Role-Oriented Network Embedding Method Based on Local Structural Feature and Commonality

Liang Ge(✉), Xiaofeng Ye, Yixuan Jia, and Qinhong Li

College of Computer Science, Chongqing University, Chongqing 400030, China
{geliang,yexiaofeng,jiayixuan,qinhong}@cqu.edu.cn

Abstract. Role-oriented network embedding has become a powerful technique for solving real-world problems, because it can capture the structures of nodes and make node embeddings better reflect the functions or behaviors of entities in the network. At present, various role-oriented network embedding methods have been proposed. However, most of the methods ignore degree distribution and the commonalities among local structures, resulting in insufficient information of node embeddings, and some methods that preserve commonality always have high time complexity. To address the above challenges, we propose a novel model ReVaC from two aspects of extracting higher-quality local structural features and strengthening the commonalities among local structures in node embeddings. In detail, the degree distribution from node's 1-hop egonet is incorporated into the extraction process of local structural features to improve traditional ReFeX firstly, and the Variational Auto-Encoder is used to map those features to the local structural embedding space. Then, in the embedding space, we cluster nodes to model the commonalities among local structures. Finally, local structural embeddings and commonalities are fused to get node embeddings. We conduct extensive comparative experiments on real-word networks, and results show that ReVaC has better performance than other state-of-the-art approaches and adapts well to network scale.

Keywords: Role-oriented network embedding · Degree distribution · Local structural commonality

1 Introduction

In almost all networks, nodes tend to have one or more functions that largely determine their structural identity in the system. When considering the problem of learning a representation that captures the structural identity of nodes in a network, even if two nodes do not share the connection or are even far apart, but they have similar functions or occupy similar positions (similar structures) in the

S. Khanna et al. (Eds.): PRICAI 2022, LNCS 13630, pp. 32–44, 2022.
https://doi.org/10.1007/978-3-031-20865-2_3

network, then their potential representations should be close to each other. Obviously, community-oriented embedding methods can't handle such case, those are all based on the connection of nodes. The structure-based network embedding methods emerge as the times require. They encode local structural features into vectors to capture structural similarity and obtain role-oriented embedding representations, so are also called role-oriented embedding methods.

At present, role-oriented network embedding has gradually become one of the most important research hotspots. It still faces the following challenges: (1)the key to learning role-oriented network embedding is to extract high-quality local structural features, degree distribution is a very good local structural feature. According to the paper [1], it is shown that degree distribution, generalized to include the distribution in its k-hop neighborhood, may indeed be a good indicator of the structural position or role in the network. It excels in the evaluation of automorphic and regular equivalence, and achieves superior results in various experiments on real networks. However, this useful structural information is not well utilized. Few methods take advantage of it, and operations based on it are limited. For example, struc2vec [2] and XNetMF [3] determine their similarity only by computing distances between k-order degree sequences or degree vectors. (2)Some approaches only preserve the local structural features of nodes as much as possible into embeddings, ignoring the commonalities among local structures. The commonality of a class of similar local structures can be regarded as the feature a structural role, and ignoring it means losing part of the characteristics of the role, which is unfriendly to role-oriented embedding. However, the approach that retains commonalities needs to calculate structural similarities, which often has high time complexity and is not suitable for large datasets.

In order to meet the above challenges, we propose our model ReVaC from two aspects of extracting higher-quality local structural features and strengthening the commonalities among local structures. The model consists of three parts: local structural feature extraction, commonality modeling and fusion encoding. Firstly, we improve the traditional ReFeX [4] by incorporating degree distributions from nodes' 1-hop egonet into their initial features and leveraging iterative process to obtain local structural features. At the same time, to avoid the over-fitting caused by the high-order iteration, the Variational Auto-Encoder is regarded as the operator to map those features to a local structural embedding space. Secondly, in the embedding space, we model the commonalities among local structures. That is, nodes with similar local structures are captured by clustering, and then the common feature of nodes in the same cluster is modeled as the commonality of such similar nodes. Finally, to enrich the structural information of nodes and make the structural roles and embedding distances of nodes highly correlated, local structural embeddings and commonalities are fused to obtain node embeddings. Our main contributions can be summarized as follows:

- The traditional ReFeX is improved to incorporate degree distributions of nodes into their initial features, and iterate new initial features to obtain higher-quality local structural features.

