Learning to Recommend Accurate and Diverse Items

PEIZHE CHEN SHUAIQIANG WANG JUN MA JIANKAI SUN HUI XIONG

PRESENTED BY : SHIVANI SETHI MIT SHAH ASHWIN RAJANI SUNCHIT SHARMA

Abstract

In this study, we investigate diversified recommendation problem by supervised learning, seeking significant improvement in diversity while maintaining accuracy.

Trying to discover a factorization for matching the following supervised learning task. In doing this, we define two coupled optimization problems, parameterized matrix factorization and structural learning, to formulate our task.

propose a diversified collaborative filtering algorithm (DCF) to solve the coupled problems.

Introduction

In recent years, recommender systems have become a de-facto standard and a must-own tool for e-commerce to promote business and help customers find products.

With accuracy being the primary concern in recommendation tasks, diversity has been increasingly recognized as a crucial issue.

A system with broad horizons may provide a win-win situation: users can find more interesting items and e-commerce enterprises can increase their sales and improve users' satisfaction

How Traditional Methods Work !

Generally, approaches use the heuristic strategy to re-rank the items for recommendation based on certain diversity metric, which mainly involves two steps:

(1) generating a candidate set of favorable items based on the accuracy metric, and then

(2) selecting k items from the candidates by maximizing the recommendation diversity metric.

What’s New in this method ?

We formulate two coupled optimization problems to guarantee that the factorization results can match the learning task:

(1) With a set of parameters generated by structural support vector machine (SVM), we present parameterized matrix factorization, which can generate the representations of the users and items for structural SVM;

(2) With the representations of the users and items generated by parameterized matrix factorization, we utilize structural SVM to output the recommendation model as well as the parameters for parameterized matrix factorization.

(3) Proposing the DCF Algorithm

What is Collaborative Filtering ?

Collaborative filtering is a method of making automatic predictions (filtering) about the interests of a user by collecting preferences or taste information from many users (collaborating).

The underlying assumption of the collaborative filtering approach is that if a person A has the same opinion as a person B on an issue, A is more likely to have B's opinion on a different issue than that of a randomly chosen person.

1) Memory Based : these make predictions by utilizing historical ratings associated with similar users or items. (user-based , item- based)

2) Model Based : these explain the ratings by characterizing both items and users with vectors of factors inferred from item rating pattern.

What is Recommendation Diversity ?

Recommendation diversity in broad sense refers to the type or genres of items recommended.

1) Aggregate Diversity : The aggregate diversity across all users can be evaluated by absolute long-tail metrics, relative long-tail metrics, and slope of the log-linear relationship.

2) Individual Diversity : The individual diversity is oriented to each individual user, which attempts to achieve a diverse recommendation result for each target user.

Structural S.V.M.

Structural SVM generalizes the SVM classifier which allows training a classifier for general structured output labels such as trees, sequences or sets.

Used to formulate the task of diversified retrieval as the problem of predicting diverse subsets, and adopt Structural SVM to train a diversified model for search.

Problem Formulation

The problem that we have is a dual optimization problem dependent on the optimization of the following two :

1. Matrix factorization problem which requires **Σ** to give User and Item Matrices.

2. While the SVM gives **Σ** and the Recommendation requiring the User and Item Rating

These two are expalined in the following slides

Problem - 1

Definition 1. (Parameterized Matrix Factorization).Given a user-item rating matrix R and a set of parameters σ = {σ 1 , σ 2 , . . . , σ k }, parameterized matrix factorization generates two low-rank matrices U k×m and I k×n :

arg min (U,I)   S(U, I; Σ, R)

so that R (m×n) ≈ U (k×m)   Σ (k×k)   I (k×n)

Where Σ is a k–dimensional diagonal matrix, and the diagonal elements are σ 1 , σ 2 , . . . , σ k .

Problem - 2

Definition 2. (Learning-based Diversified Recommendation). Given a user matrix U where each column $u \in U$ is the vector of user u, an item matrix I where each column $i \in I$ is the vector of item i, a set of labels Y for users, and a set of possible recommendation functions H, where each $h : U \times I\ n \rightarrow I\ l$ recommending l items to each user based on users' historically rated items, learning-
based diversified recommendation attempts to discover a recommendation function h as well as a set of parameters σ for optimizing parameterized matrix factorization:

arg min (h, Σ) L(h(U, I n), Σ; Y, U, I)

Ground-Truth For Training :

