Benchmarking of Boolean Matrix Factorization models: classical and neural approaches

Dmitry I. Ignatov   Ruslan Molokanov

Laboratory for Models and Methods of Computational Pragmatics,
Higher School of Economics

dignatov@hse.ru, rmolokanov@hse.ru

Data Analytics and Management in Data Intensive Domains
Oct 29-31, 2025
Saint Petersburg, Russia

Presentation Outline

Boolean Matrix Factotization

Definition

Given two Boolean matrices $A \in \{0,1\}^{m \times r}$ and $B \in \{0,1\}^{r \times n}$, their Boolean product $A \circ B \in \{0,1\}^{m \times n}$ is defined element-wise as

$$(A \circ B)_{ij} = \bigvee_{k=1}^{r} (A_{ik} \wedge B_{kj})$$

where $\wedge$ denotes the logical AND, and $\bigvee$ denotes the logical OR operation.

Example

$$\underbrace{\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}}_{C} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}}_{A} \circ \underbrace{\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}}_{B}$$

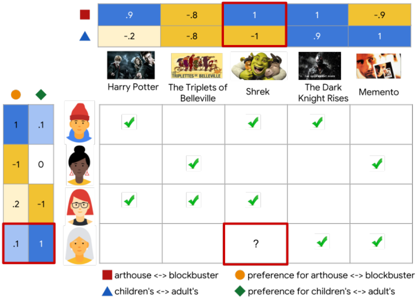


Figure: CF setup

MovieLens 20M:

- 20 M ratings (1-5)
- 27 000 movies
- 138 000 users

Google's course on Recommendation Systems

Collaborative Filtering (CF)

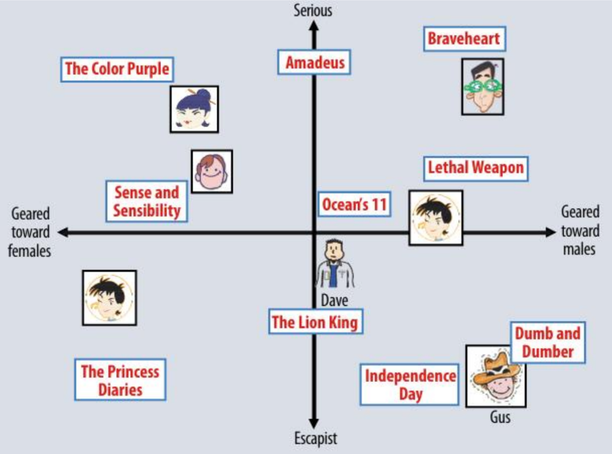


Figure: Toy example of factor space

Koren et al., 2009

Collaborative Filtering (CF)

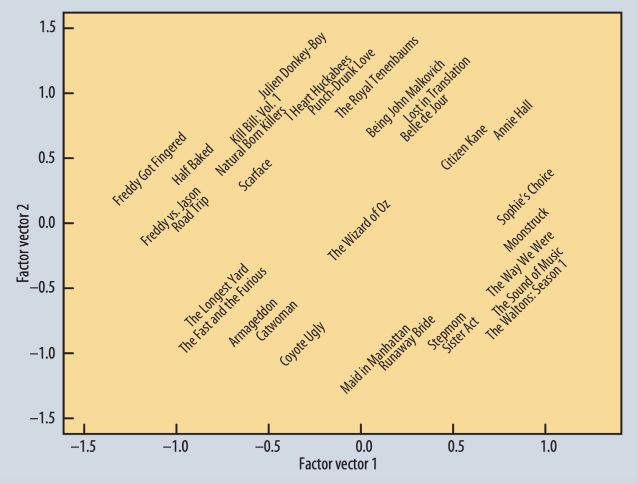


Figure: Real projection of factor space in MovieLens case

Koren et al., 2009

Matrix Factorization approaches to CF

- **Singular Value Decomposition (SVD):**
  For a matrix $A \in \mathbb{R}^{m \times n}$,:
  $$A = U \Sigma V^T,$$
  $U \in \mathbb{R}^{m \times m}$, $V \in \mathbb{R}^{n \times n}$ - orthogonal, and $\Sigma \in \mathbb{R}^{m \times n}$ - diagonal.
- **Nonnegative Matrix Factorization (NMF):**
  For a matrix $A \in \mathbb{R}^{n \times m}, A_{ij} \geq 0$, find $W \in \mathbb{R}^{n \times r}$, $H \in \mathbb{R}^{r \times m}$:
  $$\min_{W,H \geq 0} \|A - WH\|_F^2$$
- **Nonnegative/Binary Matrix Factorization (NBMF):**
  For a matrix $A \in \mathbb{R}^{n \times m}$ and mask $M \in \{0,1\}^{n \times m}$, find $W \in \mathbb{R}^{n \times r}$ with $W_{ij} \geq 0$ and binary $H \in \{0,1\}^{r \times m}$:
  $$\min_{W \geq 0, H \in \{0,1\}} \|M \odot (V - WH)\|_F^2,$$
  where $\odot$ denotes the element-wise product.

