HOSLIM: Higher-Order Sparse LInear Method for Top-N Recommender Systems

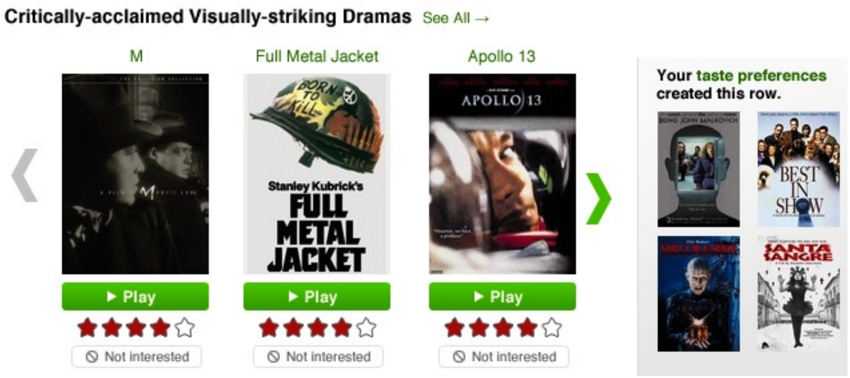
Evangelia Christakopoulou

Advisor: Prof. George Karypis
Computer Science & Engineering
University of Minnesota, Minneapolis, MN
Email: {evangel@cs.umn.edu}

January 25, 2017

Top-N Recommender Systems

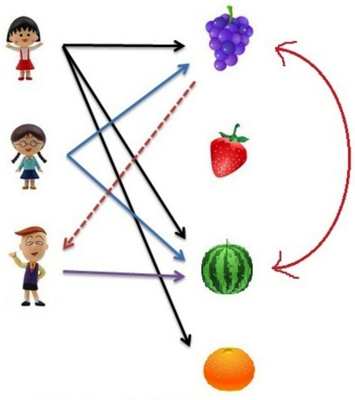
- Recommender systems are everywhere!
- They seek to predict the rating or preference that a user would give to an item.
- They recommend to the user a ranked list of N items, in which he will likely be interested in.



Notations

Def	Description
u	user
i	item
n	total number of users
m	total number of items
R	user-item purchase binary matrix, $(n \times m)$
H_u	set of items that the user has purchased

Item-based Collaborative Filtering



- The user is recommended the items that have been copurchased the most in the dataset with the items he has purchased.
- The recommendation score of user u for item i is:

$$r_{ui} = \sum_{k \in H_u} \textit{association score}(i, k)$$

- The association score between items i and k is either computed as the cosine(r_{*i}, r_{*k}) or as $P(i|k)$.
- Each item that the user has purchased independently contributes to r_{ui}.

SLIM: Sparse LInear Method for Top-N Recommender Systems

- SLIM learns a sparse aggregation coefficient matrix $S(m \times m)$, by solving an l_1 and l_2 regularized optimization problem.

$$\tilde{R} = RS$$

- SLIM casts the estimation of r_{ui} as a regression problem. This allows to capture some dependencies between the items.
- The recommendation score r_{ui} is computed as a sparse aggregation of items purchased by the user u:

$$r_{ui} = \sum_{k \in H_u} r_{uk} s_{ki}$$

- The optimization problem:

$$\underset{s_i}{\text{minimize}} \frac{1}{2} ||\mathbf{r}_i - R\mathbf{s}_i||_2^2 + \frac{\beta}{2} ||\mathbf{s}_i||_2^2 + \lambda ||\mathbf{s}_i||_1$$

$$\text{subject to } \mathbf{s}_i \geq 0 \text{ and } s_{ii} = 0$$

- SLIM outperforms other top-N recommendation methods, in a wide variety of datasets.

Limitation of the existing top-N methods

- Both the old and the new top-N recommendation methods capture only pairwise relations between the items.
- They do not capture higher-order relations.

- In some cases, purchasing a subset of the items significantly increases the likelihood of purchasing the rest.
- Ignoring this type of relations, when present, can lead to suboptimal recommendations!
- HOKNN was the 1st method that incorporated combinations of items (i.e. itemsets). The recommendation score is computed as:

$$r_{ui} = \sum_{k,m \in H_u} P(i|k,m)$$

  However, in most datasets this method did not lead to significant improvements.

