On Social Networks and Collaborative Recommendation

Ioannis Konstas
Dept. of Computing Science
University of Glasgow
United Kingdom, G12 8QQ
konstas@dcs.gla.ac.uk

Vassilios Stathopoulos
Dept. of Computing Science
University of Glasgow
United Kingdom, G12 8QQ
stathv@dcs.gla.ac.uk

Joemon M Jose
Dept. of Computing Science
University of Glasgow
United Kingdom, G12 8QQ
jj@dcs.gla.ac.uk

ABSTRACT

Social network systems, like last.fm, play a significant role in Web 2.0, containing large amounts of multimedia-enriched data that are enhanced both by explicit user-provided annotations and implicit aggregated feedback describing the personal preferences of each user. It is also a common tendency for these systems to encourage the creation of virtual networks among their users by allowing them to establish bonds of friendship and thus provide a novel and direct medium for the exchange of data.

We investigate the role of these additional relationships in developing a track recommendation system. Taking into account both the social annotation and friendships inherent in the social graph established among users, items and tags, we created a collaborative recommendation system that effectively adapts to the personal information needs of each user. We adopt the generic framework of Random Walk with Restarts in order to provide with a more natural and efficient way to represent social networks.

In this work we collected a representative enough portion of the music social network last.fm, capturing explicitly expressed bonds of friendship of the user as well as social tags. We performed a series of comparison experiments between the Random Walk with Restarts model and a user-based collaborative filtering method using the Pearson Correlation similarity. The results show that the graph model system benefits from the additional information embedded in social knowledge. In addition, the graph model outperforms the standard collaborative filtering method.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Information Filtering, Retrieval Models, Selection Process*; I.5.1 [**Computing Methodologies**]: Pattern Recognition—*Models*

.

General Terms

Algorithms, Experimentation, Measurement, Performance

1. INTRODUCTION

With the recent advances in technology, there is an emerging presence of social media and social networking systems. From the point of view of a sociologist, social media can be characterised as "collective goods produced through computer-mediated collective action" [15]. In the case of multimedia enriched social network systems, such as last.fm, the collective goods are musical tracks and the collective action is the process of crafting individual profiles of musical preference and linking them either explicitly, via bonds of friendship, or implicitly, through collaborative annotation.

This collective action leads to the creation of an implicit social networking structure, which we aim to further explore. In particular given the success of item recommendation systems in commercial websites, such as Amazon.com and Netflix, it is considered worthwhile to revisit the recommendation problem through the novel perspective of social networking. In general, recommendation systems aim to provide personalised recommendations of items to users based on their previous behaviour as well as on other information gathered by item descriptions and user profiles. However, no emphasis has been placed yet on personalisation based explicitly on social networks.

The reason is that despite there is an increasing interest in the exploration of social networks, there does not exist a concrete dataset that includes both explicit bonds of friendships among users and free-form collaborative annotation of items. This is due to that most social media systems do not allow for free access to all user profiles or lists of friends.

Given the incentives of the widespread adoption of social networks and of the lack of some previous study that directly addresses the problem of efficiently integrating the added-value knowledge provided by those networks in the field of collaborative recommendation, we propose a new methodology that tackles the aforementioned issues. Within this context we make the following contributions:

- We introduce a dataset based on data from the last.fm social network that describes a social graph among users, tracks and tags, effectively including bonds of friendship and collaborative annotation.

- We evaluate a Random Walk with Restarts (RWR) model on this dataset and show that the incorporation of friendship and social tagging can improve the performance of an item recommendation system.

- We show that the RWR method outperforms the standard Collaborative Filtering (CF) method, which we also evaluate against the same dataset.

- We show that our method using the RWR method requires no training and successfully manages to capture the knowledge present in the social network.

The rest of the paper is organised as follows. In Section 2 we review related work and provide the necessary background for the RWR and CF methods used in this study. The methodology adopted, including the way we collected the data from the social network under study, the evaluation protocol and the comparison experiments performed are included in Sections 3 and 4. In Section 5 we analyse the results of the experiments and conclude with remarks and comments of our work in Sections 6 and 7.

2. RELATED WORK

2.1 Collaborative Recommendation

We may distinguish two broad categories of collaborative recommendation systems, namely content-based and collaborative filtering. A content-based system selects items based on the correlation between the content of the items (e.g. keywords describing the items, such as album genre, artists, etc., for music tracks) and the users' preferences [6]. However, it is limited to dictionary-bound relations between the keywords used by users and the descriptions of items and therefore does not explore implicit associations between users.

Collaborative filtering systems are divided into two categories, i.e. memory-based and model-based. In the memory-based systems [9] we calculate the similarity between all users, based on their ratings of items using some heuristic measure such as the cosine similarity or the Pearson correlation score. Then we predict a missing rate by aggregating the ratings of the k nearest neighbours of the user we want to recommend to. The problem with memory-based systems is that we have to decide on a rather arbitrary basis over parameters such as the number of neighbours. What is more, in the case of social networks there is no straightforward way to introduce similarities between users based on friendships and social tagging, other than some way of ad hoc interpolation of similarity weights from those different sources.

The model-based filtering systems assume that the users build up clusters based on their similar behaviour in rating of items. A model is learned based on patterns recognised in the rating behaviours of users using clustering, Bayesian networks and other machine learning techniques [1, 21]. The problem with model-based methods is that it is necessary to fine-tune several parameters of the model as well as the fact that the models produced might not generalise well in radically different context. What is more, as in the case of memory-based systems extra effort and training needs to be done in order to introduce knowledge from social networks.

2.2 Social Media

Many research publications have been lately revolving around the area of social media. In particular, several studies foucs on dataset collection and analysis from social networks. Das et al. [5] propose sample based algorithms that capture information in the neighbourhood of a user in dynamic social networks utilizing random walks. Halpin et al. [8] study the distribution of tags in the social bookmarking site del.icio.us and propose a generative model of collaborative tagging in order to evaluate the dynamics that lie beneath the act of collaborative recommendation. Their findings prove that the dataset collected follows a power-law distribution. Even though both studies examine social networks that are based on social tagging, they do not explore the dynamics of friendships among users.

Taking into account the power of free-form tagging of items by users other than their authors/owners, researchers also focus on tag recommendation. Subramanya and Liu [17] propose a system that automatically recommends tags for blogs, using similarity ranking in a manner similar to collaborative filtering techniques. Stromhaier [16] studies a novel idea in tag recommendation, which bridges the gap between the keywords issued by a user in a query and the tags actually used by a social system. He argues that the tags used by a user when performing a query exhibit his or her *intent*, whereas the annotations of items describe content semantics. As a result, he proposes a new form of purpose tags, which extract the intent of the user and facilitate goal-oriented search in a social network. Both studies underline the importance and discriminative power of social tagging, which is also validated by our work.

2.3 Random Walk

Several studies exist in the field of applying Random Walks on bipartite graphs. Craswell and Szummer [4] study a click-through data graph in order to perform item recommendation. Nevertheless, no social content is available between users. Yildirim and Krishnamoorthy [21] propose a novel recommendation algorithm which performs Random Walks on a graph that denotes similarity measures between items. They evaluate their system using data from MovieLens. Although, the use of the Random Walk model performs well in the context of recommendation, their use of an Item-Item similarity matrix raises some issues as to the ability of the system to extend when other similarities are introduced based on social tagging.

Recent work has also been done in the field of applying Random Walks over a social graph instead of bipartite graphs, similar to what we propose in this paper. Clements et al. [3] propose a single term query system performing Random Walks on graphs including users, items and tags. They use data from LibraryThing, an online book catalogue where users rate and tag books they have read. Due to lack of ground truth, they assume that the tags assigned to an item by each user are the same as they would use as query terms to retrieve the annotated item. We argue that this assumption is rather strong and that a user experiment would be more appropriate in order to properly establish the ground truth.

Hotho et al. [10] evaluate a variation of adapted PageRank on a dataset from del.icio.us, exploring folksonomies of bookmarks based also on collaborative annotation. However, since they evaluate their proposed algorithm empirically, any comparison attempts to their results becomes cumbersome. Although both studies are close to our approach, we use a different model, namely RWR, in which we explicitly include friendships in our dataset and perform collaborative recommendations instead of queries on the graph.

2.4 Background

2.4.1 Collaborative Filtering

In traditional collaborative systems, users explicitly give preference judgments for items in the form of ratings. These ratings are usually bounded and discrete. Past user ratings are then used to predict the preference towards items not rated yet. Memory-based collaborative filtering systems can be classified in *user based* and *item based* systems depending on the way past preference judgments are used.

A user based system makes new predictions by first finding users with similar ratings to an active user (i.e. the user whose preference to new items is predicted) and then takes a weighted combination of their ratings. More formally let a be the active user and i an item which is not rated by a. Then the predicted rating of a to i, $p_{a,i}$ is obtained by

$$p_{a,i} = \bar{r}_a + \frac{\sum_{u=1}^{N}(r_{u,i} - \bar{r}_u)w_{a,u}}{\sum_{u=1}^{N} w_{a,u}} \qquad (1)$$

where $r_{u,i}$ is the rating of user u for item i, $\bar{r}_a$ and $\bar{r}_u$ are the mean ratings of users a and u and $w_{a,u}$ is the similarity weight between users a and u.

On the other hand, in an item based system predictions are made by finding similarly rated items and then calculating a weighted combination of their ratings. In other words

$$p_{a,i} = \bar{r}_i + \frac{\sum_{k=1}^{M}(r_{a,k} - \bar{r}_k)w_{i,k}}{\sum_{k=1}^{M} w_{i,k}} \qquad (2)$$

where now $\bar{r}_i$ is the mean rating of item i and $w_{i,k}$ is the similarity weight between items i and k. The main motivation behind item based systems is the computational savings in calculating the item-item similarity matrix. In real world commercial applications items tend to be significantly less than users. In our study, tracks are significantly more than users and therefore a user-based approach is considered.

