Co-purchaser Recommendation for Online Group Buying

Jihong Chen¹ · Wei Chen¹ · Jinjing Huang¹ · Jinhua Fang¹ · Zhixu Li¹ · An Liu¹ · Lei Zhao¹

Received: 11 March 2020 / Revised: 12 July 2020 / Accepted: 28 July 2020 / Published online: 9 August 2020
© The Author(s) 2020

Abstract

Online group buying is a burgeoning business model of Internet shopping, in which people with the same merchandise interests form a group and co-purchase goods with favorable prices. The buyer who launches the co-purchase is called the initiator, and other buyers are called the co-purchasers. Although recommending co-purchasers for a target buyer (co-purchase initiator) on the group buying is an interesting problem, existing studies have paid few 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 weak 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 weak 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 high recommendation performance. In addition, considering that co-purchase may last longer, the total recommendation result can be generated in multiple stages and adjust the current recommendation list based on the feedback from the recommendation of previous stages. It is a trick for all co-purchaser recommendation methods to make the total result better.

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

1 Introduction

On online group buying, buyers round up some like-minded people to purchase the same products, which can leverage a large number of people's collective bargaining power and achieve group discounts [1, 2]. In recent years, benefit from the advanced electronic payment technology and convenient express service, we have witnessed the prosperity of it in some online shopping services (e.g., TaoBao,¹ Groupon,² and PDD).³

In real applications, the co-purchase usually includes the following steps: Firstly, merchants promise to offer products or services with a discount on the condition that a certain number of customers would make the purchase. As shown in Fig. 1a, there are multiple buyers in a transaction, which

is also the essential feature of the co-purchase; then, an initiator manually invites friends, followers, and like-minded people to participate in the purchase; finally, co-purchasers accept the invitation and benefit from the lower price which is unavailable to the individual buyer [3].

Finding the right co-purchasers is a key step in the co-purchase process. 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 [3], 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 [4–7].

We consider using a recommendation system to recommend co-purchasers. As shown in Table 1, we can see

✉ Lei Zhao
zhaol@suda.edu.cn

¹ School of Computer Science and Technology, Soochow University, No.1 Shizi Street, Suzhou 215006, Jiangsu, China

¹ <https://www.taobao.com/>.

² <https://www.groupon.com/>.

³ <https://www.pinduoduo.com/>.

Fig. 1 A toy example of co-purchase: **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

Table 1 Difference between co-purchaser recommendation and other recommendation

Recommendation(rec)	For whom	Rec object	Application
Item rec	Individual	Item	News rec [8] Movie rec [20]
Group rec	Group	Item	Gyms music rec [13] Teaching material rec [11]
Collaborator rec	Individual	Individual	Co-author rec [14] Co-star rec [19]
Co-purchaser rec	(Individual+Item)	Individual	Co-purchaser rec [21]

that co-purchaser recommendation is totally different from traditional recommendation task. The existing recommendation methods are also not suitable for co-purchase scenarios.

Recommending items for a user is a hot topic, and previous researchers have made a lot of great contributions [8–10]. However, group buying is a distinctive online shopping pattern, where choosing a product is just the beginning of group buying. We need to pay more attention to who is the appropriate co-purchaser for this deal.

The purpose of group recommendation is to recommend acceptable items to a group [11, 12], such as gyms music recommendation [13], teaching material recommendation [11]. They all have a premise that there is an existing stable group, such as all members of a club, all students in a class. But on group buying, as mentioned above, the group is gradually gathered by the initiators, rather than existing before recommendation.

Collaborator recommendation is more close to real co-purchase situation than group recommendation. A co-purchase transaction may be regarded 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 [14–17] on collaborator recommendation systems as well as their real-world applications, such as co-author recommendation in the academic social network [14, 16], developer recommendation in the

open-source community [15, 18], and co-star recommendation in the film industry [19].

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 [14, 22, 23]. It is also a classic idea in many recommendation algorithms, such as the typical user-based collaborative filtering approach distinguishing the target users interests and preferences by aggregating the highest similar users [8]. 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 weak 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 weak similar users is not easy. To tackle the problem, two embedding strategies are proposed, which capture weak similar co-purchasers from different perspectives. In the first one, we propose a multi-layered learning architecture with PathSim [22] diffusion, namely PathSim diffused structural deep network embedding (PDSNE), which connects weak similar users by PathSim and directly imposes a penalty to the mapping error of the weak similar users. Obviously, it is a forthright strategy that is beneficial to the weak 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 weak 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 weak similar to the initiator.

In this paper, we refer to this approach but add more delicate recommendation processing. Compared with [21], the specific extension work may summarize as two parts.

In the first one, we compare four classic recommendation tasks, including item recommendation, group recommendation, collaborator recommendation, and co-purchaser recommendation. According to the comparison, we can see that the co-purchaser recommendation is a challenging new recommendation. In the second one, we improve the existing recommendation processing. Specifically, the co-purchaser recommendation model has multiple rounds of recommendation and adjusts the current recommendation list based on the feedback from the recommendation of previous stages.

Generally, 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 PSDNE and cop2vec, two efficient co-purchaser recommendation methods, which effectively perceive weak similar co-purchasers. In addition, we improve the original co-purchaser recommendation strategy, which makes the recommendation model more flexible.
- Through extensive experiments, we demonstrate 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 (PSDNE and cop2vec) with the details of how to capture the weak 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

The similarity search is a basic operation in collaborative filtering, and it can be directly used as a simple strategy to find a collaborator [22]. When the input is in the form of a scoring matrix, the common similarity search approach including cosine similarity and Pearson's correlation coefficient [24]. In the network analysis task [25], 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. [22] 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 [16, 26, 27], 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 [26]. Network embedding aims at learning low-dimensional vectors for the vertices of a network [27–29], such that the proximities among the original network are preserved in the low-dimensional space [30].

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 [31]. Specifically, skip-gram with negative sampling (SGNS), also known as word2vec, set new records in various NLP tasks. Inspired by it, DeepWalk [27] 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 [26, 32]. On this basis, WALKLETS [33] and node2vec [29] further extend DeepWalk by using high-order proximities and bias walk. LINE [28] is a 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 [34].

Fig. 2 Co-purchaser recommendation

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.

3.1.1 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.

3.1.2 Co-purchaser Recommendation

As shown in Fig. 2, co-purchaser recommendation aimed at recommending people who may participate in the co-purchase to the initiator when he selects a product and initiates a purchase order. Network embedding is to learn a low-dimensional vector for each node. A dense vector $R \in \mathbb{R}^K$ is used to represent a node in the network, and K is far less than $|V|$. In the group buying, $R = \{U, P\}$, let U be the vector set of all buyer nodes and let P be the vector set of all possible product nodes. 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:

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

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.

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 [22] as follows:

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

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. 3, 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 , so 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. 3, 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 [34, 35]. An autoencoder performs two actions, i.e., the encoder and decoder. The encoder consists of multiple nonlinear functions $f(\cdot) = f_{\theta_k}(\cdot \cdot f_{\theta_1}(\cdot))$ that map the input data to the representation space. The decoder also consists of multiple functions $g(\cdot) = g_{\hat{\theta}_1}(\cdot \cdot g_{\hat{\theta}_k}(\cdot))$ mapping the representations in representation space to reconstruction space. Let us assume that $f_{\theta_1}(x) = \sigma(W_1 x + b_1)$ and $g_{\hat{\theta}_1}(x) = \sigma(\hat{W}_1 x + \hat{b}_1)$, where

Fig. 3 The PathSim diffused network (the initiator links to the weakly similar users by PathSim)

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:

$$L_1 = \sum_{i,j} p_{i,j} \|f(s_i) - f(s_j)\|_2^2 \quad (4)$$

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

$$L = L_1 + \alpha L_n \quad (5)$$

As shown in previous works [34], 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 [27] 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:

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

where c is the context window size. Applying negative sampling [27], P is defined as:

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

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, and it usually refers to the window in paths, not just the one-hop neighbors in networks.) and get similar embedding [27, 29]. For example

Fig. 4 Overview of co-occurrence model

(see Fig. 4a), 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 weak similar co-purchasers 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.

To address the weak 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.

3.3.1 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 allows 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.

3.3.2 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).

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 [29]. Setting it to a low value ensures 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. 4a) 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. 4a, 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 has a higher value of the stay. The higher the numeric of the parameter, the larger the influence of the stay rate, and vice versa.

Fig. 5 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)

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. 5, and the weakly similar co-purchasers (cyan nodes in Fig. 5a) get a higher number of times co-occurrence. As discussed in 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.

3.4 Phased Co-purchaser Recommendation

In online group buying, the initiator launches a co-purchase order and invites other buyers (from the co-purchaser recommendation list). For some time after that, such as 24 h [3], these buyers can choose to accept or reject the invitation of the initiator.

Co-purchase may last longer, but previous studies only recommended co-purchasers once when the initiator launches the co-purchase order [21]; therefore, the overall recommendation results are rigid and inflexible. Also, the buyers in the recommendation list may fail to satisfy the co-purchase requirements. To globally control the flexibility of recommendations, we designed a phased co-purchaser recommendation strategy. It can be applied to all co-purchaser recommendation methods and make the total result better. Specifically, as shown in Fig. 6, the co-purchaser recommendation list is constantly updated by the recommendation methods for a period of time, from the beginning of the initiator's shopping submission to the end of the last confirmation of co-purchase status.

Fig. 6 Differences between the two strategies

$$f(G) = \frac{1}{|G|} \sum_{i=1}^{|G|} \mathbf{u}_i \quad (10)$$

According to the phased recommendation strategy, the initiator first invites other buyers to participate in a co-purchase transaction. After some buyers accept the invitation of co-purchase, they form a temporary group G with the initiator. The recommendation model will use the average function in Eq. 10 to aggregate the vector representation of this group members and then adjust the recommendation list for the temporary group and the selected products. Finally, the recommendation results are different in different co-purchase states. With the phased co-purchaser recommendation strategy, the recommendation model is more flexible and the recommendation performance is better.

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 trusted edge between two buyers when they have multiple co-purchase. Table 2 shows some statistics about datasets.

4.2 Visualization of the Embeddings

In this part, we visualize the embeddings of buyers learned by PSDNE and cop2vec. We compared two classic

Table 2 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

Fig. 7 Visualization of purchaser

Fig. 8 The performance evaluation of Top- k purchaser recommendation

embedding methods like SVD and DeepWalk (DW). The results are shown in Fig. 7 where the buyers of the same item were 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 have 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.

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 [28] and DeepWalk [27], 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 [28], 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 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 is 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. 8, 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 the term 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

Table 3 Area under curve (AUC) scores for co-purchaser prediction

	SVD	CN	LINE	SDNE	DW	node2vec	PDSDNE	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

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.

We conduct experiments on three different scale datasets and compare them with two traditional methods including singular value decomposition (SVD), common neighbors (CN), and classic network embedding methods. The performances on three datasets are summarized as Table 3. 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 PDSDNE 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, and the AUC score is 7% higher than PDSDNE, 11% higher than DeepWalk, and 19% higher than common neighbors.

4.5 Gain of N -Phased Recommendation

In this section, we demonstrate the gain obtained by using the phased recommendation strategy in a variety of co-purchaser recommendation algorithms. Given a co-purchase transaction, we divide the whole transaction into N stages

according to the time. In the phased recommendation task, we want to select the possible co-purchaser candidates in each stage. To achieve the goal, the recommendation model needs to adjust the current recommendation list based on the feedback of recommendation results from the previous stage. In order to simulate this behavior, we choose the true positives of the previous stage to join the current temporary group and then generate the recommendation list of the current phase for the temporary group and the selected products.

We use the Epinions dataset for the experiment, because there are more buyers for each product in the data set, and even after the division, there are still enough buyers in each stage to verify the recommendation performance. We choose 20% of the items from datasets as test dataset and remove their all purchaser except the initiator. For each stage of each test item, we will select the users who participated in the transaction in the next stage as a true buyer set, and the size of the true buyer set in each stage is t . Additionally, we add t unpurchased users as a false buyer set for each stage. For the validity of the experiment, we guarantee that the value of t is greater than 1 and less than n , n is the buyer removed from test item, and the total true buyer set of all stages is also n . At each phase, the recommendation model first aggregates the vector representation of this group members and then select the group and a buyer from true buyers set and generate k positive samples and then select the group and a buyer from false buyers set, and generate k negative samples. In each recommendation phase, we use area under curve (AUC) score to evaluate co-purchase intentions of positive and negative samples.

The gains of using the phased co-purchaser recommendation strategy are summarized as Table 4, and we observe that all co-purchaser recommendation methods can benefit from it. Using the basic recommendation strategy, that is, without partition, the recommendation performance is still relatively high, because we have changed the test cases, which only include n co-purchase samples between the initiator and other co-purchasers, rather than $n(n - 1)/2$ co-purchase samples between all co-purchases. The former is obviously easier to determine. We can see that the gain of DeepWalk and cop2vec is higher. In addition, the performance of cop2vec is consistently better than all the comparison methods, which shows that cop2vec method is also efficient in the phased co-purchaser recommendation.

Fig. 9 Parameter sensitivity

Table 4 Area under curve (AUC) scores for phased co-purchaser prediction

	Phase	PDSNE	DW	node2vec	cop2vec
$N = 1$	Mean	0.831	0.821	0.824	0.855
$N = 3$	1	0.836	0.841	0.829	0.863
	2	0.829	0.876	0.866	0.901
	3	0.839	0.860	0.867	0.889
	Mean	0.834	0.859	0.860	0.884
	Gain	0.3%	4%	3%	3%
$N = 5$	1	0.833	0.840	0.845	0.861
	2	0.832	0.862	0.865	0.884
	3	0.845	0.850	0.869	0.874
	4	0.842	0.847	0.853	0.870
	5	0.849	0.851	0.853	0.871
	Mean	0.840	0.850	0.855	0.873
	Gain	1%	3%	3%	2%

Bold values indicate that three methods achieve a notable performance by using phased recommendation strategy

Italic values indicate the optimal comprehensive performance

4.6 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 the AUC score on Epinions in Fig. 9. Intuitively, we can see that the performance raises when the parameter p increases, as shown in [29], and 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. Setting too large parameter k will generate a favorable path for weakly co-purchasers, but the impact on the original network will gradually appear. Interestingly, we keep the parameter l at a small figure and get a good performance.

This experiment suggests that we do not 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 are not suit to solve the problem. 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 methods, 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. Considering that co-purchase may last longer, we improve the existing recommendation strategy, which makes the recommendation model more flexible. 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.

Funding Funding was provided by National Natural Science Foundation of China (Grant Nos. 61572335, 61572336, 61902270), Natural Science Foundation, Educational Commission of Jiangsu Province, China (Grant No. 19KJA610002), Natural Science Foundation, Educational Commission of Jiangsu Province, China (Grant Nos. 19KJB520052, 19KJB520050).

