A Hybrid B2B App Recommender System

Alexandru Oprea¹, Thomas Hornung², Cai-Nicolas Ziegler³,
Holger Eggs¹, and Georg Lausen²

¹ SAP Commercial Platform, St. Leon-Rot & SAP Research, Darmstadt, Germany

{alexandru.dorin.oprea,holger.eggs}@sap.com

² Institute of Computer Science, Albert-Ludwigs-Universität Freiburg, Germany

{hornungt,lausen}@informatik.uni-freiburg.de

³ American Express, PAYBACK GmbH, München, Germany

cai-nicolas.ziegler@payback.net

Abstract. Recommender systems are integral to B2C e-commerce, with little use so far in B2B. We present a live recommender system that operates in a domain where users are companies and the products being recommended B2B apps. Besides operating in an entire new domain, the SAP Store recommender is based on a weighted hybrid design, making use of a novel confidence-based weighting scheme for combining ratings. Evaluations have shown that our system performs significantly better than a top-seller recommender benchmark.

1 Introduction and Motivation

The SAP Store caters to SME companies that aim to drive their business via B2B apps, e.g., for customer relation management or compliance. Many of these apps are geared towards specific industries and their needs. As the number of partners producing them is growing, so is the number of apps in the store itself and thus the complexity for the user (who represents a company) to actually *find* what he is looking for.

To actively help the user, we propose a hybrid recommender system that addresses exactly the needs of this specific B2B scenario. The system puts to use both knowledge-based, collaborative, and content-based sub-recommenders. Moreover, we present a novel hybrid weighting scheme [1] that incorporates *confidence scoring* for the predictions produced, so that sub-recommenders contribute for recommendations according to their confidence weight.

The system is live and can be used by logged-in users¹. We have conducted empirical evaluations via hold-out testing that show that the recommender outperforms the non-personalized top-seller recommender.

2 Recommender System Architecture

The architecture of the recommender is depicted in Figure 1. Overall, we have three different information sources for generating new recommendations: the

¹ See <http://store.sap.com>

Fig. 1. SAP Store recommender system architecture

user profiles (e.g., company size, industry, country), the app profiles (e.g, supported industries, business areas), and the transactional customer data (e.g., sales orders, downloads).

Initially, the knowledge-based component filters the list of relevant apps by a set of plausibility rules resulting in an unsorted set of candidate apps (1). These are fed to an item-item (2a) and user-item collaborative filter (CF), see (2b) [2].

To deal with the cold-start problem in cases where only sparse ratings are available for apps, a content-based augmentation scheme computes similarities based on the *cosine similarity measure* [3] between properties of the apps. For users that are new to the system, the similarity can be determined by comparing their profiles to other users based on their cosine similarity. This way, the two matrices will contain meaningful entries for all users and apps known to the system, and recommendations get more personalized once more context data is available. The scores of the two CF algorithms are combined by a *weighted mean* (cf. Section 2.1), and a sorted top- k recommendation list is returned.

The calculation of the matrices is done off-line as the computation is quadratic in the number of users or apps, respectively.

2.1 Weighting by Confidence Scores

The score of a recommended app is based on a weighted mean of the constituent item-item and user-item scores. Each of these gives an estimate of how much a user might like an app; e.g., Eq. 1 shows how a prediction score for the item-item case is determined for app a_m for user u : The ratings $r_u(b)$ of u for apps $b \in R_u$

he has already rated are weighted by their similarity to a_m , denoted $s(b, a_m)$, as an indicator if this app might be relevant for the user².

$$p_i(u, a_m) = \frac{\sum_{b \in R_u} s(a_m, b) \cdot r_u(b)}{\sum_{b \in R_u} s(a_m, b)} \quad (1)$$

Now, for each recommender score a *confidence score* is calculated, denoted c_i and c_u respectively, which is based on the number of supporting items or users of each prediction. These weights are used to determine the overall score p :

$$p(u, a_m) = \frac{c_i \cdot p_i(u, a_m) + c_u \cdot p_u(u, a_m)}{c_i + c_u} \quad (2)$$

The confidence score c_u for the prediction $p_u(u, a_m)$ tells us how *reliable* a prediction is. It grows with a growing number of supporting data points: For each user u_i , we calculate the z -score of his similarity with our current user u . We now sum these z -score similarities for all k users in user u 's neighborhood [2]. The sum is divided by k and the resulting value gives us the average normalized similarity of all the users whose ratings have been taken into account for $p_u(u, a_m)$. The same is done for the item-based case.

Since we are making use of standard z -scores, the linear combination shown in Eq. 2 based on the two confidence weights is sound. The confidence scheme represents a powerful means to adjust the hybrid recommender's weighting according to the predicted reliability of each of the two sub-recommenders.

3 Performance Evaluation

In order to test the performance of the presented hybrid recommender using our novel confidence-based weighting scheme, we conducted an empirical evaluation with real-world data of 5,233 users (e.g., companies registered for and using the SAP Store) having purchased or expressed interest in 615 app solutions.

The frequency distribution in Fig. 2(a) shows leads per app, i.e., how many companies have purchased or expressed interest in each app, sorted in descending order. The log-log plotted graph exhibits a power-law distribution, so a small number of apps attracts a high number of leads. This is confirmed by Fig. 2(b), showing that the top-5 apps accumulate 29% of all leads, and top-100 capture 90%. We thus conjecture that a non-personalized top-seller recommender, which only recommends the top- N most popular apps, will perform very well.

We adopted a hold-out cross-validation approach for testing, where one rating r_v of a user is withheld and all others are used to define his profile and calculate predictions, aiming to recommend exactly r_v . For baselining, we compared our recommender's performance with that of the top-seller recommender. The evaluation task for each of the two recommenders was to produce a list of top- N recommendations and count in how many cases the produced list contained r_v .

The evaluation is shown in Tab. 1. All results exhibit statistical significance at the $p < .05$ level, so we see the hybrid approach outperforms the top-seller.

² The score p_u for the user-item case is computed in an analogous way, with additional consideration of the user's average rating to level the effect of subjective ratings.

Fig. 2. Log-log frequency distribution of leads per app (a) and cumulative share of leads by number of apps (b)

Table 1. Performance benchmark results

	Top-1	Top-3	Top-5	Top-10
Hybrid recommender	10.9%	24.4%	33.5%	51.2%
Top-seller	6.6%	18.9%	27.6%	43.4%

4 Conclusion and Outlook

We have presented our recommender for the new domain of B2B apps, making use of a novel hybrid weighted scheme based on confidence scoring. Our first evaluations have shown very promising results and the system has gone live into operational use at SAP. In the future, we want to tune the recommending algorithms further and aim at doing the matrix calculations in real-time, using HANA [4], SAP’s new high-performance in-memory database.

References

1. Burke, R.: Hybrid Web Recommender Systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) Adaptive Web 2007. LNCS, vol. 4321, pp. 377–408. Springer, Heidelberg (2007)
2. Adomavicius, G., Tuzhilin, A.: Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. *IEEE Trans. Knowl. Data Eng.* 17(6), 734–749 (2005)
3. Baeza-Yates, R.A., Ribeiro-Neto, B.A.: Modern Information Retrieval - The Concepts and Technology Behind Search, 2nd edn. Pearson Education Ltd., Harlow (2011)
4. Färber, F., May, N., Lehner, W., Große, P., Müller, I., Rauhe, H., Dees, J.: The SAP HANA Database – An Architecture Overview. *IEEE Data Eng. Bull.* 35(1), 28–33 (2012)