Matrix Factorization approaches to CF

Issue

Upper matrix factorization methods do not leverage binary nature of a context matrix A. They are focused on constraints of factor matrices (W, H).

Idea

Treat A as binary matrix and leverage some logical mechanisms.
(..like *"trade information completeness for potential binary patterns"*).

Formal Concept Analysis (FCA)

Definition

A formal context is a triple (G, M, I), where G is a set of objects, M is a set of attributes, and $I \subseteq G \times M$ is a binary relation indicating which object has which attribute.

Definition

A formal concept of the context (G, M, I) is a pair (A, B) with $A \subseteq G$, $B \subseteq M$ such that

$$A' = B \quad \text{and} \quad B' = A,$$

where

$$A' := \{m \in M \mid \forall g \in A : (g, m) \in I\},$$

$$B' := \{g \in G \mid \forall m \in B : (g, m) \in I\}.$$

(A_1, B_1)

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

(A_3, B_3)

(A_2, B_2)

FCA-based factorization

Consider a set $\mathcal{F} \subseteq \mathcal{B}(X, Y, I)$, a subset of all formal concepts of context (X, Y, I), and introduce matrices $P_{\mathcal{F}}$ and $Q_{\mathcal{F}}$:

$$(P_{\mathcal{F}})_{il} = \begin{cases} 1, i \in A_l, \\ 0, i \notin A_l, \end{cases} \quad (Q_{\mathcal{F}})_{lj} = \begin{cases} 1, j \in B_l, \\ 0, j \notin B_l. \end{cases},$$

where (A_l, B_l) is a formal concept from F.

We can consider decomposition of the matrix I into binary matrix product $P_{\mathcal{F}}$ and $Q_{\mathcal{F}}$.

FCA-based factorization

$$\underbrace{\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{bmatrix}}_{I} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{P_{\mathcal{F}}} \circ \underbrace{\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{bmatrix}}_{Q_{\mathcal{F}}}$$

GreCon

Algorithm

Require: Boolean matrix $I \in \{0,1\}^{m \times n}$
Ensure: Factor concepts $F = \{(C, D)\}$

$U \leftarrow \{(i,j) \mid I_{ij} = 1\}$ ▷ Uncovered entries
$F \leftarrow \emptyset$
while $U \neq \emptyset$ **do**
  Pick a seed uncovered element $(i,j) \in U$
  $D \leftarrow \{j\}$ ▷ Start intent with attribute column

  repeat
    $C \leftarrow D^{\downarrow} = \{i' \mid \forall j' \in D : I_{i'j'} = 1\}$
    $D_{\text{new}} \leftarrow C^{\uparrow} = \{j' \mid \forall i' \in C : I_{i'j'} = 1\}$
    if $D_{\text{new}} = D$ **then**
      break ▷ Formal concept reached
    else
      $D \leftarrow D_{\text{new}}$
    end if
  until false
  Add factor (C, D) to F
  for all $(i', j') \in C \times D$ **do**
    Remove (i', j') from U
  end for
end while
return F

Belohlavec et. al., 2010

GreConD (more efficient one)

Algorithm

Require: Boolean matrix $I \in \{0,1\}^{m \times n}$
Ensure: Set of factor concepts $F = \{(C, D)\}$

$U \leftarrow \{(i,j) \mid I_{ij} = 1\}$ ▷ Uncovered ones
$F \leftarrow \emptyset$
while $U \neq \emptyset$ **do**
  $D \leftarrow \emptyset$ ▷ Intent of new factor
  $V \leftarrow 0$
  while there exists $j \notin D$ such that **do**
  $\left|((D \cup \{j\})^{\downarrow} \times (D \cup \{j\})^{\downarrow\uparrow}) \cap U\right| > V$
    Select $j \notin D$ that maximizes coverage
    $D \leftarrow (D \cup \{j\})^{\downarrow\uparrow}$ ▷ Close intent
    $V \leftarrow \left|(D^{\downarrow} \times D) \cap U\right|$
  end while
  $C \leftarrow D^{\downarrow}$ ▷ Extent of the factor
  Add (C, D) to F
  for all $(i,j) \in C \times D$ **do**
    Remove (i,j) from U
  end for
end while
return F

Belohlavec et. al., 2010

GreConD (more efficient one)

Issue

GreConD is efficient, but deterministic and NP-complete. So, it would be nice to propose a more flexible BMF approach...

Idea

Try to use neural network and derive from collaborative filtering setup.

Neural Collaborative Filtering (NCF)

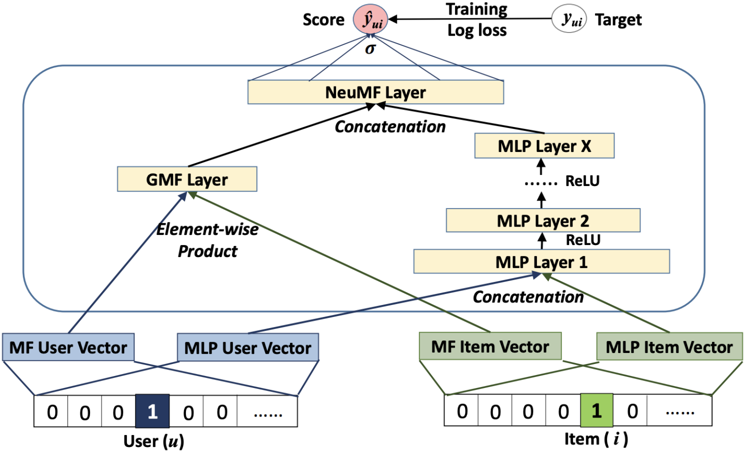


Figure: NCF architecture

He et al., 2017

Neural Collaborative Filtering (NCF)

Issue

NCF operates only in the paradigm of collaborative filtering - i.e. predicts $y \in \{0, 1\}$ value for a pair of user and item.

Idea

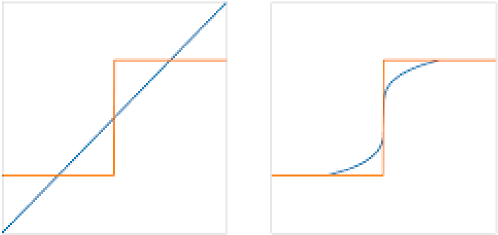
Why not to focus on the GMF branch and try to build an architecture for modeling Boolean matrix factiorization.

Straight-Through Estimator (STE)

Definition

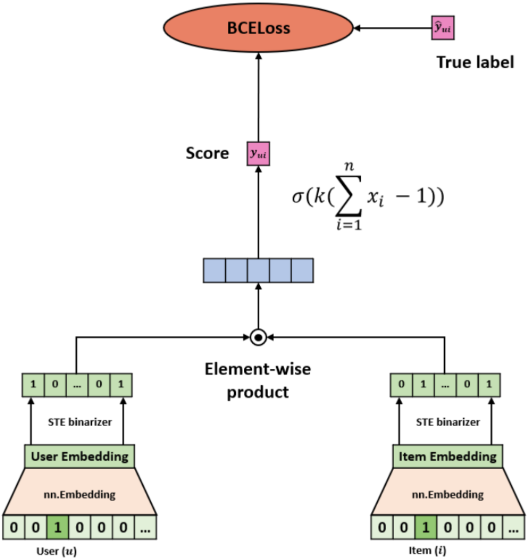
Given a non-differentiable function $y = q(x)$, the Straight-Through Estimator approximates the gradient during backpropagation as:

$$\frac{\partial \mathcal{L}}{\partial x} \approx \frac{\partial \mathcal{L}}{\partial y} \cdot 1 = \frac{\partial \mathcal{L}}{\partial y}$$