- We propose to explicitly model the commonalities among local structures by clustering in the local structural embedding space, and fuse them with local structure embeddings. That enriches the information of node embeddings and improves the expressive ability of node embeddings.
- We conduct several extensive experiments on real-world networks via our model ReVaC, and compare the results with other state-of-the-art methods. The results demonstrate the superiority of our model, and prove that our model scales well with network size.

2 Related Work

Obtaining high-quality structural features is the key to learning role-oriented embeddings, and current methods are diverse. ReFeX [4] (Recursive Feature eXtraction) extracts local and egonet features and aggregates the features of neighbors recursively. As an effective method to capture structural features, ReFeX is widely used in many other role-oriented embedding methods. For example, RolX [5] and GLRD [6] leverage the structural features extracted by ReFeX and uses matrix factorization to get low-dimension node representation. In RESD [7] and RDAA [8], ReFeX is proposed to extract structural features and utilizes encoder framework to map the network to the latent space. The key idea of GAS [9] is to extract some key structural features based on ReFeX as the guidance information to train the model. There are other methods directly based on degree features, such as SIR-GN [10] encodes the degree of each node as a one-hot vector. RoINE [11] also concatenates the degree of a node and the sum of its immediate neighbors' degree as structural feature. Besides, HONE [12] generates the high-order network embeddings by decompose a set of motif-based matrices. GraphWave [13] is based on heat-wavelet diffusion patterns, it treats graph diffusion kernels as probability distributions over networks. DRNE [14] is designed to leverage a layer-normalized LSTM to process the sequences of nodes' degree-based direct neighbors, which are treated as structural features. Gralsp [15] captures structural patterns by generating w anonymous random walks starting from one node with length L.

Structural properties also are contained in pair-wise similarities, and there are various ways to calculate them. XNetMF [3] take advantages of Singular Value Decomposition to encode the similarities based on the K-order degree vector and attribute vector as embeddings. Struc2vec [2] constructs a hierarchy of complete graphs by transforming similarities of the k-order ordered degree sequences to weights of edges. SEGK [16] decomposes the similarity matrix computed by graph kernels. REACT [17] aims to obtain node representations by applying non-negative matrix database on RoleSim [18] similarity matrix and adjacency matrix, respectively. Struc2gauss [19] generates structural contexts based on the RoleSim similarity matrix, and learns node representations in the space of Gaussian distributions. SPaE [20] computes cosine similarity between the standardized Graphlet Degree Vectors of nodes, and generates role-based embeddings via

Laplacian eigenmaps method. Role2vec [21] also recommends Motif-based features, such as mapping nodes to multiple disjoint roles based on Graphlet degree vectors.

To sum up the above, we have to admit that degree, degree-based sequences, and related degree vectors are recognized good indicators of structures, and most methods directly or indirectly utilize them. However, it is obvious that only a few methods involve degree distribution in the process of constructing feature matrix, and their operations on it are too limited.

3 Methodology

In this section, we declare the concepts used in this paper, and then introduce our framework ReVaC in detail.

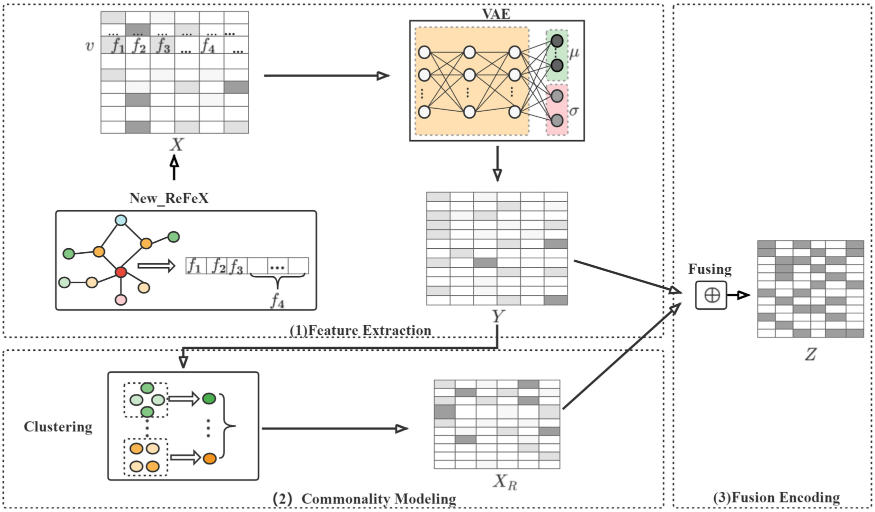


Fig. 1. An overview of the proposed ReVaC: (1) extract local structural features X with improved ReFeX and map them to local structural embedding space Y by the VAE, (2) explicitly model the commonalities among local structures in the space by clustering and obtain the common features X_R, (3) fuse local structure embeddings Y and common features X_R to obtain the final node embeddings Z.

