Set $\mathbf{I}_{prev}^+ \leftarrow \mathbf{I}_Q^+$;

ITERATIVE PROCESS OF QUERY EXPANSION

repeat

 Calculate \mathbf{R}_Q according to $dis_{agg}(\overrightarrow{DV}_I, \overrightarrow{DV}_R)$, with $R \in \mathbf{I}$ and $I \in \mathbf{I}_{prev}^+$;

for $(j = 1 : |\mathbf{R}_Q|)$;

do

if $(SSE(T_{in}, \mathbf{T}_{R_j}) \leq SSE_{thres})$ and $(R_j \notin \mathbf{I}_Q^+)$;

then

 Update $\mathbf{I}_Q^+ \leftarrow \{\mathbf{I}_Q^+, R_j\}$;

end

end

 Update $SSE_{thres} = \min_{\forall R_j \in \mathbf{I}_Q^+} SSE(T_{in}, \mathbf{T}_{R_j})$;

 Set $\mathbf{I}_{prev}^+ \leftarrow \mathbf{I}_Q^+$;

until $(|\mathbf{I}_Q^+| = |\mathbf{I}_{prev}^+|$ or $|\mathbf{I}_Q^+| = |\mathbf{R}_Q|)$;

return \mathbf{I}_Q^+ ;

step we set \mathbf{I}_{prev}^+ equal to \mathbf{I}_Q^+ , in order to denote the list of the previously identified positives.

Iterative process of Query Expansion

Next, provided the updated SSE_{thres} threshold and the previously identified positive results in the \mathbf{I}_{prev}^+ list, the iterative process of the Query Expansion strategy starts, which relies on the query expansion techniques presented by Chakrabarti et al. [2004], Porkaew et al. [1999], Wu et al. [2000] and French and Jin [2004]. In the aforementioned techniques, the previously identified positives in the \mathbf{I}_{prev}^+ list are considered as a queries' set \mathbf{Q} . Then, the distance between the set \mathbf{Q} and each R , with $R \in \mathbf{I}$, is computed, using the aggregate distance dis_{agg} :

$$dis_{agg}(\overrightarrow{DV}_Q, \overrightarrow{DV}_R) = \sum_{I \in \mathbf{Q}} dis(\overrightarrow{DV}_I, \overrightarrow{DV}_R), \quad (3)$$

Thus, based on dis_{agg} a new \mathbf{R}_Q result list is generated. Next, similar to the initialization process of the algorithm, each result $R_j \in \mathbf{R}_Q$ is examined based on condition $SSE(T_{in}, \mathbf{T}_{R_j}) \leq SSE_{thres}$, where SSE_{thres} is the threshold derived by the previously

identified positives in the I_{prev}^+ list. If the condition is satisfied then R_j is identified as positive and is appended to the end of list I_{Q}^+ . Then, after examining all results $R_j \in R_{\text{Q}}$, a new threshold SSE_{thres} is calculated based on the minimum $SSE(T_{in}, \mathbf{T}_{R_j})$, with $R_j \in I_{\text{Q}}^+$, i.e. the minimum SSE of the currently identified positive results in the I_{Q}^+ list of the current iteration. By doing so, threshold SSE_{thres} is reduced and becomes stricter for the next iteration. This is of great importance because the content-based (visually) similarity to the initial query $Q = I_{in}$ declines over the iterations and thus, a stricter threshold is required when a new iteration starts. The iterative process terminates when the number of positive results remains the same or when all results $R_j \in R_{\text{Q}}$ have been marked as positives.

The outcome of the relevance feedback step is for each input triplet $(U_{in}, T_{in}, I_{in}) \in \mathbf{Y}$, the I_{Q}^+ list of positives. In order to denote the relevance degree of each $R \in I_{\text{Q}}^+$ to query Q , each R is associated with a pos_i variable, equal to the respective ranking of R in list I_{Q}^+ . For example, in Figure 8 the second positive ranked result in the I_{Q}^+ list is I_3 , denoted as pos_2 .

2.3. The Content-based Tag Propagation Method

In the second step of the proposed approach, content-based tag propagation is performed for each input triplet $(U_{in}, T_{in}, I_{in}) \in \mathbf{Y}$ based on (a) user's U_{in} personal preferences for using tag T_{in} and annotating MMD I_{in} (b) user's U_{in} ‘‘cold start’’ degree and (c) the respective I_{Q}^+ list of positives, derived by Algorithm 1. Each positive $pos_i \in I_{\text{Q}}^+$ contains a set of tags \mathbf{T}_i . In the example of Figure 8, it holds that $|I_{\text{Q}}^+|=2$ and sets $\mathbf{T}_1 \equiv \{\text{airplane,cloud,sky}\}$ and $\mathbf{T}_2 \equiv \{\text{fighter,sky}\}$ are the tag sets of $pos_1=I_5$ and $pos_2=I_3$, respectively. In particular, the content-based tag propagation for each input triplet (U_{in}, T_{in}, I_{in}) results in a set of triplets \mathbf{Y}_{pos} as follows:

$$\mathbf{Y}_{\text{pos}}(U_{in}, T_{in}, I_{in}) \equiv \{(U_{in}, T_p, pos_i) | T_p \in \mathbf{T}_i, pos_i \in I_{\text{Q}}^+\} \quad (4)$$

For the example of Figure 8 the set of propagated triplets $\mathbf{Y}_{\text{pos}}(U_3, T_4, I_4)$ is presented in Table II. However, in order to consider the personal preferences of user U_{in} and the issue of ‘‘learning tag relevance’’, the w weight of each propagated triplet (U_{in}, T_p, pos_i) is calculated according to the following equation:

$$w(U_{in}, T_p, pos_i) = \text{Score}(U_{in}) \cdot \text{Score}(T_p) \cdot \text{Score}(pos_i) \quad (5)$$

The three factors denote the relevance degrees of U_{in} , T_p and pos_i to the input triplet (U_{in}, T_{in}, I_{in}) . In particular, $\text{Score}(U_{in})$ captures user's U_{in} personal preferences for using tag T_{in} and annotating MMD I_{in} , by considering user's U_{in} ‘‘cold start’’ degree:

$$\text{Score}(U_{in}) = \frac{\text{freq}(U_{in}, T_{in})}{\text{freq}(U_{in})} \cdot \frac{\text{freq}(U_{in}, I_{in})}{\text{freq}(U_{in})} \quad (6)$$

where the first factor expresses user's U_{in} personal preference for using tag T_{in} , as the ratio of the $\text{freq}(U_{in}, T_{in})$ times that user U_{in} annotated an MMD with tag T_{in} over the total $\text{freq}(U_{in})$ annotations of U_{in} in \mathbf{Y} . The second factor expresses user's U_{in} personal preference for annotating MMD I_{in} , which is the ratio of the $\text{freq}(U_{in}, I_{in})$ times that user U_{in} annotated MMD I_{in} over the total annotations of U_{in} in \mathbf{Y} . In the dominator of each fraction, user's U_{in} ‘‘cold start’’ degree is also considered, denoted by $\text{freq}(U_{in})$. Therefore, $\text{Score}(U_{in})$ expresses the importance of each input triplet (U_{in}, T_{in}, I_{in}) for user U_{in} based on his personal preferences and ‘‘cold start’’ degree and thus, it respectively expresses the importance of the propagated triplets (U_{in}, T_p, pos_i) , derived by Algorithm 1 and Equation (4). By doing so, for a ‘‘cold start’’ user, with relative low $\text{freq}(U_{in})$ annotations,

