Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)

Exploiting k-Degree Locality to Improve Overlapping Community Detection

Hongyi Zhang[1,2], Michael R. Lyu[1,2], Irwin King[1,2]
[1]Shenzhen Key Laboratory of Rich Media Big Data Analytics and Applications,
Shenzhen Research Institute, The Chinese University of Hong Kong, Shenzhen, China
[2]Department of Computer Science and Engineering,
The Chinese University of Hong Kong, Shatin, N.T., Hong Kong
{hyzhang, lyu, king}@cse.cuhk.edu.hk

Abstract

Community detection is of crucial importance in understanding structures of complex networks. In many real-world networks, communities naturally overlap since a node usually has multiple community memberships. One popular technique to cope with overlapping community detection is *Matrix Factorization (MF)*. However, existing MF-based models have ignored the fact that besides neighbors, "local non-neighbors" (e.g., my friend's friend but not my direct friend) are helpful when discovering communities. In this paper, we propose a *Locality-based Non-negative Matrix Factorization (LNMF)* model to refine a preference-based model by incorporating locality into learning objective. We define a subgraph called "k-degree local network" to set a boundary between local non-neighbors and other non-neighbors. By discriminately treating these two class of non-neighbors, our model is able to capture the process of community formation. We propose a fast sampling strategy within the stochastic gradient descent based learning algorithm. We compare our *LNMF* model with several baseline methods on various real-world networks, including large ones with ground-truth communities. Results show that our model outperforms state-of-the-art approaches.

1 Introduction

An individual in a social network can not only be regarded as an individual. One's behaviors are influenced by people around her, especially close friends. And her activities will influence others as well. A person always appears in a social network with multiple social identities, e.g., a (former) graduate student, a family member, a club member, a star fan, a company employer, etc. In most cases, her behaviors are related to one or several of these identities. Since identities can be defined by communities, discovering such overlapping communities in social networks becomes an important task for understanding social relationships and activities. This task is known as *overlapping community detection* [Fortunato, 2010; Gavin *et al.*, 2002; Newman, 2001].

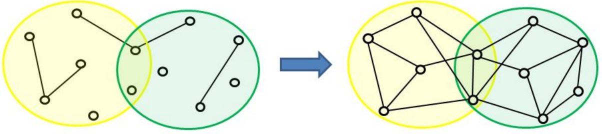

Figure 1: **Community is actually the reason behind links.**

Unlike classic community detection assuming that communities are mutually exclusive, overlapping community detection cannot be directly turned into the traditional graph clustering (i.e., node clustering) problem. Thus, many heuristic methods have been proposed in the past decade to deal with this task. Early approaches pay more attention on links. *Clique Percolation* [Palla *et al.*, 2005; Kumpula *et al.*, 2008] tries to find all k-cliques (a complete graph with k nodes) and combine those sharing $k-1$ nodes to be communities. *Link clustering* [Ahn *et al.*, 2010], on the other hand, cluster links instead of nodes and assign each node to all communities that its corresponding links belong to. Other recent works such as [Coscia *et al.*, 2012; Whang *et al.*, 2013] select some seed node and use links to expand communities. These methods aim to seek communities via links, but do not address the issue that communities are the actual reason behind links (see Figure 1). Considering a user's ego network [McAuley and Leskovec, 2012], i.e., a network of connections between her friends where communities are social circles categorized manually, the reason for two nodes to build a link is that they are in the same category. For example, the probability of one's college mates to be friends are usually much higher than that of one's random friends.

Based on the idea that "communities generate links", *Matrix Factorization* based model has been employed for overlapping community detection. To apply this model, we need to set the number of communities and randomly assign users to each community in advance. Then a particular objective function will be adopted to update the community membership for each node. Previously, a typical objective function is to minimize $||A - FF^T||$, where A is the adjacency matrix of the network and F is the node-community membership matrix. However, this value-approximation based objective function is problematic in that A only has 0 or 1 in its entry, which is more like a label (i.e., whether there is a link or not) than a real value. In order to tackle this issue, a *Preference-based Non-negative Matrix Factorization (PNMF)* model [Zhang *et al.*, 2015] has been proposed very recently. Instead of approximating the value, it maintains a pairwise preference order for each node. To be specific, we assume that $A_{i,j} = 1$ and $A_{i,k} = 0$. Previous models try to make $F_i F_j^T$ close to 1 and $F_i F_k^T$ close to 0 while the preference based model only expects $F_i F_j^T$ to be larger than $F_i F_k^T$ without considering their actual values.