3.1 Notions

A network is represented by an undirected unweighted graph $G = (V, E)$, where $V = \{v_1, ..., v_n\}$ is the set of nodes and E is the set of edges. For each node $v \in V$, the set of node v's neighbors is defined as $N(v)$, d(v) denotes the degree of node v. The 1-hop egonet of node v is defined as $G_v = \{V(g_v), E(g_v)\}$, where

$V(g_v) = \{v\} \bigcup \{u \in V|(v,u) \in E\}$ and $E(g_v)$ is the set of edges in the 1-hop egonet of v. D_v represents the degree distribution from the 1-hop egonet of node v. The extracted features of nodes are denoted as $X \in R^{n\times f}$, where f is the dimension of features. $Y \in R^{n\times d}$ are the local structural embeddings. $X_R \in R^{k\times d}$ are the common features of similar local structures, where k is the number of node structure roles. $Z \in R^{n\times d}$ represent the final node embeddings, where d is the dimension of embedding.

3.2 Model

In this section, we introduce the proposed method ReVaC. The framework is shown in Fig. 1. The ReVaC consists of three parts: (1) local structural feature extraction, (2) commonality modeling, (3) fusion encoding.

Feature Extraction. ReFeX is an effective method to capture structural features, which firstly computes initial features and then aggregates neighbors' initial features with sum- and mean-aggregator recursively to get local structural features. The initial feature of the traditional ReFeX is mostly composed of node degree and egonet-based information. It is still hard to be applied to discover node roles and complex tasks as simple statistical is preserved. Recent research [2] shows that node's degree distribution may indeed be a good indicator of the structural position or role of the node in the network and degree distributions of higher-order local neighborhoods are also sufficiently expressive structural descriptors. The 1-hop egonet of node is the smallest local structure, degree distribution from it can intuitively reflect connection pattern. So, we draw on the experience of ReFeX to incorporate degree distribution features from 1-hop egonet of nodes into initial features to help to enrich local domain information and participate in recursive process to capture higher-quality local structural features.

For each node v, the initial features extracted in this paper are as follows:

- (1) The degree of v: $f_1 = |N(v)|$
- (2) The sum of node's degree in the 1-hop egonet of v: $f_2 = \sum_{u\in V(g_v)} d(u)$
- (3) The number of edges from the 1-hop egonet of v: $f_3 = |E(g_v)|$
- (4) The degree distribution in the 1-hop egonet of v: $f_4 = D_v$

We represent D_v with the degree distribution of node v's 1-hop neighbors. To prevent one high-degree node from inflating the length of these vectors and make their entries more robust, we bin nodes together into $b = [log(d_{max} + 1)]$ logarithmically scaled buckets, where d_{max} is the maximum degree in the original graph. So that the i-th item of the degree distribution vector D_v of node v is the number of nodes that satisfy $[log(d(u) + 1)] = i, u \in V(g_v)$. Namely, $D_v^i = |\{u \in V(g_v)|[log(d(u) + 1)] = i\}|$, where the dimension of D_v is b. And then, based on the initial features, an iterative process similar to traditional ReFeX is used to obtain local structural features, so we call the above process as New_ReFeX, and the features are denoted as: $X = New\_ReFeX(f_1, f_2, f_3, f_4)$.

At the same time, we also noticed that with the increase of the number of iterations, each node can meet fairly high-order neighbors, which may cause over-fitting. For this reason, the Variational Auto-Encoder is acted as the operator to encode local structural features to get more compact and robust local structural embeddings. Specifically, the structural feature reconstruction loss of VAE is defined as follows:

$$L_{VAE} = ||X - \hat{X}||_2^2 = \sum_{v=1}^{n} ||X_v - \hat{X}_v||_2^2 \tag{1}$$

At the same time, to prevent over-fitting and better preserve key local structural information, referring to RESD, we add a degree-based regularizer, as follows:

$$L_{reg} = \sum_{v=1}^{n} (log(d(v) + 1) - MLP(Y_v))^2 \tag{2}$$

where $MLP(\cdot)$ is also a Multi-Layer perceptron model with rectified linear unit activation $ReLU(\cdot)$.

We train our model ReVaC by jointly minimizing the loss of feature reconstruction and degree-regularized constraint as follows:

$$L = L_{VAE} + \alpha L_{reg} \tag{3}$$

where α is the weight of the degree-based regularize. Through the above process, we get the local structural embeddings, we define: $Y = VAE(X)$.

Commonality Modeling. When looking at the similarity from a global perspective, the local structural information extracted are preserved as much as possible in the local structural embeddings in the above process, while commonalities among local structures are ignored. The similar local structures always correspond to the same structural role, so the commonality can be regardes as a common feature of a class of similar local structures, and can also be regarded as the feature of a structural role. If commonalities preserved in node embeddings, there is no doubt that we have captured different structural roles to which the nodes belongs, which helps to make nodes with similar local structures have similar embeddings. However, most of the current role-oriented methods ignore commonalities, and some methods that preserve commonalities tend to have high time and space complexity.

To solve the above problem, we propose to explicitly model commonalities among local structures. We were inspired by two things: (1) Clustering algorithms can cluster nodes with similar local structures. So we find the nodes with the same structural role by clustering in the embedding space. (2) Clusters describe the main structural roles that exist in the local structural embedding space, we can model their commonalities according to the set of nodes in the cluster. So in the commonality modeling part of ReVaC, the details are as follows: we use K-Means clustering based on Euclidean distance in the local structural embedding space to make nodes with similar local structures have the same cluster label. The cluster label of the i cluster is denoted i, and all nodes in this cluster form the node set R_i. Because the centroid of a cluster is the mean of the local structural embeddings of all nodes in the cluster, it represents the common feature of the cluster to a certain extent. Thus, the centroid of a cluster can be modeled as the commonality among a class of similar local structures. That is for the node v, the label i of its cluster is obtained by K-means algorithm, and then the cluster center is modeled as a commonality with a similar local structures to v, and its feature is denoted as:

$$X_{R_i} = \frac{\sum_{u \in R_i} Y_u}{|\{u|u \in R_i\}|} \tag{4}$$

where Y_u is the local structural embedding of node u and R_i denotes the set of nodes of cluster i.

For the K-mean algorithm, since the degree of real network always follows the power-law distribution, so we set K as the logarithm of the maximum degree in the network, $K = [log(d_{max}+1)]$, that is, assuming the number of potential main structural roles in the network is K. Then, we finally get the features of all structural roles via modeling, which are defined as follows: $X_R = clustering(Y, K)$

Fusion Encoding. The key idea of our algorithm is to strengthen the structural role features of nodes on the basis of preserving local structural information, that is, to explicitly preserve commonalities among local structures in node embeddings. In detail, the modeled commonalities and the local structural embeddings of nodes are fused to get node embeddings. For node v, its node embedding is defined as follows:

$$Z_v = \beta * Y_v + \gamma * X_{R_i} \tag{5}$$

where Y_v is the local structural embedding of node v, and X_{R_i} is the common feature of similar local structures of the i-th cluster which v belongs. And β and γ are hyperparameters. We think the local structural embedding and common feature of node to be equally important, so both β and γ are set to 0.5. The above is the whole process of the algorithm.

3.3 Complexity Analysis

Given a network G, let n denote the number of nodes, e denote the number of edges, m denote the feature aggregation number of New_ReFeX, f be the dimension of extracted feature matrix X, d represent the dimension of local structural embedding Y. For the local structural feature extraction part, firstly, it takes $O(n+f \cdot m \cdot (e+nf))$ to iteratively capture the local structural features of nodes by improving the traditional ReFeX method, and then map the extracted features to the local structural embedding space through the VAE, which requires $O(nf^2d + nd^2)$. Therefore, the time complexity of this part is $O(n + f \cdot bin \cdot (e + nf) + nf^2d + nd^2)$. For the commonality modeling part, the time complexity of

K-Means clustering is $O(nkt)$, where k is the number of cluster centroids and t is the number of clustering iterations. At last, for the fusion encoding part, fusing local structural embeddings and commonalities to get node embeddings takes $O(n)$. To sum up, the whole computation of ReVaC is $O(2n + ktn + f^2 \cdot bin \cdot n + f \cdot bin \cdot e + f^2 dn + d^2 n)$. Since k, t, bin are always very small and $k < f < d \ll n$, our model has an advantage over other methods for large-scale networks, such as the complexity of struc2vec is $O(n^3)$.

