Pairwise structural role mining for user categorization in information cascades

Sarvenaz Choobdar, Pedro Ribeiro, Fernando Silva
CRACS and INESC-TEC
University of Porto, Portugal
Email: {sarvenaz,prebeiro,fds}@dcc.fc.up.pt

***Abstract*—The tendency of users to connect with peers of similar interests and social demography (homophily) is one of the sources of information for user behavior modeling and classification. However this is yet an open question for structural roles where nodes at similar structural position in the network play the same roles: are structurally equivalent nodes more prone to have connections between themselves? In this paper, we tackle this open question by studying the patterns of homophily for structural roles. We propose a new method named SR-Diffuse to simultaneously identify structural roles in a network and to model the role membership matrix of users. In this method, we integrate pairwise role dependency alongside with structural features of users for role mining. We show that pairwise role dependency is necessary to distinguish some structural roles but it is a misleading factor for some others. We design an optimization model to capture structural roles with the guidance of pairwise dependency, and devise an iterative algorithm to learn structural roles simultaneously from structural properties and social dependency of users. We examine the efficacy of our new method in a users classification problem for information cascades. We compare the predictability of discovered roles by our method against some baseline methods for predicting social classes of users in different information cascades in two social networks, Flickr and Digg. The experimental results suggest that our method can improve the quality of roles membership of users and can better represent the profile of users in the network, hence it is a better predictor for social classes of users in an information cascade.**

I. Introduction

The structure of a network is determined by its connections. By observing these links, one can derive a set of features that characterizes each node's structural position. This can in turn be used to help identify the structural role of each node. Structural roles in a social network can often be associated with various functional or organizational roles such as star-center, star-edge nodes, member of-cliques or bridges in different parts of the network. This type of structural role mining has applications in many domains. For example, in the case of online social networks, it is important to know users' position in the network in order to create personalized marketing campaign. Another example is viral marketing, where the structural role of users is essential in targeting the appropriate users in order to achieve maximum coverage of the network, to spread ideas such as ads or news. In fact, structural roles are gaining increased attention in the last years, and they are now used as a tool for tasks such as node classification [1], identity resolution [2], [3], exploratory network analysis [2] and anomaly detection [4].

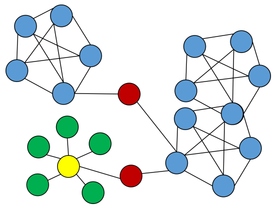


Fig. 1: Pairwise dependency across structural roles, different colors correspond to different structural roles; the pairwise role dependency exists in some structural roles such as member-of-clique (blue nodes) but it does not hold on some others such as member-of-star (green nodes).

Pairwise dependency suggests that nodes with a similar structural position may have a tendency to have connections between themselves. Figure 1 exemplifies that, with the blue nodes (member-of-clique) having connections to other blue nodes. However, this it not the case for all types of structural roles. For instance, the green nodes (member-of-star) have no connections to other green nodes, as their structural features do not give origin to pairwise connections. In this paper, one of our main goals is to incorporate pairwise dependency of different structural roles in role mining framework. For that, we first examine how actually the pairwise relations are across structural roles by running a pilot experiment on a real social network. We use the results for structural role modeling. We want to infer a role configuration L over the social network of users, using two assumptions: 1) *ego-role dependence:* the structural role of users is correlated to their ego properties X and users in the same role k have similar feature vectors; 2) *pairwise role dependence*: the role of a user is not independent from its neighbors' roles.

In the second part of this paper, we study the application of structural role mining for user classification in in the context of message communication. In our approach, we use the structural roles of users in the network instead of using the activity log of users to classify them to the social classes. In an information cascade, the capability of users in spreading information is of great interest. An important parameter for categorizing users in a cascade is their effect on the network which we measure as the consequence of users action. The time interval of involvement of users in the process is also important as the late adopters are not of interest for diffusion modeling and spreading the story. We define a new classification for users in an information cascade by two factors, detailed in section V-B.

Our main research contributions in this work are the following:

- We study the patterns of pairwise dependency for structural roles, showing that we can improve role discovery for some roles, while in others this does not happen. For example, highly connected users tend to connect to users at similar structural position (hubs attract hubs); however singular nodes connect with the hubs as well. Hence, to accurately infer a role, we cannot propagate role labels through all connections.
- We propose a new relational framework called *SR-diffusion* to jointly model pairwise dependence and structural positions of users. To the best of our knowledge this is the first study to incorporate pairwise relations in structural role mining.
- We design an algorithm to learn the SR-diffusion model in social networks, where the hidden variable role is inferred regarding the observed variables of ego properties of users and connections. We define a cost function to model the pairwise dependencies and structural similarities. This algorithm, iteratively infers the social roles of users based on structural similarities in the network and by propagating roles through connections.
- We show how information cascade modeling can benefit from role mining by predicting influential users in an information cascade solely from the structural role membership of users.

