SocRecM: A Scalable Social Recommender Engine for Online Marketplaces

Emanuel Lacic
Graz University of Technology
Graz, Austria
elacic@know-center.at

Dominik Kowald
Know-Center
Graz, Austria
dkowald@know-center.at

Christoph Trattner
Know-Center
Graz, Austria
ctrattner@know-center.at

ABSTRACT

In this paper, we present work-in-progress on *SocRecM*, a novel social recommendation framework for online marketplaces. We demonstrate that *SocRecM* is not only easy to integrate with existing Web technologies through a RESTful, scalable and easy-to-extend service-based architecture but also reveal the extent to which various social features and recommendation approaches are useful in an online social marketplace environment.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*Data mining*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Information filtering*

Keywords

social recommender engine; online marketplaces; Apache Solr

1. INTRODUCTION

Recommender systems aim at helping users find relevant information in an overloaded information space. Although various recommender frameworks are available nowadays, there is still a lack of frameworks that address important aspects in recommender systems research such as: easy integration in an existing infrastructure, scalability, hybridization and social data integration. To tackle these issues, we implemented *SocRecM*, a scalable and easy-to-integrate online social recommender framework whose purpose is not only to decrease the workload of developers via an easy-to-use framework but also to provide recommendation algorithms that utilize social data obtained from various data sources (e.g., Facebook, Twitter, etc.).

2. SYSTEM OVERVIEW

The first prototype of *SocRecM* was implemented in Java and can be found online as open-source software [1]. Figure 1 shows the system architecture of *SocRecM*. As featured, the engine can easily be integrated into a RESTful API for communicating with client applications. The *SocRecM* API provides methods for uploading marketplace and social data into the engine and for querying resource recommendations and benchmarking results. The marketplace and social data is gathered via respective connectors and pre-processed to be indexed by Apache Solr, which in turn offers powerful search and content analyzing functionalities (e.g., facets or MoreLikeThis queries) that are used by the recommendation algorithm implementations. Currently, *SocRecM* contains four types of algorithms to recommend resources (in our case products) to users, including MostPopular (MP), Collaborative Filtering (CF) [5], Content-based (C) [4] and Hybrid Recommendations (CCF) [1] (see also [3]). These algorithms are calculated based on either marketplace features, such as purchases (CF_p), title (C_t) and description (C_d) or social features, such as likes (CF_l), comments (CF_c), interactions(CF_{in}), social stream content data (C_{st}), groups (CF_g) and interests (CF_i). Additionally, the algorithms and data features are incorporated into hybrid algorithms (CCF_m for marketplace and CCF_s for social features).

The recommendation algorithms are invoked by the evaluator component and its jobs executor in order to evaluate them with respect to various IR metrics (e.g., Recall (R), Precision (P), nDCG, User Coverage (UC), etc.). The evaluation results are further used in *SocRecM* to tune the parameters of the algorithms, especially in the case of the hybrid approaches. Furthermore, the algorithms are called by the recommendation engine in order to forward the recommended resources back to the client.

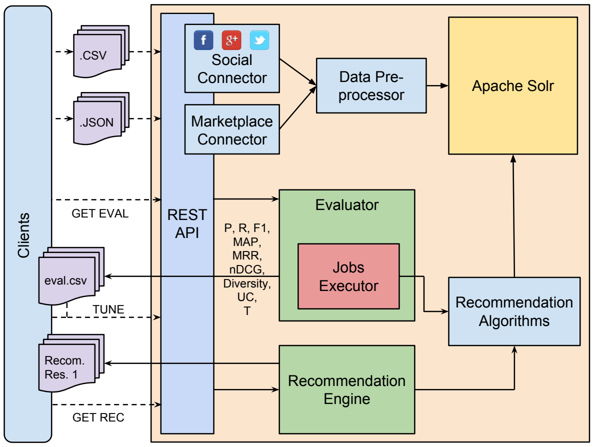
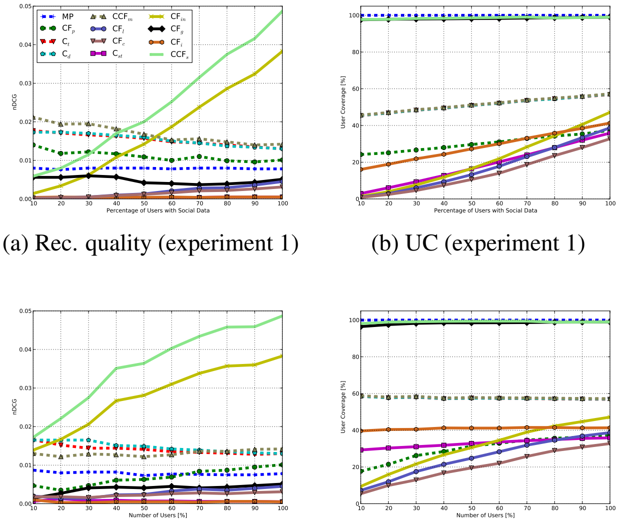


Figure 1: *SocRecM* system architecture.

