Relational learning using bilinear models and its application in E-commerce

Shenghuo Zhu
shenghuo.zhu@alibaba-inc.com

Alibaba Group

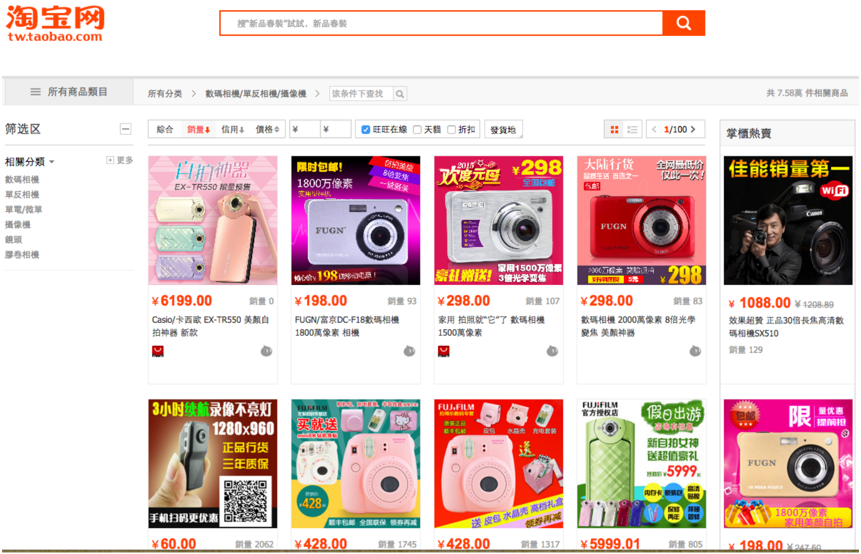
May 2, 2015

Collaborators:
Rong Jin, Qi Qian, Lijun Zhang, Mehrdad Mahdavi, Tianbao Yang

Recommendation in e-commerce

From http://ju.taobao.com/

Personalized search in e-commerce



From http://tw.taobao.com/

Recommendation to sellers

From Taobao's seller interface.

Relation and ranking

In the above tasks, we consider the relationship between

- buyer and item
- buyer/query and item
- seller and item
- seller and buyer
- ...

and rank items (or buyers) conditioned on a buyer (or a seller, a query-buyer pair).

Relation as function

- Buyer and item: u buyer, v item,
  - scoring function: $y(v; u)$
- Ranking by scores:
  - $y(v_i; u) > y(v_j; u) \Longrightarrow u$ prefers v_i over v_j.

Ranking by segmentation

- Assume that given K underlying user segment, users, u, belonging to segment k share a same scoring function:

$$y(v; u) = g_k(v)$$

- User u bought item w. Let all users that bought item w be segment k. $g_k(v)$ be the preference scores (purchase history) of item v in segument k. It is item-based collaborative filtering.

Ranking using mixture

- User u bought more than one item. The above strict segmentation assumption is relaxed. It is usually considered to use similarity between users.
- In a general term, scoring function of u is a linear combination of g_k:

$$y(v; u) = \sum_{k=1}^{K} \beta_k g_k(v),$$

where segmentation is latent.

Ranking using mixture

- User u bought more than one item. The above strict segmentation assumption is relaxed. It is usually considered to use similarity between users.
- In a general term, scoring function of u is a linear combination of g_k:

$$y(v; u) = \sum_{k=1}^{K} \beta_k g_k(v),$$

where segmentation is latent.

Ranking as matrix factorization

- For each user u,

$$y(v;u) = \sum_k \beta_k g_k(v)$$

- As each user has its own $\beta_k = f_k(u)$

$$y(v;u) = \sum_k f_k(u) g_k(v)$$

- Put $y(u,v)$, $f_k(u)$ and $g_k(v)$ as matrices

$$\mathbf{Y} = \mathbf{G}\mathbf{F}^\top \approx \mathbf{T}$$

- Convex: use low rank constraint of Y, [YLZG09].

Ranking as matrix factorization

- For each user u,

$$y(v; u) = \sum_k \beta_k g_k(v)$$

- As each user has its own $\beta_k = f_k(u)$

$$y(v; u) = \sum_k f_k(u) g_k(v)$$

- Put $y(u, v)$, $f_k(u)$ and $g_k(v)$ as matrices

$$\mathbf{Y} = \mathbf{G}\mathbf{F}^\top \approx \mathbf{T}$$

- Convex: use low rank constraint of Y, [YLZG09].

