Chapter 7
Experimental Evaluation on Tensor Decomposition Methods

Abstract In this chapter, we will provide experimental results of tensor decomposition methods on real data sets in social tagging systems (STSs). We will discuss the criteria that we will set for testing all algorithms and the experimental protocol we will follow. Moreover, we will discuss the metrics that we will use (i.e., Precision, Recall, root-mean-square error, etc.). Our goal is to present the main factors that influence the effectiveness of algorithms.

Keywords Matrix decomposition · Tensor decomposition

7.1 Data Sets

To evaluate examined algorithms, we have chosen real data sets from two different STSs: BibSonomy and Last.fm, which have been used as benchmarks in past works [3].

BibSonomy: We used a snapshot of all users, items (both publication references and bookmarks), and tags publicly available on April 30, 2007. From the snapshot, posts from the database and logic programming (DBLP) computer science bibliography are excluded since they are automatically inserted and all owned by one user and all tagged with the same tag (dblp). The number of users, items, and tags is 1,037, 28,648, and 86,563, respectively.

Last.fm: The data for Last.fm were gathered during October 2007, partly through the web services API-application program interface (collecting user nicknames), partly crawling the Last.fm site. Here, the items correspond to artist names, which are already normalized by the system. There are 12,773 triplets in the form user–artist–tag. To these triplets correspond 4,442 users, 1,620 artists, and 2,939 tags.

Following the approach of [3] to get more dense data, we adapt the notion of a p-core to tripartite hypergraphs. The p-core of level k has the property, that each user, tag, or item has/occurs in at least k posts. For both data sets we used k = 5. Thus,

P. Symeonidis and A. Zioupos, *Matrix and Tensor Factorization Techniques for Recommender Systems*, SpringerBriefs in Computer Science,
DOI 10.1007/978-3-319-41357-0_7

for the BibSonomy data set there are 105 users, 246 items, and 591 tags, whereas for the Last.fm data set, there are 112 users, 234 items, and 567 tags.

7.2 Experimental Protocol and Evaluation Metrics

For item recommendations, all tensor decomposition algorithms have the task to predict items of users' postings in the test set. Higher Order SVD algorithm is modified appropriately to recommend items to a target user. In particular, the initial tensor represents a quadruplet $\{u, t, i, p\}$ where p is the likeliness that user u will tag item i with tag t. Therefore, items can be recommended to u according to their weights associated with a $\{u, t\}$ pair.

We performed a fourfold cross-validation, thus each time we divide the data set into a training set and a test set with sizes 75 % and 25 % of the original set, respectively. Based on the approach of [2, 4], a more realistic evaluation of recommendation should consider the division of items of each test user into two sets: (i) the *past* items of the test user and (ii) the *future* items of the test user. Therefore, for a test user, we generate recommendations based only on items in his past set. This simulates the real-world applications, where users gradually tag items and receive recommendations before they provide all their tags. As most existing works ignore this division, their reported performance corresponds to the best case, because they indirectly exploit a priori known information (items in the future set). With the division into past and future sets, accuracy is expected to decrease compared to the best case when the two sets are identical. However, reported performance is more indicative of real-world applications. The default sizes of past and future sets are 50 % and 50 %, respectively, of the number of items tagged by each test user.

As performance measures for item recommendations, we use the classic metrics of precision and recall. For a test user that receives a list of N recommended items (top-N list), precision and recall are defined as follows:

- *Precision* is the ratio of the number of relevant items in the top-N list (i.e., those in the top-N list that belong to the future set of items posted by the test user) to N.
- *Recall* is the ratio of the number of relevant items in the top-N list to the total number of relevant items (all items in the future set posted by the test user).

7.3 Sensitivity Analysis of the HOSVD Algorithm

In this section, we first conduct experiments to study the influence of core tensor dimensions on the performance of the described HOSVD algorithm. If one dimension of the core tensor is fixed, we can find that the recommendation accuracy varies as the other two dimensions change, as shown in Fig. 7.1. The vertical axes denote precision and the other two axes denote corresponding dimensions. For each figure, one dimension is fixed and the other two dimensions are varied. Thus, for the leftmost figure, the tag dimension is fixed at 200 and the other two dimensions change. For the middle figure, the item dimension is fixed at 105. For the rightmost figure, the user dimension is fixed at 66.