[1] https://github.com/lacic/solr-resource-recommender

HT'14, Santiago, Chile, September 01-04, 2014.
Copyright 2014 ACM xxx-x-xxxx-xxxx-x/xx/xx ...$xx.xx.



(a) Rec. quality (experiment 1)

(b) UC (experiment 1)

(c) Rec. quality (experiment 2)

(d) UC (experiment 2)

Figure 2: Results of the two experiments in terms of recommendation quality (nDCG) and User Coverage (UC).

3. RESULTS

To demonstrate the effectiveness of our approach under different scenarios we conducted two "virtual" experiments with a social dataset gathered from the virtual world of SecondLife, which provides both: detailed marketplace purchase data and social data collected from users in the virtual world. The dataset contains 126,356 users, 122,360 products, 265,274 purchases, 1,839,783 social interactions, 510,145 social stream contents, 260,137 groups and 88,371 interests and was split into a training and test set using the method proposed in [3]. The following experiments were conducted using an IBM System x3550 server with two 2.0 GHz six-core Intel Xeon E5-2620 processors, a 1TB ServeRAID M1115 SCSI Disk and 128 GB of RAM using one Apache Solr 4.3.1 instance in the back-end.

In the first experiment we show how the recommendation quality in terms of the nDCG and User Coverage (UC) metrics [2] can be improved if the users' social data is provided in addition to the marketplace data. To simulate this, we extracted all users from our dataset that have both the marketplace and social data (10,996 users). Out of this subset, we randomly selected 10% of users and replaced them with users that only provided marketplace data. We continued until 100% of the data consisted of users with only marketplace data in their profiles. As demonstrated in plots (a) and (b) of Figure 2, the recommendation approaches based on marketplace data alone are fairly constant regardless of how many users have social data in their profiles. This is expected since the number of users neither increased nor decreased throughout the experiment. However, it is apparent that the recommenders based on the users' interactions perform best (CCF$_s$ and CF$_{in}$) but are also significantly depended on the number of social profiles in the dataset. An interesting finding in this context is that the content-based approach based on the users' social stream contents (CF$_{st}$) performed as poorly as the recommender based on the users' interests (CF$_i$).

In the second experiment we simulated a cold-start scenario for a new social marketplace system, under which we assume that all new users provide social data as it was the case with the start of Spotify, to determine if and how the recommendation quality would be affected by an increasing number of social users. In order to conduct this experiment, we again extracted all users from the dataset with both the marketplace and the social data (10,996) and eliminated the rest. Using these data, we created 10 different sets with an increasing number of users and evaluated the recommendation approaches on them, as demonstrated in plots (c) and (d) in Figure 2. As the number of users with marketplace and social data increased, the approaches based on the social data (CCF$_s$ and CF$_{in}$) delivered a much higher prediction quality than those that only used the marketplace data (e.g., CF$_p$). Moreover, the user coverage of the hybrid approaches based on social data (CCF$_s$) is much higher than the one based on the marketplace data (CCF$_m$). Comparing CF$_p$ with CF$_{in}$, shows that although both algorithms get a higher user coverage with more users, the social features ultimately provide better results.

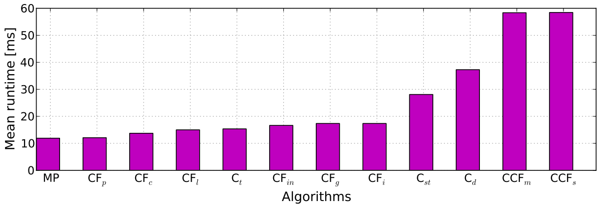


Figure 3: Mean runtime of the algorithms in *SocRecM*.

In addition to the recommendation quality, we compared the mean runtime (i.e., the time needed to calculate recommendations for a user) of the recommendation approaches shown in Figure 3. In general these results demonstrate that *SocRecM* is capable of providing near real-time recommendations for users since the maximum mean test time was only 58 milliseconds for the hybrid approaches CCF$_m$ and CCF$_s$.

4. FUTURE WORK

In the future, we plan to further extend our framework using different approaches to make sense of social data provided by users in a social marketplace environment. For example, we would like to implement a topic modeling approach based on LDA to combine comments with purchase data or to derive topics from the user's social stream to calculate similarities between users. Furthermore, we are interested in extending our framework to generate recommendations based on the user's geo-location data. Last but not least, we are interested in developing novel hybridization approaches for diverse social data sources to further increase the predictive power of our recommender framework.

Acknowledgments: This work is supported by Know-Center and the EU-funded project Learning Layers (Grant Nr. 318209).

5. REFERENCES

[1] R. Burke. Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, 12(4):331–370, 2002.

[2] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446, Oct. 2002.

[3] E. Lacic, D. Kowald, D. Parra, M. Kahr, and C. Trattner. Towards a scalable social recommender engine for online marketplaces: The case of apache solr. In *Proc.*, WWW '14. ACM, 2014.

[4] M. J. Pazzani and D. Billsus. Content-based recommendation systems. In *The adaptive web*, pages 325–341. Springer, 2007.

[5] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen. Collaborative filtering recommender systems. In *The adaptive web*, pages 291–324. Springer, 2007.