Step -1) Filter a set of high-rated items C for each user as a set of candidates

$$r_{u,i} \geq \gamma \times \overline{r}_u$$

Step - 2) Select a set of items from C by maximizing the diversity measure as the ground-truth of the target user.

$$\arg\max_{y_u \in C^k} \frac{2 \times f(y_u) \times g(y_u)}{f(y_u) + g(y_u)}.$$

Loss Functions

Parameterized MF loss. As Definitions 1 mentioned, the Parameterized MF loss function can be defined as follows:

$$\arg\min_{\mathbf{U},\mathbf{I}} \sum_{\forall r_{u,i} \in \mathbb{R}_0} (r_{u,i} - \mathbf{u}^\top \boldsymbol{\Sigma} \mathbf{i})^2 + \lambda(\|\mathbf{U}\|_F^2 + \|\mathbf{I}\|_F^2), \quad (3)$$

$\|\cdot\|_F^2$ is the Frobenius 2-norm for avoiding overfitting, and the parameter λ balances the accuracy and the regularization terms.

Learning Loss :

$$\begin{aligned} &\arg\min_{\mathbf{w},\boldsymbol{\sigma}} \quad \frac{1}{2}\left(\|\mathbf{w}\|^2 + \|\boldsymbol{\sigma}\|^2\right) + C\xi \\ &\text{s.t.} \quad \forall \mathbf{u} \in \mathbf{U}, \forall y \in \mathcal{Y} \setminus y_u : \qquad (5) \\ &\mathbf{w}^\top \sum_{u \in U} [\Psi(\mathbf{u}, \mathbf{I}_u, \boldsymbol{\sigma}, y_u) - \Psi(\mathbf{u}, \mathbf{I}_u, \boldsymbol{\sigma}, y)] \geq \sum_{u \in U} \Delta(y_u, y) - m\xi \end{aligned}$$

Joint Features

$$\Psi(\mathbf{u}, \mathbf{I}_u, \boldsymbol{\sigma}, y) = \begin{bmatrix} z_1 \sum_{\forall \mathbf{i} \in y,\ \mathbf{j} \in \mathbf{I}_u \setminus y} \left(\mathbf{u}^\top \boldsymbol{\Sigma} \mathbf{i} - \mathbf{u}^\top \boldsymbol{\Sigma} \mathbf{j} \right) \\ z_2 \sum_{\forall \mathbf{i}, \mathbf{j} \in y,\ \mathbf{i} \neq \mathbf{j}} \frac{\mathbf{i}^\top \mathbf{j}}{\|\mathbf{i}\| \|\mathbf{j}\|} \\ \|\mathbf{u}\|^2 \\ z_3 \sum_{\forall \mathbf{i} \in y} \|\mathbf{i}\|^2 \\ \mathbf{u} \\ z_4 \sum_{\forall \mathbf{i} \in y} \mathbf{i} \end{bmatrix},$$

Error Fucntion

The accuracy metric :

Given a prediction y ⊆ Iu, the accuracy metric is defined as follow :

$$f(y; \mathbf{u}, \mathbf{I}_u) = \frac{\sum_{\mathbf{i} \in y, \mathbf{j} \in \mathbf{I}_u \setminus y} \left[P(i \succ j; u) - P(j \succ i; u) \right]}{l(n_u - l)}, \quad (6)$$

where l is the number of the recommended items, and $P(i \succ j; u)$ is the preference function indicating that whether u prefers the item i to j or not:

$$P(i \succ j; u) = \begin{cases} 1, & \text{if } r_{u,i} > r_{u,j} \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

Error Function

The Diversity metric :

The diversity g(y) is defined as the average dissimilarity of all pairs of items in y :

$$g(y) = \frac{\sum_{\mathbf{i},\mathbf{j} \in y, \mathbf{i} \neq \mathbf{j}} d(\mathbf{i}, \mathbf{j})}{\frac{1}{2} l(l-1)}, \tag{8}$$

where $d(\mathbf{i}, \mathbf{j})$ is the dissimilarity between item vectors $\mathbf{i}$ and $\mathbf{j}$, and the dissimilarity between item vectors $\mathbf{i}$ and $\mathbf{j}$ is defined based on the cosine similarity:

$$d(\mathbf{i}, \mathbf{j}) = -cos(\mathbf{i}, \mathbf{j}) = -\frac{\mathbf{i}^\top \mathbf{j}}{||\mathbf{i}|| \times ||\mathbf{j}||}.$$

Error Function :

The error function. The error function includes two parts: the accuracy error $err_{acc}(y_u, y)$ and the diversity error $err_{div}(y_u, y)$.

