PLIERS: a Popularity-Based Recommender System for Content Dissemination in Online Social Networks

Valerio Arnaboldi
IIT-CNR
Via G.Moruzzi 1
Pisa, ITALY
v.arnaboldi@iit.cnr.it

Mattia Giovanni
Campana
IIT-CNR
Via G.Moruzzi 1
Pisa, ITALY
m.campana@iit.cnr.it

Franca Delmastro
IIT-CNR
Via G.Moruzzi 1
Pisa, ITALY
f.delmastro@iit.cnr.it

Elena Pagani
University of Milano and
IIT-CNR
Via Comelico 39/41
Milano, ITALY
pagani@di.unimi.it

ABSTRACT

Online social networks (OSNs) allow users to generate items and tag or rate them in order to help others in the identification of useful content. In this paper, we propose a novel tag-based recommender system called PLIERS, able to identify useful contents based on users' interests. It relies on the assumption that users are mainly interested in items and tags with similar popularity to those they already own. It reaches a good tradeoff between algorithmic complexity and the level of personalization of recommended items. To evaluate PLIERS, we performed a set of experiments on real OSN datasets, demonstrating that it outperforms the state-of-the-art solutions in terms of personalization, relevance, and novelty of recommendations.

CCS Concepts

•**Information systems → Recommender systems;** •**Human-centered computing → Social recommendation;** *Social tagging;*

Keywords

tag-based recommender systems; online social networks; content dissemination

1. INTRODUCTION

In this paper, we present *PLIERS (PopuLarity-based ItEm Recommender System)*, a novel *Tag-based Recommender systems* (TBRSs) [4] based on *folksonomies* [3]. It relies on the

assumption that a user is mainly interested in items and tags with popularity similar to that of the items she already owns, and that the similarity between items/tags can also highlight a semantic relationship between them.

In the following sections we present the reference notation for tag-based recommender systems, the existing solutions, and details on PLIERS. Then, experimental results on real OSN datasets show PLIERS performances in terms of personalisation, relevance and novelty of recommendations.

2. NOTATION AND RELATED WORK

Formally, a folksonomy can be represented with three node sets: users $U = \{u_1, \ldots, u_n\}$, items $I = \{i_1, \ldots, i_m\}$ and tags $T = \{t_1, \ldots, t_k\}$. Each binary relation between them can be described using adjacency matrices, A^{UI}, A^{IT}, A^{UT} respectively for user-item, item-tag and user-tag relations. If the user u_l has collected the item i_s, we set $a_{l,s}^{UI} = 1$, $a_{l,s}^{UI} = 0$ otherwise. Similarly, $a_{s,q}^{IT} = 1$ if i_s is tagged with t_q and $a_{s,q}^{IT} = 0$ otherwise. Furthermore, $a_{l,q}^{UT} = 1$ if u_l owns items tagged with t_q, and $a_{l,q}^{UT} = 0$ otherwise. The three matrices can be represented as a tripartite graph $G^T = (U, I, T, E)$ where U, I, and T are set of nodes representing users, items, and tags respectively, and E is the set of edges between nodes corresponding to the elements equal to 1 in the matrices. A bipartite graph $G^B = (U, V, E)$ may be used instead of a tripartite graph, with U the set of users, and V the set of either items or tags. In the following, we will consider bipartite user-item graphs with n users and m items where an edge between the user u_l and the item i_s indicates that u_l owns i_s.

ProbS [9] assigns a generic resource to each item i_s held by a target user u_t. The resource is evenly split amongst the users directly connected to the item. Subsequently, each user evenly splits the portion of the resource received amongst the items connected to her. The final score f_j^P of each item i_j is given by the sum of the portions of resources that are assigned to it after the two steps, or, more formally:

Table 1: Datasets properties.