Table 1. Detailed statistic of the datasets, including the number of nodes, edges, categories, and nodes in each category.

Dataset	Nodes	Edegs	Classes	Category0	Category1	Category2	Category3
Brazil	131	1074	4	32	32	32	35
Europe	399	5995	4	99	99	99	102
USA	1190	13599	4	297	297	297	299
Actor	7779	26733	4	1782	1787	1798	1797
Film	27312	122706	4	10101	2378	3725	11108

4 Experiments

In this section, to evaluate the effectiveness of our model, we select three tasks for the evaluation including (1)the visualization experiment by plotting the node representations in a 2-D space to observe the relationships between node embeddings and their roles, (2)the classification experiment based on the ground-truth labels of datasets by comparing the Micro-F1 and Macro-F1 scores, (3)the top-k similarity search experiment to see if nodes in the same role are mapped into close position in the embedding space.

4.1 Datasets

We conduct experiments on several real-world networks with unweighted undirected edges. The datasets we use are listed as follows and the statistics are shown in Table 1:

(1) Air-traffic networks [15]: there are three networks, consisting of American, Brazilian, and Europe air-traffic networks (Brazil, Europe, and USA for short). In these networks, nodes represent airports and edges represent the existed flights between airports.
(2) Actor co-occurrence network [22]: In Actor network, nodes represent actors and are labeled based on their influences which are measured via the number of words in their Wiki pages.
(3) English-language film network [23]: it is a film-director-actor-writer-network (Film for short). And edges denote whether two nodes appear in the same Wiki page.

4.2 Baseline

We evaluate the effectiveness of the ReVaC by comparing it with widely used role-oriented embedding algorithms. We choose eight state-of-the-art methods including struc2vec [2], ReFeX [4], RolX [5], RESD [7], RDAA [8], GraphWave [13], SEGK [16], role2vec [21]. In addition, the results of New_ReFeX on those datasets are also demonstrated in subsequent experiments.

4.3 Experiment Settings

All embedding methods using ReFeX set the number of feature aggregations to 3, the number of bins to 4, as does New_ReFeX. The number of hidden layers of the encoder and decoder are all set to 2. We apply Adam SGD optimizer with the learning rate of 0.001 and set the $L2$ regularization with weight of 0.001 to avoid over-fitting. In our later experiments, if not stated specifically, α is set to 0.3, β and γ are both set to 0.5. The dimension of node embedding is set to 128 for all methods, except ReFeX and New_ReFeX.

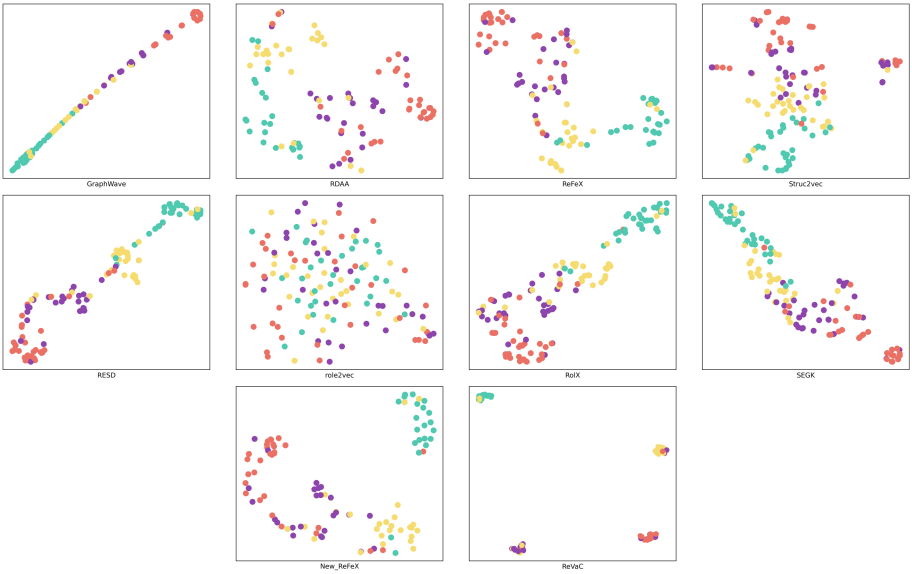


Fig. 2. Visualization of node representations on Brazil network in two-dimensional space. The label is mapped into color of point.