The remainder of the article is organized as follows. Section II reviews previous approaches and related problems. Section IV explains the problem statement and modeling and describes the proposed algorithm for SR-diffusion. Section V provides results about the experimental evaluation of our proposed method on social networks. Section VI gives the final comments on the obtained results and concludes the article.

II. Related works

A. Structural role mining

For a static network, role extraction is defined as the process of finding groups of nodes with similar properties. In other words, this is a clustering task where nodes are grouped, not based on their connectivity, but because they hold a similar position in the network. This has been studied by other researchers, where nodes with the most outstanding properties are detected as singular motifs using outliers detection methods [5]. Henderson et al. [2] found roles of nodes regarding their properties in their neighborhood by non-negative matrix factorization. In their method, the matrix of node-role is derived from matrix factorization of node-features and features-role matrices and the number of roles is determined by a Minimum Description Length (MDL) method [6].

Rossi et al. [7], used the methodology proposed by [2] for dynamic role extraction. They measure a set of features for nodes at each time snapshot. Then, by stacking all the node-by-feature matrices, they derive the matrix of feature-roles by factorizing the stacked node-by-feature matrix and iteratively generate the matrix of node-role for each time. Role discovery methods are essentially unsupervised.

However a more supervised approach for role discovery is presented by [1] where they used structural properties of users to infer their pre-defined social statuses of users. They proposed a probabilistic model to integrate users' social properties and network features for prediction of users roles. Danilevsky et al. [8] studies role discovery in hierarchical topical communities.

B. Social roles in information cascades

Information propagation in social networks has been widely studied for a number of years from different aspects. Several influence models have been proposed and studied, and the most popular ones are the linear threshold model (LT) and the independent cascade model (IC), by Kempe et al. [9]. These models study spread of influence through social networks, where the influence probabilities between users are predefined. Saito et al. [10] predict the influence probabilities in independent cascade models of propagation by maximum likelihood estimation and Goyal et al. [11] study the probabilities in the threshold model by counting the number of correlated social actions. They both consider the temporal nature of influence of users.

Other research works in this field, measure users' influence by using some structural models of influence like PageRank and in-degree centrality in the network [12], number of followers, mentions, retweets [13], [14] or the size of the information cascades [15]. Earlier studies of social influence and propagation, showed that the most influential bloggers were not necessarily the most active [16]. Temporal information has been used in modeling influence using the influence-passivity score [17]. Although the structure of the Flickr social network holds small-world properties, which in theory says a piece of information will spread quickly and widely through social links, photos on Flickr are spread with delay [18]. This study concludes that propagation is not only due to activity of users but also due to information availability at the time of users' activity. Zhou and Liu [19] integrated three sources of information to derive the influence group of users. They defined a new similarity matrix between users based on three sources of information including a social network of users, activity networks and influence networks.

All referred work use both the activity log of users and their social network to characterize the influence process. This contrasts with our proposal in which we only use topological properties of users to categorize their role in the influence spread.

III. Pairwise dependency and structural roles

In this section, we quantify the correlations between the structural roles. For the pilot experiment, we use a very basic role mining method on a network to extract a set of structural roles and examine the pairwise dependencies between roles. We use the static role mining method where the k-means algorithm is employed over the structural properties of nodes in the network, to group them into their respective roles. The measured structural roles are the same as the one explained in section IV-B.

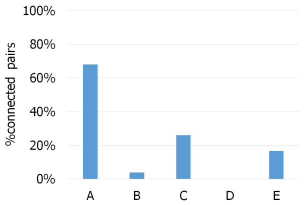


Fig. 2: Pairwise dependency across structural roles in Digg social network; The percentage of connected users varies significantly across roles; A: “cliquey”, B: ”2nd periphery”, C: ”periphery-cliquey”, D: ”periphery”, E: ”local-star”. For example, 65% of users in role “A” are connected, but users of role “D” never connect.

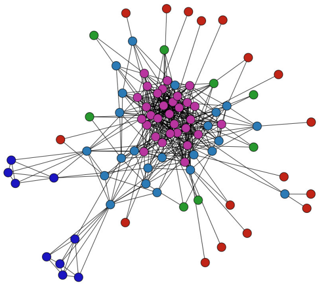


Fig. 3: A subgraph of Digg social network, including active users in one information propagation process; color-coded regarding the structural roles; A: “cliquey” (purple nodes), B: ”2nd periphery” (blue nodes), C: ”periphery-cliquey” (dark blue), D: ”periphery” (red nodes), E: ”local-star” (green nodes) . The percentage of connected users varies significantly across roles. For example, 65% of users in role “A” are connected, but users of role “D” never connect.

A. Data

For this experiment we use a dataset from Digg *Digg*[1] social network. Digg is a news aggregator in which users can submit links to interesting news stories and they can rate these stories by voting on them. Users also can designate other users as friends. More specifically, each user has a list of followers (fans who follow him) and a list of followees (friends whom he follows). All activities are visible to her fans, including all stories he submitted or voted for. We use the Digg data collected by Lerman and Ghosh [20] which contains the friendship network of users and all the posts submitted during one month, including the id, submitter id, voters for each post and the date of votes. This dataset includes 3,018,197 votes on 3,553 popular stories made by 139,409 users and the social network of active users (who have at least one vote) containing 71,367 users and 1,731,658 friendship links. We built our social network from active users and their connections, where active users are those who voted for at least one story.

