SepNE: Bringing Separability to Network Embedding *

Ziyao Li and **Liang Zhang**
Yuanpei College,
Peking University, China
{leeeezy, zl515}@pku.edu.cn

Guojie Song†
Key Laboratory of Machine Perception,
Ministry of Education, Peking University, China
gjsong@pku.edu.cn

Abstract

Many successful methods have been proposed for learning low dimensional representations on large-scale networks, while almost all existing methods are designed in inseparable processes, learning embeddings for entire networks even when only a small proportion of nodes are of interest. This leads to great inconvenience, especially on super-large or dynamic networks, where these methods become almost impossible to implement. In this paper, we formalize the problem of separated matrix factorization, based on which we elaborate a novel objective function that preserves both local and global information. We further propose SepNE, a simple and flexible network embedding algorithm which independently learns representations for different subsets of nodes in separated processes. By implementing separability, our algorithm reduces the redundant efforts to embed irrelevant nodes, yielding scalability to super-large networks, automatic implementation in distributed learning and further adaptations. We demonstrate the effectiveness of this approach on several real-world networks with different scales and subjects. With comparable accuracy, our approach significantly outperforms state-of-the-art baselines in running times on large networks.

1 Introduction

Learning low dimensional representations of network data, or network embedding (NE), is a challenging task on large networks, of which the scales can reach billion-level and is growing rapidly. For example, the number of monthly-active users of Facebook reaches 2.23 billion and increases 11% yearly. [1] At the same time, although sizes of networks may infinitely grow as data accumulates, it is often the case that only small proportions of nodes are of interest in downstream applications. This is the starting point of this paper: can we respectively learn representations for different subsets of nodes-very small compared to the collectivity-while preserving information of the entire network? If so, we can obtain good representations for the requested nodes without the redundant efforts to embed irrelevant ones.

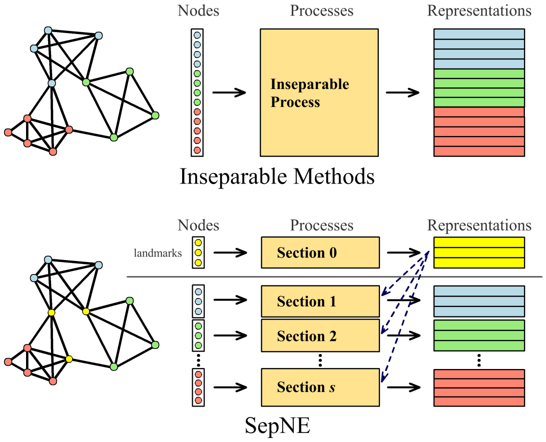

Figure 1: Inseparable and separable NE processes.

Efficiency is a major aspect of contemporary NE studies, and various methods that are applicable to large-scale networks have been proposed (Perozzi, Al-Rfou, and Skiena 2014; Tang et al. 2015; Grover and Leskovec 2016). Almost all of these methods embed entire networks with inseparable processes, in which the representation of one node depends on represented outcomes of every other node. A globally-defined optimum can be achieved under this framework, while it also causes great inconvenience: the maximum network size such methods can handle is eventually limited. For example, it takes LINE (2015), one of the fastest algorithms, several hours to embed a million-level network. Thousands of hours may be spent to achieve equivalent performance on billion-level networks. Another type of methods learns models in an inductive manner and conduct inferences over unseen data (Hamilton, Ying, and Leskovec 2017). These models have convenient inference processes, while they cannot variate over time and rely on large training data and time to achieve good performance.

The efficiency problem over super-scale networks is impossible to solve directly, as the running time of algorithms inevitably grows proportionally to problem scales. Therefore, we bring up a new perspective of solving efficiency problems: *separability*. The separability of an algorithm indicates an ability of being conducted by different workers

*This work was supported by the National Natural Science Foundation of China (Grant No. 61876006 and No. 61572041).

†Corresponding author. Email: gjsong@pku.edu.cn

[1] https://investor.fb.com/investor-news/

without exchanging information and merging outputs. In plain words, a separable algorithm divides the original problem into different *self-standing* sub-problems and separately solves each, and the solution to the sub-problems are directly usable answers instead of intermediate results. As networks are naturally composed of nodes and their relationships, an instinctive way to design separable NE algorithms is to partition the entire node set into small subsets and to separably embed each set. The solutions to the sub-problems yield direct meanings as the representations of the corresponding set of nodes.

In this paper, we implement separability in NE problems under matrix-factorization-based framework. We formalize the problem of separated matrix factorization (SMF) and elaborate a novel loss function that preserves matrix information on local, global and landmark level. We then propose **SepNE** (SEParated Network Embedding), a separable NE method based on SMF. Figure 1 illustrates the major difference between SepNE and existing methods. SepNE first partitions the node set into small subsets. A special *landmark* set is then established as references for partitioned subsets to implement *comparability*, that representations of different sets lie in the same linear space. After the landmarks are embedded, representations of different subsets are derived from the objective function defined in SMF.