Score(U_{in}) becomes high, resulting in an analogous increase of weights $w(U_{in}, T_p, pos_i)$ of the propagated triplets. In the example of Figure 8, since user U_3 has only one annotation, denoted by triplet (U_3, T_4, I_4) , it holds $\text{freq}(U_3, T_4) = \text{freq}(U_3, I_4) = \text{freq}(U_3) = 1$ and thus, all propagated triplets in Table II are important for user U_3 , with respect to his personal preferences and “cold start” degree. The relevance degree of tag T_p is calculated as follows:

$$\text{Score}(T_p) = \frac{\text{freq}(T_p)}{\sum_{T \in \mathbf{T}_{pos}} \text{freq}(T)} \quad (7)$$

$(\mathbf{T}_{pos}, \text{freq})$ is a multiset ⁷, derived by the aggregation of the \mathbf{T}_i tag sets of the positives in list I_Q^+ , where \mathbf{T}_{pos} is the underlying set of tags and \forall tag $T_p \in \mathbf{T}_{pos}$ the function $\text{freq}(T_p)$ is the multiplicity (i.e. number of occurrences) of T_p in the \mathbf{T}_{pos} multiset. For example, in Figure 8, the multiset $(\mathbf{T}_{pos}, \text{freq})$ is equal to:

$$\{(\text{airplane}, \text{cloud}, \text{sky}, \text{fighter})\}, \{(\text{airplane}, 1), (\text{cloud}, 1), (\text{sky}, 2), (\text{fighter}, 1)\}$$

Thus, according to Equation (7) the respective scores for user U_3 and tags (airplane, cloud, sky and fighter) are $(1/5, 1/5, 2/5$ and $1/5)$, respectively. In order to denote the relevance degree of pos_i , $\text{Score}(pos_i)$ is calculated based on the reciprocal rank of pos_i , which is equal to the inverse of the rank of pos_i :

$$\text{Score}(pos_i) = \frac{1}{i} \quad (8)$$

For example, in Figure 8, since $pos_1 = I_5$ is more relevant than $pos_2 = I_3$, the respective scores for positives pos_1 and pos_2 are equal to $1/1$ and $1/2$, respectively. According to Equations (5), (6), (7) and (8) weights w of the propagated triplets are presented in Table II. The final outcome of the content-based tag propagation step is the final \mathbf{Y}^+ set of triplets, with $\mathbf{Y}^+ \equiv \mathbf{Y} \cup \mathbf{Y}_{pos}(U_{in}, T_{in}, I_{in}), \forall (U_{in}, T_{in}, I_{in}) \in \mathbf{Y}$, where weights w in \mathbf{Y} equal 1 and weights w in $\mathbf{Y}_{pos}(U_{in}, T_{in}, I_{in})$ are calculated according to Equation (5). Moreover, based on the Relevance Feedback mechanism of Algorithm 1 and the tag propagation of Equation (4), there is the case of having a set of N different weights $\{w_1, w_2, \dots, w_N\}$ for the same triplet $(U, T, I) \in \mathbf{Y}^+$. To handle this case, triplet (U, T, I) is associated with weight $w(U, T, I) = \max\{w_1, w_2, \dots, w_N\}$ and stored in \mathbf{Y}^+ . By doing so, the weight $w(U, T, I) = 1$ of an initial triplet in \mathbf{Y} is also preserved in \mathbf{Y}^+ , since based on Equations (5),(6), (7) and (8) the propagated triplets in $\mathbf{Y}_{pos}(U_{in}, T_{in}, I_{in})$ are associated with a weight $w \leq 1$.

2.4. Personalized Item Recommendation based on High Order Singular Value Decomposition (HOSVD)

In the third and final step of the proposed approach, the personalized recommendation method comprises (a) modeling the set \mathbf{Y}^+ in tensor \mathcal{A} and (b) applying the tensor factorization method of HOSVD to produce the reconstructed tensor $\hat{\mathcal{A}}$ and to reveal the latent associations among users, tags and MMDs. The final goal is to recommend MMDs according to the detected latent associations in the reconstructed tensor $\hat{\mathcal{A}}$. The procedure of HOSVD is illustrated in Figure 9, where $I_1 = \mathbf{U}$, $I_2 = \mathbf{T}$, $I_3 = \mathbf{I}$ are the user, tag and MMD dimensions, respectively and S is the core tensor that captures the 3-way relations. HOSVD can be decomposed in 6 parts presented as follows.

⁷Following the definition of set theory, a multiset is a generation of the notion of an ordinary set, where multiple but finite occurrences of any element are allowed.

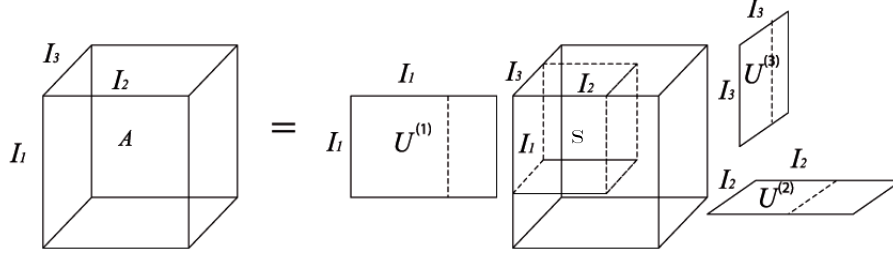


Fig. 9. Visualization of the result of HOSVD.

Initial construction of tensor \mathcal{A} : Based on the set \mathbf{Y}^+ , with $\mathbf{Y}^+ \equiv \mathbf{Y} \cup \mathbf{Y}_{pos}$, an initial 3-order tensor $\mathcal{A} \in \mathbb{R}^{U \times T \times I}$ is constructed. The initial values assigned to each entry of \mathcal{A} equals (a) to the pre-computed weights according to Equation (5) for the set of propagated triplets in \mathbf{Y}_{pos} and (b) to weights equal to 1 for the initial set of triplets in \mathbf{Y} .