B. Experiment results

We derived five structural roles in the network. Then we study the pairwise relations for different roles by counting the number of connected users in each role. Figure 2 depicts the results of this experiment for users in the social network of Digg. The percentage of connected users varies significantly across roles. For example, 65% of users in role “A” are connected, however users of role “D” are never connected. These two roles are depicted in Figure 3, with role “A” shown in orange and role “D” in yellow, corresponding to a subgraph of active users in an information cascade in a Digg social network including all of their connections to the rest of the network. We can clearly see that pairwise dependency is valid for role “A” but not for role “D”. Therefore, this dependency will be of great help in categorizing the users for some roles, but it can also be highly misleading for some other roles. In the next section we show how we can take advantage of pairwise dependency in role mining modeling.

IV. Pairwise structural role mining

In this section we explain our proposed method for role mining where both ego- and pairwise-role dependencies are considered following the framework of probabilistic graphical models [21]. Our approach aims to detect groups of users that have the same structural properties and are socially connected. The likelihood of the data is higher when users in the same group have the same structural properties, and it is also higher when users have interactions.

We model these two dependencies in the framework of role mining. We first introduce the variables in the problem that are utilized for defining the objective function of our model. The first variable x_i represents the ego features of user u_i and it is derived by measuring a set of structural properties such as degree centrality and clustering coefficient. We define the set of ego features to be utilized in our model in section IV-B . All features in this vector are normalized to the interval $[0, 1]$. The latent variable l_i shows the role label of user u_i and has value from 1 to K to indicate to which role the user belongs to. We quantify the pairwise dependency by variable λ_{kr} which measures the non-compatibility of roles k and r. Last, to represent the association between roles and ego features, we use an association variable μ_k for each role. Each dimension of this vector variable indicates the corresponding feature in the ego feature vector of x_i in the role k. Since we do not know which ego features are associated to role k, μ_k is an unknown vector and need to be learned.

For every two users u_i and u_j in the same role their ego feature vector x_i and x_j should be close on the dimensions designed by μ_k. Hence, by using a distance measure between ego feature vectors and association variable, we want to minimize:

$$\sum_{k=1}^{K} \sum_{u_i \in R_k} D(x_i, \mu_k) \tag{1}$$

where D is a distortion measure between users and R_k is the set of nodes with the label role k. Our model should also minimize the cost of pairwise role assignment to achieve a minimum role conflict between connected users:

$$\sum_{(u_i,u_j) \in E} \lambda_{kr} \mathbb{I}[(l_i = k, l_j = r)] \tag{2}$$

where λ_{kr} is the cost of non-compatibility of role k, r and $\mathbb{I}$ is the indicator function showing if the role labels of the

[1] http://www.isi.edu/ lerman/downloads/digg2009.html

connected users u_i, u_j are k and r. As discussed before pairwise role dependency is more important for some roles than for others. We tune $\Lambda = \{\lambda_{11}, ..., \lambda_{kk}\}$ in a way that it does not sacrifice the ego-role dependency for the sake of the pairwise dependency.

Our final objective function is derived from the linear combination of the two elements:

$$obj = \sum_{k=1}^{K} \sum_{u_i \in R_k} D(x_i, \mu_k) + \sum_{(u_i,u_j) \in E} \lambda_{kr} \mathbb{I}[(l_i = k, l_j = r)] \tag{3}$$

A. SR-Diffuse Algorithm

In this section we introduce our algorithm to find the values of unknown variables such that they minimize Eq. 3. We have three sets of unknown variables, the role label l_i of user u_i, the association vector f_k for each role and the pairwise dependency cost λ_{kr} between the two roles k and r. Since the association vector as well as the role labels for the users are unknown, minimizing Eq. 3 is an "incomplete-data problem", for which a popular solution method is Expectation Maximization (EM) [22]. In the following we describe a soft role assignment (SR-Diffuse) algorithm which iteratively updates each set of variables.

The algorithm starts with an initialization of the three sets of variables and then in the E-step, given the association vectors $F = \{f_1, ..., f_K\}$ and the pairwise dependency cost λ_{kr} for every pair of roles, every user is re-assigned to the roles that minimize her contribution to obj. In the M-step, the association vectors and the pairwise dependency cost are re-estimated from the role assignments $L = \{l_i, ..., l_N\}$ to minimize obj for the current assignment. Note that this corresponds to the generalized EM algorithm [22], where the objective function is reduced but not necessarily minimized in the M-step.