However, *PNMF* simply separates nodes into two parts, i.e., neighbors and non-neighbors, ignoring the fact that all non-neighbors are not supposed to be treated equally. Inspired by the famous saying "my friend's friend is also my friend", in this paper, we propose a *Locality-based Non-negative Matrix Factorization (LNMF)* model to refine the *PNMF* model by further splitting the non-neighbors into two parts, namely "local non-neighbors" and "distant non-neighbors". We define a "k-degree local network" to distinguish these two kinds of non-neighbors. Given the two assumptions that (1) neighbors are preferred to local non-neighbors and (2) local non-neighbors are preferred to distant non-neighbors, we obtain the objective function by maximizing a product of likelihood. We use the popular stochastic gradient descent as our learning method and provide an efficient sampling strategy. Experiments conducted on real-world datasets show that our *LNMF* model does outperform the state-of-the-art approaches, indicating that our model assumption makes sense.

The rest of this paper is organized as follows. In Section 2, we formally define the community detection problem and introduce some most related work in more details. Our $LNMF$ model and parameter learning process are illustrated in Section 3. Section 4 includes experimental details and interpretation of results, followed by the conclusions in Section 5.

2 Problem Definition and Related Work

In this section, we first define the problem of overlapping community detection and then provide an overview of *Matrix Factorization* based approaches to which our proposed solution belongs. In addition, we will particularly take a look at a *Preference-based Non-negative Matrix Factorization (PNMF)* model on which our proposed model builds.

2.1 Problem Definition

To formally define the problem of *community detection*, we need to have a graph in the first place. We denote the graph as $G(V, E)$, where V is the node set and E is the link or edge set.

Definition 2.1 (Community). *A community C is a subset of V which consists of all nodes with a certain feature.*

Since all nodes in a community share a common feature, they are more likely to make friends with each other. Therefore, a community usually has stronger internal connections and weaker external connection, which matches another definition proposed in [Girvan and Newman, 2002].

Definition 2.2 (Community Detection). *Given a graph $G(V, E)$, community detection aims to find a set of communities $\mathbb{S} = \{C_i | C_i \neq \emptyset, C_i \neq C_j, 1 \leq i, j \leq p\}$, which maximizes a particular objective function f, i.e.,*

$$\arg\max_{\mathbb{S}} f(G, \mathbb{S}), \quad (1)$$

where p is the number of communities.

While traditional community detection finds exhaustive and disjoint communities, i.e., $C_1 \bigcup \cdots \bigcup C_p = V$ and $C_i \bigcap C_j = \emptyset$ for any $i \neq j$, overlapping community detection has no such constraints, which is more general and realistic in real world.

2.2 Matrix Factorization Based Approaches

As we mentioned, matrix factorization is a popular class of methods to deal with overlapping community detection problem. It sets the number of communities in advance and then learns to assign each node to its corresponding communities. The objective can be formally defined as follows.

Definition 2.3 (Overlapping Community Detection via Matrix Factorization). *Given a graph $G(V, E)$ with its adjacency matrix $A \in \{0, 1\}^{n \times n}$, the objective of overlapping community detection via matrix factorization is to find a node-community membership matrix $F \in \mathbb{R}^{n \times p}$ whose entry $F_{u,c}$ represents the weight of node $u \in V$ in community $c \in C$ so that F can minimize a particular loss function l, i.e.,*

$$\arg\min_{F} l(A, FF^T), \quad (2)$$

where n is the number of nodes, p is the number of communities and C is the set of communities. In the end, we obtain the final set of communities C according to F.

Here we would like to review several important work in this class. [Psorakis *et al.*, 2011] is the earliest method which uses the basic $||A - WH^T||$ as its optimization objective. Due to the vague social meaning of W and H, [Wang *et al.*, 2011] refines the objective function to $||A - FF^T||$. [Zhang and Yeung, 2012] extends the matrix factorization model to matrix tri-factorization model by incorporating a community interaction matrix B, which results in a objective function of $||A - FBF^T||$. [Yang and Leskovec, 2013] explicitly defines the probability of having an edge between u and v by a function of F_u and F_v, then generates the likelihood function by fitting the original graph. Though these objective functions are different, they are all based on value-approximation, which is problematic because the $0/1$ value in adjacency matrix is more like a label than a value.