Ranking as matrix factorization

- For each user u,

$$y(v;u) = \sum_k \beta_k g_k(v)$$

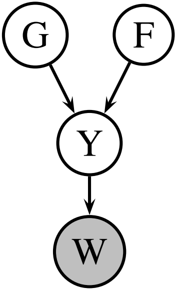
- As each user has its own $\beta_k = f_k(u)$

$$y(v;u) = \sum_k f_k(u) g_k(v)$$

- Put $y(u,v)$, $f_k(u)$ and $g_k(v)$ as matrices

$$\mathbf{Y} = \mathbf{G}\mathbf{F}^\top \approx \mathbf{T}$$

- Convex: use low rank constraint of Y, [YLZG09].



Ranking as matrix factorization

- For each user u,

$$y(v;u) = \sum_k \beta_k g_k(v)$$

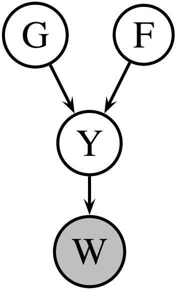
- As each user has its own $\beta_k = f_k(u)$

$$y(v;u) = \sum_k f_k(u) g_k(v)$$

- Put $y(u,v)$, $f_k(u)$ and $g_k(v)$ as matrices

$$\mathbf{Y} = \mathbf{G}\mathbf{F}^\top \approx \mathbf{T}$$

- Convex: use low rank constraint of Y, [YLZG09].



Ranking in bilinear model

- User feature x_u, and item feature z_v.

$$f_k(u) = \langle a_k, x_u \rangle\,, \qquad g_k(v) = \langle b_k, z_v \rangle$$

$$y(v; u) = \sum_k f_k(u) g_k(v) = \langle x_u, W z_v \rangle$$

where $W = \sum_k a_k b_k^\top$.

- Put $y(v; u)$, $f_k(u)$ and $g_k(v)$ as matrices

$$\mathbf{Y} = X^\top W Z$$

Ranking in bilinear model

- User feature x_u, and item feature z_v.

$$f_k(u) = \langle a_k, x_u \rangle, \qquad g_k(v) = \langle b_k, z_v \rangle$$

$$y(v; u) = \sum_k f_k(u) g_k(v) = \langle x_u, W z_v \rangle$$

where $W = \sum_k a_k b_k^\top$.

- Put $y(v; u)$, $f_k(u)$ and $g_k(v)$ as matrices

$$\mathbf{Y} = X^\top W Z$$

Issues

- How to control the complexity of learning space?
  - Rank of W, or nuclear norm $\|W\|_*$.
- When features have high dimensions, can we take the advantage of low complexity of W to reduce the computational complexity?
  - The model is essentially a linear model:

$$y \equiv \mathrm{vec}(\mathbf{Y}) = (Z \otimes X)^\top \mathrm{vec}(W) \equiv x^\top w.$$

  - A projection apporach of linear model is presented in this talk.

Issues

- How to control the complexity of learning space?
  - Rank of W, or nuclear norm $\|W\|_*$.
- When features have high dimensions, can we take the advantage of low complexity of W to reduce the computational complexity?
  - The model is essentially a linear model:

$$y \equiv \mathrm{vec}(\mathbf{Y}) = (Z \otimes X)^\top \mathrm{vec}(W) \equiv x^\top w.$$

  - A projection apporach of linear model is presented in this talk.

Solve via random projection

- To solve:

$$w = \arg\min_w \frac{\lambda}{2}\|w\|^2 + \sum_i \ell(x_i^\top w, y_i).$$

- Approach
  - Generate a random projection R of rank m, and let $\tilde{x}_i = Rx_i$.
  - Solve:

$$v = \arg\min_v \frac{\lambda}{2}\|v\|^2 + \sum_i \ell(\tilde{x}_i^\top v, y_i).$$

  - Recover: $\hat{w} = R^\top v$.

Solve via random projection

- To solve:

$$w = \arg\min_w \frac{\lambda}{2}\|w\|^2 + \sum_i \ell(x_i^\top w, y_i).$$

- Approach
  - Generate a random projection R of rank m, and let $\tilde{x}_i = Rx_i$.
  - Solve:

$$v = \arg\min_v \frac{\lambda}{2}\|v\|^2 + \sum_i \ell(\tilde{x}_i^\top v, y_i).$$

  - Recover: $\hat{w} = R^\top v$.

Issue

- $\hat{w}$ is limited in the subspace spanned by R, as $\hat{w} = R^\top v$.

