Recommender Systems for Banking and Financial Services

Andrea Gigli
MPS Capital Services
Viale Mazzini, 23
Siena, Italy 53100
andrea.gigli@mpscs.it

Fabrizio Lillo
Università di Bologna
Viale Quirico Filopanti 5
Bologna, Italy 40126
fabrizio.lillo@unibo.it

Daniele Regoli
Scuola Normale Superiore
Piazza dei Cavalieri 7
Pisa, Italy 56126
daniele.regoli@sns.it

ABSTRACT

In this work we demonstrate the usefulness of the application of Recommender Systems in the financial domain. Specifically we investigate a dataset, made available by a major European bank, containing the purchases of a large set of investment assets by 200k investors. We also present some preliminary results of the application of network analysis via statistical validation to identify clusters of investment assets.

KEYWORDS

Finance, Collaborative Filtering, Networks, Statistical Validation

ACM Reference format:
Andrea Gigli, Fabrizio Lillo, and Daniele Regoli. 2017. Recommender Systems for Banking and Financial Services. In *Proceedings of RecSys 2017 Posters, Como, Italy. Copyrights held by the authors. , August 27-31,* 2 pages.

Introduction

Banking and Financial Services, being them provided by incumbent Banks or by FinTech companies, are looking seriously at machine learning and information retrieval fields in order to leverage the data at their disposal to provide tailored services and customized experiences to their customers.

One of the fields of computer science which can support this attempt is the one represented by Recommender Systems (*RecSys*), which has been heavily investigated in the last years by the research community as well as the most promising companies in the e-commerce and entertainment fields.

In this work we show the usefulness of some *RecSys* algorithms in suggesting investment assets to a large panel of investors. This is done by using a large dataset provided by a major European bank and comparing the performance of three different *RecSys* against two baseline models in the task of suggesting investment assets.

The Dataset

The recommender system implementation and analysis have been done on a dataset with financial investment information, made available to us by a European bank during a research collaboration program, which contains 224,885 clients, 1,288,315 transactions and information related to 7 different asset types, 23 rating levels, 6 order channels, 12 industrial sectors, 8 maturity buckets, 5 coupon types, 2 product complexity levels.

The records span a period of twelve months and all data entries are properly hashed, anonymized and organized as a table, where each record represents a purchase defined by: execution date, *user* data (client, branch and account identifiers) and traded *item* data (type of asset, transaction currency, asset country, time to maturity, complexity, industrial sector, industrial group, industrial sub-group, rating, coupon type, trading channel, buy-sell type).

Having no information on the traded volume per transaction nor the client total wealth at the time of each trade, we model the recommendation problem on the basis of the binary information purchased/not purchased item.

RecSys 2017 Poster Proceedings, August 27-31, Como, Italy.

Implicit feedback recommender for financial investments

To better capture clients' preferences we compare three different *RecSys* algorithms. All of them required to test different combinations of features at our disposal in order to define *user* and *item* entities. After some trials and analysis, we defined the *user* as a combination of client ID and bank branch, and the *item* as a combination of asset type, country, time to maturity, coupon type, industrial sector and rating.The results for other aggregations are qualitatively analogous.

The first algorithm we tested is the Bayesian Personalized Ranking algorithm [3] where we use a matrix factorization method maximizing the posterior probability of user preference structure, and tune model's parameters via 5-fold cross-validation. The second one is the Alternating Least Squares algorithm [1] using 30 latent factors and a regularization factor equal to 0.01. The third one is an adaptation of the Word2Vec algorithm [2] that we call *Asset-Embedding* in the following. In this case we treated the clients' portfolios as they were documents, each asset as a word, and vector-represented each asset by the portfolio it belongs to via continuous bag of words in a 300 dimension space.

The *RecSys* algorithms mentioned above are evaluated through various tests against two benchmark algorithms based on most popular items by number of users (POP.u) or by number of transactions (POP.trans):

(1) Average Accuracy of the user preference structure (see [3]);
(2) Expected percentile ranking, as defined in [1] (the lower, the better);
(3) the Area Under the ROC curve.

Other metrics (e.g. *novelty* and *coverage*) have been calculated but are left out for lack of space. Different train/test sampling methodologies were used:

(1) leave-one-out: removing randomly from train one purchased asset for each user (who has at least 5 purchases);
(2) leave-last-out: removing from train the last (in time) asset purchased by each user;
(3) 20% level sampling: removing randomly from train 20% of interactions.