$$err_{acc}(y_u, y) = f(y_u; \mathbf{u}, \mathbf{I}_u) - f(y; \mathbf{u}, \mathbf{I}_u)$$
$$err_{div}(y_u, y) = g(y_u) - g(y).$$

Given a user vector $\mathbf{u}$, the vectors of the items $\mathbf{I}_u$ rated by u, and a prediction y, the error function $\boldsymbol{\Delta}(y_u, y)$ can be defined as the trade-off between $err_{acc}(y_u, y)$ and $err_{div}(y_u, y)$ with the formulation of the F-measure:

$$\Delta(y_u, y) = \frac{2 \times err_{acc}(y_u, y) \times err_{div}(y_u, y)}{err_{acc}(y_u, y) + err_{div}(y_u, y)}. \qquad (9)$$

DCF Algorithm - 1

Algorithm 1: Parameterized matrix factorization
Input : A user-item rating matrix R, a set of parameter σ, the trade-off parameter λ, and the learning rate η.
Output: The low-rank user and item matrices U and I.

1. U, I ⟵ Initialize();
2. repeat
3. forall the r u,i ∈ R 0 do
4. u ⟵ u − η∇ u S
5. i ⟵ i − η∇ i S
6. end
7. until converge;
8. return: U, I

DCF Algorithm -2

Algorithm 2: One optimization iteration of the cutting plane algorithm

Input : A user matrix $\mathbf{U}_{k\times m}$ where each column $\mathbf{u}$ is the vector of user u; A set of vectors $\mathbf{I}_u$, each representing the item rated by u; The recommendation ground-truth y_u for u; The set of the working constraints W ; The trade-off parameter C, and the threshold ϵ.

Output: A vector of weights $\mathbf{w}$ and the working constraints W

foreach *column* $\mathbf{u}$ *of* $\mathbf{U}$ **do**

    $H(y;\mathbf{w}) \equiv$

    $\Delta(y_u, y) + \mathbf{w}^\top \left[\Psi(\mathbf{u}, \mathbf{I}_u, \boldsymbol{\sigma}, y) - \Psi(\mathbf{u}, \mathbf{I}_u, \boldsymbol{\sigma}, y_u)\right];$

    $\hat{y}_u \leftarrow \arg\max_{y\in\mathcal{Y}} H(y;\mathbf{w})$; `// Find cutting plane`

end

$\xi \leftarrow \max\{0, \frac{1}{m}\sum_{u\in U} \max_{y\in W} H(y;\mathbf{w})\};$

if $\frac{1}{m}\sum_{u\in U} H(\hat{y}_u;\mathbf{w}) > \xi + \epsilon$ **then**

    $W \leftarrow W \cup \{\hat{y}_1, \hat{y}_2, \cdots, \hat{y}_m\};$

    $\mathbf{w} \leftarrow$ optimize Eq.(5) with C over $\bigcup W$;

end

return: $\mathbf{w}$, W

DCF Algorithm -3

Algorithm 3: Finding cutting plane

Input : A vector of weights $\mathbf{w}$, a user vector $\mathbf{u}$ for representing the user u, a set of vectors of items rated by u, the label y_u for u

Output: The most violated constraints $\hat{y}_u$ for u

$\hat{y}_u \leftarrow \varnothing$;
for $l = 1$ **to** k **do**
  $\mathbf{j} \leftarrow \underset{\mathbf{i} \in \mathbf{I}_u, \mathbf{i} \notin \hat{y}}{\arg\max}\, \Delta(y_u, \hat{y}_u \cup \{\mathbf{i}\}) + \mathbf{w}^\top \Psi(\mathbf{u}, \mathbf{I}_u, \boldsymbol{\sigma}, \hat{y}_u \cup \{\mathbf{i}\})$;
  $\hat{y}_u \leftarrow \hat{y}_u \cup \{\mathbf{j}\}$;
end
return: $\hat{y}_u$