Separability in NE problems yields several specific advantages. Firstly, separability makes it available to embed only the requested nodes and thus reduces the vain efforts in embedding irrelevant ones; in addition, the optimization complexity of SepNE is relevant only to the number of requested nodes instead of the entire network scale, leading to scalability to super-large networks. Secondly, even if entire networks are on request, SepNE shows higher speed than state-of-the-art algorithms due to its simplicity, while yielding comparable accuracy. Thirdly, separability leads to automatic implementations in multi-agent systems and further adaptations in dynamic networks. We evaluated SepNE with real-world networks of different sizes and topics. With equivalent or better accuracy, SepNE is at least 50% faster than current NE methods including DeepWalk (2014) and LINE.

A potentially more important contribution of this paper is the generalization of SMF. Maintaining competent performance, SMF reduces the complexity of MF problems from cubic to almost linear. This leads to intriguing further applications in the massive collectivity of MF-based algorithms.

2 Separated Matrix Factorization

2.1 Preliminaries

Given a matrix M, matrix factorization (MF) aims to find two matrices W and C that both satisfy given constraints and minimize the residuals of reconstructing M with $\widetilde{M} = W^TC$. Denoted in formulas, we have

$$\min_{W,C} \|M - W^TC\|. \tag{1}$$

W, C have lower ranks than M. In the embedding task of an n-node network, M is of size $(n \times n)$, in which each entry indicates a *proximity* between the two corresponding nodes. The proximities can be defined in various metrics, such as edge weights between nodes. Columns in W are desired representations. Columns in C are interpreted as representations of nodes when regarded as *contexts*.

2.2 Problem Definition

Directly factorizing matrices of large-scale networks can be unacceptably time-costly. Therefore, we propose SMF, a new optimization problem, as a trade-off between speed and accuracy. To implement separability, SMF divides the problem with a partition over the node set, and correspondingly partitions the proximity matrix. Below is a formal definition of separated matrix factorization (SMF) in NE scenarios.

SMF takes a network $G = (V, E), |V| = n$, its proximity matrix M, and a partition setup $f : V \mapsto \mathcal{V}, \mathcal{V} = \{V_1, \cdots, V_s\}$ as inputs. The task is to derive representations $(W_1, \cdots, W_s)$ and $(C_1, \cdots, C_s)$ for the partitioned sets that optimally reconstruct M. The loss of the reconstruction is defined the same as Problem (1). [2] Without loss of generosity, we permute and partition M according to $\mathcal{V}$ as

$$M = \begin{pmatrix} M_{11} & \cdots & M_{1s} \\ \vdots & \ddots & \vdots \\ M_{s1} & \cdots & M_{ss} \end{pmatrix},$$

where M_{ij} indicates the proximities between V_i and V_j. To achieve independence between sections, SMF restricts that the embedding section of every set is conducted (i) with only the proximities related to itself (the section of embedding V_i can leverage only M_{ij} and M_{ji}, $j = 1, \cdots, s$), and (ii) without any outcomes of other sections.

2.3 Method

Partitioning the nodes is the first step to separability. However, representations in the partitioned sets can be incomparable due to the limitation over the access to proximities. In another word, representations of different sets do not have a unified constraint that bounds them in the same linear space. To implement comparability, we establish landmarks with highly interactive nodes in the network, which serve as invariant references for different subsets. For the factorization process, preserving only local information is a simple way to reconstructs micro-structures of networks at a loss of global references. Combining landmark solves the comparability problem, while it still ignores the interactions between different subsets. Therefore, we elaborate a novel objective function for SMF that preserves local, global and landmark information, which achieves state-of-the-art performances.

Local information. The proximities in the partitioned matrix are naturally divided into two types, namely *local information* and *global information*. Local information refers to the proximities within every set, or sub-matrices on the diagonal; global information refers to the proximities between all pairs of different sets, or the off-diagonal sub-matrices.

[2] Frobenius forms are usually adopted in SMF, since the Frobenius form of a matrix is additive of all its entries, and therefore can be decomposed into sums of all the Frobenius forms of its partitioned sub-matrices.

We start modeling SMF with a naïve simplification that preserves only local information by factorizing s matrices on the diagonal:

$$\min_{W_i, C_i} \|M_{ii} - W_i^T C_i\|, i = 1, \cdots, s.$$

This primitive approach discards all interactions between different sets, leading to incomparable representations.

Landmark information. To implement comparability, we resort to a third type of information, *landmark information*. Landmark information indicates the proximities between subsets and manually established *landmarks*, a special set of nodes (denoted as V_0) that are chosen as references for different subsets. The improved approach sets a unified constraint over landmark information in different sets and solves the problem in two stages, formulated as

$$\min_{\Phi, \Psi} \|M_{00} - \Phi^T \Psi\|, \tag{2}$$

$$\min_{W_i, C_i} \left\| \begin{pmatrix} M_{00} & M_{0i} \\ M_{i0} & M_{ii} \end{pmatrix} - \begin{pmatrix} \Phi^T \Psi & \Phi^T C_i \\ W_i^T \Psi & W_i^T C_i \end{pmatrix} \right\|, \quad i = 1, \cdots, s. \tag{3}$$