Matrix unfolding of tensor \mathcal{A} : Tensor \mathcal{A} can be unfolded i.e., transformed to a two dimensional matrix, by arranging the corresponding fibers of \mathcal{A} as columns of A_n ($1 \leq n \leq 3$) [Lathauwer et al. 2000]. In our approach, the initial tensor \mathcal{A} is unfolded to all its three modes-dimensions. Thus, after the unfolding of tensor \mathcal{A} , we create three new matrices A_1, A_2, A_3 , as follows:

$$A_1 \in \mathbb{R}^{U \times TI}, \quad A_2 \in \mathbb{R}^{T \times UI}, \quad A_3 \in \mathbb{R}^{UT \times I}$$

Application of SVD on each unfolded matrix: Next, SVD is applied on the three matrix unfoldings A_n ($1 \leq n \leq 3$), resulting in the following decomposition:

$$A_n = U^{(n)} \cdot \Sigma^{(n)} \cdot (V^{(n)})^T, \quad 1 \leq n \leq 3 \quad (9)$$

To reveal latent associations and reduce noise, the dimensionality of each array containing the left-singular vectors (i.e., matrices $U^{(1)}, U^{(2)}, U^{(3)}$) has to be reduced. Therefore, we maintain the dominant c_n left singular vectors in each $U^{(n)}$, $1 \leq n \leq 3$, matrix based on the corresponding singular values in $\Sigma^{(n)}$. The resulting matrix is denoted as $U_{c_n}^{(n)}$. The values of c_n parameters are usually chosen by preserving a percentage of information in $\Sigma^{(n)}$.

Construction of the core tensor \mathcal{S} : The core tensor \mathcal{S} governs the interactions among the three examined modes. Its construction is implemented as:

$$\mathcal{S} = \mathcal{A} \times_1 \left(U_{c_1}^{(1)} \right)^T \times_2 \left(U_{c_2}^{(2)} \right)^T \times_3 \left(U_{c_3}^{(3)} \right)^T, \quad (10)$$

where \mathcal{A} is the initial tensor, $\left(U_{c_n}^{(n)} \right)^T$ is the transpose of $U_{c_n}^{(n)}$, \times_n is the n -mode product of a third order tensor [Lathauwer et al. 2000] and \mathcal{S} is a $c_1 \times c_2 \times c_3$ tensor.

The reconstructed tensor $\hat{\mathcal{A}}$: Finally, the reconstructed tensor $\hat{\mathcal{A}}$ is computed by:

$$\hat{\mathcal{A}} = \mathcal{S} \times_1 U_{c_1}^{(1)} \times_2 U_{c_2}^{(2)} \times_3 U_{c_3}^{(3)} \quad (11)$$

where $\hat{\mathcal{A}}$ is a tensor with the same size as \mathcal{A} . $\hat{\mathcal{A}}$ is a good approximation of \mathcal{A} (also known as low rank approximation of \mathcal{A}), in the sense that the Frobenius norm $\|\mathcal{A} - \hat{\mathcal{A}}\|_F^2$ is small

(element-wise squared differences [Lathauwer et al. 2000]). Moreover, $\hat{\mathcal{A}}$ contains less noise and additionally, latent associations are revealed by keeping only a subset of the dominant left singular vectors.

The generation of personalized item recommendation: The elements of the reconstructed tensor $\hat{\mathcal{A}}$ represent the final triplets (U, T, I) associated with the recalculated weight w' , which corresponds to the likelihood that user U will annotate MMD I with tag T . Therefore, the personalized recommendation process functions as follows: if user U poses tag T , then the N MMDs are selected that have the highest weights w' from triplets that contain both U and T .

3. EXPERIMENTAL EVALUATION

3.1. Datasets

For evaluation purposes, we prepared two real datasets ⁸, denoted as I-SEARCH and FLICKR datasets. The I-SEARCH dataset is created by researchers from 6 different European research institutes and universities within the multimodal engine of I-SEARCH, which consists of 3,532 triplets in the form user-MMD-tag with 358 users, 734 MMDs and 1,336 tags. For each MMD in the I-SEARCH dataset the multimodal descriptor is extracted following the extraction strategy of [Daras et al. 2012]. The FLICKR dataset is created by using Flickr's web services, where random text queries were posed to crawl the data. The FLICKR dataset consists of 63,172 triplets in the form user-image-tag with 8,262 users, 10,049 images and 21,866 tags. For each image in the FLICKR dataset the image descriptor is extracted based on the SIFT variant described in [Uijlings et al. 2010]. In both datasets tags are treated as regular text and thus, three preprocessing steps are followed: (a) tokenization based on a standard stop list (e.g. in, the, of, at, etc.); (b) tags are turned into lower case; and (c) all non-letter or non-digit characters in the tags are removed (e.g. dots, commas, question marks, etc.).

3.2. Evaluation Protocol

For the task of item recommendation, where in case of I-SEARCH and FLICKR the respective recommended items are in the form of MMDs and images, the following evaluation protocol was used: for each user, one of its triplets was randomly selected. The set of all selected triplets formed the test data, whereas the remaining triplets formed the training data. The task of recommendation is to predict the item in the hidden triplets. Following the evaluation protocol of similar works [Tso-Sutter et al. 2008; Nanopoulos et al. 2010], the quality of recommendations was measured in terms of recall. Thus, for a test user U that receives a list of N recommended items (top- N list), by posing a tag query T , recall is defined as the ratio of the number of relevant items in top- N list over the total number of relevant items (all items in the hidden triplets containing test user U and tag T). Other commonly used measures are precision and $F1$. However, according to Nanopoulos et al. [2010], the following two factors should be clearly mentioned: (a) For each user/tag combination in the test data, a constant number of items has to be predicted (items annotated with tag T by user U); and (b) only a pre-specified number N of recommendations is taken into account. Therefore, for this kind of evaluation protocol, it is redundant to evaluate precision (thus $F1$ too) because it is just the same as recall up to multiplicative constants. In all experiments, mean values are reported, where each experiment was repeated ten times. In order to speed up the training process for the FLICKR dataset, we performed each examined methodology into batches equal to the number of text queries that were used to

⁸<http://vcl.itl.gr/tiis/data.zip>

retrieve the images through Flickr’s web services and therefore, average recall is reported, weighted according to the number of users per batch.

3.3. Experimental Organization

The tensor factorization technique of HOSVD was implemented in Matlab Tensor Toolbox by Kolda and Sun [2008], where we had to conclude to the optimal percentage of the retained singular vectors c_1, c_2, c_3 of Equation (9), in order to reveal the latent associations among users, tags and MMDs/images. A percentage of $c_1 = c_2 = c_3 = 40\%$ suffices in terms of recall, because we found that higher values increase the HOSVD’s computational time [Lathauwer et al. 2000], without paying off in terms of the accuracy of prediction. The proposed Tag Propagation by Learning tag relevance and Tensor Factorization method (TPL-TF) is compared against Musicbox (MB) of Nanopoulos et al. [2010], an ad-hoc retrieval method of a pseudo-relevance scheme (AD-HOC) and HOSVD in both datasets.

In the MB method, the propagation of tags is performed as follows. Given a pair (U, T) , let \mathbf{I}_1 be the set of items (MMDs/images) that have been tagged by user U with tag T , and let \mathbf{I}_2 be the set of items that have not been tagged by user U with tag T . Then, for each $I_x \in \mathbf{I}_2$, a weight w is measured according to $w = \max_{I_y \in \mathbf{I}_1} \text{Sim}(I_x, I_y)$, where Sim is the similarity (normalized to $[0, 1]$) between MMD/image I_x and I_y , based on the L2 distance of the multimodal descriptors of MMDs and the SIFT descriptors of images for the I-SEARCH and FLICKR datasets, respectively. Thus, given the threshold parameter a , if it holds that $w \geq a$, then in the MB method a new triplet is propagated in the form (U, T, I_x) associated with the calculated weight $w = \text{Sim}(I_x, I_y)$. Thus, $\text{MB}(a)$ denotes the MB method, where a is the parameter to control the amount of propagated triplets. Note that the extreme case of $\text{MB}(1)$ equals the HOSVD method.