2.3 A Preference-based NMF Model

The *Preference-based Non-negative Matrix Factorization* model [Zhang *et al.*, 2015] is based on the intuitive idea that two nodes are more likely to become friends if they share more common communities. Recall the aforementioned "communities generate links" assumption, this model wants to obtain communities by extracting node preference information from links.

The basic assumption of this model can be represented as

$$r_{u,i} \geq r_{u,j}, \text{if } A_{u,i} = 1 \text{ and } A_{u,j} = 0, \quad (3)$$

where $r_{u,i}$ is the preference of node u on node i, i.e., how much u wants to build a link with i, and $A_{u,i}$ is the corresponding entry in adjacency matrix A.

Notation	Meaning
$G(V, E)$	Graph G with node set V and edge set E
$L_k(u)$	u's k-degree local network in G
$V_k(u)$	node set of $L_k(u)$ except u itself
$S_k(u)$	node set of u's k-degree local non-neighbors
$T_k(u)$	node set of u's k-degree distant non-neighbors
$N^+(u)$	node set of u's neighbors
$N^-(u)$	node set of u's non-neighbors

Table 1: **A summary of notations.**

For each node u, the objective function is to maintain a preference order of all the other nodes given the node-community membership matrix, which is denoted as

$$\mathcal{P}(>_u |F).$$

By applying the results from [Rendle *et al.*, 2009] and denoting the set of u's neighbors as $N^+(u)$ and the set of u's non-neighbors as $N^-(u)$, the above objective function can be simplified to the form of

$$\prod_{i \in N^+(u), j \in N^-(u)} \mathcal{P}(r_{u,i} > r_{u,j}|F). \tag{4}$$

$\mathcal{P}(r_{u,i} > r_{u,j}|F)$ are defined as $\sigma(F_i \cdot F_j^T - F_i \cdot F_k^T)$, where $\sigma(\cdot)$ is the sigmoid function. The rest steps are quite standard so we will not go through details.

3 A Locality-based Non-negative Matrix Factorization (LNMF) Model

In this section, we first define the concept of k-degree locality and then formalize our LPNMF model in the scenario of community detection. We will also briefly talk about the process of parameters learning and provide several candidates of sampling strategy.

3.1 Preliminaries

Definition 3.1 (k-Degree Local Network). *Given an undirected and unweighted graph G, for a node $u \in G$, u's k-degree local network $L_k(u)$ is the subgraph consisting of all nodes whose shortest path length to u is less than or equal to k.*

According to the definition above, $L_0(u)$ consists of only node u, $L_1(u)$ is the subgraph including node u and all its neighbors, $L_\infty(u)$ is the whole graph, etc. We denote the node set of $L_t(u)$ except u itself as $V_t(u)$, where $t = 1, 2, \cdots$.

Now we further define the terms of "local non-neighbors" and "distant non-neighbors".

Definition 3.2 (k-Degree Local Non-neighbors). *Given a k-degree local network $L_k(u)$, the set of k-degree local non-neighbors $S_k(u)$ is defined as $S_k(u) := L_k(u) \backslash L_1(u)$, where $k \geq 1$.*

Definition 3.3 (k-Degree Distant Non-neighbors). *Given a k-degree local network $L_k(u)$, the set of k-degree distant non-neighbors $T_k(u)$ is defined as $T_k(u) := L_\infty(u) \backslash L_k(u)$, where $k \geq 1$.*

We can see that when $k = 1$, $S_k(u) = \emptyset$ and $T_k(u) = N^-(u)$. In this case, our model degrades to the *PNMF* model. When $k \geq 2$, our model will have a new class of nodes in preference system. Thus, our model is actually a generalization of the *PNMF* model.

A summary of notations is shown in Table 1. Four simple propositions can be drawn from the above notations.

Proposition 3.1. $V_k(u) = N^+(u) \bigcup S_k(u)$.

Proposition 3.2. $N^+(u) \bigcap S_k(u) = \emptyset$.

Proposition 3.3. $N^-(u) = S_k(u) \bigcup T_k(u)$.

Proposition 3.4. $S_k(u) \bigcap T_k(u) = \emptyset$.

3.2 Model Assumption

Recall the basic assumption of *PNMF* in Equation 3. Incorporating the concept of k-degree local network, we can exploit k-degree local non-neighbors to enhance the old model assumption. The new model assumption for our *LNMF* model can be represented as

$$r_{u,i} \geq r_{u,j}, r_{u,j} \geq r_{u,d}, i \in N^+(u), j \in S_k(u), d \in T_k(u), \tag{5}$$

where $r_{u,p}$ is still the preference of node u on node p. It means (1) neighbors are preferred to local non-neighbors; (2) local non-neighbors are preferred to distant non-neighbors. These two assumptions are quite intuitive. Notice that when $k = 1$, the new model assumption degrades to the old one.