The first stage embeds the landmarks ($W_0 = \Phi$, $C_0 = \Psi$) in Problem (2), and the second stage derives representations for rest sets by solving Problem (3) with calculated Φ and Ψ. If Frobenius forms are used, the loss in Problem (3) can be explicitly decomposed into local and landmark loss as

$$\begin{aligned} \mathcal{L}_i^{lc}(W, C) &= \frac{1}{2}\|M_{ii} - W^T C\|_F^2, & (4) \\ \mathcal{L}_i^{lm}(W, C) &= \frac{1}{2}\|M_{0i} - \Phi^T C\|_F^2 \\ &\quad + \frac{1}{2}\|M_{i0} - W^T \Psi\|_F^2. & (5) \end{aligned}$$

Global information. To further combine global information into the objective function, we elaborate a global loss by first transforming Problem (3) into an equivalent form. We denote $k := |V_0|$ and assume calculated $\Phi, \Psi \in \mathbf{R}^{(d \times k)}$ are of rank d. [3] $W_i, C_i \in \mathbf{R}^{(d \times |V_i|)}$ can then be represented as linear combinations of columns in Φ, Ψ, formulated in matrix denotation as

$$\begin{aligned} W_i &= \Phi A_i \\ C_i &= \Psi B_i, \quad i = 1, 2, \cdots, s, \end{aligned}$$

where $A_i, B_i \in \mathbf{R}^{(k \times |V_i|)}$ are the coefficient matrices.

Consider a simple case where $s = 2$, V_0 is the set of landmarks and V_1 is the target subset to embed. After the transformation, global information is preserved through

$$\min_{A_1, B_2} \|M_{12} - A_1^T \Phi^T \Psi B_2\| \tag{6}$$

$$\min_{A_2, B_1} \|M_{21} - A_2^T \Phi^T \Psi B_1\|. \tag{7}$$

Problem (6)(7) are not separable, for the results of embedding V_2 (A_2,B_2) exist in V_1 problem. However, a surprising property emerges after the transformation, that $\Phi^T \Psi B_2 = W_0^T C_2 = \tilde{M}_{02}$ can be well-approximated with M_{02} if representations of V_2 are required to preserve landmark information, $A_2^T \Phi^T \Psi$ similarly. Therefore, Problem (6)(7) can be substituted as

$$\min_{A_1} \|M_{12} - A_1^T M_{02}\| \tag{8}$$

$$\min_{B_1} \|M_{21} - M_{20} B_1\|, \tag{9}$$

and separability is achieved.

The idea can be generalized to any given s and V_i by simply substituting all V_2-related variables to $V_{\bar{i}}$-related ones, where $V_{\bar{i}} = \bigcup_{j \notin \{0,i\}} V_j$. The approximation still holds if landmark information is preserved in all sets. For any set V_i, the global loss function is defined as

$$\mathcal{L}_i^{gb}(A, B) = \frac{1}{2}(\|M_{i\bar{i}} - A^T M_{0\bar{i}}\|_F^2 + \|M_{\bar{i}i} - M_{\bar{i}0} B\|_F^2).$$

2.4 Final Optimization Problem

Combined with λ-scaled global loss and regularization over A, B, the final loss function of SMF becomes

$$\begin{aligned} \mathcal{L}_i(A, B) &= \mathcal{L}_i^{lc}(A, B) + \mathcal{L}_i^{lm}(A, B) & (10) \\ &\quad + \lambda \mathcal{L}_i^{gb}(A, B) + \frac{\eta}{2}(\|A\|_F^2 + \|B\|_F^2), \end{aligned}$$

where $\mathcal{L}_i^{lc}$ and $\mathcal{L}_i^{lm}$ are redefined in A, B-denotation, namely

$$\begin{aligned} \mathcal{L}_i^{lc}(A, B) &= \frac{1}{2}\|M_{ii} - A^T \Phi^T \Psi B\|_F^2, & (11) \\ \mathcal{L}_i^{lm}(A, B) &= \frac{1}{2}\|M_{0i} - \Phi^T \Psi B\|_F^2 \\ &\quad + \frac{1}{2}\|M_{i0} - A^T \Phi^T \Psi\|_F^2. & (12) \end{aligned}$$

Accordingly, the final optimization problem of SMF is formulate as

$$\begin{aligned} & W_0 = \Phi, \quad C_0 = \Psi. \\ & W_i = \Phi A_i, \quad C_i = \Psi B_i, \\ & A_i, B_i = \arg\min_{A,B} \mathcal{L}_i(A, B), \quad i = 1, 2, \cdots, s. \end{aligned}$$

3 SepNE: Separated Network Embedding

In this section, we propose SepNE, a simple and separable NE approach based on SMF. A general framework of SepNE is presented in Algorithm 1. We then illustrate the details of SepNE, including the partition setups, landmark-selecting approaches and optimization method.

SepNE takes a given network as input and outputs node representations. In the preparation stage, landmarks are selected and embedded, and rest nodes are partitioned under a certain setup. In the second stage, partitioned sets are independently embedded by optimizing the SMF problem. The second stage is designed in a separable manner, so that if a small proportion of nodes are requested, Loop 4 in Algorithm 1 can be conducted only on the sets containing these nodes. Besides, separability allows cycles in the loop to be run distributedly.

[3] This can be guaranteed with SVD decomposition if $k \geq d$.