For calculating the similarity between users, or items, several similarity measures can be used. The most popular and the one used in this study, is the Pearson correlation score which is defined in (3), where σ_a is the standard deviation of user's a ratings.

$$w_{a,u} = \frac{\sum_{i=1}^{M}(r_{a,i} - \bar{r}_a)(r_{u,i} - \bar{r}_u)}{\sigma_a \sigma_u} \qquad (3)$$

It has been previously reported in [9] that penalising correlation values based on the number of the items that two users have co-rated can improve prediction accuracy. This approach decreases the confidence of correlation scores obtained from very few source of evidence. In this study, we use the significance weighting method proposed in [9]. More specifically, if the number of co-rated items between two users, n, is less than a thershold number of tracks Tr, then we multiply their similarity weight with $\frac{n}{Tr}$. It is also common in the literature, for both performance and accuracy, to use only a subset of users when making predictions. The selection is usually made by either setting a threshold on the similarity weight or by selecting the k most correlated users. In our experiments we use the latter approach.

In our data we do not have explicit preference judgments in the form of ratings. Instead we use the number of times a user has listened to a track (*playcount*) as an implicit indicator of preference. This is similar to the work of Morita and Shinoda [12] where the time spend on reading Usenet articles

was used as an implicit indicator of preference. Moreover, in order to incorporate the information from the users' social interactions and tagging, we adopt the following ad hoc procedure. We calculate three similarity weights based on the users playcount, users tag and users friendships respectively using the Pearson correlation coefficient and then use their weighted sum in place of $w_{a,u}$ in equation (3). More specifically,

$$w_{a,u} = \alpha w_{a,u}^{(I)} + \beta w_{a,u}^{(F)} + \gamma w_{a,u}^{(T)} \qquad (4)$$

where $\alpha + \beta + \gamma = 1$ and $w_{a,u}^{(I)}$, $w_{a,u}^{(F)}$, $w_{a,u}^{(T)}$ are the similarity weights obtained from the user tracks, user friendships and user tags respectively. The CF method described is going to be used as the baseline system in our study.

2.4.2 Random Walk with Restarts

A graph is a natural representation of data with some inherent relational structure. In a graph, objects and their relationships can be represented as nodes and weighted edges respectively, where weights denote the strength of a relationship. This abstraction allows us to integrate heterogeneous sources of data in a principled manner.

Measuring the relatedness of two nodes in the graph can be achieved using the Random Walks with Restarts (RWR) theory [11]. Starting from a node x, a RWR is performed by randomly following a link to another node at each step. Additionally, in every step there is a probability a to restart at x. Let $\mathbf{p}^{(t)}$ be a column vector where $p_i^{(t)}$ denotes the probability that the random walk at step t is at node i. $\mathbf{q}$ is a column vector of zeros with the element corresponding to the starting node set to 1, i.e. $q_x = 1$. Also let $\mathbf{S}$ be the column normalized adjacency matrix of the graph. In other words $\mathbf{S}$ is the transition probability table where its elements $S_{i,j}$ give the probability of j being the next state given that the current state is i. The stationary, or steady-state, probabilities for each node can be obtained by recursively applying (5) until convergence,

$$\mathbf{p}^{(t+1)} = (1-a)\mathbf{S}\mathbf{p}^{(t)} + a\mathbf{q} \qquad (5)$$

The stationary probabilities give us the long term visit rate of each node given a bias towards a particular starting node. Therefore, $p_i^{(l)}$, where l is the state after convergence, can be considered as a measure of relatedness between nodes x and i.

Random Walks with Restarts has recently attracted the interest of researchers in many different areas within Information Retrieval, starting from link analysis [13] to image annotation and retrieval [14, 19], text classification [20], click-through data analysis [4] and collaborative recommendations [7].

In this study we aim to study the effect of social networking and social tagging in collaborative recommendation using data crawled from the social networking service of last.fm. RWR allows us to directly predict the preference of users to particular tracks from the database by taking into account not only their music taste but also their tagging behaviour, social network as well as similarly tagged tracks. Specifically, we create our graph by representing users, tracks and tags as nodes in the graph. Relationships between users are encoded using bi-directional edges between the corresponding nodes. For each track that a

user has listened we also add an edge weighted by the number of times the user has listened the track. Similarly we add edges between tags and tracks as well as tags and users. More details are given in a later section.

3. METHODOLOGY

3.1 Data Collection

Since there are no publicly available datasets for experimentation that include both friendships and social tagging, we have collected live data from one of the most prominent social network sites. Last.fm [1] is a music social network that allows users to create a profile and augment it with the music tracks that they listen to, either from within the website itself or from their own private music collections. The most common practice is to initiate the automatic creation of a playlist with tracks similar to each other in some sense, much as a real radio would do via a simple stimulus search of mainly artists, tracks and tags. While the users listen to a track they have the ability to either move to the next track of the playlist or keep listening to the same. These actions can be interpreted as explicit negative and implicit positive feedback respectively. This feedback results in the population of activities and preferences in a user's profile. Users also have the opportunity to annotate tracks as well as explicitly befriend other users of the network. What makes last.fm different from other social networks is the fact that all of the statistics of each user's profile, including friendships, are available for everyone using the system to see and interact with.

As a result, we extracted a representative enough portion of the last.fm social network comprising of 3148 users, 30520 tracks, 12565 tags and 5616 unique bonds of friendship among the users collected, which we made freely available for other researches at [2] collected in November 2008. The process that led to this dataset is divided into two parts:

For the first part our intention was to collect users of the social network that have intense activity in their profile and thus were prone to participate more in the tasks of collaborative annotation and establishing friendships. We collected the top 50 most popular artists in the 34 largest countries, taking also care that these countries are scattered around the globe. For each of these artists we then collected their 50 most popular albums and for each track we extracted its 50 top fans. Finally, we reduced the final set of users by ensuring that each user collected had at least one friendship within the existing list. This resulted in a User-User (**UU**) matrix with a density of $1.2 * 10^{-3}$.