Motivation

- SLIM improves upon k-NN.
- Could Higher-Order SLIM impove upon Higher-Order k-NN and SLIM?

Table : Extra Notations

Def	Description
j	itemset
σ	minimum support threshold
p	total number of itemsets
I	set of itemsets
R'	user-itemset purchase matrix, $(n \times p)$

HOSLIM: Higher-Order Sparse LInear Method for Top-N Recommendation

- The model: $\tilde{R} = RS + R'S'$
  - Recommendation score: a sparse aggregation of both the **items** purchased by the user and the **itemsets** that it supports.

$$\tilde{r}_{ui} = \mathbf{r}_u^T \mathbf{s}_i + \mathbf{r}_u'^T \mathbf{s}_i'$$

  - S: the aggregation coefficient matrix corresponding to items. $(m \times m)$
  - S': the aggregation coefficient matrix corresponding to itemsets. $(p \times m)$

HOSLIM: The optimization problem

$$\underset{s_i, s'_i}{\text{minimize}} \frac{1}{2}||\mathbf{r}_i - R\mathbf{s}_i - R'\mathbf{s}'_i||_2^2 + \frac{\beta}{2}(||\mathbf{s}_i||_2^2 + ||\mathbf{s}'_i||_2^2) + \lambda(||\mathbf{s}_i||_1 + ||\mathbf{s}'_i||_1)$$

$$\begin{aligned}
\text{subject to} \quad & \mathbf{s}_i \geq 0 \\
& \mathbf{s}'_i \geq 0 \\
& s_{ii} = 0, \text{ and} \\
& s'_{ji} = 0 \text{ where } \{i \in \mathcal{I}_j\}.
\end{aligned}$$

- The optimization problem is solved using the BCLS library.

Datasets

Name	#Users	#Items	#Non-zeros	Density
groceries	63,035	15,846	1,997,686	0.2%
synthetic	5000	1000	68,597	1.37%
delicious	2,989	2,000	243,441	4.07%
ml	943	1,681	99,057	6.24%
retail	85146	16470	820,414	0.06%
bms-pos	435,319	1,657	2,851,423	0.39%
bms1	26,667	496	90,037	0.68%
ctlg3	56,593	39,079	394,654	0.017%

Experimental Evaluation

- $10\times$Leave-one-out Cross Validation
- Extensive search over the parameter space of the various methods was performed, in order to find the set of parameters that lead to the best results.
  - k-nn: number of neighbors
  - HOKNN: number of neighbors, support threshold (σ)
  - SLIM: l_2 parameter (β), l_1 parameter (λ)
  - HOSLIM: l_2 parameter (β), l_1 parameter (λ), support threshold (σ)
- Evaluation Metric: Hit Rate (HR)

$$HR = \frac{\#hits}{\#users}$$

Results Outline

The experimental analysis focuses on answering the following questions:

- Do higher-order relations exist in real-world datasets?
- Incorporating them in modern top-N methods could improve the recommendation quality?

Table : Itemset Statistics

Name	#Itemsets	$\frac{\text{itemset nnz}}{\text{original nnz}}$
groceries	551,333	7.97
synthetic	8,614	2.55
delicious	47,362	16.38
ml	895	1.94
retail	42,925	1.85
bms-pos	73,382	5.99
bms1	19,055	6.39
ctlg3	5,245	0.51

Verifying the existence of Higher-Order Relations

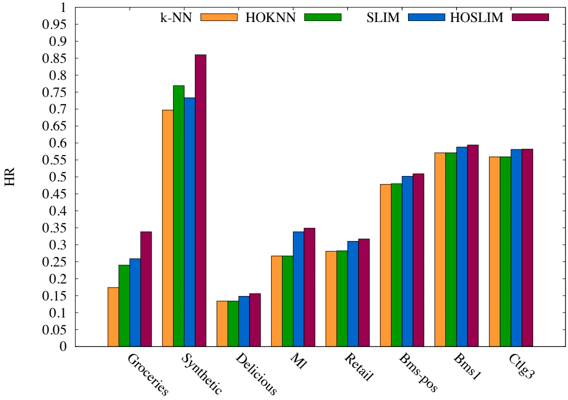
We measured how prevalent are the itemsets with strong association between the items that comprise it (beyond pairwise associations).