4.4 Visualization

In this section, we visualize the learned embeddings, which can directly reflect the performance of different methods. The Brazil network is selected, and we apply t-SNE to reduce the dimension of embeddings to 2 for visualization. Each node is represented as a point and the color indicates its role label. Ideally, points in the same color should be close together, and those in different colors should be farther away from each other. As shown in Fig. 2, we observe that role2vec cannot extract role information well as the points in different colors are mixed up. Graphwave may be over-fitting to one specific structure characteristic as the points are almost lined up. The other methods achieve that points in the same color are clustered in varying degrees, such as RDAA, RESD, RolX, SEGK, struc2vec. We note that the New_ReFeX extracts higher quality features than ReFeX as expected, because points in the same color are closer and points in different colors are further apart. Obviously, the ReVaC divides the points with different colors into different clusters, and the clusters in different colors are far apart.

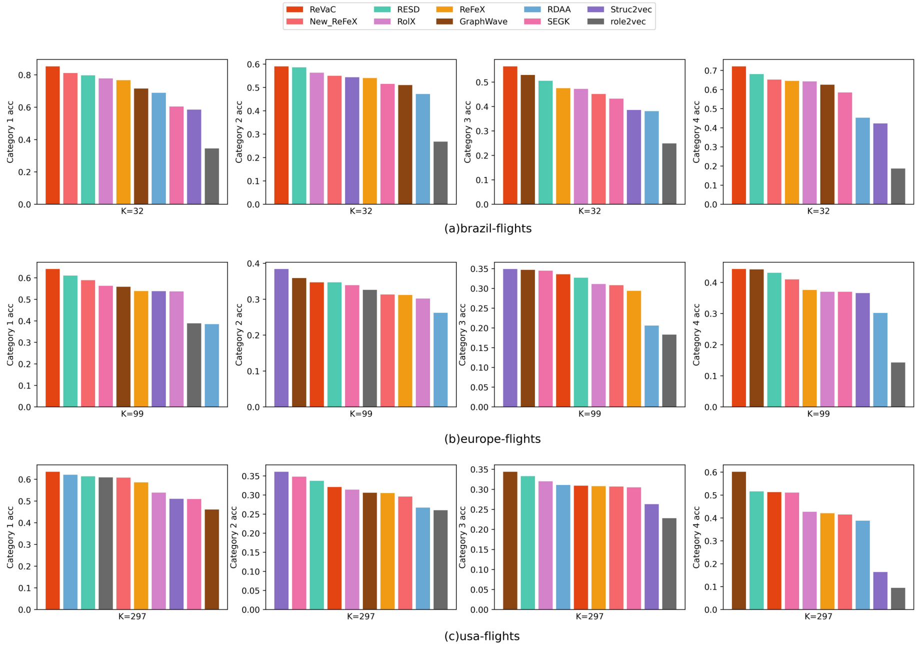
Table 2. Node classification average F1-micro score(F1 for short) and F1-macro score(F2 for short) on different networks. For each column, we mark the values with significant advantages, i.e. the top results of these methods. OM means that it cannot be calculated in fixed memory, and OT means that the result cannot be calculated within 12 h.