Algorithm 1 General framework of SepNE.

```
Input: $G = (V, E), |V| = n$.
Output: Node embeddings for partitioned sets of nodes.
Partition rest nodes in set $V$ into $s$ subsets as $\mathcal{V}$;
Sample $k$ landmarks as set $V_0$;
Conduct SVD on calculated proximity matrix $M_{00}$ and calculate $\Phi, \Psi$:
    $M_{00} = U_d \Sigma_d V_d^T,$
    $\Phi = U_d \sqrt{\Sigma_d}, \Psi = V_d \sqrt{\Sigma_d};$
$W_0 = \Phi$;
for $i = 1, 2, \cdots, s$ do
    Calculate relevant proximity matrices $M_{0i}$, $M_{i0}$, $M_{ii}$, $M_{i\bar{i}}$, $M_{\bar{i}i}$;
    Optimize the loss functions:
        $A_i, B_i = \arg\min_{A,B} \mathcal{L}_i(A, B);$
    Calculate embeddings for set $V_i$:
        $W_i = \Phi A_i, C_i = \Psi B_i$
end for
return $(W_0, W_1, \cdots, W_s)$
```

For the proximity matrix to be factorized, two metrics are adopted in SepNE. The first metric defines $M = A + A^2$, where A is the transition matrix of PageRank (1999) [4]. The second metric simplifies the first one with $M = I + A$. Our metrics are similar to TADW (2015), which proved an equivalency between factorizing A and $A + A^2$ and DeepWalk (2014) with very short walks. A more instinctive understanding of the metrics can be derived from the perspective of *proximity orders*. A can be interpreted as a measurement of first-order proximity, and $A + A^2$ a combination of first-order and second-order proximity. These concepts were proposed and further discussed in (Tang et al. 2015).

3.1 Partition Setups

We propose three different partition setups for SepNE in this paper. *SepNE-LP* (Louvain Partition) partitions a network according to its communities using Louvain (2008). Leveraging community structures conforms *matrix local information* to *network local information* [5], which serves as an empirical approach to improve performance.

However, as SepNE leverages all the information of the proximity matrix, community-based partitions are not necessary. We further propose *SepNE-RP* (Random Patition) which randomly assigns nodes to sets, and *SepNE-IO* (Interested Only) which simply puts the requested nodes into one or more sets and ignores all unrequested ones.

3.2 Landmark Selection

Landmark-selecting approaches influence not only representations of the landmarks, but also the loss of all sets in the entire SMF problem. As the key intention of setting landmarks is to establish references for different sets, landmarks are expected (i) to have as much connection with rest nodes as possible; (ii) to have the connection cover as many sets as possible.

Approaches that select nodes with high degrees generally work well if k is loosely controlled. However, on real-world networks, nodes with the highest degrees tend to distribute in a few giant and highly connected communities. When k is strictly confined, choosing these nodes actually limits the number of sets these landmarks adjoin. To relieve this problem, we propose *GDS* (Greedy Dominating Set), an approach that greedily maximizes the number of nodes the landmarks adjoins.

GDS first forms a maximum heap using degrees of nodes and initialize the landmark set as empty. After initialization, GDS iteratively examines the top of the heap. The top is simply removed if dominated by the current landmark set, otherwise added into the set and then removed. The process continues until the heap is empty or the size reaches k. Experiments show that GDS well captures the informative structure of a network when k is strictly confined.

While serving as good references, landmarks selected with GDS are completely one-hop isolated. As a consequence, if only one-hop proximity is leveraged, M_{00} of GDS is supposed to be a diagonal matrix. Furthermore, if $k > d$, SVD will generate null representations for some landmarks. Therefore, we only use GDS when higher order proximity is adopted or $k = d$. Otherwise, we implement degree-based approaches.

3.3 Optimization

The optimization problem in SepNE is solved similarly to (Yu et al. 2014), where A, B are iteratively optimized as

$$
\begin{aligned}
A^{(t+1)} &= \arg\min_{A} \mathcal{L}_i(A, B^{(t)}), \\
B^{(t+1)} &= \arg\min_{B} \mathcal{L}_i(A^{(t+1)}, B).
\end{aligned}
$$

As explicit calculation of the loss function value involves large matrix multiplications, A, B are calculated by solving the gradient-minimization problem $\frac{\partial \mathcal{L}}{\partial A} = \frac{\partial \mathcal{L}}{\partial B} = 0$ in each iteration. Cholesky decomposition is adopted as the matrix in the gradient problem is always positive-definite.

3.4 Complexity Analysis

The complexity of the preparation stage is $O(nlogn + k^3)$, while empirically the time expenses are low especially when random partitions are implemented. If $M = I + A$, the average complexity of each section is $O(k \times (deg + iter \times k) \times n_i)$, including both the time in calculating proximity matrices and in optimization.

With separability implemented, SepNE is available to only embed a proportion of nodes. Besides, when small proportions of nodes are requested, the complexity of SepNE in the second stage is irrelevant to the entire scale of the network. This property yields strong scalability of SepNE to super-scale networks.

4 Experiments

We evaluated SepNE on several publicly available real-world networks with different sizes and topics, including

[4] $A_{ij} = 1/d_i$ if $(i, j) \in E$ and 0 otherwise, where d_i is the degree of node i.

[5] Which refers to connections within real-world communities.

