Chapter 2
Related Work

In this chapter, we introduce the general related work for context-aware ranking with factorization models. Related work on specific issues like tag recommenders, Markov chains, etc. is discussed in detail in the corresponding chapters. Here, we discuss three general topics. The first one is recommender systems because the standard task of personalized item recommendation (a two mode problem) can be seen as context-aware ranking where the context is the user. Nevertheless in recommender systems, the term 'context' is usually used only for cases with at least three modes and furthermore the first mode is typically assumed to be the user. Thus, in the discussion about recommender systems we stick to the definition within the recommender community and use the term *context-aware recommender system* only for ranking problems with at least three modes. In contrast to this, in this book we use the term *context-aware ranking* for any number of modes. Secondly, we investigate factorization models on which our proposed approach is based. Finally, we discuss the literature about ranking in general and context-aware ranking in particular.

2.1 Recommender Systems

Traditionally, recommender systems are designed for two-mode problems. The task is to predict how much a user likes an item. In context-aware recommender systems, additional modes are available such as time, mood, etc.

Recommender settings can be distinguished into rating prediction (regression) and item recommendation (ranking). Both settings differ in the type of observations that are available. For rating prediction the observations are explicitly given real values. Item recommendation problems usually have only implicit binary feedback – where only positive feedback is observed. The later problem is the more challenging one as the interpretation of the observation is difficult. Furthermore, it appears more often in practice because implicit feedback is easy to gather – e.g. almost every web server records implicit behavior in log files by default. This book focus on the recommendation problem from implicit feedback.

S. Rendle: Context-Aware Ranking with Factorization Models, SCI 330, pp. 9–15.
springerlink.com

2.1.1 Two-Mode Recommender Systems

Recommender systems are a well-studied field with many different approaches. One of the most popular approaches is k-nearest-neighbor collaborative filtering (Sarwar et al, 2001). Recently, matrix factorization methods have become very popular because of their success on the Netflix challenge (Srebro et al, 2005; Koren, 2008). The item recommendation problem can also been transformed into a multi-class classification problem and standard binary classifiers like SVMs can be applied using the 1-vs-1 or 1-vs-rest scheme (Schmidt-Thieme, 2005). Other well-known approaches for recommender systems are Boltzmann machines (Salakhutdinov et al, 2007) or the PLSA (probabilistic latent semantic analysis) model (Hofmann, 2004). We will discuss these two-mode recommender systems in more detail in chapter 6.

2.1.2 Context-Aware Recommender Systems

In comparison to the vast literature in traditional recommender systems over two modes, context-aware recommender systems over more modes have attracted much less attention. Moreover, only simple methods have been presented so far for context-aware recommender systems. The approaches for context-aware recommenders can be categorized into three types: (1) contextual pre-filtering, (2) contextual post-filtering and (3) contextual modelling.

2.1.2.1 Contextual Pre-filtering

Given a context, in contextual pre-filtering a relevant subset of the observations of the past is generated. Then, a standard two-mode recommender method can be applied on this subset – both for training and prediction. The advantage of this approach is that any traditional two-mode method can be used because the training set is made context-aware and thus the method has not to be adjusted. But this approach has two important limitations:

1. For each context, an own recommender model has to be applied. For lazy methods without a training phase (like typical cosine kNN) this is no problem. But for parametrized models (that are known to provide better quality) like factorization models, Boltzmann machines, etc. for each subset a model has to be learned. This is not feasible for a huge number of context.
2. By creating a subset of the training data, a lot of information is withheld from the recommender system. This is especially important as we deal with a highly sparse setting. Furthermore the skipped data is supposed to contain important information because observations between two context are not completely independent. To tackle the issue of sparsity, Adomavicius et al (2005) suggest to generate a broader subset that includes data not only corresponding exactly to the given context but also to other related context.