Algorithm 1 SR-Diffuse

```
procedure SR-DIFFUSE($G = (V, E), K, \sigma$)
    $X \leftarrow egoFeatures(G)$
    $L^0 \leftarrow initialize(X)$
    $\Lambda \leftarrow updateVariables(L^0)$        ▷ $\Lambda = \{\lambda_{11}, ..., \lambda_{KK}\}$

    while (not Converged) do
        $L^t \leftarrow roleAssignment(\{f_1, ..., f_K\}, \Lambda)$        ▷ E-step
        $\Lambda, \{f_1, ..., f_K\} \leftarrow updateVariables(L^t)$        ▷ M-step
        if $||L^t - L^{t-1}|| < \sigma$ then
            Converged $\leftarrow True$
        end if
    end while
    return $L^t$
end procedure
```

1) Initialization: To initialize the model, we applied the fuzzy k-means clustering algorithm [23] to the data set resulting in a partitioning of users into K clusters. We use this assignment to provide the values to the association vector, and compute the variables relative to that assignment. These variables form the starting point for EM, which is then run to convergence.

2) Role assignment (E-step): The assignments of users to roles are updated using the current estimates of the association vector and the pairwise dependency cost. In simple role assignment when pairwise interactions of users is not considered, and the E-step is a simple assignment of every user to the role representative that is nearest to it according to the distance function. In contrast, our model incorporates interaction between the users. As a result, computing the assignment of users to cluster representatives to minimize the objective function is computationally intractable in any non-trivial model [24]. We follow the iterated conditional modes (ICM) [25], [26] approach, which is a greedy strategy to sequentially update the role assignment of each user, keeping the assignments for the other users fixed. The algorithm performs role assignments in random order for all users. Each user u_i is assigned to the role label k that minimizes the user's contribution to the objective function. Optimal assignment for each user is the one that minimizes the distance between the users in the same role and maximizes the association between roles and ego features (first term of obj) with a minimal penalty for pairwise dependence assumption violations caused by this assignment (second term of obj). After all users are assigned, they are randomly re-ordered, and the assignment process is repeated. This process proceeds until no user changes its role assignment between two successive iterations. ICM is guaranteed to reduce obj or keep it unchanged (if obj is already at a local minimum) in the E-step [25]. Overall, the assignment of points to roles incorporates pairwise supervision by discouraging assumption violations proportionally to their severity, which guides the algorithm towards a desirable role configuration over the network.

3) Update variables (M-step): The M-step of the algorithm consists of two parts. First we discuss the update of the association vector f_k for users in role k when labels $L = \{l_i, ..., l_N\}$ for all users are fixed. The association variables $\{f_1, ..., f_k\}$ are re-estimated from users currently assigned to the roles to decrease the objective function obj in Eq. 3. Each role association calculated in the M-step of the EM algorithm is equivalent to the expectation value over the points in that cluster, which is essentially their arithmetic mean.

$$f_k = \frac{\sum_{x_i \in R_k} x_i}{|R_k|} \tag{4}$$

The second set of variables that we discuss is the pairwise dependency cost λ_{kr} for the roles k and r. The main intuition for this variable is that users of certain roles tend to connect to each other but some others do not. Hence for fixed association vectors $F = \{f_1, ..., f_K\}$ and role assignment $L = \{l_i, ..., l_N\}$, we estimate the pairwise dependency cost λ_{kr} as follows:

$$\lambda_{kr} = \frac{|u_i : l_i = k|.|u_i : l_i = r| * \alpha}{|(u_i, u_j) \in E : l_i = k, l_j = r|} \tag{5}$$

where the denominator measures the number of pairs of (k, r) in the network and it is normalized by the number of connections if these roles where always connected. The basic idea is that cost of having same role for connected nodes is higher if it is a rare case in the network.

To complete the model parameterization, we need to specify α, the variable used in Eq. 3 to represent the strength of the preference towards assigning connected users to the same role. We experimented with a range of values for α for both data sets, measuring both the number of connections in each role and the coherence of the clusters with respect to the structural properties. We evaluated the structural coherence of a role as the average distance between every pair of users that were assigned to the role. As expected, increasing α results in a larger number of connections among users in the same role. Our method results in roles configuration consistent with the pairwise role dependence assumption, while not sacrificing the structural properties quality. This parameter also helps to find appropriate number of roles for a network, we discuss this issue more in section IV-C.

B. *Ego features*

In this section we define the ego feature vector for the users. It is possible to use a different feature set for role mining such as local features [27] or recursive feature aggregation [28] We selected the structural properties of users that have been shown to be correlated to social classes of users [5], [27]:

- the normalized node degree (K): quantifies the linkage of node i; it is the degree of node i divided by the sum of all nodes' degree in the network.
- the normalized average degree (r): shows the intensity of connectivity in the neighborhood of node i; it is calculated by averaging over all degree of immediate neighbors of node i.
- the standard deviation of degree (cv): coefficient variation of the degrees of the immediate neighbors of a node characterizes the coherence of the connectivity; it is measured by the standard deviation of the degrees in the neighborhood of node i.
- the clustering coefficient (cc): quantifies the connectivity between neighbors; it is measured as the proportion of existing connections between neighbors of node i to the number of all possible links between them [29].
- the locality index (loc): characterizes the structure of neighbors' connectivity to the rest of the network; it is the ratio of links to the nodes outside of neighborhood to the number of links within the neighborhood to.
- the common neighbors (CN): measures the commitment of users to the neighborhood. This feature shows if neighborhood of a user has an overlap with its neighbors. It is the number of common neighbors between a user's direct connections.