DCF Algorithm - 4

Algorithm 4: The DCF Algorithm

Input : A user-item rating matrix $\mathbf{R}$

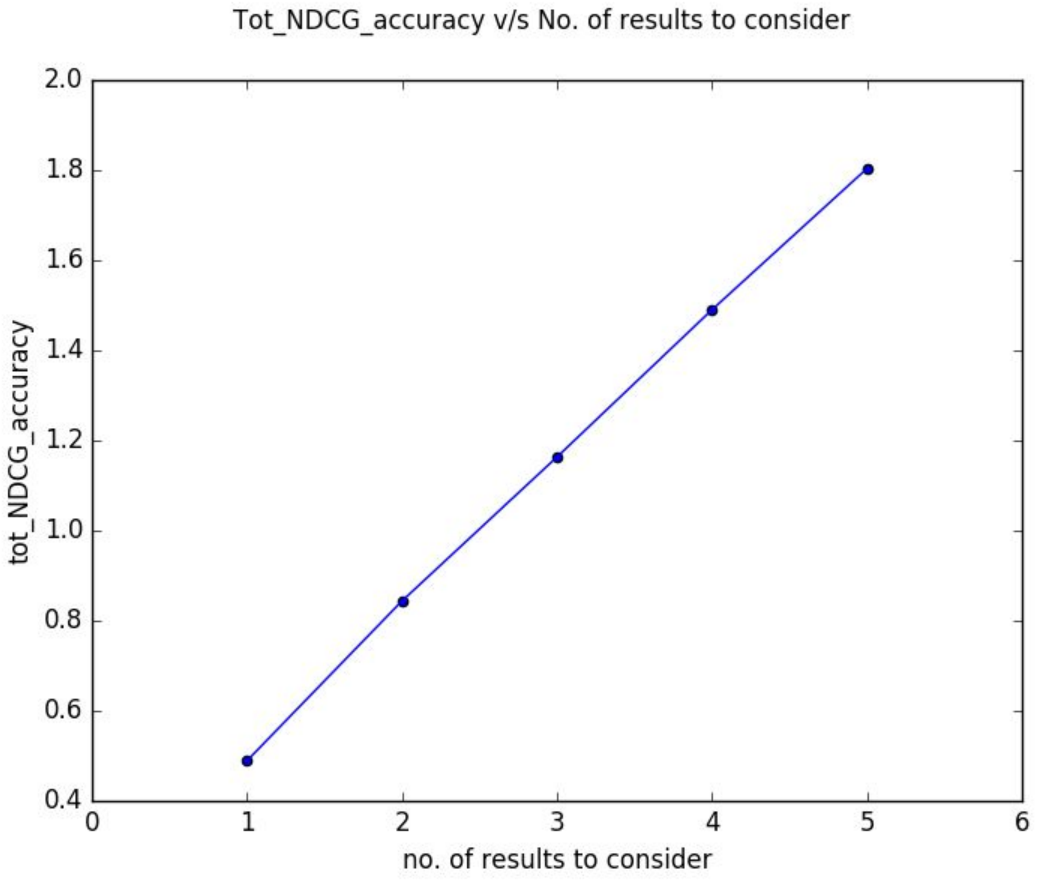
Output: A vector of weights $\mathbf{w}$ for Structural SVM classifier, a user matrix $\mathbf{U}_{k\times m}$, a item matrix $\mathbf{I}_{k\times n}$ and a diagonal matrix $\mathbf{\Sigma}$

$\mathcal{Y} \leftarrow \texttt{GenerateLabels}(\mathbf{R})$;
$\mathbf{w}, \mathbf{\Sigma} \leftarrow \texttt{Initialize}()$;
$\mathbf{U}, \mathbf{I} \leftarrow \texttt{OptimizeParaMF}(\mathbf{R}, \mathbf{\Sigma})$;
$W \leftarrow \varnothing$;
repeat
    $\mathbf{w}, W, \hat{Y} \leftarrow \texttt{CutPlane}(\mathbf{U}, \mathbf{I}, \mathbf{w}, \mathbf{\Sigma}, \mathcal{Y}, W)$;
    $\mathbf{\Sigma} \leftarrow \texttt{Update}(\mathbf{U}, \mathbf{I}, \mathbf{w}, \mathbf{\Sigma}, W, \hat{Y})$;
    $\mathbf{U}, \mathbf{I} \leftarrow \texttt{OptimizeParaMF}(\mathbf{R}, \mathbf{\Sigma})$;
until *converge*;
return: $\mathbf{w}$, $\mathbf{U}$, $\mathbf{I}$, $\mathbf{\Sigma}$

Experimental Results

The diversified recommendation algorithms can significantly improve recommendation diversity, but DCF and MF-pop cannot avoid loss of accuracy resulting from consideration of the diversity issue. In our experiments, DCF demonstrates significantly greater improvement with less accuracy loss. For example, DCF obtains 22.4% and 12.8% improvement on N T C for the two datasets respectively, with 1.9% and 0.9% loss on P A correspondingly, compared with the best performing traditional algorithm.

Our Experiments with Accuracy Coff.



THANK YOU