Decision Support Systems 54 (2013) 880–890

Contents lists available at SciVerse ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss

Recommendation as link prediction in bipartite graphs: A graph kernel-based machine learning approach

Xin Li [a,*], Hsinchun Chen [b]

[a] Department of Information Systems, City University of Hong Kong, 83 Tat Chee Avenue, Kowloon Tong, Hong Kong
[b] Department of MIS, University of Arizona, 1130 East Helen St., Rm. 430, Tucson, AZ, USA

ARTICLE INFO

Article history:
Received 23 August 2011
Received in revised form 21 June 2012
Accepted 18 September 2012
Available online 4 October 2012

Keywords:
Recommender systems
Kernel-based methods
Link prediction
Bipartite graph
Collaborative filtering

ABSTRACT

Recommender systems have been widely adopted in online applications to suggest products, services, and contents to potential users. Collaborative filtering (CF) is a successful recommendation paradigm that employs transaction information to enrich user and item features for recommendation. By mapping transactions to a bipartite user–item interaction graph, a recommendation problem is converted into a link prediction problem, where the graph structure captures subtle information on relations between users and items. To take advantage of the structure of this graph, we propose a kernel-based recommendation approach and design a novel graph kernel that inspects customers and items (indirectly) related to the focal user–item pair as its context to predict whether there may be a link. In the graph kernel, we generate random walk paths starting from a focal user–item pair and define similarities between user–item pairs based on the random walk paths. We prove the validity of the kernel and apply it in a one-class classification framework for recommendation. We evaluate the proposed approach with three real-world datasets. Our proposed method outperforms state-of-the-art benchmark algorithms, particularly when recommending a large number of items. The experiments show the necessity of capturing user–item graph structure in recommendation.

1. Introduction

Consumers nowadays have an increasing amount of experience with online recommender systems, such as when buying books from Amazon, borrowing movies from Netflix, or setting up friend circles on Facebook. Compared with search engines, recommender systems apply information filtering mechanisms that may lead users to items they are not aware of or cannot access using a keyword search. Well-designed recommender systems save users' time, improve customer satisfaction [29], and promote sales [12]. To better capture users' interests and make effective recommendations, it is necessary to combine multiple models [35] and make effective use of different types of data, such as user information, item information, and transaction information (business transactions, browsing activities, review activities, etc.) [32,49]. The collaborative filtering (CF) recommendation paradigm models users' collaborative behaviors reflected in transactions and cross-recommends products to users.There are generally two types of recommendation tasks, predicting purchase vs. predicting rating. Transaction/purchase is essentially an implicit and coarse rating on preferring an item [22]. Furthermore, it does not differentiate the statuses of "unknown" vs. "unlike." Thus, the two tasks are quite different from each other according to their computational nature. In real-world e-commerce applications, there generally exists more transaction information than explicit rating information. In this research, we focus on the modeling and prediction of transactions.

Given a set of users U and a set of objects/items O, we represent transactions T between them as a user-object interaction graph $G=(V, E)$, where $V=U\cup O$ and $E=\{(u,o)\colon u\in U,\, o\in O,\, u\rightarrow o\in T\}$. This is a bipartite graph since user nodes can connect only to item nodes and vice versa. Under this representation, recommendation is equivalent to link prediction between users and items based on the existing graph (of previous transactions). The graph representation reveals subtle relations between indirectly connected users and items. Several collaborative filtering heuristic algorithms have explored the structure of user–item interaction graphs to improve recommendation performance [27,70]. However, few learning-based methods explicitly utilize the graph in constructing effective personalized recommendation models. Existing learning-based recommendation algorithms usually rely on explicit feature extraction, which is difficult to apply onto graph-structured data due to the requirements of superior computational capacity and extensive domain knowledge (to design features).

In this research, we propose a generic kernel-based machine learning approach of link prediction in bipartite graphs and apply it in recommender systems. We inspect nodes and links that are close to a focal user–item pair as its context to predict the possibility for the pair to interact, i.e., the possibility that the user may use/buy the item. We propose a novel graph kernel to capture the structure and user/item features in the context of focal user–item pairs and feed it into the one-class SVM algorithm to build the prediction

* Corresponding author. Tel.: +852 34424446; fax: +852 34420370.
E-mail addresses: xin.li@cityu.edu.hk (X. Li), hchen@eller.arizona.edu (H. Chen).

0167-9236/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.dss.2012.09.019

model. We examine the validity and computational efficiency of the graph kernel. We demonstrate the performance of this recommendation approach using three real-world datasets.

The paper is organized as follows. The second section reviews related studies on recommendation algorithms. The third section introduces the proposed graph kernel-based recommendation framework. The fourth section describes experiments on three real-world datasets and discusses the experimental results. The last section summarizes the findings and proposes directions for future research.

2. Literature review

2.1. Recommendation algorithms

There have been several survey papers on recommender system studies [3,24]. In this research, we focus on the utilization of graph structure in the design of recommendation algorithms and review previous studies according to the feature types and computational techniques.

First, previous studies generally use two types of features in designing the recommendation algorithms: collective local features and graph-related features.