For the second part we populated the rest of the dataset with respect to the users collected in the previous part. In detail, for each user we downloaded the top 500 tracks in terms of times listened to and for each of those their 50 most popular tags. Then, we collected the top 50 tags annotated by each user ensuring that they are a subset of those captured from the tracks already acquired. In order to avoid sparsity problems in the final matrix we removed all tracks that had been listened to by less than 8 users along with their corresponding tags. The derived User-Track (**UTr**), User-Tag (**UTg**) and Track-Tag (**TrTg**) matrices have a density of $7.8 * 10^{-4}$, $4.6 * 10^{-4}$ and $3.2 * 10^{-4}$ respectively.
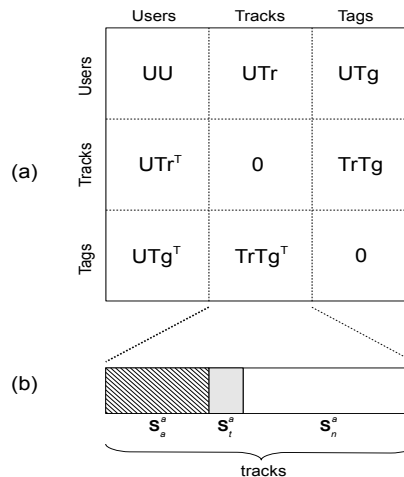
Figure 1: a. Social graph S and the sub-matrices that comprise it. b. Returned vector of tracks. S_t^a is the 20% randomly selected set of tracks user u_a has listened to used for evaluation, S_a^a is the remaining 80% that gets removed before the re-ordering of the vector and S_n^a is the set of tracks that the user has not listened to.

3.2 Dataset

The inclusion of all the matrices derived by the data collection method described resulted in the full social graph (**S**) depicted in Figure 1 with a density of $8.3 * 10^{-4}$. The **UTr** sub-matrix consists of the playcount of every track by each user, i.e. the number of times it has been listened to. Contrary to standard collaborative systems which use bounded ratings (e.g. from 1 to 5) for the corresponding sub-matrix, in our case playcount ranges from 1 to 11640, following a power-law distribution common to human activities [2, 5, 8].

In order to normalise the rest of the matrices that by definition contain binary information, e.g. whether a user has used a particular tag or not, we applied the following technique: We replace each bond of friendship per user in the **UU** sub-matrix with the average user playcount. This is crucial because, as stated above, the **UTr** sub-matrix may contain very large values and thus the binary values in **UU** will be suppressed.

In the case of the **UTg** sub-matrix, the associated tags for each user have been collected based on popularity as described in the previous section. Taking into account both this useful property and the potential underestimation due to the values of playcount, we applied an exponential decay function to the values of the tags of each user in the matrix, where the most popular tag gets the average user's playcount. The same process was performed for the **TrTg** sub-matrix accordingly.

3.3 Evaluation

We evaluated both the RWR and CF methods using either a part of **S** or as a whole, as will be clarified in the following section. In every experiment we follow the same per user evaluation protocol so as to allow for fair comparisons between the methods. For each individual user in the dataset we randomly select a list of 20% of the tracks he or she has listened to –which from now on we shall refer to as

($\mathbf{S}_t^a$)– and set zeros to the corresponding elements of $\mathbf{UTr}$ and $\mathbf{UTr}^\mathsf{T}$ sub-matrices. ($\mathbf{S}_a^a$ is the remaining 80% of tracks user u_a has listened to and $\mathbf{S}_n^a$ corresponds to the tracks u_a has not listened to.)

In the case of the RWR method we then create a query vector $\mathbf{q}$ so that $q_i = 1$ if $S_{u,i} > 0$, and $q_{ua} = 1$, where $i = [1..N]$, u_a is the current user under evaluation and N is the total number of columns of $\mathbf{S}$. In this way we impose the restarts of the model to be biased towards the elements (users, tracks and tags) that comprise the personalised profile of the current user (corresponding to row $\mathbf{S}_a$ of the adjacency matrix) and increase the stationary probability of those that are in the neighbourhood of it. Next, we normalise $\mathbf{q}$ so that $\|\mathbf{q}\|_1 = 1$. Then we perform a Random Walk on $\mathbf{S}$ which returns the stationary probability vector corresponding to u_a of all the tracks in the dataset. From this vector we remove the tracks user u_a has not listened to ($\mathbf{S}_a^a$) and re-order the remaining tracks in $\mathbf{S}_t^a$ and $\mathbf{S}_n^a$ (Figure 1) in descending order, with the first element having been assigned the highest probability, denoting higher preference. This vector is pruned to contain the top 1000 tracks.

In the case of the CF method, we predict the playcount of the tracks of $\mathbf{S}_t^a$ and $\mathbf{S}_n^a$ for each user u_a and return a ranked vector of those tracks in descending order of playcount. This ranked vector is also pruned to the top 1000 tracks.