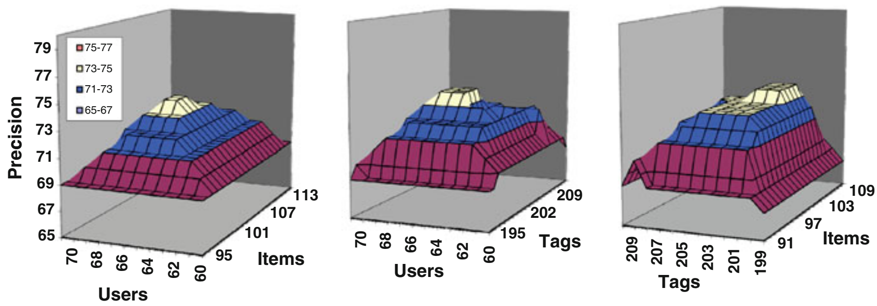

Fig. 7.1 Performance of the HOSVD algorithm as dimensions of the core tensor vary for the BibSonomy data set. For the *leftmost figure*, tag dimension is fixed at 200 and the other two dimensions change. For the *middle figure*, item dimension is fixed at 105. For the *rightmost figure*, user dimension is fixed at 66

Our experimental results indicate that a 70 % of the original diagonal of $S^{(1)}$, $S^{(2)}$, $S^{(3)}$ matrices can give good approximations of A_1, A_2, A_3 matrices. Thus, the numbers c_1, c_2, and c_3 of left singular vectors of matrices $U^{(1)}$, $U^{(2)}$, $U^{(3)}$ after appropriate tuning are set to 66, 105, and 200 for the BibSonomy data set, whereas they are set to 40, 80, and 190 for the Last.fm data set.

Next, we study the influence of the proposed kernel smoothing scheme on recommendation accuracy of the HOSVD algorithm in terms of precision. We present our experimental results in Fig. 7.2a, b, for both the BibSonomy and Last.fm data sets. As shown, the smoothing kernel method can improve the performance accuracy. The results are consistent in both data sets.

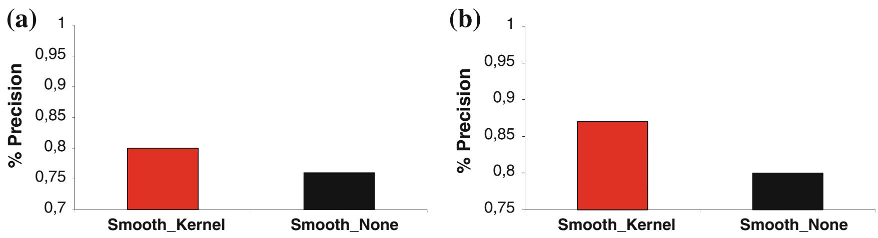

Fig. 7.2 Precision of the HOSVD algorithm associated with and without a smoothing scheme for the **a** BibSonomy data set and **b** Last.fm data set

7.4 Comparison of HOSVD with Other Tensor Decomposition Methods in STSs

In this section, we compare several tensor decomposition methods in data sets that concern the domain of STSs. To evaluate examined algorithms, we have chosen real data sets from two different social tagging systems: BibSonomy and Last.fm [7]. We have tested the following state-of-the-art methods:

- ClustHOSVD(tfidf + semantics): This is the ClustHOSVD algorithm [7], which incorporates TFIDF as a weighting schema and it is combined with the semantic similarity of tags.
- ClustHOSVD(tfidf): This is the ClustHOSVD algorithm [7], which incorporates only the term frequency-inverse document frequency.
- TFC: Rafailidis and Daras [6] proposed the Tensor Factorization and Tag Clustering model, which is a tensor factorization and tag clustering model that uses a TFIDF weighting scheme.
- HOSVD: This is the Tucker's tensor decomposition method [5].
- LOTD: Cai et al. [1] proposed low-order tensor decomposition (LOTD), which is based on low-order polynomial terms on tensors (i.e., first and second order).
- FOTD: Full Order Tensor decomposition (FOTD) proposed by Cai et al. [1] which incorporates, except the first and second terms, also the third-order polynomial term.

The parameters we used to evaluate the performance of ClustHOSVD(tfidf + semantics), ClustHOSVD(tfidf), HOSVD, Tensor Factorization and Tag Clustering model, LOTD, and FOTD are identical to those reported in the original papers. We measure precision versus recall for all six algorithms. The results for the BibSonomy and Last.fm data sets are depicted in Figs. 7.3 and 7.4, respectively.

For both data sets, ClustHOSVD(tfidf + semantics) outperforms the other comparison methods. The reason is that it exploits both the conventional cosine similarity and the semantic similarity of tags. In contrast, the Tensor Factorization and Tag Clustering model incorporates the TFIDF weighting scheme without exploiting also semantic information. FOTD presents the worst results, which are according to what

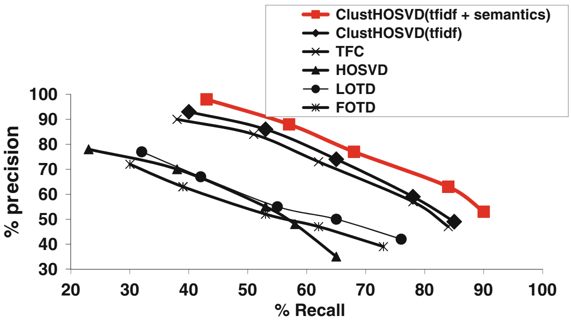

Fig. 7.3 Comparison between variations of ClustHOSVD(tfidf + semantics), ClustHOSVD(tfidf), HOSVD, and LOTD/FOTD algorithms in terms of precision–recall curve for BibSonomy data set

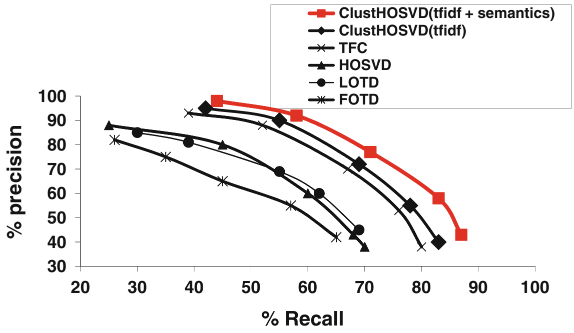

Fig. 7.4 Comparison between variations of ClustHOSVD(tfidf + semantics), ClustHOSVD(tfidf), HOSVD, and LOTD/FOTD algorithms in terms of precision–recall curve for Last.fm data set

Cai et al. [1] have reported in their paper. That is, the LOTD method had better results than FOTD in terms of precision–recall diagram, because of the overfitting problem which existed in all data sets.

References

1. Cai, Y., Zhang, M., Luo, D., Ding, C., Chakravarthy, S.: Low-order tensor decompositions for social tagging recommendation. In: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining (WSDM'11), pp. 695–704. ACM (2011)
2. Herlocker, J., Konstan, J., Terveen, L., Riedl, J.: Evaluating collaborative filtering recommender systems. ACM Trans. Inf. Syst. **22**(1), 5–53 (2004)
3. Hotho, A., Jaschke, R., Schmitz, C., Stumme, G.: Information retrieval in folksonomies: search and ranking. In: The Semantic Web: Research and Applications, pp. 411–426 (2006)
4. Huang, Z., Chen, H., Zeng, D.: Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering. ACM Trans. Inf. Syst. **22**(1), 116–142 (2004)
5. De Lathauwer, L., De Moor, B., Vandewalle, J.: A multilinear singular value decomposition. SIAM J. Matrix Anal. Appl. **21**(4), 1253–1278 (2000)
6. Rafailidis, D., Daras, P.: The tfc model: tensor factorization and tag clustering for item recommendation in social tagging systems. Trans. Syst. Man Cybern.—Part A: Syst. Humans (2012)
7. Symeonidis, P.: ClustHOSVD: Item recommendation by combining semantically enhanced tag clustering with tensor HOSVD. IEEE Trans. Syst. Man Cybern. (2015)