  $$CN_i = \sum_{u_j \in N_{u_i}} \frac{\left|N_{u_i} \cap N_{u_j}\right|}{\left|N_{u_i} \cup N_{u_j}\right|} \tag{6}$$

  where N_{u_i} is the set of neighbors of user i.
- the eigenvector centrality ($eig-cntr$): ranks users regarding their importance in the network. This centrality measure acts similar to degree centrality however it gives higher score to the nodes which are themselves connected to high score nodes.

This feature vector has the advantage of measuring the connectivity of a node in its neighborhood structure and also it is fast to calculate. However any other feature set can also be used in our method.

C. *Determining number of structural roles*

The number of roles is one of the challenges in role mining. Our role mining method solves this issue by initializing the number of roles to a relatively large number ($n/2$) and when it stops the non-empty roles are the final roles. The final number of non-empty roles is determined by the value of α in equation 5. We study the effect of value of α by measuring the quality of roles in two terms: 1) isolation and 2) compactness. Isolation assesses how well roles are separated by calculating the distance between centers of roles and compactness assess the coherence of roles by measuring the distance between users in the same role, measured respectively by first and second part of equation 7. We calculate the quality score QS of discovered roles by:

$$QS = \underset{\forall r,k \in [1,K]^2}{min} \; dist(f_k, f_r) - \underset{\forall k \in [1:K]}{mean} \; \underset{x_i \in R_k}{max} \; dist(f_k, x_i) \tag{7}$$

The higher score shows higher quality for a role set as it shows roles are well separated by high value of isolation and have high coherency by low value for compactness component. We find the appropriate number of role K by varying value α as long as it improves the equation 7.

V. Experiments

We demonstrate the efficacy of our method through user classification in information cascades. For S cascades we label involved users in each cascade regarding the class label definition in section V-B. The classification task is to predict the labels of users in a cascade based on role membership matrix. We use logistic regression for this purpose. We compare the predictability of discovered roles by our method to three baseline methods. The first method evaluates the effect of pairwise role dependence assumption, the second evaluates the effect of ego properties of users on the roles and the third one compares the predicta bility of structural roles to ego properties.

A. *Data*

Throughout this section we will be using two different data sets, coming from two well known and established internet communities: *Digg*, as explained in section III-A and *Flickr*[2]. Flickr is a popular photo and video hosting website with a large community of users. We use data collected by Cha et al. [30], which includes a social friendship network of users and information propagation from one user to another. The associated mechanism is similar to Digg, but instead of URLs, photos are shared and voted. We sample 4000 photos from those which number of favorite marking is higher than 100. This sample contains a network with 914,400 users and 18,595,048 links. The social network includes all users who have marked the selected photos as favorites and all their connections in the original data.

Both datasets include a static social network with social relationships between users and a dynamic evolving network describing information propagation.

[2] http://socialnetworks.mpi-sws.org/datasets.html

B. Class definition in information cascade

In this section, we define a set of social classes for users in an information cascade. Different categorizations for active users in a cascade are defined in literature [31], [32], [33], [34], however all these definitions are one dimensional and only consider either time of action or influence of a user's action. In this paper, by inspiration from existing definitions, we define a new categorization of users based on two factors to capture both time and consequence of a user's action.

1) time of action: Borge-Holthoefer et al. [34] divide the lifetime of a cascade in to three phases based on the final size of th cascade: 1) slow growth: the time slot when the cascade size is less than 5% of final size; 2) explosive phase: when cascade size grow from 5 to 90% of final size; 3) saturation phase: when cascade size is above 90% of its final size
2) consequence of action: Baños et al. measure the effect of a user by multiplicative number of the given user. The multiplicative number of user u_i is the quotient of the number of listeners reached one time step after u_i showed activity, $l(t+\tau)$, and the number of nearest listeners of u_i, i.e., those who instantaneously received its message, $l(t)$ (which is given by the number of followers of u_i that are involved in the cascade). Thus, the ratio $l(t+\tau)/l$ measures the multiplicative capacity of a user: $\delta_l = l(t+\tau)/l > 1$ indicates that a user has been able to increase the number of listeners who received the message beyond her immediate followers.

