ADVERSARIAL TENSOR FACTORIZATION FOR CONTEXT-AWARE RECOMMENDATION

Huiyuan Chen and Jing Li
Electrical Engineering and Computer Science, Case Western Reserve University

We consider two public datasets from HetRec 2011: MovieLens and Last.fm. We compare with three popular tensor factorization methods: CANDECOMP/PARAFAC (CP), HOSVD and PITF. We use the common evaluation scheme of F1-measure for top-N recommendations.

In this work, we consider the robustness of tensor-based models for context-aware recommendations, and design a new method ATF, which combines tensor factorization and adversarial learning. We also develop a learning algorithm to solve our minmax optimization. In the future, we plan to further investigate the problem of incorporating more contextual factors to better understand the users’ behaviors and interests, such as time, location, and users’ social networks.

References

Conclusion

We propose a new adversarial tensor factorization (ATF) model that combines tensor factorization and adversarial learning for context-aware recommendations, which enhance the robustness of a recommender model. Empirical studies on two real-world datasets show that the proposed method outperforms standard tensor-based methods.

Problem Definition

We use the notation in tag recommender [4], in which the contextual factor is the tag. Let \( \mathcal{P}, \mathcal{Q}, \) and \( \mathcal{T} \) denote the set of users, items, and tags, respectively. The historical user-item-tag events can be represented by the set \( \mathcal{D} \), i.e., if \( (p, q, r) \in \mathcal{D} \) means that user \( p \) has tagged an item \( q \) with the tag \( r \). We can use a tensor \( X \in \mathbb{R}^{P \times Q \times R} \) to indicate the interactions among the users, items and tags, where \( P, Q, R \) are the number of users, items and tags, respectively. If \( (p, q, r) \in \mathcal{D} \), \( X_{pq}^{\mathcal{D}} = 1 \), otherwise, \( X_{pq}^{\mathcal{D}} = 0 \). In tag-aware recommendations, for a given user-item pair \( (p, q) \), the goal is to provide a list of tags that user \( p \) is likely to label item \( q \). This can be achieved by predicting the missing entries in the tensor \( X \).

Tensor Factorization

The PITF [4] decomposes tensor \( X \) via three pairs of inner products among the latent vectors of the user \((u_p^k, u_p^k, t_r)\), item \((v_q^k, v_q^k, t_r)\) and tag \((l_r^k, l_r^k, t_r)\):

\[
A_{TIF}(\Theta) = \langle u_p^k, v_q^k, l_r^k \rangle,
\]

where \( u_p^k, v_q^k \in \mathbb{R}^K \) denote the latent factors of the user \( p \) interacting with the item \( q \) and tag \( r \), respectively. The PITF model is optimized with Bayesian Personalized Ranking (BPR) criterion from implicit feedback [5]. The core idea is to optimize rankings by considering \( (p, q, r, r') \in \mathcal{D}_Q \) where

\[
\mathcal{D}_Q = \{(p, q, r, r') | (p, q, r) \in \mathcal{D} \land (p, q, r') \notin \mathcal{D}\}
\]

The BPR tries to minimize the following objective function:

\[
\mathcal{L}_{BPR}(\Theta) = \sum_{(p, q, r, r') \in \mathcal{D}_Q} - \ln \sigma(A_{TIF}(\Theta) + \lambda \|\Theta\|^2),
\]

where \( \sigma(\cdot) \) is the sigmoid function, \( \|\cdot\| \) is the Frobenius norm, \( \lambda \) is the regularization parameter, and \( \lambda_{u_p^k}, \lambda_{v_q^k}, \lambda_{l_r^k} \) for short.

Learning Algorithm

As the intermediate variable \( \Theta \) maximizes the objective function that is minimized by \( \Theta \), the optimization in Eq. (5) can be formulated as a minmax objective function:

\[
\Theta^\ast, \Delta^\ast = \arg \min_{\Theta} \max_{\Delta \in \mathbb{R}^{\|\Theta\|}} \mathcal{L}_{BPR}(\Theta + \Delta) + \lambda \|\Delta\|^2,
\]

where the optimization of model parameters \( \Theta \) is the minimizing player and adversarial perturbations \( \Delta \) is the maximizing player. The two players alternately play the minmax game until convergence.

Updating \( \Theta \): Given a training instance \( (p, q, r, r') \), the adversarial perturbations \( \Delta \) can be updated by maximizing:

\[
\Delta^\ast = \arg \max_{\Delta} \mathcal{L}_{BPR}(\Theta + \Delta) + \lambda \|\Delta\|^2,
\]

where \( \lambda_{u_p^k}, \lambda_{v_q^k}, \lambda_{l_r^k} \) for short.

Updating \( \Theta \): The model parameters \( \Theta \) can be obtained by minimizing:

\[
\min_{\Theta} \mathcal{L}_{ATF}(\Theta) = - \ln \sigma(A_{TIF}(\Theta) + \lambda \|\Theta\|^2),
\]

where \( A_{TIF}(\Theta) = X_{pq}^{\mathcal{D}} - X_{pq}^{\mathcal{D}} + \Delta_{pqrr}^{\mathcal{D}} \).

Fig. 1: Adversarial Tensor Factorization.