HybridSVD

When Collaborative Information is Not Enough

Evgeny Frolov¹, Research Scientist
Ivan Oseledets²,®, Associate Professor

¹Skolkovo Institute of Science and Technology
²Institute of Numerical Mathematics of the Russian Academy of Sciences

1. BACKGROUND

Proper tuning of simpler baselines vs. modern SOTA shows surprising results.
Part of the reason: tuning can be hard and cumbersome.
Can we reduce tuning hassles at least for the baselines?

Example PureSVD $\{A - UΣU^T\} \rightarrow \min$

Advantages:
- simple tuning via rank truncation
- minimal storage requirements
- supports real-time & session-based recs
- stable, deterministic output
- highly optimized implementations
- scales to ‘billion-size’ problems

Data-debiasing trick [Nikolakopoulos et al. 2019]:
$A = AD^D \land D = \text{diag}[\{d_1, \ldots, d_k]\]$
- $d$ emphasizes the significance of popular items,
- $d$ improves sensitivity to rare/niche items,
- often leads to a significant quality improvement.

Interesting fact: the 3rd most cited work (via Scopus) among all ACM RecSysConfrence proceedings papers!

2. OBJECTIVES

To develop efficient LRA method that
- allows incorporating side information into the model,
- inherits the key advantages of the PureSVD approach,
- admits minimal tuning,
- outperforms strong baselines in typical tasks.

3. KEY IDEA

“Similarity” of users $i$ and $j$ depends on co-occurrence of items in their preferences.
$\gamma = AA^T = UΣU^T \iff u_{ij} \sim \text{sim}(i,j)$

Replace scalar products with a bilinear form.

$$\text{sim}(i,j) \sim a_i^T S a_j$$

Creates “virtual” links based on side features.

7. RESULTS

- 5-fold crossvalidation (averaged);
- 95% confidence intervals (vertical black bars).

Additional results:
-优于PureSVD的性能
-与SOTA方法相比，性能提高显著
-支持实时及会话式推荐
-稳定的，确定性输出
-高度优化的实现
-适用于‘十亿规模’的场景

4. SOLUTION

The problem takes generalized form:
$\{A A^T = UΣU^T \iff \text{Matrix “roots”:} \}
\begin{align*}
A A^T &= UΣU^T & S &= L_L L_L^T \\
A^T A &= ΣV^T V^T & K &= L_L L_L^T
\end{align*}$

Solution:

via SVD of an auxiliary matrix

$\text{link to the original latent space:}$
$\begin{align*}
L_L^T U &= Σ, \\
L_L^T V &= V
\end{align*}$

Properties

- latent space structure:
  $U^T K U = I, \quad V^T S V = I$
- “hybrid” folding-in:
  $p = \frac{1}{2} V^T V L_L^T \alpha$

Controlling side features weight and similarity structure

5. ADDRESSING COLD START

Finding a transformation between latent and real features is a hard task.
Hybrid MF models often incorporate this task into main optimization.

Several representative examples:
- Collective factorization [Singh/Gordon 2008, Arora/Kolda/Dunlavy 2011],
- Local Collective Embeddings (LCE) [Savelski/Martrach 2014],
- Factorization Machines (FM) [Rendle 2010].

It is a lot simpler with SVD!

6. EFFICIENT COMPUTATIONS

PureSVD part

“hybrid” part

Overall computational complexity:
$O(\text{Rank} \cdot r) + O((M + N)^{\frac{3}{2}}) + O((L_L + L_S)^2)$

Finding Cholesky/square root factors:
- efficient schemes for sparse and dense representations;
- computational complexity is adjustable, depends on data structure;
- can be computed on GPU https://developer.nvidia.com/cholmod.

For sparse feature representations:
- symbolic Cholesky decomposition, can reuse sparsity pattern for different $\alpha$ and $\beta$;
- can also use incomplete or thresholded variant of Cholesky decomposition.

For dense feature representations:
- fast symbolic factorization for computing matrix square root [Ambikasaran/O’Neill/Singh 2014].

8. CONCLUSIONS

HybridSVD offers a set of practical advantages:
- Allows generating structured latent feature space.
- Has a small number of intuitive hyper-parameters.
- Enables simplified tuning.
- Supports quick online and session-based recommendations.
- Effective in standard, warm start, and cold start regimes.

May not be the best in all cases; however, definitely is a strong baseline!

9. SOME REFERENCES

Improving PureSVD with EigenRec model:

A structural view on generalized SVD: