



Co-purchaser Recommendation Based on Network Embedding

Jihong Chen¹, Wei Chen^{1,2}, Jinjing Huang¹, Jinhua Fang¹, Zhixu Li¹, An Liu¹, and Lei Zhao¹(✉)

¹ School of Computer Science and Technology, Soochow University, Suzhou, China
jhchen@stu.suda.edu.cn,

{robertchen, huangjj, jhfang, zhixuli, anliu, zhao1}@suda.edu.cn

² Institute of Artificial Intelligence, Soochow University, Suzhou, China

Abstract. Although recommending co-purchasers for a target buyer on the group buying is an interesting problem, existing studies haven't paid attention to this topic. Different from the collaborator recommendation that only considers users with high similarity to the target user, co-purchaser recommendation takes both users with high and weak similarity into account, and the recommendation results can achieve high recall and diversity. However, the task turns out to be a challenging problem since it is hard to make a precise recommendation for buyers with weak similarity. To address the problem, we propose the following two methods. In the first one, we directly impose a penalty to the weakly similar co-purchasers in the embedding space. To further improve the recommendation performance, in the second one, we smoothly increase the co-occurrence probability of the weakly similar co-purchasers by truncated bias walk. Our experimental results on real datasets show that the proposed methods, particularly the latter, can effectively complete the co-purchaser recommendation and has a high recommendation performance.

Keywords: Group buying · Collaborator recommendation · Network embedding · Truncated walk

1 Introduction

Co-purchase, also known as group buying, in which people with the same merchandise interests form a group and conduct the purchase together to achieve discounts [1]. In recent years, we have witnessed the prosperity of it in some online shopping services (e.g., Taobao¹, Groupon², and PDD³) benefit from the advanced electronic payment technology and convenient express service.

In real applications, the co-purchase usually includes the following steps: Firstly, merchants promise to offer products or services with discount on the

¹ <https://www.taobao.com/>.

² <https://www.groupon.com/>.

³ <https://www.pinduoduo.com/>.

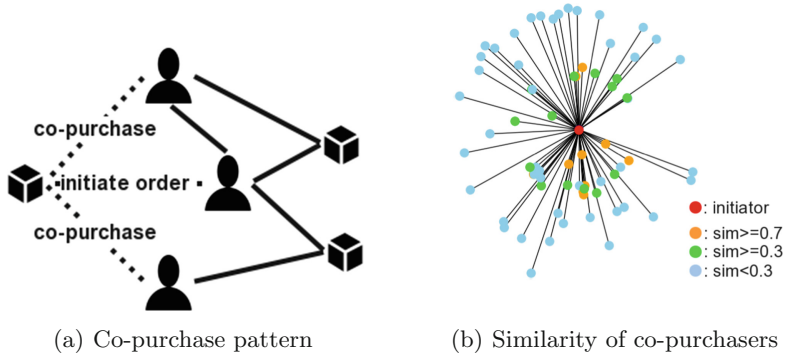


Fig. 1. A toy example of copurchase: (a) simple co-purchase pattern; (b) similarity of co-purchasers in one co-purchase on the TaoBao. The distance between nodes \propto their shortest path and sim represents cosine similarity of their shopping history.

condition that a certain number of customers would make the purchase; Then, an initiator manually invite friends, followers, and like-minded people to participate in the purchase; Finally, co-purchasers accept the invitation and benefit from lower price which is unavailable to the individual buyer [2].

Most group recommendation algorithms generate an item suggestion for a group [3]. However, as the above discussion shows, the co-purchase is a distinctive group activity, in which members of the group are uncertain until the purchase order is submitted, that is to say, we need to pay more attention to who is the appropriate co-purchaser for an initiator, rather than focusing on what is the right item for a group like the traditional group recommendation. How to choose the appropriate co-purchasers? There are two methods for the problem: the manual invitation of the initiator and the automatic recommendation of the recommendation system. Although the former is a classic solution commonly used by industry [2], it still has many flaws, such as inefficiency, insufficient demand for co-purchaser, and the limited quantity of participants. Compared with the first method, the latter one is a more promising method, as the superiority of the recommendation system has been proven in many other areas [4].

If we consider a co-purchase transaction as a collaboration between the initiator and the co-purchaser, then we can solve the co-purchaser problem build upon the positive experiences of previous collaborator recommendation tasks. Much literature has been published [5–7] on collaborator recommendation systems as well as their real-world applications, such as co-author recommendation in the academic social network [5, 6, 8], developer recommendation in the open source community [7], and co-star recommendation in the film industry [9].

In the above collaborator recommendation tasks, finding robustly similar users for the target user is a core task, for example, in an academic network, people tend to repetitively collaborate with fellow researchers with close researcher topics [6, 8]. It is also a classic idea in many recommendation algorithms, such as

the typical user-based collaborative filtering approach distinguish the target users interests and preferences by aggregating the highest similar users [4]. However, the co-purchaser recommendation is a special scenario, in which not all co-purchasers have high similarity with the initiator. As shown in Fig. 1b, a large number of weakly similar users also participated in the co-purchase transaction, but they are usually not noticed by existing recommendation methods.

The co-purchaser recommendation is a challenging task, since the identification of potential co-purchasers from the weakly similar users is not easy. To tackle the problem, two embedding strategies are proposed, which capture weakly similar co-purchasers from different perspectives. In the first one, we propose a multi-layered learning architecture with PathSim [5] diffusion, namely PathSim Diffused Structural Deep Network Embedding (PDSNE), which connects weakly similar users by PathSim and directly impose a penalty to the mapping error of the weakly similar users. Obviously, it is a forthright strategy that is beneficial to the weakly similar user, but it will inevitably damage the original network structure. In the second one, we devise a co-occurrence model based on truncated walking paths, namely co-purchasers to vectors (cop2vec). More specifically, cop2vec can smoothly improve the co-occurrence probability of the weakly similar co-purchasers by truncated bias walk, and thus learn a more reasonable representation for co-purchasers. In this way, not only those co-purchasers who are highly similar to the initiator are close to the initiator, but also the potential co-purchasers with weakly similar to the initiator.

The contributions of our paper are summarized as follows:

- To the best of our knowledge, this is the first work that shows how to recommend co-purchasers in group buying. This is an important subject because co-purchaser recommendation has been proved to be more effective than the handcrafted invitation.
- We propose PDSNE and cop2vec, two efficient co-purchaser recommendation strategies, which effectively perceive weakly similar co-purchasers.
- Through extensive experiments, demonstrates the efficacy and scalability of the presented methods in the co-purchaser recommendation task.

The rest of the paper is organized as follows. Section 2 presents related work. Section 3 introduces the embedding methods (PDSNE and cop2vec) with the details of how to capture the weakly similar users for the co-purchaser recommendation. Section 4 describes the experimental setup and presents qualitative and quantitative results. Section 5 gives the conclusion with future work.

2 Related Work

2.1 Similarity Search

Similarity search is a basic operation in collaborative filtering, and it can be directly used as a simple strategy to find a collaborator [5, 11]. When the input is in the form of a scoring matrix, the common similarity search approach including Cosine Similarity and Pearson Correlation Coefficient [11]. In the network

analysis task [12], a large number of similarity search methods with different definitions of similarity have been proposed such as Common Neighbors, Jaccard Index, and Adamic-Adar Index, In addition to the above mentioned local-based function, Sun et al. [5] proposed a path-based similarity measure to suit peer objects.

2.2 Network Embedding

The low-dimensional representation learning of recommendation objects is a classic approach to the recommendation system [8,13,14], for example, one of the most efficient and best used recommend methods is matrix factorization in which users and items are represented in a low-dimensional latent factors space [4]. Network embedding aims at learning low-dimensional vectors for the vertices of a network [10,15,16], such that the proximities among the original network are preserved in the low-dimensional space.

Recent progress in neural embedding methods for linguistic tasks has dramatically advanced state-of-the-art Natural Language Processing (NLP) capabilities. These methods attempt to map words and phrases to a low dimensional vector space that captures semantic relations between words [17]. Specifically, Skip-gram with Negative Sampling (SGNS), also known as word2vec, set new records in various NLP tasks. Inspired by it, DeepWalk [15] is proposed as a method for learning the latent representations of the nodes of a social network. The method aims to transplant the word-context concept in documents into networks, and combines truncated random walk with Skip-gram model to achieve this. We can utilize the model to learn the low-dimensional and distributed embedding of nodes as it facilitates the preservation of its structural context—local neighborhoods—in the original network. On this basis, WALKLETS [18] and node2vec [19] further extend DeepWalk utilize high-order proximities and bias walk. LINE [16] is recently proposed embedding approach for large scale networks. By design, LINE learns two representations separately, one preserving first-order proximity and the other preserving second-order proximity. Then, Wang et al. extended the method using a deep autoencoder [20].

3 Co-purchaser Recommendation

3.1 Formalizations

In this section, we first introduce the concept of interaction networks, and then give a formal definition of the co-purchase recommendation problem.

Interaction Networks. An interaction network is defined as a graph $G = (V, E)$, where V and E represent the node set and the edge set. For example, one can represent the interaction network in Fig. 1a with buyers and products as nodes, wherein edges indicate the interactions, such as the purchase (buyer to product) and the trust (buyer to buyer). In order to ensure data consistency, the edges are unweighted.

Co-purchaser Recommendation. Network embedding is to learn a low-dimensional vector for each node. Let U be the vector set of all buyers and let P be the vector set of all possible products. On the basis, we define the co-purchaser recommendation goal as follows: Given a buyer i and a product j , we can now assign a co-purchase score S to each buyer c , it can be written as

$$S(c, (i, j)) = U_c \cdot U_i^T + U_c \cdot P_j^T \tag{1}$$

3.2 Multi-layered Learning Architecture with PathSim

In this section, we first define the notation of PathSim diffusion. Then we introduce the multi-layered learning model of PDSNE. At last we present some discussions and analysis on the model.

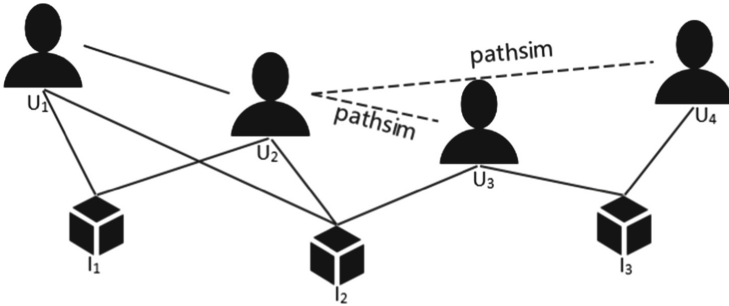


Fig. 2. The PathSim diffused network (The initiator links to the weakly similar users by PathSim).

Given a network $G = (V, E)$, we can obtain its adjacency matrix $S \in \mathbb{R}^{V \times V}$. we have $s_{ij} = 1$ if there exists a link between i and j , and $s_{ij} = 0$ otherwise. for each row $s_i = \{s_{ij}\}_{j=1}^n$. In reality, the observed links only account for a small portion. There exist many co-purchasers who have some connectivity with the initiator but no direct links, especially weakly similar co-purchasers. we define the PathSim diffused matrix $P \in \mathbb{R}^{V \times V}$ by extending PathSim measurement proposed in [5] as follows:

$$p_{ij} = \begin{cases} s_{ij} & \text{if } s_{ij} \neq 0 \\ \frac{2 \times \text{path}(i,j)}{\text{path}(i,i) + \text{path}(j,j)} & \text{if the shortest length between } i \text{ and } j < R \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

Where $\text{path}(i, j)$ is the number of paths between i and j , $\text{path}(i, i)$ is the number of paths between i and i , $\text{path}(j, j)$ is the number of paths between j and j . Notice the length of all paths is the shortest length of the path between i and j . Where R is the range of the PathSim diffusion. In theory, the score of the p can

measure the connectivity between vertexes and normalized by the visibility of vertexes. As shown in Fig. 2, there is 1 path between u_2 and u_4 , 1 path between u_4 and u_4 and 3 paths between u_2 and u_2 . We can calculate that the score of the p is 0.5.

Intuitively, if two vertexes share many common neighbors, they tend to be similar. As shown in Fig. 2, u_1 has the same shopping history as u_2 , so they are similar and can be purchasing together. to model the neighbor structure, also known as the second-order proximity, autoencoders have emerged as one of the commonly used building blocks [20,21]. An autoencoder performs two actions, i.e. the encoder and decoder. The encoder consists of multiple non-linear functions $f(\cdot) = f_{\theta_k}(\dots f_{\theta_1}(\cdot))$ that map the input data to the representation space. The decoder also consists of multiple functions $g(\cdot) = g_{\hat{\theta}_1}(\dots g_{\hat{\theta}_k}(\cdot))$ mapping the representations in representation space to reconstruction space. Let us assume that $f_{\theta_1}(x) = \sigma(W_1x + b_1)$ and $g_{\hat{\theta}_1}(x) = \sigma(\hat{W}_1x + \hat{b}_1)$, where σ is the activation function, $\theta = (W, b)$ are the parameters involved in the encoder, and $\hat{\theta} = (\hat{W}, \hat{b})$ are the parameters involved in the decoder. The goal of the autoencoder is to minimize the following reconstruction loss function

$$L_n = \sum_i \|s_i - g(f(s_i))\|_2^2 \quad (3)$$

Naturally, it is necessary for network embedding to preserve the link structure. We wish to see that the stronger the link between the two vertexes, the more similar their embedding vectors. Many classical recommendation algorithms have the objective, for example, in matrix factorization techniques, the higher the user's rating of the item, the more overlapping their latent vectors. In addition, by adding the penalty of PathSim score, these weakly similar co-purchasers will be close to the initiator in the embedding space. The loss function for this goal is defined as follows:

$$L_l = \sum_{i,j} p_{i,j} \|f(s_i) - f(s_j)\|_2^2 \quad (4)$$

To preserve both neighborhood structure and link structure, we jointly minimize the objective function by combining Eqs. 4 and 3:

$$L = L_l + \alpha L_n \quad (5)$$

As shown in previous works [20], we use stochastic gradient descent (SGD) to optimize the model. The key step is to calculate the partial derivative of the parameters $\{\theta, \hat{\theta}\}$. Ultimately, the embedding vectors can be computed by the encoder. However, While the PathSim diffusion can be beneficial to weakly similar co-purchasers. It can also damage the original network structure and bring some negative impact on the general reconstruction of the network. We want to use a smoother way to perceive weakly similar co-purchasers and minimize the impact on the basic network features.

3.3 Co-occurrence Model Based on Truncated Walk

For the consideration of being self-contained, we briefly review the key idea of the co-occurrence model. The co-occurrence model is first used for linguistic tasks, and attempt to map words to a low dimensional vector space that captures semantic relations between words. Specifically, the SGNS model aims to maximize the co-occurrence probability among the words that appear within a window. Inspired by it, DeepWalk [15] is proposed as a method for learning the latent representations of the nodes of a social network. The method samples a set of paths from the input graph using the truncated random walk. Each path sampled from the graph corresponds to a sentence from the corpus, where a node corresponds to a word. Given a path consisting of nodes $w_1 - w_k$, The co-occurrence model objective is to maximize the following term:

$$\frac{1}{K} \sum_{i=1}^K \sum_{-c < j < c} \log P(w_{i+j}|w_i) \tag{6}$$

Where c is the context window size. Applying negative sampling [17], P is defined as:

$$P(w_{i+j}|w_i) = \sigma(\mathbf{u}_i^T \mathbf{u}_j) + \sum_{t \in NS} \sigma(-\mathbf{u}_i^T \mathbf{u}_t) \tag{7}$$

Where $\sigma(x) = 1/(1 + \exp(-x))$, and NS is the negative samples for w_i .

By applying the co-occurrence model in formula 6, Frequently co-occurring nodes in a path share similar neighborhoods (In this section, the definition of neighborhoods is slightly different from PDSNE, it usually refers to the window in paths, not just the one-hop neighbors in networks.) and get similar embedding [13,15]. For example (see Fig. 3a), u_1 may co-occur with u_2 most frequently, we will naturally recommend u_1 as a co-purchaser to u_2 . However, it is still challenging to recommend weakly similar co-purchaser like u_3 . Unfortunately, a large number of co-purchaser have distributed the long tail of similarity, and they are difficult to be perceived by the existing recommendation approaches.

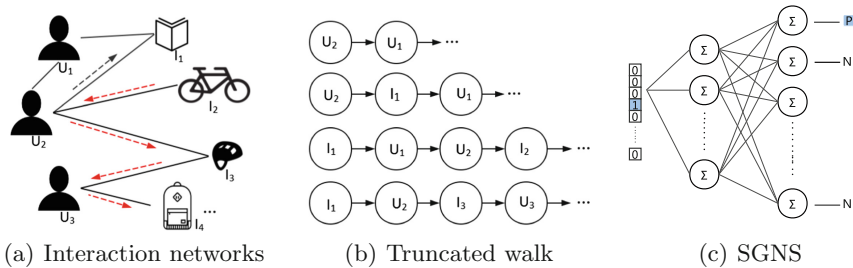


Fig. 3. Overview of co-occurrence model

To address the weakly similar co-purchaser problem, We propose a novel neighborhoods sampling strategy that is beneficial to the weakly similar co-purchasers, Which can smoothly improve the co-occurrence probability of the weakly similar co-purchasers by truncated bias walk.

General Neighborhoods Sampling Strategy. Network embedding methods based on the SGNS architecture reconstructs network features by learning the notion of neighborhoods. We first briefly introduce the general neighborhoods sampling strategy—truncated random walk, formally, a random walk begins at the source node s and gets a node sequence of fixed length le , let n_i denote the i th node in the sequence, starting with $n_0 = s$. The node n_i is generated by the following distribution.

$$P(n_i = v | n_{i-1} = u, i < le) = \begin{cases} \frac{\pi_{uv}}{\sum_{x \in \Gamma(u)} \pi_{ux}} & \text{if } v \in \Gamma(u) \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

where $\Gamma(u)$ is the one-hop neighbors of node u , and π_{uv} is the unnormalized transition probability between nodes u and v (e.g., the edge weights w_{ux}).

However, the simple way not allow us to account for the network structure and guide our search procedure to explore different types of network neighborhoods. Additionally, the farther nodes are difficult to capture, and may not even be touched in the finite number of the truncated walk. As shown in Fig. 1a, consider a truncated random walk arrived at the purchaser node u_2 , after which the walk will have multiple paths to reach another purchaser node u_1 , that is, u_2 will frequently coexist with u_1 in the node sequence generated by walks, and finally the SGNS model maps two nodes that frequently coexist into two close feature vectors. In contrast, there are rare opportunities to travel from u_2 to u_3 , that is, u_2 will rarely coexist with u_3 , and finally the SGNS model maps two nodes that rarely coexist into two irrelevant feature vectors.

Biased Neighborhoods Sampling Strategy. Prior studies have found the equivalence between word-context and node-neighborhood and transplanted the SGNS model to the network embedding. The daily corpus can only represent the common word feature, likewise, the truncated random walk can only preserve the basic and general network feature. We want to get more information that benefits weakly similar users. For example, a student may face the following scenarios on the group buying platform: he may co-purchase with his classmates, which is very intuitive because they are robustly similar; he may also co-purchase with buyers of a safety helmet because they have a consistent need for helmets; he might even co-purchase with buyers of a rucksack, however, there are incongruities between their shopping behavior. In reality, the last scenario is very common. There is no aligned preference between co-purchasers, just an intersection under a large category (e.g., outdoor activities).

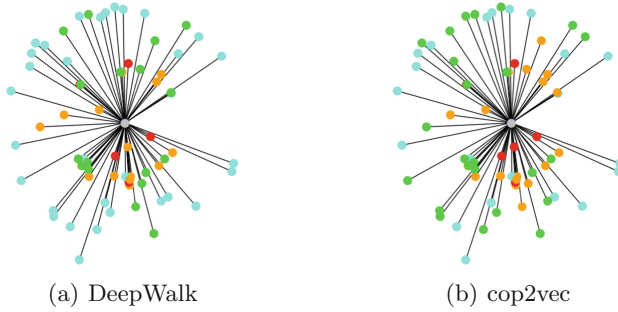


Fig. 4. The number of times co-occurrence of co-purchaser in one co-purchase on the TaoBao (●: initiator, ●: co-occurrence > 100, ●: co-occurrence > 50, ●: co-occurrence > 10, ●: co-occurrence ≤ 10). (Color figure online)

Building on the above observations, we design a flexible neighborhood sampling strategy which allows us to perceive the weakly related nodes effectively and sensitively. We achieve this by developing a flexible bias walk procedure that can explore farther neighborhoods with co-purchase tendencies. For example, a bias walk that just traversed edge (t, u) and now resides at node u . The walk now needs to decide on the next step so it evaluates the transition probabilities π_{ux} on edges (u, x) leading from u . We set the unnormalized transition probability to $\pi_{ux} = \alpha_{pkl}(t, u, x) \cdot w_{ux}$, where

$$\alpha_{pkl}(t, u, x) = \begin{cases} p & \text{if } t == x \\ k \cdot \text{sim}(t, x) & \text{if } t \in I \text{ and } x \in I \\ \frac{lw}{1+|w_t-w_x|} & \text{if } x \in \Gamma(t) \\ 1 & \text{otherwise} \end{cases} \quad (9)$$

In the equation, p , k , and l are the preset biased parameters that control the tendency of truncated walks. w_t is the purchase edge associated with t . U and I are the users set and the items set. $\text{sim}(t, x)$ denotes the approximate index between item nodes t and x . We simply set the approximate index to $\text{sim}(t, x) = (\Gamma(i) \cap \Gamma(j)) / (\Gamma(i) \cup \Gamma(j))$, although we can calculate a more accurate approximate index using side information attached to products.

Parameter p controls the likelihood of immediately revisiting a node in the walk. If we set a value greater than 1, it would lead the walk to explore the nodes that have already visited, and this would keep the walk “local” close to the starting node [19]. Setting it to a low value ensure the walk spreads out at a faster rate and avoids “bigram” redundancy in node sequences.

Parameter k is the key to ensure that the initiator node was able to perceive the weakly similar co-purchaser. Setting k to a high value, the walking strategy encourages the walk to diffuse along the chains (the red line in Fig. 3a) that are composed of related goods. These chains are like backbones in the network, by approaching the chains, the paths generated by walks makes more meaningful

when the walk is moving far away. That is, the farther co-purchasers attached to the chain will more likely coexist with the initiator.

Going back to Fig. 3a, Buyer u_2 bought a product i_2 in online shopping, consider a random walk that just traversed edge (i_2, u_2) and now resides at node u_2 . There are several alternative nodes (i_3, u_1, i_2, i_1) on the next step. At this point, we could observe that i_3 (safety helmet) has a high similarity with i_2 (bicycle) because buyers of the two commodities are almost overlapping. The similarity between two items is amplified by the biased parameter k , and then propagated to the biased factor, and the transition probability is adjusted to a larger value. That is to say, the walk has the high possibility to choose i_3 on the next step, and the walking path is like a backbone of the interaction network. Finally, the purchasers of i_3 , such as u_3 , will appear in the walking path and form a co-occurrence with u_1 and u_2 . SGNS model will capture the phenomenon of co-occurrence and map it to the embedding space.

Parameter l allows us to adjust the stay rate of the walk. If two buyers have a consistent preference for one item or two items get a consistent rating by one buyer, the item or buyer have a higher value of the stay. The higher the numeric of the parameter, the larger the influence of the stay rate, and vice versa.

By adjusting the biased parameters, the biased strategy of walking can flexibly explore the neighborhoods of nodes in interaction networks. In particular, the parameters allow our walk procedure to generate more meaningful co-occurrence paths for the co-purchaser recommendation. A toy example is shown in Fig. 4, the weakly similar co-purchasers (cyan nodes in Fig. 4a) get a higher number of times co-occurrence. As discussed in the formula 6, the weakly similar co-purchasers will gain better embedding vectors because they have higher relevance to the initiator. In addition, the biased walk is a smooth strategy and does not damage the original network information. That is, the original structure of networks and the adaptability to weakly similar co-purchasers both can be taken into account.

4 Experiments

In this section, we conduct various experiments to demonstrate the effectiveness of our proposed methods. First, we describe three real-world datasets on online shopping and visualize the embedding of a small number of purchasers. Secondly, we evaluate the methods by the top- k purchaser recommendation task. Finally, we report the co-purchaser detection experimental results on multiple online shopping datasets and present the influence of biased parameters.

4.1 Datasets

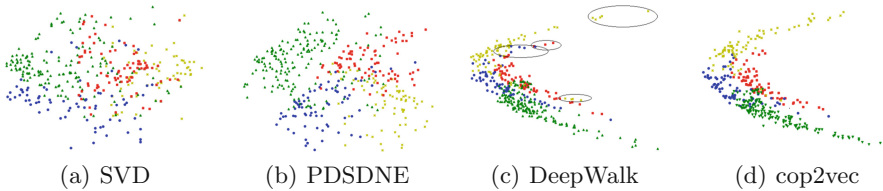
We design experiments on three widely adopted online shopping datasets, including Epinions, Amazon Electrol and TaoBao IJCAI16. Note, Amazon and TaoBao are processed into 5-core subsets, which all users and items have at least five records. Additionally, to enhance the diversity of truncated walk, we add a trust edge between two buyers when they have multiple co-purchase. Table 1 shows some statistics about datasets.

Table 1. Statistics of datasets

	Buyer nodes	Product nodes	Purchase edges	Trust edges
Epinions	40163	61273	664823	269649
Amazon	158694	61848	1612208	574868
TaoBao	35295	17617	250489	71244

4.2 Visualization of the Embeddings

In this part, we visualize the embeddings of buyers learned by PDSNE and cop2vec. We compared two classic embedding methods like SVD and DeepWalk. The results are shown in Fig. 5 where the buyers of the same item are highlighted with the same color. While the PDSNE can be effective for the 2D embedding, it can also present a sparse form. Network embedding methods based on the truncated walk has a natural advantage in dealing with this problem, DeepWalk can map purchasers of the same item more closely. On that basis, cop2vec can further compact these nodes that are mapped to remote locations due to the weak similarity.

**Fig. 5.** Visualization of purchaser (Color figure online)

4.3 Top- k Purchaser Recommendation

Although purchaser recommendation is not common on many e-commerce platforms, it is a critical part of group buying because we need to decide whom to recommend products to. To split the test set, We randomly selected 20% items from the TaoBao dataset and removed their 50% purchaser. After the model training, we choose Top- k close purchasers for the item in the embedding space, which are considered to be the most likely buyers to purchase the item. In order to comprehensively evaluate the effectiveness of the recommendation, we not only employ two state-of-the-art embedding models as baselines, including LINE [16] and DeepWalk [15]. But also fully compared the two proposed methods.

For a fair comparison, we use a 128 dimensions vector to denote a node in all methods. In LINE, as suggested in [16], the representation is directly concatenated by first-order (dimension 64) and second-order (dimension 64). In addition, we still use the same parameters for the truncated walk. The number

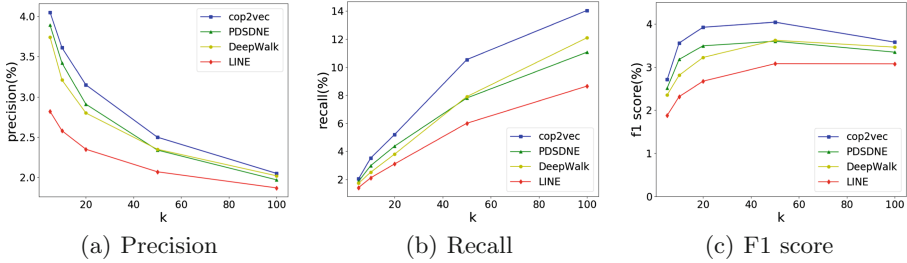


Fig. 6. The performance evaluation of Top- k purchaser recommendation.

of walks per node is 50, and the walk length is 30. The context window is 8, and the size of negative samples is 5. In PDSNE, the structure of two-layer encoder are 1000 and 128, and this is also the case with the decoder. In cop2vec, the biased parameters are tuned to be optimal.

