Context-aware Preference Modeling with Factorization

Balázs Hidasi
balazs.hidasi@gravityrd.com

http://hidasi.eu

Context-aware factorization on implicit feedback [1,2]
Implicit feedback
- Observation of user activity, inferred feedback
  - The practical scenario (explicit feedback is scarcely available in practice)
  - Noisy positive feedback
  - Missing negative feedback

Context
- Any information besides the user-item transactions
  - Property of the transaction rather than that of users or items
  - Helps recommendations
    - (1) Noise filtering during training (context-related patterns)
    - (2) Recommendation lists are adapted to fit the users’ needs better

iTALS/iTALSx factorization algorithms
- Context-aware data organized into a tensor
- Pairwise preference estimation
  - User-item-context(s) combination in training data: 1 as preference value
  - Missing (negative) feedback: 0 as preference value
- Weight function: lower weights for missing feedback
- Loss: weighted sum of squared errors

Features
- No restriction on the context
  - Based on the SA-MDM data model, GFF works on any context-aware context dimensions.
  - The extended is fully compatible with MDM, enabling the recommendation problem independently of the number and the meaning of classes used by the state.
- Noisy not at random hypotheses and more.

General Factorization Framework – experimentation with preference models [4]

Aims & Goals
Create an algorithm that allows experimentation with novel models for the context-aware recommendation problem, that are not restricted to the two main model classes used by the state-of-the-art.

Typical interactions
User-item: Core interaction.
User-item-context: Context-dependent reweighting of interactions.
User-context: Context-dependent user bias.
Item-context: Context-dependent item bias.
Context-context: Required for symmetric models.

Models beyond the state-of-the-art
Traditional MF
Interaction model (3D)
Interaction model (4D)
Pairwise interaction model (iTALSx)
N-way model (iTALS)
Reduced model (3D)
Pairwise model (iTALS)

Features
No restriction on the context
- Based on the SA-MDM data model, GFF works on any context-aware recommendation problem independently of the number and the meaning of context dimensions. The extended is fully compatible with MDM, enabling the usage of additional data (e.g. session information, item metadata, etc.).
Large preference model classes
- The only restriction on the preference model is that it must be linear in the dimensions of the problem. (Meaning that a dimension can not directly interact with itself in the model.) This intuitive restriction does not restrict the applicability to real-world problems.

Data type independence
- Besides the practically more useful implicit case, explicit problems can also be addressed by changing the weighting scheme in the loss function.

Flexibility
- The weighting scheme of GFF is very flexible, enabling to incorporate extra knowledge through the weights such time decay, dwell time dependent weighting, missing not at random hypotheses and more.

Scalability
- GFF scales well both in terms of the number of interactions in the training set and in the number of features. This makes it applicable in real life recommender systems.

Data Mining and Knowledge Discovery (2015)

Example setting
- U – Users
  - I – Items
  - S – Seasonality
  - Q – Sequentiality

Dataset
- Best model
- Improvement over traditional
- Novel better than traditional

Grocery
- U+I+US+UQI
  +20.14% 3 of 5

TV1
- U+I+US+UQI
  +15.37% 2 of 5

TV2
- U+I+US+UQI
  +30.30% 4 of 5

LastFM
- U+I+US+UQI
  +12.40% 3 of 5

VoD
- U+I+US+UQI
  +19.02% 2 of 5

![Graph showing the comparison between ALS methods and GFF.](image-url)

Scaling of ALS
- Number of transactions: linear
- Number of features: cubical / practically quadratic

Approximations
- ALS-CD (Coordinate Descent – learn one feature at a time)
- ALS-CG (Conjugate Gradient – approximate the LS solution)

Practically linear scaling in the number of features (for commonly used values)
- High factor models can be used
- More frequent retrainings
- Better trade-offs between accuracy and training times

Accuracy is similar to ALS

Comparison
- CG is better: faster, more stable, a little bit more accurate and directly approximates the ALS solution

![Graph showing the comparison between ALS methods and GFF.](image-url)

Worse
Similar
Better

<table>
<thead>
<tr>
<th>Method</th>
<th>Similar</th>
<th>Worse</th>
<th>Better</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALS-CG</td>
<td>10 of 75 (13.33%)</td>
<td>3 of 75 (4%)</td>
<td></td>
</tr>
<tr>
<td>ALS-CG</td>
<td>16 of 75 (21.33%)</td>
<td>2 of 75 (2.67%)</td>
<td></td>
</tr>
</tbody>
</table>

Considerable speed-up (up to 10 times)

- Improved speed (up to 10 times)
- Better trade-offs between accuracy and training times

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Best model</th>
<th>Improvement over traditional</th>
<th>Novel better than traditional</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grocery</td>
<td>U+I+US+UQI</td>
<td>+20.14%</td>
<td>3 of 5</td>
</tr>
<tr>
<td>TV1</td>
<td>U+I+US+UQI</td>
<td>+15.37%</td>
<td>2 of 5</td>
</tr>
<tr>
<td>TV2</td>
<td>U+I+US+UQI</td>
<td>+30.30%</td>
<td>4 of 5</td>
</tr>
<tr>
<td>LastFM</td>
<td>U+I+US+UQI</td>
<td>+12.40%</td>
<td>3 of 5</td>
</tr>
<tr>
<td>VoD</td>
<td>U+I+US+UQI</td>
<td>+19.02%</td>
<td>2 of 5</td>
</tr>
</tbody>
</table>