Sample	Users	Tags	Links	$\overline{k}(T)$	$\overline{p}(T_U)$
MovieLens	5 K	17 K	105.6 K	6.14	52.75
Delicious	1.9 K	40.6 K	230.5 K	5.67	121.80
Twitter	5 K	194 K	508.5 K	2.62	74.24

$$f_j^P = \sum_{l=1}^{n} \sum_{s=1}^{m} \frac{a_{l,j} a_{l,s} a_{t,s}}{k(u_l) k(i_s)} \qquad j = 1, 2, \ldots, m \qquad (1)$$

where $k(u_l) = \sum_{j=1}^{m} a_{l,j}$ is the number of items collected by the user u_l and $k(i_s) = \sum_{j=1}^{n} a_{s,j}$ is the number of users interested in the item i_s. The set of f_j^P values determines a ranking of contents concerning the interests of u_t. ProbS tends to recommend items with the highest popularity.

HeatS [9] uses rules opposite to those of ProbS. Each resource is first split amongst the items related to each user, and then amongst the users connected to each item. The score of the item i_j for the target user u_t is:

$$f_j^H = \frac{1}{k(i_j)} \sum_{l=1}^{n} \sum_{s=1}^{m} \frac{a_{l,j} a_{l,s} a_{t,s}}{k(u_l)} \qquad j = 1, 2, \ldots, m \qquad (2)$$

HeatS tends to recommend non-popular items.

Hybrid (ProbS + HeatS) [2] calculates a linear combination of ProbS and HeatS using an *hybridization parameter* $\lambda \in [0, 1]$ such that by setting $\lambda = 0$ we obtain the pure HeatS, and with $\lambda = 1$ we get instead ProbS. The value of λ may be difficult to select in real situations.

PD and BHC [5] try to correct ProbS and HeatS. Preferential Diffusion (PD) divides the ProbS scores by the degree of the recommended item, with an exponent ϵ used as a parameter to control the normalization. Biased Heat Conduction (BHC) multiplies the HeatS score of each recommended item by its popularity, using an exponent γ similar to ϵ. An optimal tuning of the parameters could be difficult to achieve in practice.

3. PLIERS

PLIERS is inspired by ProbS and shares with it the same two steps. In addition, PLIERS normalizes the value obtained by ProbS when comparing an item i_j with one of the items of the target user, i_s, by multiplying the score by the cardinality of the intersection between the set of users connected to i_j and the set of users connected to i_s, divided by $k(i_j)$ (i.e., the popularity of i_j). In this way, items with popularity similar to the popularity of the items of the target user, and which possibly share the same set of users, are preferred.

The score of the item i_j is then:

$$f_j^{PL} = \sum_{l=1}^{n} \sum_{s=1}^{m} \frac{a_{l,j} a_{l,s} a_{t,s}}{k(u_l) k(i_s)} \frac{|U_s \cap U_j|}{k(i_j)} \quad j = 1, \ldots, m \qquad (3)$$

where U_j is the set of users connected to the item i_j and $k(i_j)$ is the popularity degree of the item i_j. The normalization introduced in PLIERS favours items whose popularity (i.e. number of connected users) is similar to that of the items already owned by the target user. All the procedures above can be equally applied to user-tag graphs, leading to the same considerations.

4. EXPERIMENTAL RESULTS

We compared PLIERS with reference TBRSs: HeatS, ProbS, Hybrid with $\lambda = 0.5$; PD with $\epsilon = -0.85$ and BHC with $\gamma = 0.8$ as in [5]. We used three benchmark datasets containing user-tag bipartite graphs. We assessed the accuracy of the obtained recommendations by calculating the level of personalization in terms of popularity of the recommended tags and the appropriateness of recommendations with respect to the users' interests. We performed also a link prediction task on the datasets [9, 8, 7]. It consists in randomly removing a few links from the graph and to calculate the degree to which the recommendations coincide with the removed links. A good recommender system should be able to approximate the original graph, although removing links changes the structure of the graph, and a complete reconstruction is not possible, particularly with sparse graphs.