As shown in Fig. 6, the walk-based network embedding method (DeepWalk and cop2vec) outperforms the proximity-based method LINE. When k is taken as 50, the performance of cop2vec is better. Compared with PDSNE, precision is improved by 6%. Compared with LINE, the precision is improved by 23%, which is partly due to the first-order and the second-order not well coordinated in LINE. In terms of recall, cop2vec was significantly higher than the contrast methods, which is increased by 41% compared to DeepWalk and 67% higher than LINE. This shows that the bias walk strategy can effectively perceive the purchasers who are weakly associated with items.

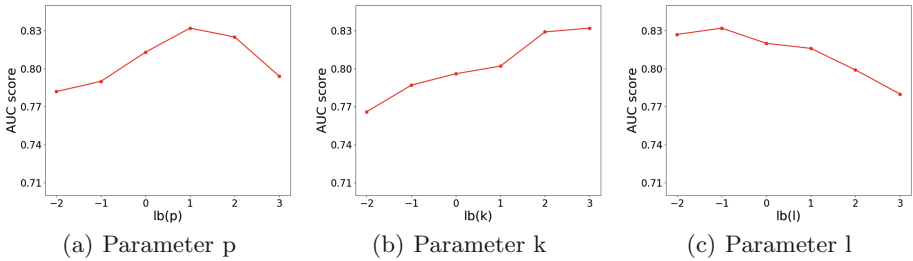
4.4 Co-purchaser Detection

In this section, we evaluate our proposed method on the co-purchaser recommendation task. Given a purchase initiator and his order for a certain item, we want to select the possible co-purchaser candidates. Note that current group buying platforms encourage buyers to sign in using social accounts, that is, we can give priority to recommending co-purchaser from a group of social accounts, rather than recommending co-purchaser from the whole buyers.

We choose 20% of the items from datasets and remove their n purchaser as a true buyer set. Additionally, we add $(n-1)/2$ unpurchased users as a false buyer set for each selected item. Where n is 50% of an item's total number of buyers. Select two buyers from the true buyers set, one as the initiator and one as the co-purchaser to form a positive sample, and finally generate $n(n-1)/2$ positive samples. Select a buyer from true buyers set and false buyers set respectively, one as the initiator and one as the co-purchaser to form a negative sample, and finally generate $n(n-1)/2$ negative samples. We use AUC (Area Under Curve) score to evaluate co-purchase intentions of positive and negative samples, where the co-purchase intention can be represented by the Hadamard product of the embedding vectors.

Table 2. AUC (Area Under Curve) scores for co-purchaser prediction

	SVD	CN	LINE	SDNE	DeepWalk	node2vec	PDSNE	cop2vec
Epinions	0.789	0.708	0.765	0.744	0.802	0.806	0.815	0.832
Amazon	0.620	0.549	0.546	0.589	0.656	0.655	0.582	0.687
TaoBao	0.703	0.645	0.668	0.681	0.683	0.696	0.717	0.764

**Fig. 7.** Parameter sensitivity

We conduct experiments on three different scale datasets and compare them with two traditional methods including SVD (Singular Value Decomposition), CN (Common Neighbors), and not just network embedding methods. The performances on three datasets are summarized as Table 2. We observe that cop2vec is consistently better than all the comparison methods.

On Epinions dataset, the performance of co-purchaser recommendations is the best, and we attribute this to a large number of real trust edges on the dataset. The AUC score of PDSNE is 9% higher than SDNE, 6% higher than LINE, and 2% lower than cop2vec. On Amazon dataset, the walk-based network embedding method is significantly higher than other types of methods, and the worst-performing DeepWalk is also 20% better than the proximity-based embedding method LINE. We can see that the performance of cop2vec gain is more significant on Taobao dataset, the AUC score is 7% higher than PDSNE, 11% higher than DeepWalk, and 19% higher than Common Neighbors.

4.5 Parameter Sensitivity

We investigate the parameter sensitivity in this section. Specifically, we mainly evaluate how the different choices of biased parameters affect the results of the co-purchaser recommendation. We report AUC score on the Epinions in Fig. 7. Intuitively, we can see that the performance raises when the value of parameter p increase, as shown in [19], a high p ensures that the walk does not go too far from the start node. We also observe that performance tends to saturate once the biased parameter k reaches around 8. Interestingly, we keep the parameter l at a small figure and get a good performance. This experiment suggests that

we don't need to pay too much attention to the "closed-loop" structure in the co-purchaser recommendation task.