2.1.2.2 Contextual Post-filtering

In contrast to pre-filtering, post-filtering takes the contextual information at the end into account. First, all contextual information is discarded and a traditional two-mode recommender system is applied. For predicting, the learned two-mode model is used without contextual information. Context-awareness is ensured by post-processing the recommender list. Panniello et al (2009) suggest to use a weighting or filtering approach. For both approaches a contextual probability for the entity is computed. In the weighting approach, the recommender list is reordered by multiplying with this probability. The filtering approach removes entities with a probability that is smaller than a certain threshold.

An advantage of contextual post-filtering is that in the first step any recommender system can be used. And in comparison to pre-filtering only one model has to be built which makes it more applicable. An open issue for post-filtering is how to obtain the contextual probability. Panniello et al (2009) use a simple estimator by counting occurrences of the item in identical context of users in the neighbourhood. This estimator will be unreliable in sparse settings, because it matches for exact context and items. Again a generalization of the context and/ or item might help.

Panniello et al (2009) have empirically compared the performance of pre- and post-filtering. Their results indicate that neither of the two of them outperform the other.

2.1.2.3 Contextual Modelling

Instead of using two-mode recommender systems and applying just a pre/post-processor, in contextual modelling the recommender system uses context information directly in the model.

Adomavicius et al (2005) present a multidimensional model based on OLAP cubes. As sparsity is a main problem for generating estimations in context-aware settings, aggregation of dimensions is used to generate more reliable estimations. In total, this approach is rather simple because it does not subsume any strong (recommender) model. Another approach is to apply SVMs where the context is part of the feature space (Oku et al, 2006). In general, applying standard classifiers does not scale because the categorical variables have to be encoded as many binary variables resulting in a huge dataset even for mid-sized problems.

All the methods that we develop in this work can be classified as contextual modelling. A strong point of our proposed approach is that it subsumes the best performing methods for two-mode item recommendation and three-mode tag recommendation.

2.2 Factorization Models

Our context-aware models are based on factorization models. As we are dealing with categorical variable domains, the problem can be seen as predicting the entries of a multi-mode tensor. With factorization models, each variable is described by a vector of (latent) variables which are called the factors. The entries of a multi-mode tensor can then be constructed by combining the factors. There are two things to consider when applying factorization models: (1) How the factors interact – this is determined by the model structure/ equation. (2) How the factors are obtained – this is defined by the optimization criterion.

Tucker (1966) suggest to factorize the multi-mode tensor into a smaller core tensor and one factor matrix for each mode. Higher-order singular value decomposition (HOSVD) is one method for obtaining the factors (Lathauwer et al, 2000). HOSVD corresponds to a least-square optimization on a tensor without missing values. Parallel factor analysis (PARAFAC) (Harshman, 1970; Carroll and Chang, 1970) is a special case of the Tucker decomposition (TD) where the core tensor is diagonal. The advantage of PARAFAC over TD is that the model equation has no nested sums and thus is much faster. In the two-mode case PARAFAC corresponds to matrix factorization (MF). Singular value decomposition (SVD) is well-known method for estimating factors of a two-mode PARAFAC/ MF model. Analogously to HOSVD, SVD also optimizes for least-square and does not allow missing values. Another approach is to learn MF with a sparse optimization and ridge regression terms as regularization. This has been introduced as maximum-margin matrix factorization (Srebro et al, 2005) using the hinge loss.

For rating prediction in two-mode recommender systems, sparse matrix factorization with regularization and least-square optimization is known to be one of the best approaches (Koren, 2008). Salakhutdinov and Mnih (2008) have extended this to Bayesian Probabilistic Matrix Factorization where the parameters are learned with Markov Chain Monte Carlo (MCMC). For the three-mode problem of tag recommendation, Symeonidis et al (2008) have used HOSVD. To apply this, the missing values have been imputed with 0.

We will discuss factorization models in detail in chapter 5. The optimization is discussed in chapter 4. Applications and comparisons to state-of-the-art models are provided in part III.

