Learning Label Trees for Probabilistic Modelling of Implicit Feedback

Andriy Mnih & Yee Whye Teh
Gatsby Computational Neuroscience Unit, University College London

Summary

An efficient probabilistic approach to collaborative filtering with implicit feedback, based on modelling the user's item selection process.

- Tree-structured distributions over items for scalability.
- A principled and efficient algorithm for learning effective item trees from data.
- A fix for the standard evaluation protocol for implicit feedback models, addressing its unrealistic assumptions.

Introduction

Collaborative filtering is the method of choice for inferring complex user preference patterns from large collections of feedback data.

- **Explicit feedback**: ratings given by users to items
  - Received a lot of attention: several very effective methods
  - Ratings can be scarce or expensive to collect
- **Implicit feedback**: user purchase or click history
  - Easier to collect than explicit feedback: produced by common user actions
  - The existing methods are not fully probabilistic

Modelling item selection

Our approach is to model the item selection process:

- Treat chosen items as samples from a user-specific distribution.
- The probability of an item under the user's distribution $P(i|u)$ quantifies the degree of the user's interest.
- User and item properties are captured by latent factor vectors:
  - U_u for user u, V_i for item i.
- The probability of user u choosing item i is given by

$$P(i|u) = \frac{\exp(U_u^\top V_i + b_i)}{\sum_k \exp(U_u^\top V_k + b_k)}.$$

- Computing the probability of an item takes too long, as it requires considering all available items.
- **Idea: Associate items with the leaves of a binary tree and exploit its structure to speed up normalization exponentially.**

Tree-structured item space

- One-to-one correspondence between root-to-leaf paths and items.
- Choosing an item now involves a sequence of $\Theta(\log_K N)$ K-way decisions, instead of a single N-way decision.
- Making the K-way decisions probabilistic induces a distribution over items.

Hierarchical item selection model (HIS)

- For user u, the probability of moving from node n_j to node n during a root-to-leaf tree traversal is given by

$$P(n|n_j,u) = \frac{\exp\left(U_u^\top Q_n + b_n\right)}{\sum_{m\in C(n_j)} \exp\left(U_u^\top Q_m + b_m\right)},$$

  if n is a child of n_j and 0 otherwise.
  - $C(n_j)$ is the set of children of node n_j.
  - Q_n and b_n are the factor vector and the bias of node n.
- The probability of selecting item i is the probability of following the path $n_0^i, ..., n_{l_i}^i$ that starts at the root and stops at the leaf containing i:

$$P(i|u) = \prod_{j=1}^{l_i} P(n_j^i|n_{j-1}^i,u).$$

Learning item trees

- We would like to learn the tree structure jointly with the model parameters, but maximizing the log-likelihood w.r.t. the tree structure is intractable.
- **We learn the tree greedily, one level at a time.**
  - For simplicity, we assume that user factor vectors are known and fixed.
- **Top-down hierarchical model-based clustering of items.**
  - Start with all items assigned to the root node.
  - Recursively, partition the set of items at each node among its K children.
  - Update the node assignment for one item at a time as to approximately maximize the log-likelihood.
- Difficulty: The effect of moving an item between nodes at level l on the log-likelihood depends on the future nodes $n_{l+1}^i, ..., n_{l_i}^i$ of the item paths.
  - We approximate the user-dependent tree-structured distribution over items below a node at the current depth by a user-independent flat distribution.
  - This produces a lower-bound on the achievable likelihood for the complete tree-structured model.

Learning a tree level

Suppose we have learned the first $l-1$ nodes of each item path and would like to learn the l^{th} node. The contribution of item i to the log-likelihood of the still-to-be-learned levels of the tree is

$$L_i^l = \sum_{u\in U_i} \left(\log P(n_l^i|n_{l-1}^i,u) + \log P(i|n_l^i,u)\right),$$

where U_i is the set of users who rated item i in the training set and $P(i|n_l^i,u) = \prod_{j=l+1}^{l_i} P(n_j^i|n_{j-1}^i,u)$. Adding up the contributions from all items gives

$$L^l = \sum_i \left(\sum_{u\in U_i} \log P(n_l^i|n_{l-1}^i,u) + \sum_{u\in U_i} \log P(i|n_l^i,u) \right).$$

Approximating the tree-structured user-dependent $P(k|n_l^k,u)$ with a flat user-independent distribution $P(k|n_l^k)$ gives

$$\tilde{L}^l = \sum_i \left(\sum_{u\in U_i} \log P(n_l^i|n_{l-1}^i,u) + |U_i| \log P(i|n_l^i) \right).$$

$\tilde{L}^l$ can be maximized w.r.t. n_l^i in $O(K)$ time since the user factor vectors are fixed and the model is log-linear in them.

Algorithm for learning a tree level

- Initialize $\{n_l^i\}$ randomly
- Repeat until convergence:
  - Pick a user/item pair from the training set
  - Set n_l^i and $P\left(i|n_k^i\right)$ to the values that jointly maximize $\tilde{L}^l$
  - Update $Q_{n_l^i}$ using an online estimate of the gradient of $\tilde{L}^l$

Training procedure

1. Train a model based on a random tree and extract user factor vectors.
2. Learn a tree from the (fixed) user factor vectors.
3. Train a model based on the learned tree, updating both user and item factor vectors.

Note: Each of the three stages is online as model parameters are updated after each user/item pair. However, the set of items has to be fixed in advance.

Evaluation protocol

Implicit feedback models are evaluated using information retrieval metrics.

- Need to know which items are relevant and which are not.
- Typically items that are not selected by the user are assumed to be irrelevant.
- Problematic, as some of those items are actually relevant.

Our approach: use a small quantity of explicit feedback to identify the truly not relevant items.

Results

- MovieLens 10M dataset
  - Ratings on a scale from 0 to 5
  - 69878 users and 10677 movies
  - Keep user/item pairs with ratings 4 and higher
- We compare to Binary Matrix Factorization (BMF) and Bayesian Personalized Ranking (BPR).
- Not relevant $\equiv$ rated 2 or lower

Model	PPL	MAP	P@1	P@10	R@1	R@10
BMF	–	70.80	75.66	49.77	20.94	77.21
BPR	865	**72.75**	75.75	50.63	21.50	**78.39**
HIS (Random)	921	70.68	74.65	49.91	20.66	77.31
HIS (LearnRI)	822	72.50	76.64	50.64	21.51	78.22
HIS (LearnCI)	**820**	72.61	**76.68**	**50.69**	**21.54**	78.27

- Not relevant $\equiv$ unrated

Model	MAP	P@1	P@10	R@1	R@10
BMF	**16.13**	**22.10**	**12.94**	**4.66**	**23.55**
BPR	12.73	14.27	9.89	3.06	18.86

Conclusion and future work

- We introduce a new approach to modelling implicit feedback using tree-structured distributions over items, along with a principled algorithm for learning the item trees.
- Competitive with the best existing methods.
- Future work:
  - A fully online version of the tree-learning algorithm
  - Multiple leaves per item for greater flexibility
  - Application to classification with many classes

Support: Gatsby Charitable Foundation; European Community's Seventh Framework Programme (FP7/2007-2013) grant agreement No. 270327.