We also adopt two independence assumptions, i.e., node independence and pair independence assumptions, listed in [Zhang *et al.*, 2015] in order to formalize our model.

- **Node independence**. The preference order of each node is independent with that of any other node. There will be a link between u and v if and only if u prefers to build relationship with v and symmetrically v prefers to build relationship with u.
- **Pair independence**. For a fixed node i, its preference on j and k is independent with its preference on u and v when $j, u \in N^+(i)$ and $k, v \in N^-(i)$.

3.3 Model Formulation

Given the above model assumptions, we are ready to present our *LNMF* model in a formal way. Since nodes are independent of each other, we can consider one node at first.

For each node u, the optimization criterion is to maximize the likelihood of preference order which can be represented as a product of pairwise preferences, i.e.,

$$\begin{aligned} \prod_{i,j \in V_k(u)} & [\mathcal{P}(r_{u,i} \geq r_{u,j}|F)^{\delta(u,i,j)} \\ & (1 - \mathcal{P}(r_{u,i} \geq r_{u,j}|F))^{1-\delta(u,i,j)}] \cdot \\ \prod_{j,d \in N^-(u)} & [\mathcal{P}(r_{u,j} \geq r_{u,d}|F)^{\xi(u,j,d)} \\ & (1 - \mathcal{P}(r_{u,j} \geq r_{u,d}|F))^{1-\xi(u,j,d)}], \end{aligned} \tag{6}$$

where $\delta(\cdot)$ and $\xi(\cdot)$ are two indicator functions that

$$\delta(u, i, j) = \begin{cases} 1 & \text{if } i \in N^+(u) \text{ and } j \in S_k(u), \\ 0 & \text{otherwise} \end{cases}$$

Algorithm 1 Community Detection via *LNMF*

```
Input: $G$, the adjacency matrix of original graph
Output: $F$, the node-community membership matrix
initialize $F$
compute initial loss
repeat
    for $num\_samples = 1$ to $sample\_size$ do
        sample $(u, i, j, d)$ according to Algorithm 2
        for each entry $\Theta$ in $F_u, F_i, F_j$ and $F_d$ do
            update $\Theta$ according to Equation (10)
            $\Theta \leftarrow \max(\Theta, 0)$
        end for
    end for
    compute loss
until convergence or $max\_iter$ is reached
```

and

$$\xi(u, j, d) = \begin{cases} 1 & \text{if } j \in S_k(u) \text{ and } d \in T_k(u), \\ 0 & \text{otherwise} \end{cases}.$$

Recall the four propositions in preliminaries that $V_k(u)$ and $N^-(u)$ can be split into two disjoint sets with different levels of preference. Following the scheme argued in [Rendle *et al.*, 2009; Zhao *et al.*, 2014], we can simplify Equation 6 to

$$\frac{\sum_{i \in N^+(u), j \in S_k(u)} \mathcal{P}(r_{u,i} \geq r_{u,j}|F)}{|N^+(u)| \cdot |S_k(u)|} + \frac{\sum_{j \in S_k(u), d \in T_k(u)} \mathcal{P}(r_{u,j} \geq r_{u,d}|F)}{|S_k(u)| \cdot |T_k(u)|}. \tag{7}$$

Applying the sigmoid function $\sigma(x) := \frac{1}{1+e^{-x}}$ to interpret $\mathcal{P}(r_{u,i} \geq r_{u,j}|F)$, i.e., $\mathcal{P}(r_{u,i} \geq r_{u,j}|F) = \sigma(\hat{x}(u,i) - \hat{x}(u,j))$, we sum up the log-likelihood functions of all nodes:

$$\sum_u [\sum_{i \in N^+(u), j \in S_k(u)} \ln \sigma(\hat{x}(u,i) - \hat{x}(u,j)) + \lambda(u) \cdot \sum_{j \in S_k(u), d \in T_k(u)} \ln \sigma(\hat{x}(u,j) - \hat{x}(u,d))], \tag{8}$$

where $\hat{x}(u,i) := F_u \cdot F_i^T$ can be regarded as the correlation between u and i, and $\lambda(u) := \frac{|N^+(u)|}{|T_k(u)|}$ can be regarded a coefficient of local influence.