3.3.1 Issues with the Evaluation Protocol

In the standard evaluation methodology for CF, we would predict the user playcount only on the withheld tracks of ($\mathbf{S}_t^a$) and calculate the Mean Average Error (MAE) in comparison to the original values. The reason for this choice to predict the playcount only for some of the tracks that have already been listened to, is because we usually do not have relevance judgement for all tracks by each user. Nevertheless, in the case of the RWR method we cannot directly compute the playcount for each track and instead are only able to get a rank of the tracks in order of predicted preference. As a result, we are forced to evaluate the method as if it was being used in real-time purposes, i.e. judging both the predictive accuracy and precision of the system at the same time by using standard information retrieval metrics. Consequently, this leads to the following problems:

Firstly, due to the sparsity of the $\mathbf{UTr}$ sub-matrix, namely that each user has actually listened to far fewer tracks than those existing in the dataset, it is relatively difficult to boost the $\mathbf{S}_t^a$ in the topmost positions of the ranked vector for each u_a. Secondly, some of the tracks belonging to $\mathbf{S}_n^a$ might actually be relevant to the tastes of the user, i.e. belong to the neighbourhood of the tracks in his or her profile and thus shall be ranked highly in the stationary probability vector. However, since they have not been listened to yet, they shall be considered as non-relevant. Taking into account that $\|\mathbf{S}_n^a\|_1 >> \|\mathbf{S}_t^a\|_1$, this phenomenon tends to be rather common. These problems are unavoidable but do not seem to play a significant justifying of our arguments.

4. EXPERIMENTS

In this section we illustrate the methodology for the individual experiments conducted for both the RWR and CF methods. The outcome of each experiment was the aggregation of the lists containing the top 1000 tracks in descending order for each user, which were tested against the ground truth, i.e. the $\mathbf{S}_t$ for all 3148 users.
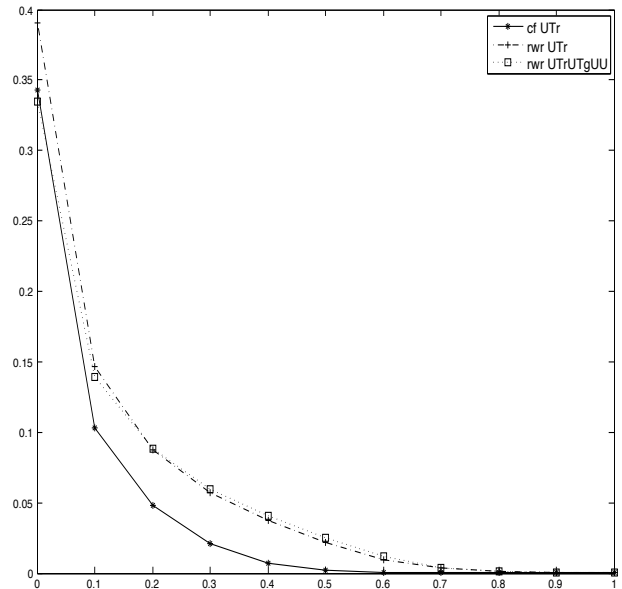

Figure 2: Interpolated Recall-Precision Curves for baseline CF method, RWR method using UTr sub-matrix only and RWR method using the full social graph S

4.1 Graph model

Parameter Estimation To find the optimal value for the restart probability (α) we performed a series of three preliminary experiments on a withheld part of the dataset using $\mathbf{S}$ by setting each time the parameter equal to 0.3, 0.5 and 0.8 respectively. The resulted MAP were 0.0105, 0.0123 and 0.0480 and both the interpolated precision-recall graphs and P@N had a similarly increasing tendency as we moved towards the method with $\alpha = 0.8$. This can be ascribed to the fact that by setting a higher restart probability we essentially suppress the model to go back to the initial query vector $\mathbf{q}$ more frequently and thus perform the random walk in the neighbouring elements of u_a, what can be interpreted as a stronger *personalised* model.

Having determined $\alpha = 0.8$, we performed 4 experiments using incrementally a larger part of $\mathbf{S}$, which was column normalised so that $\|S_j\|_1 = 1$, where $j = [1..N]$ and N is the number of columns of $\mathbf{S}$, as explained in an earlier section. The first experiment used only the $\mathbf{UTr}$ (and $\mathbf{UTr}^\mathsf{T}$) sub-matrix being the direct equivalent to the baseline CF method without any extra knowledge of social content. The second experiment was an extension of the previous with the additional inclusion of $\mathbf{UU}$, introducing in that way the bonds of friendship among the users. The third was also an extension of the first experiment but this time adding the $\mathbf{UTg}$ and $\mathbf{TrTg}$ (along with $\mathbf{UTg}^\mathsf{T}$ and $\mathbf{TrTg}^\mathsf{T}$), therefore including the role of social tagging. Finally, the last one used the whole social graph $\mathbf{S}$.