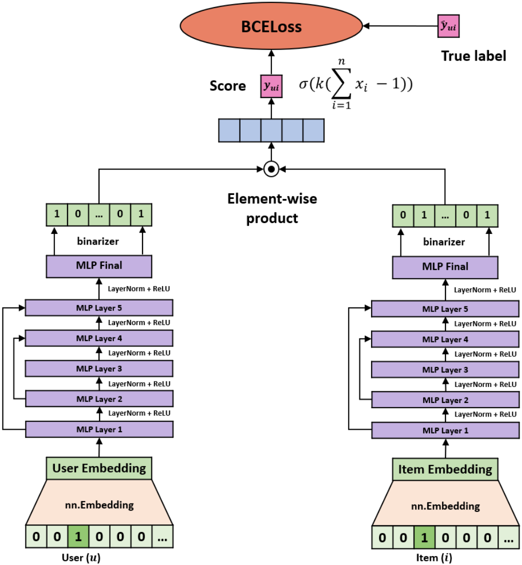
Courbariaux et al., 2016

NBMFShalow



BCELoss
$\hat{y}_{ui}$
True label
Score
y_{ui}
$\sigma(k(\sum_{i=1}^{n} x_i - 1))$
Element-wise product
1 0 ... 0 1
0 1 ... 0 1
STE binarizer
STE binarizer
User Embedding
Item Embedding
nn.Embedding
nn.Embedding
0 0 1 0 0 0 ...
0 0 1 0 0 0 ...
User (u)
Item (i)

NBMFLarge



BCELoss
$\hat{y}_{ui}$
True label
Score
y_{ui}
$\sigma(k(\sum_{i=1}^{n} x_i - 1))$
Element-wise product
1 0 ... 0 1
binarizer
MLP Final
LayerNorm + ReLU
MLP Layer 5
LayerNorm + ReLU
MLP Layer 4
LayerNorm + ReLU
MLP Layer 3
LayerNorm + ReLU
MLP Layer 2
LayerNorm + ReLU
MLP Layer 1
User Embedding
nn.Embedding
0 0 1 0 0 0 ...
User (u)
0 1 ... 0 1
binarizer
MLP Final
LayerNorm + ReLU
MLP Layer 5
LayerNorm + ReLU
MLP Layer 4
LayerNorm + ReLU
MLP Layer 3
LayerNorm + ReLU
MLP Layer 2
LayerNorm + ReLU
MLP Layer 1
Item Embedding
nn.Embedding
0 0 1 0 0 0 ...
Item (i)

NBMFShallow (vs) NBMFLarge

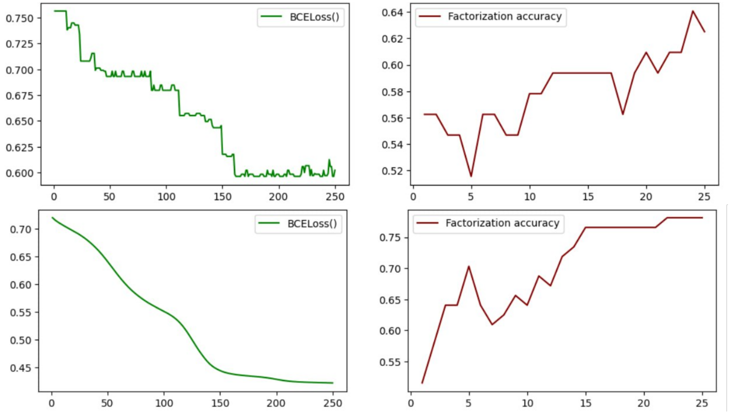


Figure: Training dynamics on primitive setup: upper pair of graphs - NBMFShallow, lower pair of graphs - NBMFLarge (in simplest version)

GreConD: synthetic data

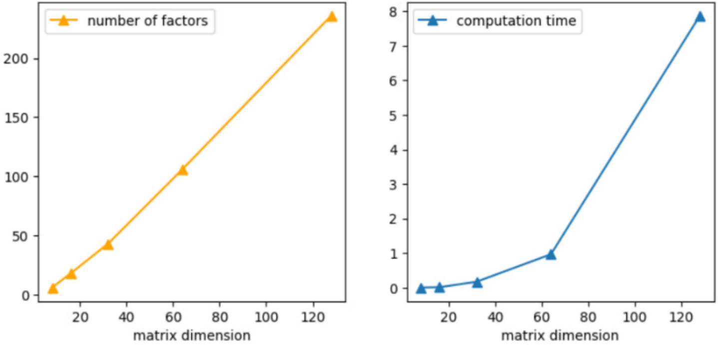


Figure: left - number of extracted factors, right - computation time

GreConD: MovieLens data

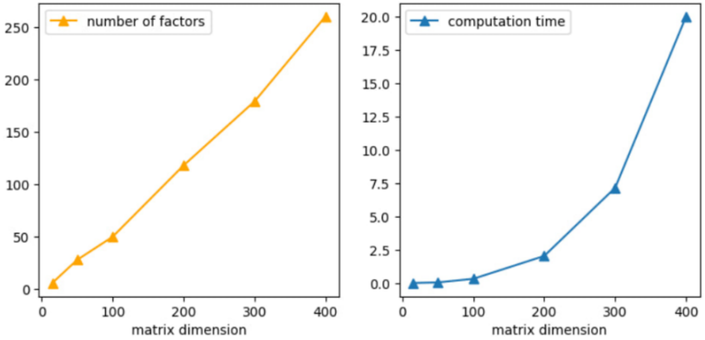


Figure: left - number of extracted factors, right - computation time

NBMFLarge: synthetic data

Table: 16×16 **random** matrices (50% density).