2.3 Ranking

Besides recommender systems, there is several work on general ranking. First, we discuss approaches that learn models for optimal ranking. Secondly, we investigate context-aware approaches for ranking.

2.3.1 Global Ranking

There are several approaches that try to optimize models for ranking. Both Kondor et al (2007) and Huang et al (2008) model distributions over permutations. Burges et al (2005) optimize a neural network model for ranking using gradient descent. All these approaches learn only one ranking – i.e. they are not context-aware. In contrast to this, our models are collaborative models that learn context-aware rankings, i.e. one individual ranking per context. In the application part of this work, we show empirically that in our settings it is important to take context into account and that our BCR optimized factorization models outperform even the upper bound for non context-aware ranking.

One way to make the global ranking models context-aware is to apply the idea of contextual-prefiltering (see section 2.1.2.1). That means for each context an individual model is used. For example, for item recommendation each user has an own model or for tag recommendation each post (user-item combination) has an own model. Each individual model is learned from the subset of the training data that matches to the model's context. Obviously, the training data for each model is very small which results in poor parameter estimates. The reason is that the parameters for each model are independently and thus no inference across context is possible. Moreover this approach is not able to learn for unobserved context, e.g. an unobserved post. In contrast to this, the factorization approach that we propose in this book does not require that the values of a context have been jointly observed – instead it can infer across context.

2.3.2 Context-Aware Ranking

On the other hand there are context-aware ranking approaches. Agrawal et al (2006) investigate context-sensitive ranking. But their problem setting differs substantially from ours. In addition to a dataset like in our case, they assume that a set of contextual preferences is given in advance (alternatively they can also be learned, e.g. by association rule mining). A contextual preference is a binary relation over two variable instances given a context. The type of context they investigate is given by a conjunction of equality constraints. Thus it is possible that the context is sparse, i.e. some of the variables are not defined. The task they solve is for a given query (e.g. SQL) to take the contextual preferences into account and to rank the resulting tuples. If we would apply this to our sparse problem setting, the where-clause of the query would contain the complete context. As we are dealing with very sparse settings, the selection would typically be empty, because it is very unlikely that there are observations for exactly this context. Furthermore, we are not interested to rank for context that has been observed already but rather in ranking for non-observed context. In total, the problem settings of our work and in (Agrawal et al, 2006) are too different and thus their method does not make sense in our setting and vice versa, our approach is supposed to perform bad in their setting.

Haveliwala (2003) describes a context-sensitive extension of the famous Pagerank (Brin and Page, 1998) algorithm. The idea is to generate a context-aware Pagerank in three steps: (1) A small set of 'topics' (e.g. 16 in his experiments) is selected and one Pagerank for each of these topics is generated. (2) A probabilistic classifier is learned to map a context to the topics. For this they use a Naive Bayes classifier. (3) The context-sensitive Pagerank is now the weighted average of the topic Pageranks where the probabilities of the topic classifier are used as weights. In our settings no topics are given in advance. Thus one could model the topics as latent topics. The factorization dimensions of our factorization models can be seen as a kind of latent topics. Instead of finding latent topics just for the entities to rank, factorization models also generate factors for the variables in the context. Furthermore, our factorization models optimize all 'topics'/ factors jointly and also do not need to learn any mapping from topics to rankings.

References

Adomavicius, G., Sankaranarayanan, R., Sen, S., Tuzhilin, A.: Incorporating contextual information in recommender systems using a multidimensional approach. ACM Transactions on Information Systems 23(1), 103–145 (2005)

Agrawal R, Rantzau R, Terzi E (2006) Agrawal, R., Rantzau, R., Terzi, E.: Context-sensitive ranking. In: SIGMOD 2006: Proceedings of the 2006 ACM SIGMOD international conference on Management of data, pp. 383–394. ACM, New York (2006)