Theorem 3 of [ZMJ+13]

For any $0 < \varepsilon \leq 1/2$, with a probability $1 - \exp(-(d-r)/32) - \exp(-m/32) - \delta$, we have

$$\|\hat{w} - w_*\|_2 \geq \frac{1}{2}\sqrt{\frac{d-r}{m}}\left(1 - \frac{\varepsilon\sqrt{2(1+\varepsilon)}}{1-\varepsilon}\right)\|w_*\|_2.$$

Dual space

- Dual variable and function

$$\ell_*(\alpha, y_i) = \sup_{\xi} \{\alpha_i \xi - \ell(\xi, y_i)\}$$

- Dual problem

$$\alpha = \arg\min_{\alpha} \frac{1}{2\lambda} \alpha^\top X^\top X \alpha + \sum_i \ell_*(\alpha_i, y_i).$$

- Dual problem after random projection

$$\hat{\alpha} = \arg\min_{\alpha} \frac{1}{2\lambda} \alpha^\top X^\top R^\top R X \alpha + \sum_i \ell_*(\alpha_i, y_i).$$

Proposition

For any $0 < \varepsilon \leq 1/2$, with a probability at least $1 - \delta$, we have

$$\|\tilde{\alpha} - \alpha_*\|_K \leq \frac{\varepsilon}{1-\varepsilon}\|\alpha_*\|_K,$$

provided $m = \Omega(\varepsilon^{-2} \log \delta^{-1})$.

Random Projection Dual Recovery (DuRP)

Alibaba Group 阿里巴巴集团

- Generate a random projection R of rank m, and let $\tilde{x}_i = Rx_i$.
- Solve:

$$\arg\min_v \frac{\lambda}{2}\|v\|^2 + \sum_i \ell(\tilde{x}_i^\top v, y_i).$$

- Obtain dual variables: $\tilde{\alpha}_i = \ell'(\tilde{x}_i^\top v, y_i)$
- Recover primal solution: $\tilde{w} = -\frac{1}{\lambda}\sum_i \tilde{\alpha}_i x_i$.

Random Projection Dual Recovery (DuRP)

- Generate a random projection R of rank m, and let $\tilde{x}_i = Rx_i$.
- Solve:

$$\arg\min_{v} \frac{\lambda}{2}\|v\|^2 + \sum_i \ell(\tilde{x}_i^\top v, y_i).$$

- Obtain dual variables: $\tilde{\alpha}_i = \ell'(\tilde{x}_i^\top v, y_i)$
- Recover primal solution: $\tilde{w} = -\frac{1}{\lambda}\sum_i \tilde{\alpha}_i x_i$.

Approximation error of DuRP

Theorem 2 of [ZMJ+13]

For any $0 < \varepsilon \leq 1/2$, with a probability at least $1 - \delta$, we have

$$\|\tilde{w} - w_*\|_2 \leq \frac{\varepsilon}{1 - \varepsilon}\|w_*\|_2,$$

provided $m \geq \frac{(r+1)\log(2r/\delta)}{c\varepsilon^2}$.

Dual variables in bilinear model

- Dual variables in α can be reshaped to a matrix A, where the nonzero entries correpsond to the user-item pairs having interaction.
- Then the recovered matrix is written as

$$W = XAZ^{\top}.$$

DuRP for high dimensional bi-linear model

- Very high dimension in its linear representation, i.e. $Z \otimes X$
  - Random projection: $R = R_2 \otimes R_1$.[QJZL13].
- Recovered matrix $W = XAZ^\top$ is of high dimension and usually dense, thus is difficult to apply to online service.
  - Approximated by multiplication of two low rank matrices, using approximate SVD [HMT11].

Summary

- Many applications using relational data
- A bilinear model is a straightforward approach
- To learn from massive data, dual recovery random projection.

Reference

N. Halko, P. Martinsson, and J. Tropp.
Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions.
SIAM Review, 53(2):217–288, 2011.

Qi Qian, Rong Jin, Shenghuo Zhu, and Yuanqing Lin.
An integrated framework for high dimensional distance metric learning and its application to fine-grained visual categorization.
Technical Report 2013-TR102, NEC Laboratories America, 2013.
arXiv:1402.0453.

Kai Yu, John Lafferty, Shenghuo Zhu, and Yihong Gong.
Large-scale collaborative prediction using a nonparametric random effects model.
In *ICML'09: Proceedings of the 26th Annual International Conference on Machine Learning*, pages 1185–1192, New York, NY, USA, 2009. ACM.

Lijun Zhang, Mehrdad Mahdavi, Rong Jin, Tianbao Yang, and Shenghuo Zhu.
Recovering optimal solution by dual random projection.
In *COLT'13: The 26th Annual Conference on Learning Theory*, 2013.