Method	Brazil		Europe		USA		Actor		Film	
	F1	F2	F1	F2	F1	F2	F1	F2	F1	F2
GraphWave	0.762	0.757	0.521	0.490	0.523	0.472	0.477	0.448	OM	OM
RDAA	0.790	0.783	0.462	0.436	0.610	0.597	0.473	0.456	0.509	0.406
ReFeX	0.763	0.758	0.567	0.555	0.630	0.625	0.479	0.460	0.513	0.409
New_ReFeX	0.786	0.782	0.572	0.563	0.641	0.635	0.480	0.461	0.539	0.411
RESD	0.791	0.787	0.557	0.545	0.631	0.622	0.471	0.458	0.477	0.375
role2vec	0.323	0.312	0.350	0.348	0.422	0.418	0.311	0.304	0.338	0.302
RolX	0.796	0.793	0.551	0.528	0.627	0.618	0.467	0.452	0.487	0.383
SEGK	0.733	0.726	0.536	0.524	0.615	0.606	OT	OT	OT	OT
struc2vec	0.742	0.737	0.578	0.560	0.647	0.644	OT	OT	OT	OT
ReVaC	**0.835**	**0.831**	**0.582**	**0.563**	**0.660**	**0.654**	**0.481**	**0.463**	**0.526**	**0.401**

4.5 Role-Oriented Node Classification

We conduct the task of role-based node classification on five real-world networks to quantitatively evaluate role-oriented embedding methods. To be specific, for each dataset, a linear logistic regression classifier trained and tested using embeddings generated by each base-line and our model. We randomly sample 70% node embeddings as the training set and the other embeddings are used as the test set with 20 random runs. The performance on the Micro-F1(F1 for short) and Macro-F1(F2 for short) is shown in Table 2, for each column, we label the values of methods with significant advantages.

We have following observations: (1) Role2vec gets the worst performance. RESD achieves competitive results on this task. Struc2vec shows superiority on small networks, like USA, Europe, while struc2vec and SEGK have high computational complexity. (2) As expected, the classification results of New_ReFeX on multiple datasets are better than that of ReFeX, where New_ReFeX outperforms others on Actor and gets the highest score on Film. This further illustrates that the New_ReFeX extracts higher quality local structural features and performs well on large datasets. (3) In general, Our ReVaC overperforms all of the baselines on all the datasets, which verifies the correctness of the idea extracting degree distribution features and strengthening the commonalities among local structures in node embeddings. ReVaC is a state-of-the-art method for role-oriented network representation learning.



(a)brazil-flights

(b)europe-flights

(c)usa-flights

Fig. 3. Accurate values of Top-k similarity search for different embedding methods on three datestes.

4.6 Top-k Similarity Search

In this section we demonstrate the effectiveness of our model in finding the top-k nodes that are most structurally similar to the query node. We apply the top-k similarity search task on the three air-traffic datasets, respectively. In specific, we find the k most similar nodes for the central node by computing the euclidean distance. Then we count the nodes with the same label as the central node among the K nodes and calculate the accurate value of top-k. We expect that the embedding distance of nodes with similar local structures is closer, and the number of nodes with the same label in the K nodes is greater, that is, the larger the accurate value, the better. Referring to Table 1, the number of nodes in different categories of the three air-traffic datasets is different, so we set K = 32, K = 97, K = 297. Figure 3 shows the performance of top-k search on four categories of three datasets by different embedding methods.

We can come to this conclusion gradually through the following observations: (1)none of the compared methods can produce top results on all categories across the three air traffic networks. Some methods have very high accuracy in one category but low accuracy in other categories, which leads to poor overall performance of embedding methods, such as the performance of role2vec and struc2vec on the USA network. (2)our model achieves excellent and stable results. Firstly, the accuracy of the ReVaC on all four categories on the Brazil network is significantly higher than other baseline methods. Secondly, on Europe and USA networks, although the accuracy of ReVaC on all four categories is not better than that of other methods, the average accuracy of those is significantly higher.

5 Conclusion

In this paper, aiming at the challenges of the existing role-oriented network embedding methods, we propose solutions from two aspects. On the one hand, we incorporate degree distributions of nodes into the extraction of local structural features to improve the traditional ReFeX, and then we use the Variational Auto-Encoder as an operator to obtain noise-reduced and more robust local structural embeddings. On the other hand, in the local structural embedding space, we exploit a clustering algorithm to model the common features among similar local structures and fuse them into the local structural embeddings. This makes it possible to strengthen the commonality of local structural roles on the basis of keeping local structural features, so as to achieve the purpose of enriching structural information and improving the expression power of node embedding. At the same time, we also introduce the framework of the model, carried out theoretical analysis and experiments. Extensive experiments confirm the effectiveness of ReVaC, and also demonstrate that our framework can adapt well to network scale and dimensions.