In the AD-HOC method, a pseudo-relevance feedback scheme is used, where the top- k content-based similar items (MMDs/images) are considered as positives, based on which the tag propagation is performed. Given a pair (U, T) , let \mathbf{I}_1 be the set of items (MMDs/images) that have been tagged by user U with tag T and for each $I_y \in \mathbf{I}_1$ let \mathbf{I}_2 be the set of the top- k content-based similar items. Then, let \mathbf{T}_{I_x} be the set of tags that have been assigned to item $I_x \in \mathbf{I}_2$. For each $T_x \in \mathbf{T}_{I_x}$, a new triplet is propagated in the form (U, T_x, I_x) , associated with a weight $w = \text{Sim}(I_x, I_y) * \text{Sim}(T_x, T)$, where $\text{Sim}(I_x, I_y)$ is the similarity between MMD/image I_x and I_y , as in the case of the MB method and $\text{Sim}(T_x, T)$ is the similarity between tags T_x and T according to Equation (2). In our experiments we set $k = |\mathbf{R}_Q|$.

For both datasets, we conducted three set of experiments: (a) the impact of the $|\mathbf{R}_Q|$ length on the proposed TPL-TF method; (b) comparison of TPL-TF against HOSVD, AD-HOC and MB for different number of recommended MMDs/images; and (c) performance of TPL-TF on users’ cold start degree. All experiments were conducted on a desktop PC with 4-core Intel i7-3770K CPU 3.4GHz and 16GB RAM, running Windows 7.

3.4. Impact of the $|\mathbf{R}_Q|$ Length

In Figure 10, we present the experimental results for the proposed TPL-TF method by varying length $|\mathbf{R}_Q|$ as a percentage of the total number of MMDs/images in the I-SEARCH and FLICKR datasets, respectively. In the top panel, we present the average number of identified positives based on the relevance feedback step of Section 2.2. As expected, by increasing length $|\mathbf{R}_Q|$, more positives are identified. This happens because in the initialization process of Algorithm 1 by increasing length $|\mathbf{R}_Q|$ we maximize the probability of retrieving a result R in \mathbf{R}_Q of equal or higher association degree of tag T_{in} with result R , than the respective association degree of tag T_{in} with query Q , expressed by the condition $SSE(T_{in}, \mathbf{T}_R) \leq SSE(T_{in}, \mathbf{T}_Q)$. By doing so, more positives are identified, along with the increase of the $|\mathbf{R}_Q|$ length.

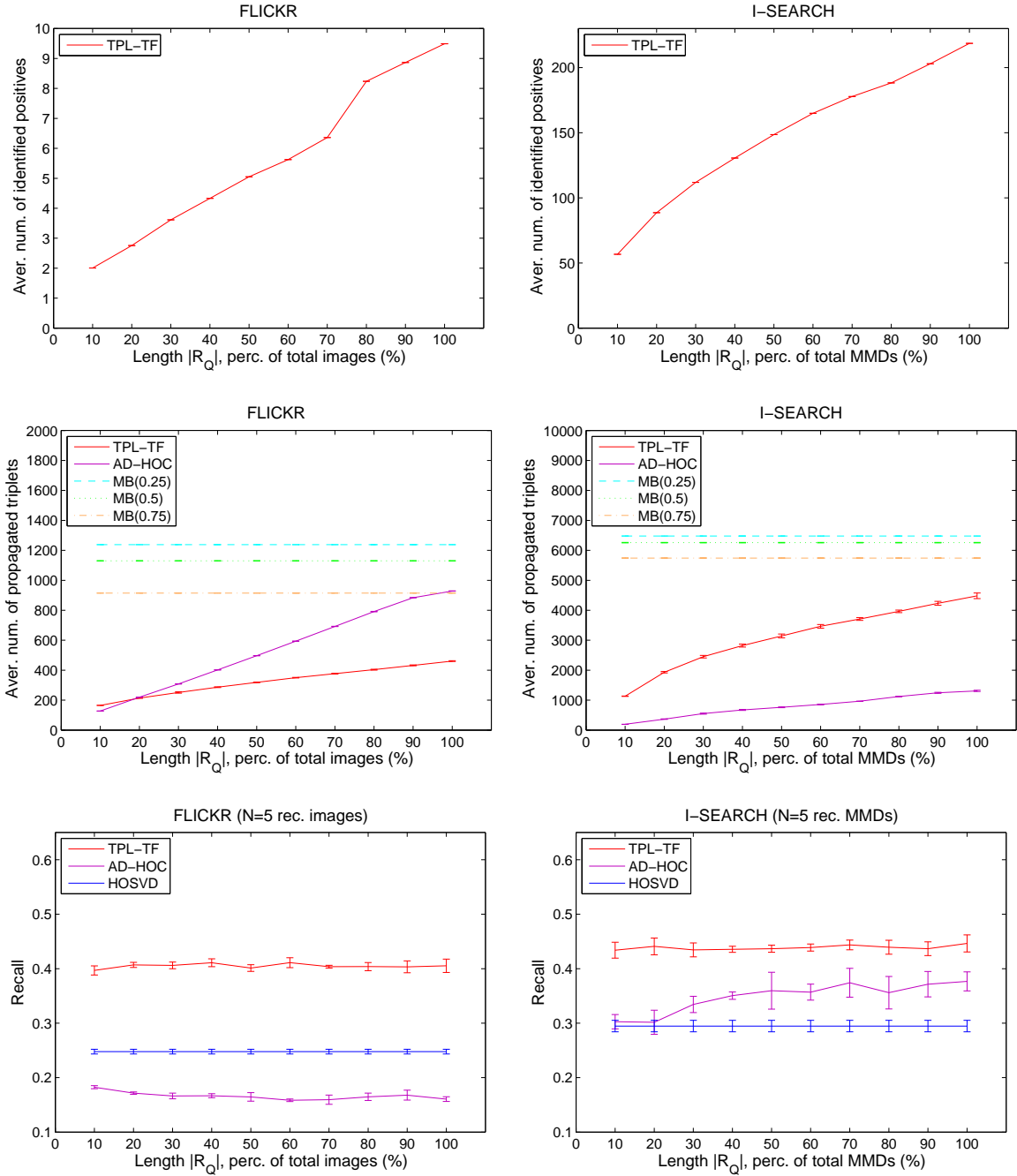


Fig. 10. Impact of the $|R_Q|$ on the proposed TPL-TF method. Top panel: Average number of identified positives. Middle panel: Average number of propagated triplets. Bottom panel: Average recall.