Dataset	Directed	Nodes	Links
Wiki	directed	2,405	17,981
Cora	directed	2,708	5,429
Citeseer	directed	3,312	4,732
Flickr	undirected	1,715,255	22,613,981
Youtube	undirected	1,157,827	4,945,382
Wiki-Gen	directed	2^i	$\approx 7.5 \times 2^i$

Table 1: Statistics of datasets used in this paper.

three document networks and two social networks. Performances over three benchmark tasks were evaluated: (i) matrix reconstruction on document networks, (ii) multi-class classification tasks on document networks and (iii) multi-label classification tasks on social networks.

4.1 Experiment Setups

Datasets. Five real-world networks were used in this paper. *Wiki*, *Cora* and *Citeseer* are thousand-level document networks. [6] Wiki contains Wikipedia pages of 19 classes; Cora contains machine learning papers from 7 classes and *Citeseer* contains publications from 6 classes. Links between documents are pointers or citations. *Flickr* and *Youtube* are million-level social networks. [7] Users and their relationships are represented as nodes and links on the networks, and real-world communities are available. *Wiki-Gen*s are a series of networks generated by implementing Kronfit (2010) on Wiki. Table 1 shows the statistics of all datasets.

Comparison Algorithms and Parameters. Algorithms and their parameters are briefly introduced below. We did not compare SepNE with algorithms that are not scalable to large networks. Except otherwise noted, the representation dimension for all algorithms was $d = 128$.

- **SVD** was conducted over the full proximity matrices. As SVD theoretically generates the optimal rank-d approximations in F-norm, it was proposed as the strongest possible baseline for matrix reconstruction.
- **Nyström Method** (Drineas and Mahoney 2005) is a fast monte-carlo method of approximating matrix multiplications. It was taken as a representative of probabilistic MF algorithms. The number of *landmarks* in Nyström method is set the same as SepNE for fair comparison.
- **LINE** (2015) embeds a network by optimizing an objective function of edge reconstruction. The parameters were set the same as the original paper, namely $\rho_0 = 0.025$, negative sampling $K = 5$ and sample size $T = 10^{10}$.
- **DeepWalk** (2014) embeds nodes by regarding them in random walk sequences as words in sentences. The parameters were set as window size $win = 10$, walk length $t = 40$ and walks per node $\gamma = 40$.
- **SepNE** was evaluated under all three partition setups (SepNE-LP, SepNE-RP and SepNE-IO). On document networks, parameters were set as $iter = 100$, $\lambda = 0.4$ and $\eta = 0.1$, $M = A + A^2$ and $k = 200$; on social networks, parameters were $iter = 5$, $\lambda = 50$, $\eta = 1$, $M = I + A$ and $k = 1000$.

Table 2: Running time comparison over flickr.

SepNE-IO	6.2mins
SepNE-RP	43.8mins
SepNE-LP	68.8mins
LINE(1st)	138.1mins
DeepWalk	>24hrs

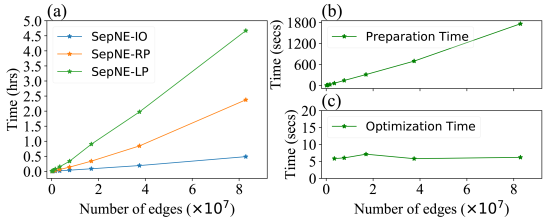

Figure 2: Scalability of SepNE, demonstrated on Wiki-Gens.

4.2 Running Time

To demonstrate the speed advantage of SepNE, we compared the running time of SepNE, LINE and DeepWalk on Flickr network. [8] The nodes in the five biggest communities of Flickr (75,338 nodes, 4.39%) were regarded as interested for SepNE-IO. Results are presented in Table 2. With better performance (introduced below), our method was significantly faster than LINE (for 50.2%) and DeepWalk even in embedding the entire network; when requested to embed only the nodes of interest, SepNE completed the task in a very short time. [9]

We also evaluated the trend of running time with network scales increasing on Wiki-Gens and the number of requested nodes for SepNE-IO fixed as 10,000. Figure 2 (a) shows a linear trend in all three setups. Figure 2 (b)(c) show the trends of preparation and optimization time in SepNE-IO. Preparation time increased linearly mainly due to the time used in reading data, while optimization time remained invariant. The results all corroborate that SepNE is scalable to super-large networks.

4.3 Matrix Reconstruction

The performance of reconstructing proximity matrix is a direct metric of representation quality. We evaluated matrix reconstruction performances of different algorithms on all document datasets. As the results were similar, we took Wiki as a representative. Two metrics were used, including the R^2

[6] Available at `https://linqs.soe.ucsc.edu/data`

[7] Available at `http://socialnetworks.mpi-sws.org/datasets.html`

[8] All efficiency experiments were conducted on a single machine with 128GB memory, 32 cores 2.13GHz CPU with 16 workers.

[9] Data are saved as edge lists in the experiments for fair comparison. If adjacent lists are available, the time of SepNE-IO can still be reduced significantly.