In the end, to prevent our model from overfitting, we add a regularization term $reg(F) = ||F||_F^2$, which is the Frobenius norm of the node-community membership matrix. The final objective function l is

$$l(F) = \sum_u [\sum_{i \in N^+(u), j \in S_k(u)} \ln \sigma(\hat{x}(u,i) - \hat{x}(u,j)) + \lambda(u) \cdot \sum_{j \in S_k(u), d \in T_k(u)} \ln \sigma(\hat{x}(u,j) - \hat{x}(u,d))] - \lambda_r reg(F), \tag{9}$$

where λ_r is a regularization coefficient.

3.4 Parameter Learning

As an efficient and widely-used paradigm for parameter learning, *stochastic gradient descent (SGD)* is employed as our learning algorithm. Distinguished from the traditional batch gradient descent which computes Equation 9 in each iteration, *SGD* only picks a small number of random samples to perform update. In our case, a sample is a (source, neighbor, local non-neighbor, distant non-neighbor) quadruple. Mathematically, we calculate the derivative of our final objective function l by

Algorithm 2 Sampling Strategy

```
Input: $G$, the adjacency matrix of original graph
Output: $(u, i, j, d)$, a quadruple to perform a step in
    stochastic gradient descent
sample node $u$ from $V$ uniformly at random
sample node $i$ from $N^+(u)$ uniformly at random
sample node $j$ from $S_k(u)$ uniformly at random
sample node $d$ from $T_k(u)$ uniformly at random
```

$$\Theta^{t+1} = \Theta^t + \alpha \frac{\partial l}{\partial \Theta}, \tag{10}$$

where is Θ can be any entry of the node-community membership matrix F. For the non-negative constraints, we apply a projected gradient method proposed in [Lin, 2007], which maps the parameter vector back to the nearest point in projected space, in our case, the non-negative space.

The whole process is described in Algorithm 1. Let sample size be t. The time complexity of each iteration is $O(tp)$ and the space complexity is $O(np)$, where n is the number of nodes and p is the number of communities.

3.5 Sampling Strategy and Other Issues

Due to the nature of stochastic gradient descent, sampling strategy matters to both running time and performance. More than what *PNMF* did, we need to sample a set of quadruples for each learning step. The process is described in Algorithm 2.

For the sampling of j, we need to pre-process the whole graph to record a set of local nodes of each u in the graph. By using the fact that $N^-(u) = S_k(u) \bigcup T_k(u)$, we keep sampling a random node until we get a node neither in $N^+(u)$ nor in $S_k(u)$ and let d be this node.

Moreover, there are several remaining issues to be discussed.

- **The number of communities.** The nature of matrix factorization needs us to set the number of communities which are unknown in advance. A cross-validation paradigm is used. In details, we reserve 10% of nodes as validation set at first. After learning the node-community membership matrix F, we compute the sum of log-likelihood function for all nodes in validation set via Equation 7. Since the computational cost is huge for cross-validation, only a small set of quadruple will be sampled.
- **The community membership threshold.** Obtaining the node-community membership matrix F is still one step away from getting the final node-community correspondence. We need to set a threshold to decide whether a community accepts a node. We employ the approach

Dataset	V	E
Dolphins	62	159
Les Misérables	77	254
Books about US politics	105	441
Word adjacencies	112	425
American college football	115	613
Coauthorship in network science	1,589	2,742

Table 2: **Statistics of six Newman's datasets. V: number of nodes, E: number of links.**

in [Zhang *et al.*, 2015] to deal with this issue. In short words, we set a probability threshold to $\mathcal{P}(r_{u,i} \geq r_{u,j}|F)$ and use the sigmoid function to reversely compute the lower bound of community membership weight assuming that u, i share one community but u, j do not share any community.

- **The convergence criterion.** First, we randomly generate a subset of quadruples to be our loss sample and compute initial loss on this set according to Equation 9. After each iteration, we need to compute loss again and we stop stochastic gradient descent when the absolute difference between current loss and previous loss is smaller than a very small percentage, say ϵ, of initial loss.

4 Experiments

In this section, we compare our $LNMF$ model with both classic and state-of-the-art overlapping community detection methods on various real-world datasets. We will show our experimental results with two metrics, namely modularity and F_1 score, and have a brief discussion.

4.1 Data Description

Six benchmark networks collected by Newman[1] are used as our datasets. These networks are relatively small and have no ground-truth communities. Basic information of these datasets can be found in Table 2.