In the middle panel of Figure 10, we present the average number of propagated triplets for the proposed TPL-TF method based on the tag propagation step of Section 2.3. The increase of length $|R_Q|$ and thus the number of identified positives, results in an analogous increase of the average number of propagated triplets based on Equation (4). Therefore, the average numbers of propagated triplets for the FLICKR and I-SEARCH datasets are at different scale, since the respective average numbers of identified positives are also at different scale. Moreover, the average number of propagated triplets is presented for the AD-HOC method, where the increase of $|R_Q|$ results in analogous increase of propagated triplets. However, by considering the tag similarity $Sim(T_x, T)$ in the weights of the propagated triplets, due to “cold start” and sparsity problems it often holds that $Sim(T_x, T) = 0$ and thus the respective weights are also equal to 0. By doing so, the propagated triplets of $w = 0$ are discarded. This explains the different number of propagated triplets of AD-HOC in the evaluation datasets along with the increase of $|R_Q|$, where in the case of FLICKR overcomes the number of propagated triplets of TPL-TF, whereas in the case of I-SEARCH remains lower than TPL-TF. Additionally, the average number of propagated triplets is presented for the MB method by varying the a control parameter in the same way as presented in [Nanopoulos et al. 2010], following conservative (low number of propagated triplets) or aggressive (high number of propagated triplets) strategy, corresponding to high or low values of the control parameter a , respectively.

In the bottom panel of Figure 10, we demonstrate the impact of length $|R_Q|$ on the proposed TPL-TF method in terms of average recall for $N = 5$ recommended images/MMDs. In this set of experiments, HOSVD and AD-HOC are considered as baseline methods. For both datasets we can make the following observations. For all different values of length $|R_Q|$, the performance of TPL-TF is preserved higher than HOSVD, since the issues of “cold start”, sparsity and “learning tag relevance” are handled through the relevance feedback and tag propagation steps. Additionally, by increasing length $|R_Q|$, the recall of TPL-TF is preserved. This happens because despite the fact that the average number of propagated triplets is increased along with length $|R_Q|$, the calculated weights w of the propagated triplets become relative small according to Equations (5), (6), (7), (8) and thus, the performance of the TPL-TF method in terms of recall is not affected significantly. On the contrary, the AD-HOC method achieves different recommendation accuracy in the evaluation datasets. In the FLICKR dataset, the recall of the AD-HOC method is reduced along with the increase of length $|R_Q|$. In the I-SEARCH dataset, AD-HOC’s recall is increased along with the increase of length $|R_Q|$, where after a certain point it remains the same ($|R_Q|=70\%$). This happens because for large values of length $|R_Q|$ the content-based similarities become relatively low. Thus, the combination of content-based and tag similarities results in the propagated triplets’ low weights, having insignificant impact on the recommendation accuracy of AD-HOC. In contrast to TPL-TF, the AD-HOC method considers all results in R_Q as positives based on which the tag propagation is performed. In the AD-HOC method the issue of “learning tag relevance” is considered through the combination of the content and tag similarities, whereas in the TPL-TF method the issue of “learning tag relevance” is efficiently handled (a) by identifying positives through the Relevance Feedback of Algorithm 1 and (b) by calculating the weights of propagated triplets based on Equation (5).

In Figure 11, we evaluate the impact of length $|R_Q|$ on TPL-TF, in terms of computational time.⁹ In order to evaluate the tag propagation step, the computational time of HOSVD in TPL-TF is omitted, since it is similar to the HOSVD method. The increase of length $|R_Q|$ in the TPL-TF method results in an analogous increase of the computation time, where after a certain point (60% and 70% in FLICKR and I-SEARCH) starts to decrease. This happens because for large values of length $|R_Q|$, the Relevance Feedback method of Algorithm 1 identifies more positives in the first iterations, and thus making the stopping

⁹In the FLICKR dataset, the computation times of all batches are aggregated.

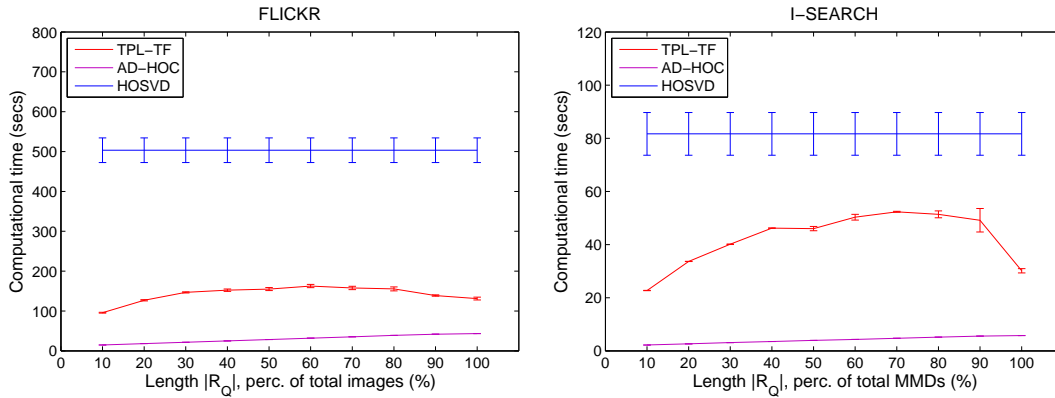


Fig. 11. Impact of the $|R_Q|$ on the proposed TPL-TF method, in terms of computational time (secs).

criterion of $SSE(T_{in}, \mathbf{T}_R) \leq SSE(T_{in}, \mathbf{T}_Q)$ more strict for the next iterations. By doing so, the algorithm terminates in the first iterations, resulting in the decrease of TPL-TF’s computational time. As we can observe the computation time of AD-HOC is lower than the one of TPL-TF, since it does not include the Relevance Feedback step. Regarding the MB method the computational times are 5.7 and 1.2 secs in the FLICKR and I-SEARCH datasets, respectively, which are extreme low since in the MB method the calculation of tag similarities of Equation (2) is omitted. Nevertheless, all tag propagation methods are of much lower complexity than the cubic complexity of HOSVD [Kolda and Sun 2008]. For example, the TPL-TF method for $|R_Q| = 10\%$ requires a percentage of 20% and 25% of HOSVD’s computational times in FLICKR and I-SEARCH, respectively. Therefore, based on the experimental results of Figures 10 and 11, in order to consider the lowest computational time for the TPL-TF method and the highest recommendation accuracy for the AD-HOC method of low complexity, in the rest of experiments for TPL-TF we set $|R_Q|$ equal to 10% in both datasets and for AD-HOC we set $|R_Q|$ equal to 10% and 70% in FLICKR and I-SEARCH, respectively.

