Title: Contextual Factors Involvement for Score Modeling under LambdaRank in Recommender Systems

Author Keywords:
Context-Aware Recommender
Learning to Rank
Tensor Factorization
Gradient Boosted Regression Trees

Abstract: For most recommender systems, the ultimate task is to generate a ranked item list for a user under specific context environment. The basis for sorting such an item list is the score yielded by the model regarding features of user, item and context respectively. Loss/utility definition on the whole list and score modeling are then the two main components for learning to rank (L2R) in recommender systems.

L2R is a hot research area in Information Retrieval (IR) community. In order to apply L2R in recommender scenarios, we use the user-context combination to replace the explicit query in IR. On the other hand, LambdaRank, a powerful L2R framework that combines gradient on pair-wise loss and specific evaluation metrics as utility functions for learning, is chosen to wrapping up the scoring models.

In this work, we extend the SVDFeature framework[1], which has won the 2012 KDD Cup Track1[2] and has implemented LambdaRank. Our scoring model is developed tailored to the consideration of contextual factors within this framework. Thanks to the natural factorization effect of LambdaRank, we can separate the weights updating direction calculation to partial differential of loss on score and partial differential of score on weight[3]. Thus more attention can be paid on the score modeling while ranking loss does not need to be cared about anymore. As to score modeling, aside from users and items, contextual factors (e.g. time, location, company by friends) are also influential auxiliary information we need to consider for a recommender. Two modern score modeling strategies, Tensor Factorization (TF) and Gradient Boosted Regression Trees (GBRT) are chosen to involve contextual factors in the score modeling in recommender. Compared to the complexity of learning tree structure in function space of GBRT, TF has the advantage of conveniently calculating derivatives following Matrix Factorization thought. Meanwhile, features belonging to these contextual factors can be in different formats, either as discrete categorical forms or as continuous values. In the view of this point, TF is more bound to discrete format data input, yet GBRT holds its strength as flexibly splitting on feature values.

There have already been successful attempts of TF and LambdaMART (additive trees in LambdaRank) in recommender strategies. However, to the best of our knowledge, this is the first time that tensor-based approach and additive trees are implemented in the same LambdaRank framework.

The experiment is conducted on datasets (TV1 and TV2) that have been collected by two IP-television providers in Europe. The two providers have 200,000 and 600,000 television subscribers respectively. For the fairness of comparing the scoring models, we apply the same contextual factors, “time of day” and “day of week” for each algorithm. The experiment illustrates the performance on Mean Average Precision (MAP) when applying TF and GBRT towards user-context query unit in LambdaRank. The work also aims to inspire other model solutions within L2R framework to cope with missions in recommender systems.

Reference:

[1] http://svdfeature.apexlab.org/wiki/Main_Page, Wiki page for svdfeature.

[2] T. Chen, L. Tang, et al., *Combining Factorization Model and Additive Forest for Collaborative Followee Recommendation*, KDD challenge workshop 2012.

[3] C.J.C. Burges, *From RankNet to LambdaRank to LambdaMART: An Overview*, Microsoft Research Technical Report, MSR-TR-2010-82.