References

1. Jin, J., Heimann, M., Jin, D., Koutra, D.: Toward understanding and evaluating structural node embeddings. ACM Trans. Knowl. Discov. Data **16**(3), 1–32 (2022)
2. Ribeiro, L., Saverese, P., Figueiredo, D.: struc2vec: Learning node representations from structural identity, vol. 129685, pp. 385–394. ACM (2017)
3. Heimann, M., Shen, H., Safavi, T., Koutra, D.: Regal: representation learning-based graph alignment, pp. 117–126. ACM (2018)

4. Henderson, K., Gallagher, B., Li, L., Akoglu, L., Eliassi-Rad, T., Tong, H., Faloutsos, C.: It's who you know: graph mining using recursive structural features, pp. 663–671. ACM (2011)
5. Henderson, K., et al.: Rolx: structural role extraction mining in large graphs, pp. 1231–1239. ACM (2012)
6. Gilpin, S., Eliassi-Rad, T., Davidson, I.: Guided learning for role discovery (glrd): framework, algorithms, and applications. vol. 128815, pp. 113–121. ACM (2013)
7. Zhang, W., Guo, X., Wang, W., Tian, Q., Pan, L., Jiao, P.: Role-based network embedding via structural features reconstruction with degree-regularized constraint. Knowl.-Based Syst. **218**, 106872 (2021)
8. Jiao, P., Tian, Q., Zhang, W., Guo, X., Jin, D., Wu, H.: Role discovery-guided network embedding based on autoencoder and attention mechanism. IEEE Trans. Cybern. **PP**, 1–14 (2021)
9. Guo, X., Zhang, W., Wang, W., Yu, Y., Wang, Y., Jiao, P.: Role-oriented graph auto-encoder guided by structural information **12113**, 466–481 (2020)
10. Joaristi, M., Serra, E.: Sir-gn: a fast structural iterative representation learning approach for graph nodes. ACM Trans. Knowl. Discov. Data **15**(6), 1–39 (2021)
11. Liang, Q., et al.: Rolne: improving the quality of network embedding with structural role proximity, vol. 12342, pp. 16–28 (2020)
12. Rossi, R., Ahmed, N., Koh, E., Kim, S., Rao, A., Abbasi-Yadkori, Y.: A structural graph representation learning framework, pp. 483–491. ACM (2020)
13. Donnat, C., Zitnik, M., Hallac, D., Leskovec, J.: Learning structural node embeddings via diffusion wavelets, pp. 1320–1329. ACM (2018;2017;)
14. Tu, K., Cui, P., Wang, X., Yu, P., Zhu, W.: Deep recursive network embedding with regular equivalence, pp. 2357–2366. ACM (2018)
15. Jin, Y., Song, G., Shi, C.: Gralsp: graph neural networks with local structural patterns, pp. 4361–4368 (2020)
16. Nikolentzos, G., Vazirgiannis, M.: Learning structural node representations using graph kernels. IEEE Trans. Knowl. Data Eng. **33**(5), 2045–2056 (2021, 2019)
17. Pei, Y., Fletcher, G., Pechenizkiy, M.: Joint role and community detection in networks via l2,1 norm regularized nonnegative matrix tri-factorization, pp. 168–175. ACM (2019)
18. Jin, R., Lee, V., Hong, H.: Axiomatic ranking of network role similarity, pp. 922–930. ACM (2011)
19. Pei, Y., Du, X., Zhang, J., Fletcher, G., Pechenizkiy, M.: struc2gauss: structural role preserving network embedding via gaussian embedding. Data Min. Knowl. Disc. **34**(4), 1072–1103 (2020)
20. Shi, B., Zhou, C., Qiu, H., Xu, X., Liu, J.: Unifying structural proximity and equivalence for network embedding. IEEE access **7**, 106124–106138 (2019)
21. Ahmed, N.K., et al.: Role-based graph embeddings. IEEE Trans. Knowl. Data Eng. **34**(5), 2401–2415 (2022, 2020)
22. Rossi, R.A., Ahmed, N.K.: The network data repository with interactive graph analytics and visualization. **6**, 4292–4293 (2015)
23. Leskovec, J., Huttenlocher, D., Kleinberg, J.: Predicting positive and negative links in online social networks, pp. 641–650. ACM (2010)