Next, in Figure 12 we evaluate the performance of the proposed TPL-TF method against HOSVD, MB and AD-HOC methods, by varying the number (N) of recommended images/MMDs. The quality of recommendations of the MB method is lower than HOSVD for all values of a , either following conservative (low number of propagated triplets) or aggressive (high number of propagated triplets) strategy, corresponding to high or low values of the control parameter a , respectively. Moreover, for the MB method, we also conducted experiments for other values of the a parameter based on which we observed that even for the extreme case of MB(0.9), where the content-based similarity is high and the numbers of mean average propagated triplets are 187 and 1,966 for FLICKR and I-SEARCH, respectively, the mean average recall of MB(0.9) is preserved close to HOSVD, by propagating few noisy tags but not adequately handling the “cold start” and sparsity problems. For example, in case of $N=5$ the mean average recall of MB(0.9) is equal to 0.23 and 0.25 in FLICKR and I-SEARCH, respectively. This happens because the issue of the “learning tag relevance” is not handled in the MB method, and therefore for all values of parameter a , noisy tags are propagated, which consequently decrease the recommendation accuracy of HOSVD. The performance of AD-HOC is solely based on the linear combination of tags and MMDs/images content-based similarities. However, AD-HOC does not necessary handle the issue of “learning tag relevance” efficiently. By considering all top- k similar MMDs/images as positives, tag propagation is performed between not necessary conceptually similar MMDs/images. Thus, the tag similarities capture the relevance degrees of

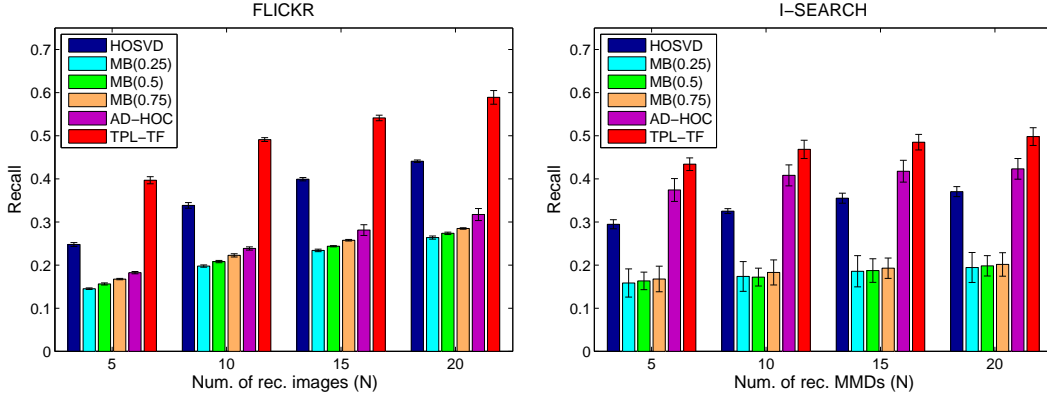


Fig. 12. Comparison of the proposed TPL-TF method against the HOSVD, MB and AD-HOC methods in terms of recall.

all top- k , not necessary conceptual similar, MMDs/images, which furthermore results in adding noisy tags. In the case of FLICKR there are few conceptually similar MMDs/images (positives) in the top- $k = |R_Q|$ content-based results (as they identified by the Relevance Feedback mechanism of TPL-TF in the top-panel of Figure 10), whereas in I-SEARCH there more conceptually similar MMDs/images in the top- k content-based results. This explains the different performance of the AD-HOC method in the evaluation datasets, where in FLICKR recall of AD-HOC remains lower than HOSVD, whereas in I-SEARCH is increased. On the contrary, the proposed TPL-TF method achieves higher recommendation accuracy than HOSVD and AD-HOC for both datasets, by efficiently detecting conceptually similar items based on the Relevance Feedback mechanism of Algorithm 1 and by capturing users' personal preferences bases on Equations (5) and (6). By doing so, TPL-TF handles the issue of “learning tag relevance” in a personalized manner, while avoiding to propagate noisy tags.

3.5. Performance Evaluation based on Users' Cold Start Degree

In the experiments of Figure 13, the performance of the proposed method is evaluated on the “cold start” problem, by controlling users' cold start degree. As presented in the top panel of Figure 13, the cold start degree is denoted by the number of users' $\text{freq}(U)$ appearances (triplets) in the training sets. The respective propagated triplets by MB, AD-HOC and the proposed TPL-TF method are illustrated in the middle panel of Figure 13, where for all examined methods the average number of propagated triplets is increased according to users' appearances in the training sets. Next, in the bottom panel of Figure 13, the proposed TPL-TF method is compared against HOSVD, AD-HOC and MB in terms of recall. We can make the following observations. Users' cold start degree affects negatively the HOSVD method, by decreasing the quality of recommendations, i.e. the recall of HOSVD is decreased along with the decrease of users' appearances in the training sets. The MB method reduces the quality of recommendations in both datasets, by performing tag propagation and ignoring the “learning tag relevance” issue. The AD-HOC method has similar performance on the evaluation datasets to the experiments of Figure 12. By not detecting efficiently conceptually similar results and thus, depending on the amount of the top- k results that are conceptually similar, AD-HOC adds noisy tags, accordingly. However, for the FLICKR dataset there is an exceptional case of cold start users (with 1-5 appearances), where the MB and AD-HOC method achieve slightly higher recall than HOSVD, by partially handling the “cold start” problem. This happens because for the FLICKR dataset in the case of cold start users, MB and AD-HOC propagate a relative low number of triplets, as depicted in the middle