Figure. 4 shows the distribution of users in defined time phased and in a multiplicative number of users for a cascade. The blue distribution shows influential users, and the red one belongs to those that were not able to affect network beyond their 1-hop neighborhood. As we can see not all the early adopters in the "slow growth" phase are influential enough to affect users for further voting. The red distribution has higher frequency but lower influence mean comparing to the blue one. Regarding the aforementioned factors and Figure. 4, we categorize users that are active in a cascade into six groups or classes: 1) initiators: active users in slow growth phase with $\delta_l > 1$. 2) promoters: active users in explosive phase with $\delta_l > 1$. 3) early adopters: active users in slow growth phase with $\delta_l < 1$. 4) common users: active users in explosive phase with $\delta_l < 1$. 5) late adopters: active users in saturation phase with $\delta_l > 1$. 6) passives: active users in saturation phase with $\delta_l < 1$.

These six groups constitute our class labels and we call them social classes to differentiate them from structural roles that we have from the structural role mining framework. In this paper we investigate how social classes correlate to the structural roles and we demonstrate the predictability of our role mining method through predicting social classes in a cascade.

C. Experiment configuration and results

The first step of structural role minins is to determine the suitable number of roles in a network regarding the method explained in section IV-C. Figure 6 shows the quality of discovered roles for different values of α. As we can see, the worst quality belong to the setting with $\alpha = 0$ which is basically when pairwise dependency has zero effect in the role mining. This demonstrates that our method improves role mining results by incorporating the pairwise dependency. Our method specially improves the quality of role sets in the network when the roles are very similar and relying only on structural features is not enough for learning the roles. For example, Figure 5 shows a subgraph of Digg social network, where nodes are positioned regarding their first and second principle component of the matrix of nodes ego features. In this figure nodes with similar ego features are located closely. In the network (a) nodes are color coded regarding their role from k-means algorithm while nodes in the network (b) are color coded by their roles discovered from SR-Diffuse. As we can see for the same number of roles, different role configuration is derived by two methods. SR-Diffuse puts connected nodes that are close regarding ego vectors in the same role while k-means can not; green and dark blue nodes in network (a) are placed in the same group (dark red) by SR-Diffuse and cyan nodes in network (a) are divided into two roles (dark and light blue) in network (b).

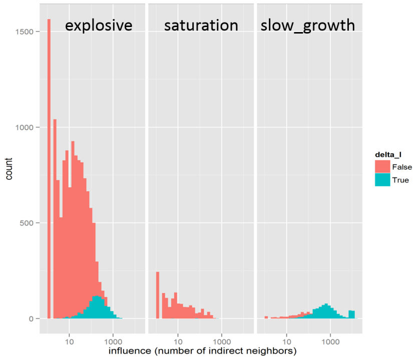
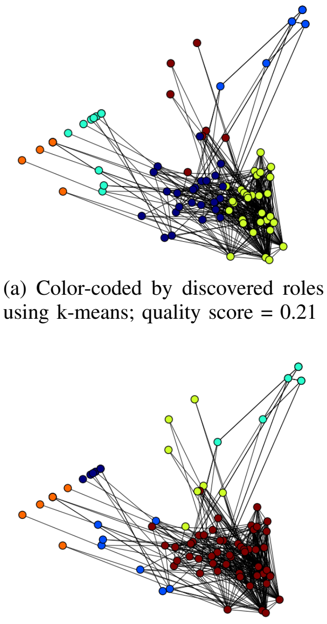


Fig. 4: The influence distribution of active users at different phase of a cascade lifetime in Digg social network; blue distribution belongs to the users with $l(t+\tau)/l > 1$, who could influence the network beyond their immediate neighbors. In the "slow growth" phase these users have larger degree (number of immediate neighbors) .

From Figure 6, we can see that SR-Diffuse finds the best roles configuration on Digg social network when $\alpha = 56$ and on Flickr network $\alpha = 72$. With this configuration the number of roles that SR-Diffuse found on these networks are respectively 8 and 11. We use the same number of roles for the baseline methods. Next we explain how discovered roles can predict social classes of users in an information cascade.

We select S disjoint cascades that do not have any active users in common. We measure the ego properties of the N active users in the cascades and then learn structural roles of users by a role mining method (our method (SR-Diffuse), pair-means and c-means). This gives us the role membership matrix of users which we use as predictor to build the classifier using logistic regression. In order to be able to evaluate the predictability and generality of discovered roles we use 50% of users to build the role membership matrix and put the rest aside as the test set. We use the role membership matrix of users in the train set to build the classifier and evaluation result



(a) Color-coded by discovered roles using k-means; quality score = 0.21

(b) Color-coded by discovered roles using SR-Diffuse; quality score = 0.43

Fig. 5: A subgraph of Digg social network, including active users in one cascade; color-coded regarding the structural roles. Nodes are positioned regarding their first and second principle component of the ego features matrix of nodes. Nodes with similar ego feature vectors are located closely; SR-Diffuse puts connected nodes that are similar regarding ego feature vectors in the same role better than the way k-means does.

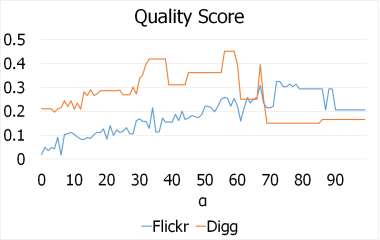


Fig. 6: The quality of discovered role set by SR-Diffuse for different values of α (pairwise dependency parameter).

is derived from the classification of users in test set.