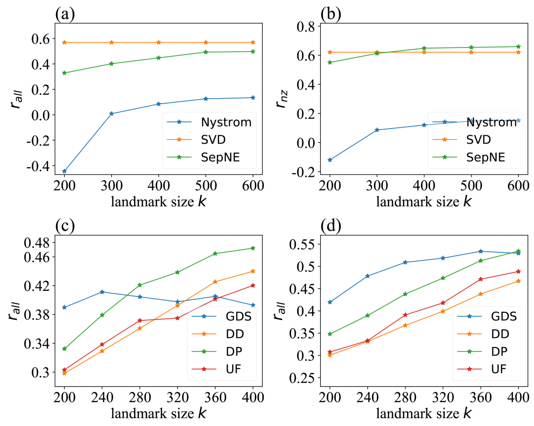

Figure 3: Performances of matrix preserving on Wiki network: (a)(b). r_{all} and r_{nz} of different algorithms versus k; (c)(d). r_{all} of different landmark-selecting approaches versus k ($d = 128$ in (c) and 200 in (d)).

score over all entries (r_{all}) and non-zero entries (r_{nz}):

$$r_{all} = 1 - \frac{\|\tilde{M} - M\|_F^2}{\|M\|_F^2},$$

$$r_{nz} = 1 - \frac{\|(\tilde{M} - M) \times B\|_F^2}{\|M\|_F^2},$$

where × indicates element-wise multiplication and $B_{ij} = 1$ if $M_{ij} \neq 0$ otherwise 0.

We evaluated Nyström method, SVD and SepNE with different k over both metrics. We then compared different landmark-selecting approaches with different k and d, including four: *DD (Degree Deterministic)* picking nodes with the k highest degrees; *DP (Degree Probabilistic)* sampling landmarks using degrees as weights; *UF (Uniform)* uniformly selecting landmarks; and GDS.

According to Figure 3 (a)(b), SepNE significantly outperforms Nyström method for up to 38.3% and shows competent performance compared with SVD. SVD shows its advantage on r_{all}, while preserving non-zero entries can be more important than zero entries on real-world networks due to the existence of unobserved links. When k is large enough, SepNE outperforms SVD on r_{nz}. This is because non-zero entries are more densely distributed inside communities and better reconstructed by SepNE.

In Figure 3 (c)(d), when k is slightly larger than d, or to say that k is strictly confined, GDS shows its significant advantages. However, the r_{all} of GDS decreases when $k > 2d$ due to the null representations generated in SVD. At the same time, degree-based approaches gradually get rid of their biases as k increases, and therefore show continuous improvements of performance.

4.4 Classification

We implemented two types of classification tasks on document and social networks. A simple logistic regression was used as the classifier for both tasks. The representations were all normalized before used as features. All results were averaged over 10 runs.

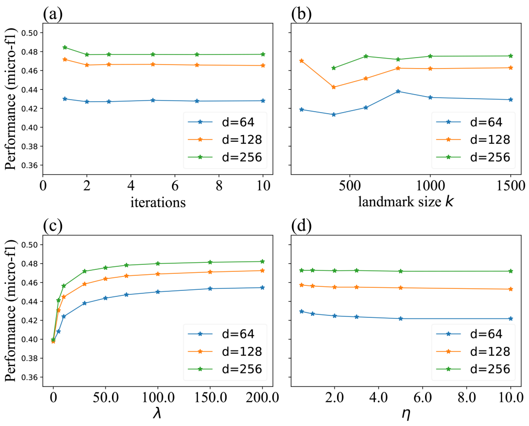

Figure 4: Performances of multi-label classification under different parameter settings on Flickr network.

Multi-class classification. We implemented multi-class classification on three document networks which predicts the subject category a given document is in. Table 3 reports the performances. Macro F1 results are not shown due to the similarity. Despite the minimum information SepNE leverages, it outperforms DeepWalk in the majority (4 out of 6) of cases. This is because SepNE incorporates a more robust and elegant way to leverage proximities between nodes. LINE is struggling to capture information on smaller networks, while SepNE is as well competent.

Multi-label classification. The multi-label classification task on social networks was defined as predicting whether a given node is in each community. The five largest in Flickr and communities with more that 1,000 members in Youtube were extracted as labels. As labels were sparse, we conducted training and predicting processes over the nodes that have at least one label. The training percentage was varied from 1% to 90%. Table 4 and Table 5 show the results.

SepNE shows significant advantages on Flickr. Using 10% training data, SepNE-LP outperforms LINE and DeepWalk using 90%. Representations from SepNE are more predictive than DeepWalk even if only one-hop proximity is leveraged. The reason may be that as Flickr has relatively high average degree, the larger window size of DeepWalk actually encumbers it in determining the importance of information. All three setups of SepNE show good performances, while SepNE-LP shows its advantage over the other two setups. This shows the effectiveness of the empirical method of partitioning networks according to communities, while the time cost of SepNE-LP is significantly higher than the other two simplified setups. The task on Youtube is more challenging as both the network and labels are much sparser. DeepWalk outperforms both LINE and SepNE due to its ability in leveraging remote proximities with its larger window size, which successfully relieves the problem of spar-

Table 3: Multi-class prediction results over document networks *(micro-averaged F1 scores)*. Best performances are bolded.

	Wiki		Cora		Citeseer	
%train	10%	90%	10%	90%	10%	90%
LINE(1st)	0.4488	0.5937	0.4657	0.6009	0.3206	0.4259
LINE(2nd)	0.3298	0.4787	0.2637	0.3297	0.2221	0.2561
DeepWalk	0.5737	**0.6893**	**0.7509**	0.8187	0.5086	0.5813
SepNE	**0.5764**	0.6867	0.7365	**0.8220**	**0.5157**	**0.6072**

Table 4: Multi-label prediction results over Flickr network *(micro-averaged F1 scores)*. Best performances are bolded.