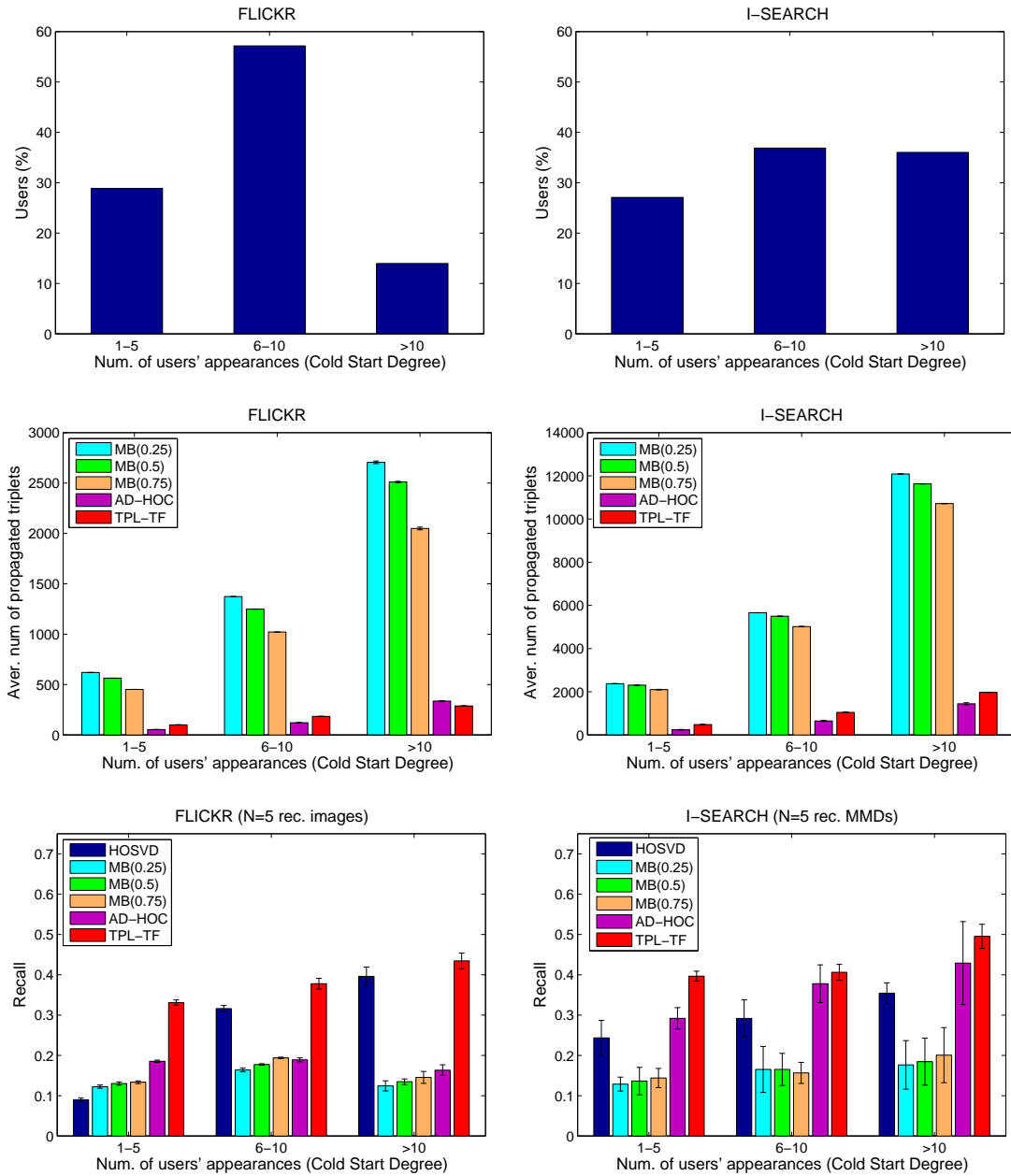


Fig. 13. Top panel: Users' cold start degree in the training sets. Middle panel: Average number of propagated triplets with respect to users' cold start degree. Bottom panel: Performance of the TPL-TF, HOSVD, MB, AD-HOC methods with respect to users' cold start degree.

panel of Figure 13 and thus, propagate few noisy tags resulting in the slightly increase of the recommendation accuracy. However for the rest cases of non-cold start users (i.e. 6-10 and > 10 appearances), the MB and AD-HOC methods highly reduce the quality of recommendations. In the I-SEARCH dataset, AD-HOC increases the retrieval accuracy of HOSVD. This happens because most of the top- k content-based similar results in the I-SEARCH dataset are conceptually similar and thus, the final weights of the propagated triplets, combined with the tag similarities, reflect on the tag relevance degrees. At this point we must mention the consideration of the tag similarities in the weights' calculations of the propagated triplets is the main difference between AD-HOC and the MB method, where in the latter only the content-based similarities are considered. Therefore, in contrast to MB, AD-HOC achieves to increase the recommendation accuracy in I-SEARCH. On the contrary, the proposed TPL-TF method, achieves high recommendation accuracy in both datasets especially for the case of cold start users, whereas for the non-cold users achieves to preserve higher or similar recommendation accuracy to HOSVD, by performing accurate tag propagation, handling the "learning tag relevance" issue and thus avoiding to propagate noisy tags. We must highlight that TPL-TF outperforms AD-HOC in the I-SEARCH dataset because (a) in contrast to AD-HOC, the TPL-TF method efficiently detects conceptually similar results and thus avoids propagating noisy tags and (b) users' personal preferences are considered according to Equations (5) and (6).

4. DISCUSSION AND CONCLUSION

The proposed TPL-TF methodology of content-based tag propagation and tensor factorization for item recommendation based on social tagging can efficiently handle several problems that exist in STS, which produce noise and decrease the quality of recommendations. Through the first step, the content information is exploited to retrieve the content-based similar results and then, a relevance feedback mechanism is activated so as to automatically identify the optimal number of conceptually similar results (positives), based on users' tag assignments. Next, through the second step, tags are propagated between the identified content-based and conceptually similar results, in order to handle the sparsity and "cold start" problems, where the relevance degrees between users, tags and items are calculated, in order to perform accurate tag propagation and consequently to address the issues of "learning tag relevance", sparsity and "cold start". Moreover, the ternary relation among users, tags and items is preserved by performing tag propagation in the form of triplets based on users' personal preferences and "cold start" degree. Finally, through the third step, the problems of polysemy and synonymity are addressed and the latent associations among users, tags and items are revealed. Thus, regarding the recommendations' quality of the proposed approach, we experimentally showed that it outperforms the related methods of Symeonidis et al. [2010], Nanopoulos et al. [2010] and an ad-hoc retrieval method of a pseudo-relevance feedback scheme, by performing the same tensor factorization technique of HOSVD on two different evaluation datasets. Finally, we present the recommendation methodology in the multimodal engine of I-SEARCH at <http://vcl.itl.gr/is/>, by simplifying users' interactions, where the recommended items are unified sets of media of different modalities in the form of MMDs.

Nevertheless, the tensor factorization technique of HOSVD has three major drawbacks (a) the cubic runtime to build the model; (b) the application of SVD on the three unfolded matrices of the tensor results in "memory overflows" for large-scale datasets; and (c) the tensor factorization technique is more biased to 0s than the 1s, since the tensor is extremely sparse and consequently the element-wise squared differences between the initial tensor and its approximation depend more on 0s than on 1s. These problems have already been addressed in several works like the methods described in [Kolda and Sun 2008]. Moreover, the tensor factorization approaches like the one of [Symeonidis et al. 2010] or the one of [Rendle et al. 2009] can only deal with prediction problems involving three categorical variables

(i. e. user, tag and item). However, in STS, additional information can be used, e. g., the words of a resource’s title, the age of a user, the time, etc. Factorization machines (FM) [Rendle 2010] have been recently proposed as a generic model that allows to encode any kind of additional data in tensor factorization techniques, in order to build Context-Aware recommender systems [Adomavicius et al. 2005]. However, the “cold start” problem is not addressed and therefore as a future work we plan to extend and evaluate the proposed methodology in FM.