Table I demonstrates the evaluation results of our method and the baseline methods. We measure the performance of each method in terms of F-score for the predicted roles in the test set. F-score is the harmonic mean of precision and recall which are respectively equal to $\frac{|p\cap r|}{|p|}$ and $\frac{|p\cap r|}{|r|}$ for the predicted role p with reference to actual role r. We can see that SR-Diffuse can better predict roles of users in an information cascade

We compare the predictability of discovered roles by our method to three baseline methods, the first one evaluates the effect of pairwise role dependence assumption, the second one evaluate the effect of ego properties of users on the roles and the third one compare the predictability of structural roles to ego properties:

- pair-means: this method uses pairwise role dependence for cluster assignments, but does not perform distance learning; This method applies majority votes on the labels of neighbors of a user to infer her role. This method is initialized by clustering a subset of users using the fuzzy k-means algorithm and then the role labels for the rest of the users are assigned by majority votes.
- c-means: the fuzzy k-means algorithm over structural properties are utilized for role discovery.
- ego-feat: this method uses the ego features of users as described in section IV-B to make the prediction model.

TABLE I: Performance of SR-Diffuse in classifying users in information cascade in comparison to baseline methods.

Digg	F1	precision	recall
SR-Diffuse	0.50	0.67	0.41
c-means	0.44	0.52	0.39
pair-means	0.46	0.67	0.36
ego-feat	0.29	0.76	0.18

Flickr	F1	precision	recall
SR-Diffuse	0.46	0.58	0.39
c-means	0.44	0.61	0.35
pair-means	0.40	0.57	0.31
ego-feat	0.33	0.62	0.23

Table I reports the classification performance of discovered roles by our method comparing to the baseline methods in terms of F1, precision and recall. We can see that worst performance (lowest F1) belongs to the ego-feat method, its precision is the highest though. This shows that the ego features are good indicators for social classes of users in information cascade. The recall is low, it suggests that ego features are not enough for predicting roles. The classifier performs better when the structural role membership is used as the predictor instead of ego features. As we can see from the Table I, we have better classification performance for all three role mining methods (SR-Diffuse, c-means and pair-means) over the ego-feat method. Overall, the role configuration discovered by SR-Diffuse is a better classifier for social classes in information cascade as we have the best classification results from the classifier trained over this role membership matrix. This suggests that combination of ego features and pairwise dependencies can improve the quality of role mining results and better detect existing structural roles in the network.

VI. Conclusions and Discussion

In this paper, we studied patterns of homophily for structural roles in a network. We showed how structural compatibility varies across different structural roles and devise a new method to take advantage of this property for discovering some of structural roles and avoiding misclassification for the others. We proposed a novel relational *structural role mining* method to find roles configuration over a network. Our method is capable of finding roles membership of users regarding their structural features and pairwise dependencies. It iteratively assigns users into structural roles in a way that the derived roles set has the most coherency in terms of including most similar users and has the least non-compatibility of roles in the neighborhood of each user. This algorithm automatically finds the appropriate number of roles in a network by controlling the pairwise dependency parameter.

The experimental results, using two real social network data sets, show that the proposed model greatly outperforms a number of baseline models and is able to effectively infer roles of users in an information cascade scenario. In our experiment, we have shown that the predictability of discovered roles by our method is higher than baselines.

In this study, we also explore how influential users modeling in information cascade can benefit from structural role mining in a network. We defined a set of class labels for active users in information propagation events on a social network based on their influence and time of action and then used structural roles membership of users to predict their class labels in an information cascade. We showed that discovered structural roles by our method are better predictors for social classes of users in a cascade comparing to a set of baseline methods.

One of the emerging challenges in structural role mining is spotting roles of a users relative to the community they belong to. As a future work we intend to extend our method in a way to be capable of finding roles of users in each community they are part of.

References

[1] Y. Zhao, G. Wang, P. S. Yu, S. Liu, and S. Zhang, "Inferring social roles and statuses in social networks," in *Proc. ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, 2013.

[2] K. Henderson, B. Gallagher, T. Eliassi-Rad, H. Tong, S. Basu, L. Akoglu, D. Koutra, C. Faloutsos, L. Li, Y. Matsubara *et al.*, "Rolx: Structural role extraction & mining in large graphs," in *Proc. ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, Beiging, China, 2012.

[3] S. Gilpin, T. Eliassi-Rad, and I. N. Davidson, "Guided learning for role discovery (glrd): framework, algorithms, and applications," in *Proc. ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, 2013.

[4] R. Rossi, J. Neville, B. Gallagher, and K. Henderson, "Modeling dynamic behavior in large evolving graphs," in *Proc. ACM Int. Conf. on Web Search and Data Mining*, 2013.