Table 1: Evaluation metrics for leave-last-out test methodology, with variable number of most purchased items excluded from test set.

most purchased excluded items	recommender system	Average Accuracy	*Rank*	AUC
0	BPR-MF	**0.961**	**3.878**	**0.970**
	Asset Embedding	0.951	4.975	0.950
	ALS	0.954	4.590	0.954
	POP.u	0.949	5.119	0.958
	POP.trans	0.950	5.044	0.958
20	BPR-MF	**0.941**	**5.919**	**0.951**
	Asset Embedding	0.906	9.389	0.903
	ALS	0.909	9.080	0.906
	POP.u	0.916	8.404	0.926
	POP.trans	0.917	8.286	0.927
50	BPR-MF	0.885	11.537	**0.914**
	Asset Embedding	0.859	14.105	0.874
	ALS	**0.917**	**8.259**	0.913
	POP.u	0.825	17.472	0.819
	POP.trans	0.820	18.032	0.817

Due to limited amount of space, we here report the results for the leave-last-out case only.

Given that a good *RecSys* should give suggestions relevant and specific to the user and expand user's taste into neighboring areas, we run the above tests after removing $n = \{0, 20, 50\}$ most popular items from the test set. In this way if a *RecSys* performs well with 0 popular items removed and poorly with 50, it is reasonable to deduce that maybe it is just good in suggesting popular items but not items related to the specific interests of the user.

Table 1 displays the results of our study for the leave-last-out train/test sampling case.It shows that all the *RecSys* we propose perform extremely well on the dataset at our disposal, in terms of both average accuracy and ranking structure (expected percentile ranking - *Rank* - and AUC). BPR-MF is the best performer when no popular items is excluded from the test set and its advantage doesn't reduce when we increase the number of popular purchased item removed from the test set. ALS performs similarly well, while Asset Embedding performs better than POPs when the number of popular items excluded from the test set is at least 50, but it never beats BPR-MF and ALS.

Toward a network-based RecSys for banking and financial services

Besides the training of the recommender system shown above and the detailed test previously mentioned, we performed an analysis of the dataset seen as it were a bipartite network users → items. We implemented a statistical validation procedure [4] to get a statistically significant projection on the item module of the network. We used this statistically filtered network:

- to identify items' communities: each community represents the set of items that are purchased together by users in a statistically over-expressed way with respect to a random rewiring of the bipartite network keeping fixed the assets' degrees;
- to identify the features that are statistically over-expressed (or under-expressed) inside communities;
- to compute, for each user, a raking of communities of items based on the p-values of the hyper-geometric distribution of the number of purchased objects in the different communities.

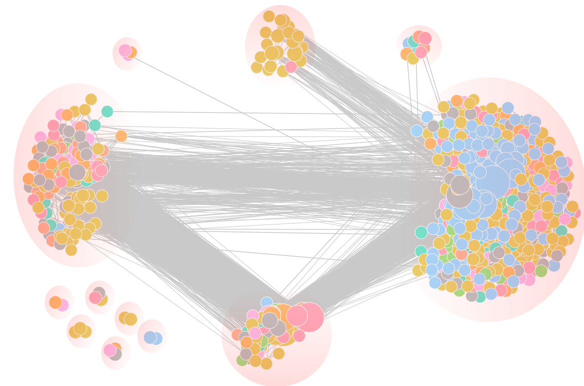

Figure 1: Communities (pink regions) of assets detected on the statistically filtered asset graph projection. Color denotes `sector` attribute.

As an example of the possible network analysis, Figure 1 shows the statistically filtered network derived by applying the validation algorithm to the bipartite network with the same specification of users and items as in Table 1, for 1% confidence threshold. There are 4 big connected communities, 2 smaller ones (but still connected) and 6 small isolated communities. Color of nodes (i.e. of assets) refers to different value of `sector` attribute. As an example, the light-blue sector, Governmental assets, results to be statistically over-expressed in the rightmost community, and under-expressed in the leftmost and in the bottom one. This evidence indicates that statistically filtered investors' decisions could be used to cluster assets: a promising starting point to build a statistically guided algorithm for recommendations. This is part of a work in progress for future publication.

ACKNOWLEDGMENTS

The authors would like to thank MPS Bank for supporting the collaboration, Francesco Mainieri (MPS) for essential support in extracting the data and Franco Maria Nardini (CNR, Pisa) for useful comments. FL and DR acknowledge support by the European Community's H2020 Program under the scheme INFRAIA-1- 2014-2015: Research Infrastructures, Grant Agreement No. 654024 SoBigData: Social Mining & Big Data Ecosystem.

REFERENCES

[1] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative filtering for implicit feedback datasets. In *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*. Ieee, 263–272.

[2] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *CoRR* abs/1301.3781 (2013). http://arxiv.org/abs/1301.3781

[3] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence*. AUAI Press, 452–461.

[4] Michele Tumminello, Salvatore Micciche, Fabrizio Lillo, Jyrki Piilo, and Rosario N Mantegna. 2011. Statistically validated networks in bipartite complex systems. *PloS one* 6, 3 (2011), e17994.