%train	1%	3%	5%	10%	20%	30%	50%	90%
LINE(1st)	0.3683	0.4118	0.4165	0.4219	0.4270	0.4273	0.4296	0.4274
LINE(2nd)	0.3450	0.3824	0.3955	0.3973	0.4032	0.4056	0.4069	0.4068
DeepWalk	0.4072	0.4353	0.4433	0.4481	0.4518	0.4564	0.4585	0.4592
SepNE-IO	0.4065	0.4341	0.4477	0.4562	0.4582	0.4607	0.4630	0.4622
SepNE-RP	0.4061	0.4388	0.4502	0.4601	0.4628	0.4634	0.4636	0.4658
SepNE-LP	**0.4269**	**0.4468**	**0.4562**	**0.4623**	**0.4645**	**0.4656**	**0.4674**	**0.4677**

Table 5: Multi-label prediction results over Youtube network *(micro-averaged F1 scores)*. Best two performances are bolded.

%train	1%	3%	5%	10%	20%	30%	50%	90%
LINE(1st)	0.1031	0.2322	0.2745	0.3141	0.3410	0.3520	0.3594	0.3673
LINE(2nd)	0.0782	0.1839	0.2158	0.2643	0.2987	0.3159	0.3280	0.3350
DeepWalk	0.2037	**0.3397**	**0.3739**	**0.4105**	**0.4355**	**0.4438**	**0.4501**	**0.4556**
SepNE-IO	0.2035	0.3325	0.3574	0.3885	0.4041	0.4129	0.4170	0.4216
SepNE-RP	**0.2256**	0.3355	**0.3633**	**0.3920**	0.4115	0.4157	0.4214	0.4273
SepNE-LP	**0.2253**	**0.3361**	0.3620	0.3882	**0.4118**	**0.4170**	**0.4218**	**0.4277**

sity at the cost of much higher time expenses. SepNE outperforms both LINE(1st) and LINE(2nd), which again corroborates its stronger ability to leverage near proximities.

4.5 Parameter Sensitivity

Figure 4 shows the effect of $iter$, k, λ, η and d. $Iter$, k and η do not show significant influences. The performance of SepNE is good with even one iteration, probably indicating that local information on Flickr is less important. Figure 4 (c) shows that larger λ generally leads to better performance, converging with $\lambda \geq 100$. The higher performances of larger λs, particularly compared with $\lambda = 0$, show the effectiveness of the elaborated global loss.

5 Related Work

There are massive literature proposed over NE problems. Traditional dimension reduction approaches (Roweis and Saul 2000; Tenenbaum, de Silva, and Langford 2000; Belkin and Niyogi 2002) are applicable on network data through Graph Laplacian Eigenmaps or proximity MF. Recently, various skip-gram-based NE models and applications were proposed (Perozzi, Al-Rfou, and Skiena 2014; Grover and Leskovec 2016; Du et al. 2018). Besides, the pioneering work of (Levy and Goldberg 2014) proved an equivalency between skip-gram models and matrix factorization, which further leads to new proximity metrics under the proximity MF framework (Yang et al. 2015; Cao, Lu, and Xu 2015; Qiu et al. 2018). Edge reconstruction algorithms (Tang et al. 2015) were proposed to gain scalability on large networks. Neural networks, including autoencoders (Wang, Cui, and Zhu 2016; Tu et al. 2018) and CNNs (Kipf and Welling 2016; Hamilton, Ying, and Leskovec 2017) were also leveraged in NE problems. There is also a new trend (Ribeiro, Saverese, and Figueiredo 2017) that leverages structural information instead of proximity in NE.

The most similar work to ours is (Ahmed et al. 2013), in which a similar partition was adopted to achieve separability, while other parts of the work had major differences with ours. Besides, it focused mainly on technical issues in distributed learning and preserved only link information, while SepNE is more generalized idea with a more elaborated optimization goal.

6 Conclusion

In this paper, we formalized the problem of separated matrix factorization, based on which we proposed SepNE, an separable network embedding method which outperforms strong baselines in both efficiency and performance.

The key contribution of SepNE is providing a novel perspective of evaluating network embedding methods: separability. A separable method is stronger than a distributable one, as partly conducting a separable task provides meaningful outputs. This property provides an option of embedding only a proportion of nodes and yields strong significance in distributed learning, super-large network embedding and dynamic network embedding. Furthermore, SMF reduces the complexity of MF from cubic to linear with the generalizability over all MF-based algorithms.

SepNE is still a simple framework. For future work, one intriguing direction is to incorporate more complex information into the framework without loss of efficiency. Also, a theoretical proof of a lower bound over the loss in matrix reconstruction, and a more theoretical explanation of SMF can be extremely informative. Should there be such work, it will be theoretically-founded to apply SMF on all matrix-factorization-based algorithms.

References

[Ahmed et al. 2013] Ahmed, A.; Shervashidze, N.; Narayanamurthy, S.; Josifovski, V.; and Smola, A. J. 2013. Distributed large-scale natural graph factorization. In *Proceedings of the 22th International Conference on World Wide Web (WWW'13)*, 37–48.