Collective local features capture the collective characteristics of individual user/item information, such as a user's demographic characteristics, content of interest [32], item's specifications, transaction contexts (environment, time, etc.), and temporal usage patterns (which reflect user's characteristics). Collective local features directly show differences between people and the products they bought. Thus, researchers design similarity measures to cross-recommend similar products among similar users [5]. Researchers also proposed several probabilistic latent variable models on users' usage/purchase histories that group users/items to (hidden) classes to estimate the probabilities of user–item interactions [21].

Graph-related features highlight interactions between related users/items. By representing transactions using a user–item interaction graph, the subtle relations between indirectly connected users and items can be modeled by the graph structure. In light of the graph theory, topological characteristics of nodes on the graph were considered an indication of items' attractiveness and importance in the network, which can be used in recommendation [18]. Graph structure can be used to design similarity measures for cross-recommendation. The bipartite user–item graph can also be projected onto a unipartite user/item graph [70] to simplify the graph structure. Furthermore, some researchers construct graphs based on user/item similarities. Such artificial graphs can help alleviate the data sparsity problem [8]. In Webpage recommendation applications, a Webpage browsing graph has been employed to capture user behaviors and interest in certain contents [62].

Second, previous recommendation algorithms generally include two types of computational techniques: heuristic (i.e., memory-based) algorithms and learning-based (model-based) algorithms.

Heuristic algorithms are designed based on rules/measures provided by domain experts. A widely adopted heuristic is to recommend the most popular items (according to number of sales) to users. User/item similarity-based cross-recommendation is also a type of heuristic, which relies on designing appropriate similarity measures. In previous research, several similarity measures have been proposed based on user/item features [32]and transactional characteristics [5]. The most popular measures were the Pearson correlation coefficient [52] and cosine-based similarity [55].

User/item similarities have been combined with the interaction graph structure to capture information in connected nodes for link prediction [36]. Liben-Nowell and Kleinberg [43] explored the algorithms that iteratively compare nodes and update node similarities to a global node similarity measure. Taking a graph view of transactions, several previous studies proposed using eigenvector-based node ranking algorithms to rank items for recommendation. These algorithms are in general similar to PageRank [18,19,23] and HITS [27], which model influence of users and attractiveness of products based on the connectivity of the graph. Furthermore, Fouss et al. proposed that closer users and items on the interaction graph (as measured by conducting random walks between them) may have a higher probability of interacting [14,15]. Huang et al. [26] also conjectured that possible links should lead to some topological changes on the graph, such as on clustering coefficients.

Compared to heuristic algorithms, learning-based methods build models based on existing data instances. Several probabilistic models, such as probabilistic latent semantic analysis (PLSA) [21], have been proposed to learn from business transactions [50,67,68] to predict future purchases. It is a common practice for such models to design and capture hidden user classes according to purchase histories and use the classes to aid recommendation. Temporal information of user purchase histories can be captured with a maximum entropy model to better tackle the recommendation problem [28,48]. Other research has explored finding the most effective features in business transactions that can be fed into mature machine learning models to do recommendation. For example, the probabilistic relational model (PRM) [17,45] was utilized to make predictions based on product features [10]. The regression model has been used with product features [60] and user rating features [2] to make recommendations. The SVM algorithm was applied on product features [64] and transaction context features, such as time, weather, etc., [4,46] to examine the probability that a product will be selected by users.

Due to its success in a recent Netflix contest, the matrix factorization method has attracted significant interest. Matrix factorization assumes that each item and user can be characterized by a set of factors whose inner product is the user's rating of the item [35], which are essentially hidden groups of users and items. These factors can be learned by minimizing the estimation error using the stochastic gradient descent or alternating least squares methods. Variations of matrix factorization techniques have been explored for computational efficiency and data characteristic concerns [16,34,47,53]. Notably, Hu et al. [22] have developed a method for matrix factorization on binary matrices, which better suits the purchase recommendation problem.

It is also possible to use graph-related features in the learning-based paradigm. Huang et al. [25] extended the PRM framework and defined rules to extract features on connected nodes in the user–item interaction graph to construct recommendation models. Yajima [65] used a Laplacian kernel to capture the positional relations among nodes on the graph and built one-class SVM models for each user to recommend items that are positionally closer to their previously bought items. Under an unsupervised learning paradigm, Reddy et al. [51] proposed to use a graph-based clustering algorithm to group similar users on the user–item graphs. The user groups help recommendation. Table 1 summarizes major research on recommendation algorithms. In general, a significant amount of effort focused on learning-based models using collective local features. Several studies explored the use of graph features in heuristics. There are limited studies using learning-based models on explicit graph data representations.

2.2. Link prediction in graphs

Using graph representations, a recommendation problem can be converted to a link prediction problem, which is an active research area in computer science. To our best knowledge, most link prediction research focused on unipartite networks, such as social networks, Webpages, and email networks. However, the recommendation problem is on bipartite networks.

In link prediction research, several heuristics have been proposed. For example, sociologists identified a strong homophily phenomenon in friendships [44]. This rationale led to a heuristic that uses actor similarity to infer social links. Liben-Nowell and Kleinberg [43] adopted and proposed several graph-based similarity measures for social