REFERENCES

- ADOMAVICIUS, G., SANKARANARAYANAN, G., SEN, R., AND TUZHILIN, A. 2005. Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Transactions on Information Systems* 23, 1, 103–145.
- BELKIN, M. AND NIYOGI, P. 2003. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Comput.* 15, 6, 1373–1396.
- CATTUTO, C., SCHMITZ, C., BALDASSARRI, A., SERVEDIO, V. D. P., LORETO, V., HOTHO, A., GRAHL, M., AND STUMME, G. 2007. Network properties of folksonomies. *AI Commun.* 20, 245–262.
- CHAKRABARTI, K., ORTEGA-BINDERBERGER, M., MEHROTRA, S., AND PORKAEW, K. 2004. Evaluating refined queries in top-k retrieval systems. *IEEE Transactions on Knowledge and Data Engineering* 16, 2, 256–270.
- DARAS, P., MANOLOPOULOU, S., AND AXENOPOULOS, A. 2012. Search and retrieval of rich media objects supporting multiple multimodal queries. *IEEE Transactions on Multimedia* 4, 3, 734–746.
- FRENCH, J. AND JIN, X.-Y. 2004. An empirical investigation of the scalability of a multiple viewpoint cbr system. In *Proceedings of the CIVR*. 252–260.
- FURNAS, G., LANDAUER, T., GOMEZ, L., AND DUMAIS, S. 1987. The vocabulary problem in human-system communication. *Communications of the ACM* 30, 11, 964–971.
- GOLDER, S. AND HUBERMAN, B. 2005. The structure of collaborative tagging systems. *CoRR*.
- HALPIN, H., ROBU, V., AND SHEPHERD, H. 2007. The complex dynamics of collaborative tagging. In *WWW’07*. 211–220.
- HERLOCKER, J., KONSTAN, J., TERVEEN, L., AND J. RIEDL. 2004. Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.*, 5–53.
- HOI, S. C. H., LYU, M. R., AND JIN, R. 2006. A unified log-based relevance feedback scheme for image retrieval. *IEEE Trans. Knowl. Data Eng.*, 509–524.
- JING, F., ZHANG, B., LIN, F. Z., MA, W. Y., AND ZHANG, H. J. 2001. A novel region-based image retrieval method using relevance feedback. In *Proceedings of the 2001 ACM workshops on Multimedia: multimedia information retrieval*. MULTIMEDIA ’01. ACM, New York, NY, USA, 28–31.
- KOLDA, T. G. AND SUN, J. 2008. Scalable tensor decompositions for multi-aspect data mining. In *ICDM’08*. 363–372.
- LAMBIOTTE, R. AND AUSLOOS, M. 2005. Collaborative tagging as a tripartite network. *CoRR*.
- LATHAUWER, L. D., MOOR, B. D., AND VANDEWALLE, J. 2000. A multilinear singular value decomposition. *SIAM J. Matrix Anal. Appl.* 21, 1253–1278.
- LI, X., SNOEK, C., AND WORRING, M. 2008. Learning tag relevance by neighbor voting for social image retrieval. In *Proceedings of the 1st ACM international conference on Multimedia information retrieval*. MIR ’08. ACM, New York, NY, USA, 180–187.
- LIU, D. AND CHEN, T. 2005. Probabilistic relevance feedback with binary semantic feature vectors. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*. 513–516.
- MARINHO, L. B., NANOPOULOS, A., SCHMIDT-THIEME, L., JASCHKE, R., HOTHO, A., STUMME, G., AND SYMEONIDIS, P. 2011. Social tagging recommender systems. In *Recommender Systems Handbook* 1st Ed., F. Ricci, L. Rokach, and B. Shapira, Eds. Springer, 615–644.
- MARKINES, B., CATTUTO, C., MENCZER, F., BENZ, D., HOTHO, A., AND G. STUMME. 2009. Evaluating similarity measures for emergent semantics of social tagging. In *Proceedings of the 18th int. conf. on World Wide Web*. WWW ’09. ACM, New York, NY, USA, 641–650.
- NANOPOULOS, A., RAFAILIDIS, D., SYMEONIDIS, P., AND MANOLOPOULOS, Y. 2010. Musicbox: Personalized music recommendation based on cubic analysis of social tags. *IEEE Transactions on Audio, Speech and Language Processing* 18, 407–412.

- PAPADOPOULOS, S., KOMPATSIARIS, Y., AND VAKALI, A. 2010. A graph-based clustering scheme for identifying related tags in folksonomies. In *Proceedings of the 12th international conference on Data warehousing and knowledge discovery*. Springer-Verlag, 65–76.
- PORKAEW, K., CHAKRABARTI, K., AND MEHROTRA, S. 1999. Query refinement for multimedia similarity retrieval in mars. In *Proceedings of the seventh ACM international conference on Multimedia (Part 1)*. ACM, 235–238.
- QI, G.-J., AGGARWAL, C., TIAN, Q., JI, H., AND HUANG, T. 2012. Exploring context and content links in social media: A latent space method. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34, 850–862.
- RENDLE, S. 2010. Factorization machines. In *Proceedings of the 2010 IEEE International Conference on Data Mining*. IEEE Computer Society, Washington, DC, USA, 995–1000.
- RENDLE, S., MARINHO, L., NANOPOULOS, A., AND SCHMIDT-THIEME, L. 2009. Learning optimal ranking with tensor factorization for tag recommendation. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 727–736.
- SCAFER, J., FRANKOWSKI, D., HERLOCKER, J., AND JEN, S. 2007. Collaborative filtering recommender systems. In *The Adaptive Web*. 291–324.
- SEVIL, S. G., KUCUKTUNC, O., DUYGULU, P., AND CAN, F. 2010. Automatic tag expansion using visual similarity for photo sharing websites. *Multimedia Tools Appl.* 49, 81–99.
- SMEULDERS, A. W. M., WORRING, M., SANTINI, S., GUPTA, A., AND JAIN, R. 2000. Content-based image retrieval at the end of the early years. *IEEE Trans. Pattern Anal. Mach. Intell.* 22, 1349–1380.
- SYMEONIDIS, P., NANOPOULOS, A., AND MANOLOPOULOS, Y. 2010. A unified framework for providing recommendations in social tagging systems based on ternary semantic analysis. *IEEE Trans. on Knowl. and Data Eng.* 22, 179–192.
- TANG, J., QI, G.-J., ZHANG, L., AND XU, C. 2012. Cross-space affinity learning with its application to movie recommendation. *IEEE Transactions on Knowledge and Data Engineering*, accepted.
- TANG, J., YAN, S., HONG, R., QI, G. J., AND CHUA, T. S. 2009. Inferring semantic concepts from community-contributed images and noisy tags. In *Proceedings of the 17th ACM international conference on Multimedia*. ACM, New York, NY, USA, 223–232.
- TSO-SUTTER, K., MARINHO, B., AND SCHMIDT-THIEME, L. 2008. Tag-aware recommender systems by fusion of collaborative filtering algorithms. In *Proceedings of the 2008 ACM symposium on Applied computing*. ACM, New York, NY, USA, 1995–1999.
- UIJLINGS, J. R. R., SMEULDERS, A. W. M., AND SCHA, R. J. H. 2010. Real-time visual concept classification. *IEEE Transactions on Multimedia* 12, 7, 665–681.
- WU, L., FALOUTSOS, C., SYCARA, K., AND PAYNE, T. 2000. Falcon: Feedback adaptive loop for content-based retrieval. In *Proceedings of the 26th International Conference on Very Large Data Bases*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 297–306.
- WU, X., ZHANG, L., AND YU, Y. 2006. Exploring social annotations for the semantic web. In *Proceedings of the 15th international conference on World Wide Web*. ACM, New York, NY, USA, 417–426.
- XU, Y., ZHANG, L., AND LIU, W. 2006. Cubic analysis of social bookmarking for personalized recommendation. In *In Frontiers of WWW Research and Development - APWeb*. 733–738.
- YANG, Y., XU, D., NIE, F., LUO, J., AND ZHUANG, Y. 2009. Ranking with local regression and global alignment for cross media retrieval. In *Proceedings of the ACM International Conference on Multimedia*. 175–184.