Brin, S., Page, L.: The anatomy of a large-scale hypertextual web search engine. In: WWW7: Proceedings of the Seventh International Conference on World Wide Web, vol. 7, pp. 107–117. Elsevier Science Publishers B. V, Amsterdam (1998)

Burges, C., Shaked, T., Renshaw, E., Lazier, A., Deeds, M., Hamilton, N., Hullender, G.: Learning to rank using gradient descent. In: ICML 2005: Proceedings of the 22nd International Conference on Machine Learning, pp. 89–96. ACM Press, New York (2005)

Carroll, J., Chang, J.: Analysis of individual differences in multidimensional scaling via an n-way generalization of eckart-young decomposition. Psychometrika 35, 283–319 (1970)

Harshman, R.A.: Foundations of the parafac procedure: models and conditions for an 'exploratory' multimodal factor analysis. UCLA Working Papers in Phonetics, 1–84 (1970)

Haveliwala, T.H.: Topic-sensitive pagerank: A context-sensitive ranking algorithm for web search. IEEE Transactions on Knowledge and Data Engineering 15(4), 784–796 (2003)

Hofmann, T.: Latent semantic models for collaborative filtering. ACM Trans. Inf. Syst. 22(1), 89–115 (2004)

Huang, J., Guestrin, C., Guibas, L.: Efficient inference for distributions on permutations. In: Platt, J., Koller, D., Singer, Y., Roweis, S. (eds.) Advances in Neural Information Processing Systems, vol. 20, pp. 697–704. MIT Press, Cambridge (2008)

Kondor, R., Howard, A., Jebara, T.: Multi-object tracking with representations of the symmetric group. In: Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics, San Juan, Puerto Rico (2007)

Koren, Y.: Factorization meets the neighborhood: a multifaceted collaborative filtering model. In: KDD 2008: Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 426–434. ACM, New York (2008)

Lathauwer, L.D., Moor, B.D., Vandewalle, J.: A multilinear singular value decomposition. SIAM J. Matrix Anal. Appl. 21(4), 1253–1278 (2000)

Oku, K., Nakajima, S., Miyazaki, J., Uemura, S.: Context-aware svm for context-dependent information recommendation. In: MDM 2006: Proceedings of the 7th International Conference on Mobile Data Management, p. 109. IEEE Computer Society, Washington (2006)

Panniello, U., Tuzhilin, A., Gorgoglione, M., Palmisano, C., Pedone, A.: Experimental comparison of pre- vs. post-filtering approaches in context-aware recommender systems. In: RecSys 2009: Proceedings of the third ACM conference on Recommender systems, pp. 265–268. ACM, New York (2009)

Salakhutdinov, R., Mnih, A.: Bayesian probabilistic matrix factorization using Markov chain Monte Carlo. In: Proceedings of the International Conference on Machine Learning, vol. 25 (2008)

Salakhutdinov, R., Mnih, A., Hinton, G.: Restricted boltzmann machines for collaborative filtering. In: ICML 2007: Proceedings of the 24th International Conference on Machine Learning, pp. 791–798. ACM, New York (2007)

Sarwar, B., Karypis, G., Konstan, J., Reidl, J.: Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th International Conference on World Wide Web, pp. 285–295. ACM Press, New York (2001)

Schmidt-Thieme, L.: Compound classification models for recommender systems. In: IEEE International Conference on Data Mining (ICDM 2005), pp. 378–385 (2005)

Srebro, N., Rennie, J.D.M., Jaakola, T.S.: Maximum-margin matrix factorization. In: Advances in Neural Information Processing Systems, vol. 17, pp. 1329–1336. MIT Press, Cambridge (2005)

Symeonidis, P., Nanopoulos, A., Manolopoulos, Y.: Tag recommendations based on tensor dimensionality reduction. In: RecSys 2008: Proceedings of the 2008 ACM Conference on Recommender Systems, pp. 43–50. ACM, New York (2008)

Tucker, L.: Some mathematical notes on three-mode factor analysis. Psychometrika 31, 279–311 (1966)