[Belkin and Niyogi 2002] Belkin, M., and Niyogi, P. 2002. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Advances in Neural Information Processing Systems 14*. 585–591.

[Blondel et al. 2008] Blondel, V. D.; Guillaume, J.-L.; Lambiotte, R.; and Lefebvre, E. 2008. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* 2008(10):P10008.

[Cao, Lu, and Xu 2015] Cao, S.; Lu, W.; and Xu, Q. 2015. Grarep: Learning graph representations with global structural information. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management (CIKM'15)*, 891–900.

[Cui et al. 2018] Cui, P.; Wang, X.; Pei, J.; and Zhu, W. 2018. A survey on network embedding. *IEEE Transactions on Knowledge and Data Engineering*.

[Drineas and Mahoney 2005] Drineas, P., and Mahoney, M. 2005. On the nystrm method for approximating a gram matrix for improved kernel-based learning. *Journal of Machine Learning Research* 6:2153–2175.

[Du et al. 2018] Du, L.; Wang, Y.; Song, G.; Lu, Z.; and Wang, J. 2018. Dynamic network embedding : An extended approach for skip-gram based network embedding. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI'18)*, 2086–2092.

[Grover and Leskovec 2016] Grover, A., and Leskovec, J. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'16)*, 855–864.

[Hamilton, Ying, and Leskovec 2017] Hamilton, W.; Ying, Z.; and Leskovec, J. 2017. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems 30*. 1024–1034.

[Kipf and Welling 2016] Kipf, T. N., and Welling, M. 2016. Semi-supervised classification with graph convolutional networks. *CoRR* abs/1609.02907.

[Leskovec et al. 2010] Leskovec, J.; Chakrabarti, D.; Kleinberg, J.; Faloutsos, C.; and Ghahramani, Z. 2010. Kronecker graphs: An approach to modeling networks. *Journal of Machine Learning Research* 11:985–1042.

[Levy and Goldberg 2014] Levy, O., and Goldberg, Y. 2014. Neural word embedding as implicit matrix factorization. In *Advances in Neural Information Processing Systems 27*. 2177–2185.

[Mislove et al. 2007] Mislove, A.; Marcon, M.; Gummadi, K. P.; Druschel, P.; and Bhattacharjee, B. 2007. Measurement and analysis of online social networks. In *Proceedings of the 5th ACM/Usenix Internet Measurement Conference (IMC'07)*.

[Page et al. 1999] Page, L.; Brin, S.; Motwani, R.; and Winograd, T. 1999. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford University.

[Perozzi, Al-Rfou, and Skiena 2014] Perozzi, B.; Al-Rfou, R.; and Skiena, S. 2014. Deepwalk: online learning of social representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'14)*, 701–710.

[Qiu et al. 2018] Qiu, J.; Dong, Y.; Ma, H.; Li, J.; Wang, K.; and Tang, J. 2018. Network embedding as matrix factorization: Unifying deepwalk, line, pte, and node2vec. In *Proceedings of the 11th ACM International Conference on Web Search and Data Mining (WSDM'18)*, 459–467.

[Ribeiro, Saverese, and Figueiredo 2017] Ribeiro, L. F.; Saverese, P. H.; and Figueiredo, D. R. 2017. Struc2vec: Learning node representations from structural identity. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'17)*, 385–394.

[Roweis and Saul 2000] Roweis, S., and Saul, L. 2000. Nonlinear dimensionality reduction by locally linear embedding. *Science* 2323–2326.

[Tang et al. 2015] Tang, J.; Qu, M.; Wang, M.; Zhang, M.; Yan, J.; and Mei, Q. 2015. Line:large-scale information network embedding. In *Proceedings of the 24th International Conference on World Wide Web (WWW'15)*, 1067–1077.

[Tenenbaum, de Silva, and Langford 2000] Tenenbaum, J. B.; de Silva, V.; and Langford, J. C. 2000. A global geometric framework for nonlinear dimensionality reduction. *Science* 2319–2323.

[Tu et al.] Tu, K.; Cui, P.; Wang, X.; Yu, P. S.; and Zhu, W. Deep recursive network embedding with regular equivalence. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'18)*.

[Tu et al. 2018] Tu, K.; Cui, P.; Wang, X.; Wang, F.; and Zhu, W. 2018. Structural deep embedding for hyper-networks. In *Proceedings of the 32nd AAAI's Conference on Artificial Intelligence (AAAI'18)*.

[Wang, Cui, and Zhu 2016] Wang, D.; Cui, P.; and Zhu, W. 2016. Structural deep network embedding. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'16)*, 1225–1234.

[Yang et al. 2015] Yang, C.; Liu, Z.; Zhao, D.; Sun, M.; and Chang, E. 2015. Network representation learning with rich text information. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI'15)*, 2111–2117.

[Yu et al. 2014] Yu, H.-F.; Jain, P.; Kar, P.; and Dhillon, I. 2014. Large-scale multi-label learning with missing labels.

In *Proceedings of the 31st International Conference on Machine Learning (ICML'14)*, 593–601.

[Zhang et al. 2018] Zhang, Z.; Cui, P.; Wang, X.; Pei, J.; Yao, X.; and Zhu, W. 2018. Arbitrary-order proximity preserved network embedding. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'18)*.