5 Conclusions

As an emerging online shopping form, group buying has been restricted by the co-purchaser recommendation problem. Both the handcrafted invitation and the classic collaborative filtering do not solve the problem well. In this paper, we present network embedding based methods to address the co-purchaser recommendation challenge. To cope with the problem that traditional algorithms are desensitized to the weakly similar nodes, we propose two novel co-purchaser recommendation method, namely PDSNE and cop2vec, particularly the latter, which effectively perceive weakly similar nodes and maintain the original network information. Experiments on real-world datasets verify the effectiveness of our proposed approaches. For future work, incorporating side information such as stores, product categories, and attributes of buyers constitutes a heterogeneous network with more diversity of the bias walk, which may further improve the co-purchaser recommendation performance.

Acknowledgment. This work was supported by the National Natural Science Foundation of China (Grant No. 61572335, 61572336, 61902270), and the Major Program of Natural Science Foundation, Educational Commission of Jiangsu Province, China (Grant No. 19KJA610002), and the Natural Science Foundation, Educational Commission of Jiangsu Province, China (Grant No. 19KJB520052, 19KJB520050), and Collaborative Innovation Center of Novel Software Technology and Industrialization, Jiangsu, China.

References

1. Liu, H., Wang, W., Liu, D., Wang, H., Du, N.: HappyGo: a field trial of local group buying. In: ACM 2012 Conference on Computer Supported Cooperative Work, pp. 505–508. ACM, Seattle (2012)
2. Prospectus of Pinduoduo (A famous group buying platform in China). <https://sec.gov/Archives/edgar/data/1737806/000104746918004833/a2235994zf-1.htm>. Accessed 20 Jan 2019
3. Qin, D., Zhou, X., Chen, L.: Dynamic Connection-based Social Group Recommendation. *IEEE Trans. Knowl. Data Eng.* (2018)
4. Yue, S., Martha, L., Alan, H.: Collaborative filtering beyond the user-item matrix: a survey of the state of the art and future challenges. *ACM Comput. Surv.* **47**(1), 1–45 (2014)
5. Sun, Y., Han, J., Yan, X., Yu, P.S., Wu, T.: PathSim: meta path-based top-k similarity search in heterogeneous information networks. *Proc. VLDB Endowment* **4**(11), 992–1003 (2011)
6. Gustavo A.P., Hector G.C., Francisco J.C., Lucia R.: Recommending intra-institutional scientific collaboration through coauthorship network visualization. In: Proceedings of the 2013 Workshop on Computational Scientometrics: Theory & Applications, pp. 7–12. ACM, San Francisco (2013)

7. Chen, X.: Study on cooperator recommendation of virtual collaborative community. *J. Softw.* **8**(11), 2908–2916 (2013)
8. Chen, T., Sun, Y.: Task-guided and path-augmented heterogeneous network embedding for author identification. In: *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pp. 295–304. ACM, Cambridge (2017)
9. Guo, Z., Li, H.: Link prediction of actor cooperation relationship in heterogeneous information network. *Comput. Eng.* **43**(1), 219–225 (2017)
10. Cui, P., Wang, X., Pei, J., Zhu, W.: A survey on network embedding. *IEEE Trans. Knowl. Data Eng.* **31**(5), 833–852 (2019)
11. McLaughlin, M.R., Herlocker, J.L.: A collaborative filtering algorithm and evaluation metric that accurately model the user experience. In: *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 329–336. ACM, Sheffield (2004)
12. Víctor, M., Fernando, B., Juan-Carlos, C.: A survey of link prediction in complex networks. *ACM Comput. Surv.* **49**(4), 1–33 (2016)
13. Wang, J., Huang, P., Zhao, H., Zhang, Z., Zhao, B., Lee, D.L.: Billion-scale commodity embedding for E-commerce recommendation in Alibaba. In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery*, pp. 839–848 (2018)
14. Wen, Y., Guo, L., Chen, Z., Ma, J.: Network embedding based recommendation method in social networks. In: *Companion of the Web Conference 2018*, pp. 11–12 (2018)
15. Perozzi, B., Al-Rfou, R., Skiena, S.: DeepWalk: online learning of social representations. In: *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 701–710. ACM, New York (2014)
16. Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., Mei, Q.: LINE: large-scale information network embedding. In: *Proceedings of the 24th International Conference on World Wide Web*, Florence, Italy, pp. 1067–1077 (2015)
17. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: *Advances in Neural Information Processing Systems*, pp. 3111–3119 (2013)
18. Perozzi, B., Kulkarni, V., Chen, H., Skiena, S.: Don’t walk, skip!: Online learning of multi-scale network embeddings. In: *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, New York, USA, pp. 258–265 (2017)
19. Grover, A., Leskovec, J.: node2vec: scalable feature learning for networks. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 855–864. ACM, San Francisco (2016)
20. Wang, D., Cui, P., Zhu, W.: Structural deep network embedding. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1225–1234. ACM, San Francisco (2016)
21. Cao, S., Lu, W., Xu, Q.: Deep neural networks for learning graph representations. In: *Thirtieth AAAI Conference on Artificial Intelligence*, Arizona, February, pp. 1145–1152 (2016)