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are

included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Liu H, Wang W, Liu D, Wang H, Du N (2012) HappyGo: a field trial of local group buying. In: CSCW '12 Computer Supported Cooperative Work, Seattle, WA, USA, February 11–15
- Yi X, Ying W, Yutian Z, Qing C (2014) Understanding the impact of service reputation on the online group-buying behaviors. In: The Thirteenth Wuhan International Conference on E-Business, WHICEB 2014, Wuhan, China, May 31–June 1, 2014
- pinduoduo. Prospectus of Pinduoduo[EB/OL]. <https://sec.gov/Archives/edgar/data/1737806/000104746918004833/a2235994zf-1.htm>. Accessed 20 Jan 2019
- Panagiotis A, Todri V (2015) Personality-based recommendations: evidence from Amazon.com. In: Poster proceedings of the 9th ACM conference on recommender systems, RecSys 2015, Vienna, Austria, September 16, 2015
- Sklar M, Concepcion KJ (2014) Timely tip selection for four-square recommendations. In: Li Chen and Jalal Mahmud. Poster proceedings of the 8th ACM conference on recommender systems, RecSys 2014, Foster City, Silicon Valley, CA, USA, October 6–10, 2014
- Hu P, Du R, Hu Y, Li N (2019) Hybrid item-item recommendation via semi-parametric embedding. In: Proceedings of the twenty-eighth international joint conference on artificial intelligence, Macao, China, August 10–16, 2019
- Wang Q, Yin H, Wang H, Nguyen QVH, Huang Z, Cui L (2019) Enhancing collaborative filtering with generative augmentation. In: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining, KDD 2019, Anchorage, AK, USA, August 4–8, 2019
- Yue S, Larson M, Hanjalic A (2014) 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
- Ma J, Zhou C, Cui P, Yang H, Zhu W (2019) Learning disentangled representations for recommendation. In: Advances in neural information processing systems 32: annual conference on neural information processing systems 2019, NeurIPS 2019, Vancouver, BC, Canada
- Wang J, Huang P, Zhao H, Zhang Z, Zhao B, Lee DL (2018) Billion-scale commodity embedding for e-commerce recommendation in Alibaba, pp 839–848
- Kompan M, Bieliková M (2014) Group recommendations: survey and perspectives. *Comput Inform* 33(2):446–476
- Seko S, Yagi T, Motegi M, Muto S (2011) Group recommendation using feature space representing behavioral tendency and power balance among members. In: Proceedings of the 2011 ACM conference on recommender systems, RecSys 2011, Chicago, IL, USA
- Qin D, Zhou X, Chen L, Huang G, Zhang Y (2020) Dynamic connection-based social group recommendation. *IEEE Trans Knowl Data Eng* 32(3):453–467
- Parada GA, Ceballos HG, Cantu FJ, Rodriguez-Aceves L (2013) Recommending intra-institutional scientific collaboration through coauthorship network visualization. In: Proceedings of the 2013 workshop on computational scientometrics: theory and applications
- Chen X (2013) Study on cooperator recommendation of virtual collaborative community. *JSW* 8(11):2908–2916
- Chen T, Sun Y (2017) 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, Cambridge, February 6–10, 2017
- Maurya A, Telang R (2017) Bayesian multi-view models for member-job matching and personalized skill recommendations. In: 2017 IEEE international conference on big data, Boston, MA, USA, December 11–14, 2017
- Xiao M, Ma K, Liu A, Zhao H, Li Z, Zheng K, Zhou X (2020) SRA: secure reverse auction for task assignment in spatial crowdsourcing. *IEEE Trans Knowl Data Eng* 32(4):782–796
- Guo Z, Li H (2017) Link prediction of actor cooperation relationship in heterogeneous information network. *Comput Eng* 43(1):219–225
- Zhao L, Lu Z, Pan SJ, Yang Q (2016) Matrix factorization+ for movie recommendation. In: Proceedings of the twenty-fifth international joint conference on artificial intelligence, IJCAI New York, NY, USA, 9–15 July 2016, pp 3945–3951
- Chen J, Chen W, Huang J, Fang J, Li Z, Liu A, Zhao L (2019) Co-purchaser recommendation based on network embedding. In: Web information systems engineering—WISE 2019—20th international conference, Hong Kong, China, November 26–30, 2019, Proceedings
- Sun Y, Han J, Yan X, Yu PS, Wu T (2011) PathSim: meta path-based top-k similarity search in heterogeneous information networks. *VLDB Endow* 4(11):992–1003
- Colomo Palacios R, Fyhn PG, Soto-Acosta P, Edvardsen K (2018) Building collaboration between academia and local authorities: a case study in Norway. *IJTM* 78(1/2):133–146
- McLaughlin MR, Herlocker JL (2004) A collaborative filtering algorithm and evaluation metric that accurately model the user experience. In: SIGIR 2004: proceedings of the 27th annual international ACM SIGIR conference on research and development in information Retrieval, Sheffield, UK, July 25–29, 2004
- Cui P, Wang X, Pei J, Zhu W (2019) A survey on network embedding. *IEEE Trans. Knowl. Data Eng.* 31(5):833–852
- Wen Y, Guo L, Chen Z, Ma J (2018) Network embedding based recommendation method in social networks. In: Companion of the web conference 2018 on the web conference 2018, WWW 2018, Lyon, France, April 23–27, 2018
- Perozzi B, Al-Rfou R, Skiena S (2014) DeepWalk: online learning of social representations. In: The 20th ACM SIGKDD international conference on knowledge discovery and data mining, KDD '14, New York, NY, USA—August 24–27, 2014
- Tang J, Qu M, Wang M, Zhang M, Yan Y, Mei Q (2015) LINE: large-scale information network embedding. In: Proceedings of the 24th international conference on world wide web, WWW 2015, Florence, Italy, May 18–22, 2015
- Grover A, Leskovec J (2016) node2vec: scalable feature learning for networks. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, San Francisco, CA, USA, August 13–17, 2016
- Goyal P, Chhetri SR, Canedo A (2020) dyngraph2vec: capturing network dynamics using dynamic graph representation learning. *Knowl Based Syst* 187:104816
- Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J (2013) Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems 26: 27th annual conference on neural information processing systems 2013. Proceedings of a meeting held December 5–8, 2013, Lake Tahoe, Nevada, United States

32. Chang S, Han W, Tang J, Qi G-J, Aggarwal CC, Huang TS (2015) Heterogeneous network embedding via deep architectures. In: Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining, Sydney, NSW, Australia, August 10–13, 2015
33. Perozzi B, Kulkarni V, Chen H, Skiena S (2017) 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 2017, Sydney, Australia, July 31–August 03, 2017
34. Wang D, Cui P, Zhu W (2016) Structural deep network embedding. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, San Francisco, CA, USA, August 13–17, 2016
35. Zhang G, Liu Y, Jin X (2020) A survey of autoencoder-based recommender systems. *Front Comput Sci* 14(2):430–450