4.2 Collaborative Filtering

As illustrated earlier, the playcount variable lies in a much wider range than the usual rating scale in collaborative filtering systems. As a result, using equation (3) would result in heavy bias towards extreme playcount values of the nearest neighbours to u_a. In order to avoid this effect we remove the bias by standardising the playcount and mean of the nearest neighbours at prediction time by subtracting the mean $\bar{r}_a$ and dividing with the standard deviation σ_a of u_a. In this way, we project $r_{u,i}$ and $\bar{r}_u$ of the nearest neighbours in the range of values of u_a. Consequently equation (1) becomes:

$$p_{a,i} = \frac{\sum_{u=1}^{n} \left(\frac{r_{u,i} - \bar{r}_a}{\sigma_a} - \frac{\bar{r}_u - \bar{r}_a}{\sigma_a} \right) * w_{a,u}}{\sum_{u=1}^{n} w_{a,u}} \qquad (6)$$

Parameter Estimation The CF method we adopted uses two different parameters: The threshold number of common tracks between users, Tr in the significance weighting method employed by [9], as described in a previous section, and the number of nearest neighbours k, being users with the highest correlation with u_a. We estimated those parameters empirically using a withheld part of the dataset to $Tr = 20$ and $k = 15$.

For the purposes of our experiments we calculated 3 User-User similarity matrices using equation (3) based on the **UTr**, **UU** and **UTg** sub-matrices of **S**. The first similarity matrix $(w_{a,u}^{(I)})$ is the standard used in CF systems, whereas the two others $(w_{a,u}^{(F)}, w_{a,u}^{(T)})$ measure the correlation between users, based independently on friendships and collaborative annotation.

For reasons of comparison we performed 4 experiments in a similar fashion to those described using the RWR method. The first experiment used only $w_{a,u}^{(I)}$ and consisted the role of baseline method in our research. The second used both $w_{a,u}^{(I)}$ and $w_{a,u}^{(F)}$ with linear interpolation weights $\alpha = \beta = 0.5$ and $\gamma = 0$ in equation 4. The third one used $w_{a,u}^{(I)}$ and $w_{a,u}^{(T)}$ with $\alpha = \gamma = 0.5$ and $\beta = 0$ and the last one incorporated all 3 similarity matrices with $\alpha = \beta = \gamma = 0.33$.

As was mentioned earlier, a major drawback of the memory-based CF method we adopted is that there are numerous parameters which need fine tuning, such as the interpolation weights on the 3 similarity matrices. In an effort to make fair comparisons, we chose to give equal significance to the similarity matrices, in accordance with what we also did in the RWR method where the **UU**, **UTr**, **UTg** and **TrTg** sub-matrices were assigned equal weights in their contribution to **S**.

5. RESULTS

In this section we discuss the results we got from all the experiments performed in this study. In the case of the experiments with the RWR method it is shown in Table 1 that the addition of social network and social tagging information increases the number of relevant retrieved (num_rel_ret) tracks, though not always statistically significant when we compare subsequent experiments. There is also a notable decrease in MAP and Precision at high ranks and in parallel to num_rel_ret increase and Precision at lower ranks. This tendency accounts for the fact that with the progressive addition of social knowledge the method retrieves more tracks thus effectively increases its recall but at lower ranks (e.g P@200), causing a harm to its precision. What is more, it
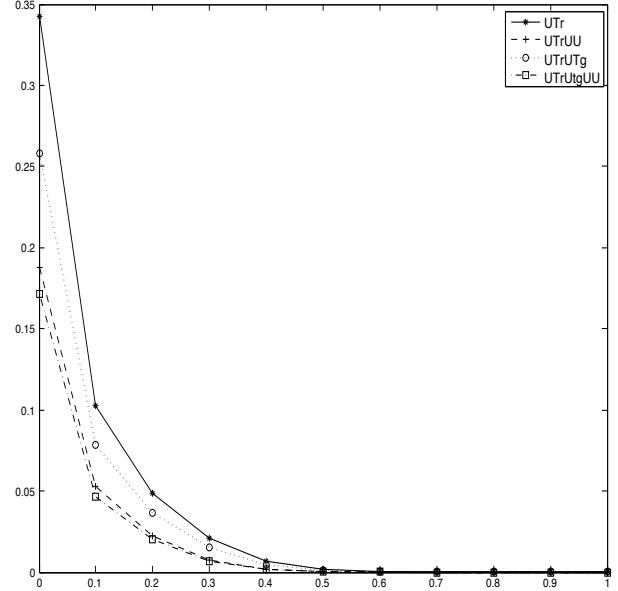

Figure 3: Interpolated Recall-Precision Curves for all 4 experiments of the CF method

is worth noting that the RWR method with the **UTrUTg** graph retrieves statistically significantly more relative tracks than the RWR method with the **UTrUU** graph (and also has higher P@200 and P@1000) compared to the simple RWR method. All three observations strongly confirm our argument highlighting the benefits of the social graph.

In Figure 2 it is shown that the RWR methods with the **UTr** and **UTrUTgUU** graphs respectively, outperform the baseline CF method. The difference is not significant at lower recall levels but it grows progressively from recall = 0.10 to around 0.5. Note that the RWR method using only the **UTr** sub-matrix is the direct equivalent to the baseline CF. The two RWR methods' curves distinguish very little with the RWR using the full social graph **S** exhibiting slightly better precision at recall levels between 0.3 and 0.6 but not statistically significant. What is more, the majority of the results of the RWR methods shown in Table 1 are statistically significantly higher than the baseline CF, further enhancing the evidence in favour of our argument, stating that the RWR method is more effective than the CF method.