FC layers	Output activation	Final Accuracy	Final Density
emb_size $\times [4, 8, 16, 8, 4]$	tanh	**0.85**	0.63
emb_size $\times [4, 8, 16, 8, 4]$	sigm	0.82	0.64
emb_size $\times [4, 8, 16, 8, 4]$	maxpool	0.81	0.54

Table: 16×16 **identity** matrix.

FC layers	Output activation	Final Accuracy	Final Density
emb_size $\times [4, 8, 16, 8, 4]$	tanh	0.937	0.08
emb_size $\times [4, 8, 16, 8, 4]$	sigm	**0.953**	0.03
emb_size $\times [4, 8, 16, 8, 4]$	maxpool	0.949	0.07

Table: 16×16 **"bar-problem"** matrix.

FC layers	Output activation	Final Accuracy	Final Density
emb_size $\times [4, 8, 16, 8, 4]$	tanh	**1.0**	0.75
emb_size $\times [4, 8, 16, 8, 4]$	sigm	**1.0**	0.75
emb_size $\times [4, 8, 16, 8, 4]$	maxpool	**1.0**	0.75

NBMFLarge: MovieLens data

Table: MovieLens matrix, NBMFLarge with **tanh** output activation.

FC layers	Output activation	*emb_size*	Final Accuracy	Final Density
$emb\_size \times [4, 8, 16, 8, 4]$	tanh	8	0.983	0.037
$emb\_size \times [4, 8, 16, 8, 4]$	tanh	16	0.988	0.033
$emb\_size \times [4, 8, 16, 8, 4]$	tanh	20	**0.992**	0.028

Table: MovieLens matrix, NBMFLarge with **maxpool** output activation.

FC layers	Output activation	*emb_size*	Final Accuracy	Final Density
$emb\_size \times [4, 8, 16, 8, 4]$	maxpool	8	0.987	0.022
$emb\_size \times [4, 8, 16, 8, 4]$	maxpool	16	0.991	0.024
$emb\_size \times [4, 8, 16, 8, 4]$	maxpool	20	**0.993**	0.026

NBMFLarge (vs) GreConD

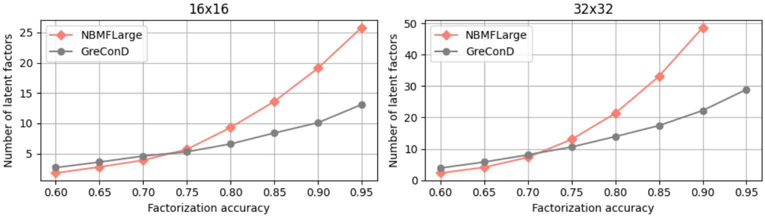


Figure: Mutual experiment on random synthetic data.

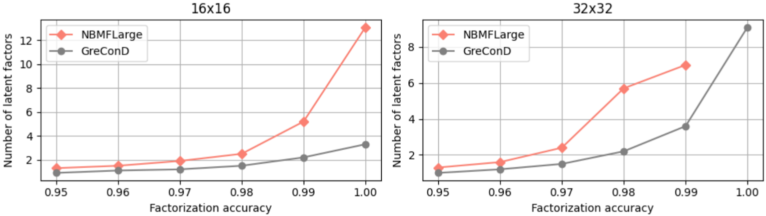


Figure: Mutual experiment on MovieLens data.

Discussion

Issues and improvements

- **Huge latency** of proposed neural-network-based method.
- **Scalability** and **sparsity** issues of NBMFLarge.
- Include **more non-neural BMF methods** into the benchmarking.
- Hidden representations of user and item **can be used beyond** BMF task: in collaborative filtering (like NCF paradigm) or for deeper analysis of matrix structure.

Conclusion

Results

- **GreConD** algorithm implemented and evaluated on synthetic and MovieLens matrices.
- Two neural-network-based BMF models proposed and compared with each other in case of training dynamics: **NBMFShallow** and **NBMFLarge**.
- **NBMFLarge** evaluated on synthetic and MovieLens matrices.
- Comparative evaluation of **NBMFLarge** and **GreConD** conducted.

Contacts

Repository of the research

Laboratory for Models and Methods of Computational Pragmatics, FCS HSE