Datasets Description. We used three bipartite user-tag graphs obtained from Twitter [1], MovieLens and Delicious [7, 9]. The graphs extracted from these datasets are very large (i.e., 1.6M users and 30.2M tags for Twitter, 1.9K users and 40.9K tags for Delicious, and 8.7K users and 39.2K tags for MovieLens). Due to memory constraints, we sampled portions of these graphs with maximum size of 5,000 users. Table 1 summarizes the characteristics of the obtained samples, where U, T, and L are respectively the number of users, tags, and links. $\overline{k}(T)$ is the average tag degree in the graph and $\overline{p}(T_U)$ is the average popularity of the tags for the average user. From Table 1, we can note that tags in Twitter are connected, on average, to fewer users than in the other datasets (i.e., $\overline{k}(T)$ is lower). This could lead to less accurate results in terms of link prediction.

Metrics. We defined an index V (*variance*), to calculate the average difference in terms of popularity between the recommended tags and those already owned by the users:

$$V = \frac{1}{n} \sum_{l=1}^{n} \frac{1}{r_l} \sum_{q=1}^{r_l} \sqrt{(k(t_q) - p(T_{u_l}))^2} \qquad (4)$$

where n is the number of users in the network, r_l is the number of recommended tags for user u_l and $p(T_{u_l}) = \frac{1}{z} \sum_{j=1}^{z} k(t_j)$ is the mean popularity of the tags originally linked to the user u_l with z the number of those tags. The *overlap* O measures the percentage of users connected to both the recommended tag and one of the tags of the target user, averaged for all the tags of the user and then for all the users. It gives us an idea of the potential interest for the users in the recommended tags. It is defined as:

$$O = \frac{1}{n} \sum_{l=1}^{n} \frac{1}{r_l} \sum_{q=1}^{r_l} \frac{1}{z} \prod_{k=1}^{z} J(U_{i_q}, U_{i_k}) \qquad (5)$$

where U_{i_q} is the set of users connected to the item i_q and $J(S_1, S_2)$ is the Jaccard's index, that measures the percentage of overlap between two generic sets S_1 and S_2. A good system should provide both a low V and a high O.

For link prediction, we used three standard metrics. The *recall* (R) index measures the number of recovered links within the first L recommendations for each user divided by L. The *precision* (P) measures the number of recovered links within the first L recommendations divided by the total number of recovered links, for each user. The *novelty* (N) index measures the capacity of a recommender system

Table 2: Experimental results. Values in bold are related either to PLIERS or to the systems that outperform it.

	PLIERS		ProbS		HeatS		Hybrid		PD		BHC	
	V	O	V	O	V	O	V	O	V	O	V	O
MovieLens	**41.90**	**0.118**	80.34	0.102	50.50	0.054	50.82	0.091	**41.54**	0.085	49.94	0.063
Delicious	**288.50**	**0.090**	422.87	0.085	**121.01**	0.007	299.052	0.087	**120.48**	0.026	**181.08**	0.044
Twitter	**91.01**	**0.017**	560.36	**0.021**	**73.22**	0.001	244.52	**0.020**	**73.00**	0.009	**73.13**	0.002

Table 3: R, P, and N for the first 10 recommended items with MovieLens, Delicious and Twitter.

	MovieLens			Delicious			Twitter		
	P	R	N	P	R	N	P	R	N
PLIERS	0.037	0.088	41.046	0.114	0.089	323.82	0.017	0.022	21.96
ProbS	0.036	0.091	105.110	0.111	0.083	492.87	0.041	0.054	569.95
HeatS	0.004	0.014	3.270	0.007	0.007	1.12	0.001	0.001	1.01
Hybrid	0.037	0.080	38.489	0.120	0.094	353.55	0.038	0.040	196.90
PD	0.023	0.048	0.005	0.036	0.032	0.00	0.002	0.002	0.00
BhC	0.019	0.032	0.005	0.081	0.051	0.14	0.002	0.002	0.00

to generate novel and unexpected results, generally related to items with low popularity, quantified by measuring the average popularity of the first L recommended items. A good system should have high P and R, and low N.

Results and Discussion. Table 2 shows values of V and O for the different datasets and TBRSs. Highlight in bold are the values better than those achieved by PLIERS. We note that PLIERS always yields the better trade-off. For V, PLIERS obtains values very close to the best results for two traces, and it always outperforms both ProbS and Hybrid. It yields the best O, or very close to the best with Twitter. With Delicious, HeatS, PD, and BHC perform better than PLIERS in terms of V. Yet, with this trace, PLIERS supplies an overlap that largely outperforms those of the solutions yielding better V. These results tell that PLIERS is able to recommend tags of comparable popularity with those of the tags already owned by the users, and also of higher (or similar) relevance than the other solutions.

Table 3 depicts the results of the link prediction task. As in [6], we removed 10% of the links. From the figures in the table, we note that PLIERS again supplies the best trade-off. Its R and P are always very similar to the results of ProbS and Hybrid. In the case of Twitter, PLIERS' P and R are worse than those of ProbS and Hybrid, but in this case tags are connected, on average, to fewer users than in the other graphs and the removal of random links has a higher impact on the graph structure, having a negative impact on the recommendations. In this case, recommending tags with high popularity (as done by ProbS and Hybrid) is probably more effective. However, the level of personalization is clearly worse than the one obtained by PLIERS, as shown by the V index. For the N index, PLIERS is always better than ProbS and Hybrid, and reaches a value that is closer to the value of $\overline{p}(U_T)$ (see Table 1). Hence, PLIERS is able to recommend tags of comparable popularity to that of the target user.

5. CONCLUSIONS

In this work, we proposed a new tag-based recommender systems called PLIERS that recommends tags or items with popularity as similar as possible to those already owned by the users. We compared PLIERS with other reference systems in the literature. The results indicate that PLIERS recommends tags with popularity closer to that of tags owned

by the users than the other solutions. In case of link prediction, PLIERS performs very well, with results comparable to the other existing recommender systems in terms of precision and recall, but providing better novelty in the recommendations.

6. ACKNOWLEDGMENT

7. REFERENCES

[1] R. Dunbar, V. Arnaboldi, M. Conti, and A. Passarella. The structure of online social networks mirrors those in the offline world. *Social Networks*, 43:39–47, 2015.

[2] C. Liu and W.-X. Zhou. An improved heats+ probs hybrid recommendation algorithm based on heterogeneous initial resource configurations. *arXiv preprint arXiv:1005.3124*, 2010.

[3] G. Quattrone, E. Ferrara, P. De Meo, and L. Capra. Measuring similarity in large-scale folksonomies. In *arXiv:1207.6037*, 2012.

[4] F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor. *Recommender systems handbook*, volume 1. Springer, 2011.

[5] F.-G. Zhang and A. Zeng. Information filtering via heterogeneous diffusion in online bipartite networks. *PlosOne*, 10, 2015.

[6] Z.-K. Zhang, C. Liu, Y.-C. Zhang, and T. Zhou. Solving the cold-start problem in recommender systems with social tags. *EPL (Europhysics Letters)*, 92(2):28002, 2010.

[7] Z.-K. Zhang, T. Zhou, and Y.-C. Zhang. Personalized recommendation via integrated diffusion on user–item–tag tripartite graphs. *Physica A: Statistical Mechanics and its Applications*, 389(1):179–186, 2010.

[8] Z.-K. Zhang, T. Zhou, and Y.-C. Zhang. Tag-aware recommender systems: a state-of-the-art survey. *Journal of computer science and technology*, 26(5):767–777, 2011.

[9] T. Zhou, Z. Kuscsik, J.-G. Liu, M. Medo, J. R. Wakeling, and Y.-C. Zhang. Solving the apparent diversity-accuracy dilemma of recommender systems. *PNAS*, 107(10):4511–4515, 2010.