Figure 3 shows the interpolated recall-precision curves of all 4 experiments of the CF method. It is evident that the addition of friendships and social tagging information deteriorates the performance increasingly. This is clearly due to the ad hoc way that this extra knowledge is introduced to the method via the linear interpolation of the User-User similarity matrices. This negative phenomenon is also illustrated in Table 1 where all of the results are constantly falling, with the induction of social content to the CF method. This supports our argument that the memory based CF method cannot provide with adequate non-trivial mechanisms to incorporate social knowledge.

6. DISCUSSION

The analysis of the results showed that the inclusion of

		MAP	P5	P20	P100	P200	P1000	num_rel_ret
cf	UTr	0.0305	0.1472	0.0934	0.0490	0.0355	0.0144	45268
	UTrUU	0.0149	0.0719	0.0470	0.0297	0.0227	0.0104	32862
	UTrUTg	0.0228	0.1046	0.0698	0.0398	0.0296	0.0129	40546
	UTrUTgUU	0.0134	0.0628	0.0416	0.0278	0.0215	0.0102	32106
rwr	UTr	**0.0498**	**0.1747**	**0.1229**	**0.0683**	**0.0505**	**0.0221**	**69514**
	UTrUU	**0.0498**	**0.1726**	**0.1229**	**0.0686**	**0.0507**	**0.0222**	**69742**
	UTrUTg	**0.0481**	0.1486	**0.1145**	**0.0678**	**0.0512**	**0.0227***	**71404***
	UTrUTgUU	**0.0480**	0.1483	**0.1139**	**0.0685**	**0.0513**	**0.0228**	**71645**

Table 1: **Summary of the results obtained by all experiments. (Bold typeset indicates statistical significance at** $p < 0.001$ **compared to the baseline CF model. * indicates statistical significane at** $p < 0.05$ **between consecutive experiments of RWR models)**

friendships and social tags in the graph does increase the recall of the RWR method. Although, the addition of friendships in the RWR method by using the **UTrUU** graph show some increase in the *num_rel_ret*, still it is not statistically significant. This is due to the sparsity of the **UU** sub-matrix, which can be ascribed to the nature of the social network we studied. The users of last.fm by definition place a stronger emphasis on the media provided and on a secondary basis on the explicit social part of it, i.e. spend time to make friends. This is a common phenomenon in the case of special interest social networks that revolve around a certain media, such as Flickr and YouTube in contrast with pure social networks such as Facebook and LinkedIn. This is also the reason for the recent addition of a feature in the former, which asks the users to integrate their lists of friends from networks belonging in the latter category, in an effort to increase the significance of the role that is being placed by the explicit indication of friendship. What could also be true is the fact that the users added in the friendship list of a single user may be friends in real life, without this always being interpreted to a consequent similarity in musical tastes.

On the other hand, the addition of social tagging when using the **UTrUTg** and then the whole social graph **S**, exhibited a relatively high and statistically significant increase in the *num_rel_ret* tracks. This can be explained in the opposite way of the aforementioned argument in the sense that users of special interest networks will spend a significant amount of their time in annotating the multimedia content. What is more, the act of collaborative annotation establishes a folksonomy [10] of tracks, that is a user-defined categorisation of items based on their own tags. Our hypothesis is that folksonomies provide with a more natural and direct link between users, than is the artificial and more implied case of similarity between users based on playcounts of similar tracks.

The RWR method using either the whole social graph **S** or some part of it, outperformed our baseline CF method. The graph model captures the relationships between users, tracks and tags in a less ad hoc way than the CF method and provides with more elaborate patterns and rules than the correlation measure between users in the case of user based CF. Even though, it is nothing more than a memory-based technique that does not try to model the given data based on an assumed distribution or a mixture of them as in the case of parametric models, it can be argued that it is superior in the personalised context it is applied on. A parametric model tries to capture the general pattern of users with similar preferences, whereas the RWR method is able to account for local personalised preferences of each user.

The results analysis also showed that the inclusion of social content in the CF method as we linearly combined more User-User similarity matrices progressively deteriorated all the results of the system. This provides with an extra hint that the model in general and the method of linear interpolation of correlation weights from User-User matrices is not appropriate enough in the case of very sparse matrices as was the case of **UU** and **UTg**. For this reason we argue that the memory-based CF method is not able to deal nontrivially with the addition of social knowledge and that the RWR method proves to be more robust in that it does not require explicit tuning or some other sort of training.

As explained earlier, the evaluation methodology adopted lies on the *de facto* base that we have a relatively small ground truth provided by the users as to the number of tracks they have actually listened to. This primarily accounts for the overall low precision numbers we get throughout Figures 2 and 3 and Table 1, though without this causing harm to the process of justifying our main arguments.

Finally, the dataset we collected from last.fm was successful in its purpose to point out the significance of the social knowledge of friendships and social tags in a track recommendation context. Although, there are studies concerning datasets on social graphs, to the best of our knowledge there does not yet exist a dataset describing a social graph that contains explicit indications of friendship.

7. CONCLUSION AND FUTURE WORK

In this work we addressed a number of issues concerning the role of social networks on collaborative recommendations. We created a dataset using data from the last.fm social network which includes information about users, tracks and tags and their inter-relationship, namely bonds of friendship among users and collaborative annotation of tracks. We then showed that the extra knowledge provided by the users' social activity can improve the performance of a recommendation system using the RWR method. We also applied a common memory-based CF item recommendation method used as a baseline, in which we included the social knowledge in a more artificial way, and showed that it is inferior to the RWR method.