[5] L. Costa, F. Rodrigues, C. Hilgetag, and M. Kaiser, "Beyond the average: detecting global singular nodes from local features in complex networks," *Europhysics Letters (EPL)*, vol. 87, no. 1, 2009.

[6] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, no. 5, 1978.

[7] R. Rossi, B. Gallagher, J. Neville, and K. Henderson, "Role-dynamics: fast mining of large dynamic networks," in *Proc. ACM Int. Conf. on World Wide Web*, Lyon, France, 2012.

[8] M. Danilevsky, C. Wang, N. Desai, and J. Han, "Entity role discovery in hierarchical topical communities," in *Proc. ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, 2013.

[9] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proc. ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, 2003.

[10] K. Saito, R. Nakano, and M. Kimura, "Prediction of information diffusion probabilities for independent cascade model," in *Knowledge-Based Intelligent Information and Engineering Systems*. Springer, 2008.

[11] A. Goyal, F. Bonchi, and L. V. Lakshmanan, "Learning influence probabilities in social networks," in *Proc. ACM Int. Conf. on Web Search and Data Mining*, 2010.

[12] H. Kwak, C. Lee, H. Park, and S. Moon, "What is twitter, a social network or a news media?" in *Proc. ACM Int. Conf. on World Wide Web*, 2010.

[13] C. Lee, H. Kwak, H. Park, and S. Moon, "Finding influentials based on the temporal order of information adoption in twitter," in *Proc. ACM Int. Conf. on World Wide Web*, 2010.

[14] M. Cha, H. Haddadi, F. Benevenuto, and P. K. Gummadi, "Measuring user influence in twitter: The million follower fallacy." in *Proc. AAAI Int. Conf. on Weblogs and Social Media*, vol. 10, 2010.

[15] E. Bakshy, J. M. Hofman, W. A. Mason, and D. J. Watts, "Everyone's an influencer: quantifying influence on twitter," in *Proc. ACM Int. Conf. on Web Search and Data Mining*, 2011.

[16] N. Agarwal, H. Liu, L. Tang, and P. S. Yu, "Identifying the influential bloggers in a community," in *Proc. ACM Int. Conf. on Web Search and Data Mining*, 2008.

[17] D. M. Romero, W. Galuba, S. Asur, and B. A. Huberman, "Influence and passivity in social media," in *Proc. of the ECML/PKDD*, 2011.

[18] M. Cha, F. Benevenuto, Y.-Y. Ahn, and K. P. Gummadi, "Delayed information cascades in flickr: Measurement, analysis, and modeling," *Computer Networks*, vol. 56, no. 3, 2012.

[19] Y. Zhou and L. Liu, "Social influence based clustering of heterogeneous information networks," in *Proc. ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, 2013.

[20] K. Lerman, R. Ghosh, and T. Surachawala, "Social contagion: An empirical study of information spread on digg and twitter follower graphs," *arXiv preprint arXiv:1202.3162*, 2012.

[21] J. Pearl, *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann, 1988.

[22] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, 1977.

[23] Y. Yang, "Information theory, inference, and learning algorithms," *Journal of the American Statistical Association*, vol. 100, no. 472, 2005.

[24] E. Segal, H. Wang, and D. Koller, "Discovering molecular pathways from protein interaction and gene expression data," *Bioinformatics*, vol. 19, 2003.

[25] J. Besag, "On the statistical analysis of dirty pictures," *Journal of the Royal Statistical Society. Series B (Methodological)*, 1986.

[26] Y. Zhang, J. M. Brady, and S. Smith, "Hidden markov random field model for segmentation of brain mr image," in *Medical Imaging 2000*. International Society for Optics and Photonics, 2000.

[27] S. Choobdar, F. Silva, P. Ribeiro, and S. Parthasarathy, "Dynamic inference of social roles in information cascades," *Data Mining and Knowledge Discovery*, to appear.

[28] K. Henderson, B. Gallagher, L. Li, L. Akoglu, T. Eliassi-Rad, H. Tong, and C. Faloutsos, "It's who you know: graph mining using recursive structural features," in *Proc. ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, 2011.

[29] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, Jun. 1998.

[30] M. Cha, A. Mislove, and K. P. Gummadi, "A Measurement-driven Analysis of Information Propagation in the Flickr Social Network," in *Proc. ACM Int. Conf. on World Wide Web*, 2009.

[31] C. Buntain and J. Golbeck, "Identifying social roles in reddit using network structure," in *Proc. ACM Int. Conf. on World Wide Web*, vol. Companion, 2014.

[32] R. A. Baños, J. Borge-Holthoefer, and Y. Moreno, "The role of hidden influentials in the diffusion of online information cascades," *EPJ Data Science*, vol. 2, 2013.

[33] E. M. Rogers, *Diffusion of innovations*. Simon and Schuster, 2010.

[34] J. Borge-Holthoefer, A. Rivero, and Y. Moreno, "Locating privileged spreaders on an online social network," *Physical review E*, vol. 85, no. 6, 2012.