- We found all frequent itemsets of size 3 with σ equal to 10. For each of the itemsets we computed the quality metric:

  $$dependency_{max} = \frac{P(ABC)}{max(P(AB)P(C), P(AC)P(B), P(BC)P(A))}$$

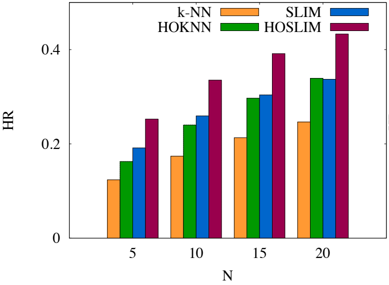
  We considered only the itemsets with dependency higher than a specified threshold (2,5).
- We found out that there are datasets like groceries with high coverage of such itemsets (a large percentage of users having at least one itemset and a large percentage of non-zeros covered by at least one itemset.)
  However, there are datasets like retail with low coverage.
- The correlation coefficient of the improvement in performance in HOSLIM beyond SLIM with the product of affected users coverage and number of non-zeros coverage = 0.88

Performance Comparison

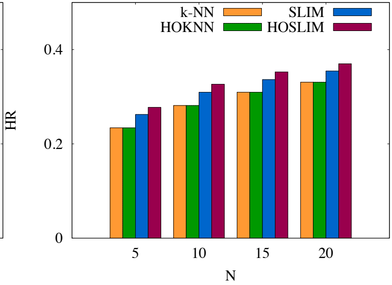


- The incorporation of higher-order information can improve the recommendation quality.
- The improvement percentage depends on the existence of higher-order relations in the dataset.

Performance for different values of N



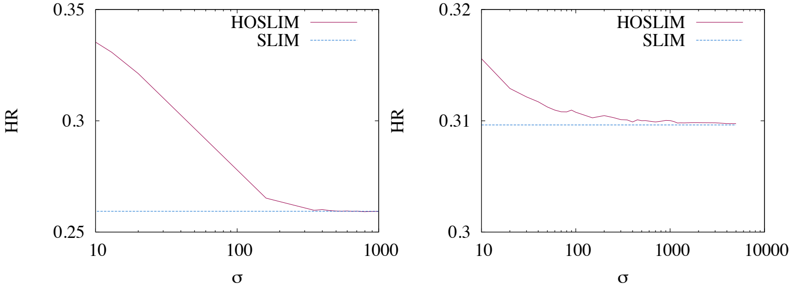
Groceries dataset



Retail dataset

- N is small, as a user will not see an item at the 100th position.

Sensitivity of the support of the itemsets

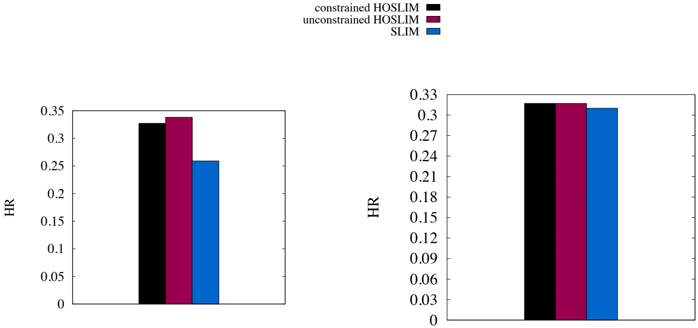


Groceries dataset

Retail dataset

- A low support means that HOSLIM benefits more from the itemsets, thus the HR is higher.

Efficient Recommendation by controlling the number of non-zeros



Groceries dataset

Retail dataset

- $$nnz(S') + nnz(S_{HOSLIM}) \leq 2nnz(S_{SLIM})$$
- The cost of computing the top-N recommendation list depends on the number of non-zeros in the model.

Concluding Remarks

- Higher-order information exists in some real-world datasets.
- Its incorporation in modern top-N methods could help the recommendation quality, especially when the dataset in question contains abundant higher-order information.

Thank you!