A major consideration for the future, especially when we are dealing with real-time response as is the case of searching, would be the scalability of the RWR method in larger social graphs. Though the original algorithm implemented

in this study, does not require storagewise any precomputations, its on-line response time is not acceptable in real life situations. The authors of [18] propose fast approximations of the original RWR method ensuring up to 90% or better quality, which takes into account properties of social graphs, namely linear correlations and community-like structure.

An interesting approach to consider would be the potential commercial value of the item recommendation system we propose, in large-scale case studies such as Amazon.com and Netflix. By encouraging their users to create a social network, these companies can benefit from providing recommendations based not only on past purchase behaviour, but also based on their customers' social behaviour. As a result, we may witness a shift of the established at the moment paradigm of "people who like this also like that" to the more social "people who like me also like this" [15].

8. ACKNOWLEDGMENTS

9. REFERENCES

[1] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on*, 17(6):734–749, 2005.

[2] A.-L. Barabasi. The origin of bursts and heavy tails in human dynamics. *Nature*, 435:207, 2005.

[3] M. Clements, A. P. de Vries, and M. J. T. Reinders. Optimizing single term queries using a personalized markov random walk over the social graph. In *Workshop on Exploiting Semantic Annotations in Information Retrieval (ESAIR)*, March 2008.

[4] N. Craswell and M. Szummer. Random walks on the click graph. In *SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 239–246, New York, NY, USA, 2007. ACM.

[5] G. Das, N. Koudas, M. Papagelis, and S. Puttaswamy. Efficient sampling of information in social networks. In I. Soboroff, E. Agichtein, and R. Kumar, editors, *SSM*, pages 67–74. ACM, 2008.

[6] A. M. Ferman, J. H. Errico, P. van Beek, and M. I. Sezan. Content-based filtering and personalization using structured metadata. In *JCDL '02: Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries*, pages 393–393, New York, NY, USA, 2002. ACM.

[7] F. Fouss, A. Pirotte, J.-M. Renders, and M. Saerens. Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation. *IEEE Trans. Knowl. Data Eng.*, 19(3):355–369, 2007.

[8] H. Halpin, V. Robu, and H. Shepherd. The complex dynamics of collaborative tagging. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 211–220, New York, NY, USA, 2007. ACM Press.

[9] J. L. Herlocker, J. A. Konstan, A. Borchers, and J. Riedl. An algorithmic framework for performing collaborative filtering. In *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 230–237, New York, NY, USA, 1999. ACM.

[10] A. Hotho, R. Jäschke, C. Schmitz, and G. Stumme. *Information Retrieval in Folksonomies: Search and Ranking*. 2006.

[11] L. Lovasz. Random walks on graphs: A survey. In *Combinatronics*, volume 2, pages 1–46. Bolyai Society for Mathematical Studies, 1996.

[12] M. Morita and Y. Shinoda. Information filtering based on user behavior analysis and best match text retrieval. In *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 272–281, New York, NY, USA, 1994. Springer-Verlag New York, Inc.

[13] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

[14] J.-Y. Pan, H.-J. Yang, C. Faloutsos, and P. Duygulu. Gcap: Graph-based automatic image captioning. In *CVPRW '04: Proceedings of the 2004 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'04) Volume 9*, page 146, Washington, DC, USA, 2004. IEEE Computer Society.

[15] M. Smith, V. Barash, L. Getoor, and H. W. Lauw. Leveraging social context for searching social media. In *SSM '08: Proceeding of the 2008 ACM workshop on Search in social media*, pages 91–94, New York, NY, USA, 2008. ACM.

[16] M. Strohmaier. Purpose tagging: capturing user intent to assist goal-oriented social search. In *SSM '08: Proceeding of the 2008 ACM workshop on Search in social media*, pages 35–42, New York, NY, USA, 2008. ACM.

[17] S. B. Subramanya and H. Liu. Socialtagger - collaborative tagging for blogs in the long tail. In *SSM '08: Proceeding of the 2008 ACM workshop on Search in social media*, pages 19–26, New York, NY, USA, 2008. ACM.

[18] H. Tong, C. Faloutsos, and J.-Y. Pan. Fast random walk with restart and its applications. In *ICDM '06: Proceedings of the Sixth International Conference on Data Mining*, pages 613–622, Washington, DC, USA, 2006. IEEE Computer Society.

[19] J. Urban and J. M. Jose. Adaptive image retrieval using a graph model for semantic feature integration. In *MIR '06: Proceedings of the 8th ACM international workshop on Multimedia information retrieval*, pages 117–126, New York, NY, USA, 2006. ACM Press.

[20] Y. Xu, X. Yi, and C. Zhang. A random walks method for text classification. In J. Ghosh, D. Lambert, D. B. Skillicorn, and J. Srivastava, editors, *SDM*. SIAM, 2006.

[21] H. Yildirim and M. S. Krishnamoorthy. A random walk method for alleviating the sparsity problem in collaborative filtering. In *RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems*, pages 131–138, New York, NY, USA, 2008. ACM.