Moreover, we choose three large networks with ground-truth communities collected by SNAP[2] [Yang and Leskovec, 2012] to test the scalability of our model. These networks are of different types:

- **YouTube** dataset: a social network of a video-sharing web site.
- **DBLP** dataset: a collaboration network of research paper authors in computer science.
- **Amazon** dataset: a products co-purchasing network based on Customers Who Bought This Item Also Bought feature of the Amazon website.

Simple statistics for these three datasets are shown in Table 3.

4.2 Experimental Setup

We conduct our experiments on a computer with a Xeon 2.60GHz CPU and 64GB memory.

Dataset	V	E	C	U
DBLP	317k	1.0M	2.5k	429.8
Amazon	335k	926k	49k	100.0
YouTube	1.1M	3.0M	30k	9.7

Table 3: **Statistics of three SNAP datasets. V: number of nodes, E: number of links, C: number of ground-truth communities, U: average number of nodes per community.**

Comparison methods. We select both classic and state-of-the-art methods to compare with our model. The latter four are *Non-negative Matrix Factorization (NMF)* based models.

- **SCP** [Kumpula *et al.*, 2008] accelerates the original **CP** method [Palla *et al.*, 2005] in a sequential manner. We set k to be 4 or 5 when finding k-cliques.
- **LC** [Ahn *et al.*, 2010] clusters link instead of node to get overlapping communities. We ignore all communities with only one or two nodes since they are meaningless.
- **BNMF** [Psorakis *et al.*, 2011] is one of the earliest work which applies *MF* into community detection. Squared loss is used as loss function.
- **BNMTF** [Zhang and Yeung, 2012] incorporates a community interaction matrix into the classic *MF* to become a *Matrix Tri-Factorization* model. Squared loss is used as loss function.
- **BigCLAM** [Yang and Leskovec, 2013] is claimed by its authors as a scalable model. It can search for the best number of communities given a range.
- **PNMF** [Zhang *et al.*, 2015] is the model on which our proposed model builds.

Evaluation metrics.

- **Modularity**. We use the classic modularity as our metric for Newman's datasets. Modularity Q is defined as

$$Q = \frac{1}{2m} \sum_{u,v \in V} (A_{u,v} - \frac{d(u)d(v)}{2m})|C_u \cup C_v|,$$

  where m is the number of links, V is the node set, A is the adjacency matrix, $d(u)$ is the degree of node u, and C_u is the set of communities to which node u belongs. This definition indicates that for each node pair (u, v) which shares communities, its contribution to modularity is positive if u, v are linked and is negative otherwise. It matches our intuition that nodes inside one community tends to build links with each other.

- $\mathbf{F_1}$ **score**. For SNAP datasets with ground-truth communities, F_1 score is obviously one of the best measurements. The F_1 score of a detected community S_i is defined as the harmonic mean of precision(S_i) and recall(S_i), where precision(S_i) and recall(S_i) are defined as

$$\text{precision}(S_i) = \max_j \frac{C_j \bigcap S_i}{|C_j|},$$

[1] http://www-personal.umich.edu/ mejn/netdata/

[2] http://snap.stanford.edu/data/

Dataset	SCP	LC	BNMF	BNMTF	BigCLAM	PNMF	LNMF(RI)
Dolphins	0.305	0.654	0.507	0.507	0.423	0.979	**1.086(10.9%)**
Les Misérables	0.307	0.773	0.125	0.103	0.540	1.103	**1.184(7.3%)**
Books about US politics	0.496	0.851	0.461	0.492	0.529	0.864	**1.270(47.0%)**
Word adjacencies	0.071	0.271	0.254	0.268	0.231	0.668	**0.701(4.9%)**
American College football	0.605	0.891	0.558	0.573	0.518	1.049	**1.235(17.7%)**
Coauthorships in network science	0.729	0.956	0.661	0.741	0.503	1.657	**2.310(39.4%)**

Table 4: **Comparison in terms of modularity. RI: Relative Improvement over PNMF.**

Dataset	BigCLAM	PNMF	LNMF(RI)
DBLP	0.039	0.098	**0.107(9.2%)**
Amazon	0.044	0.042	**0.048(11.4%)**
YouTube	0.019	**0.060**	0.057(0.0%)

Table 5: **Experimental results on SNAP datasets in terms of $\mathbf{F_1}$ score. RI: Relative Improvement over PNMF.**

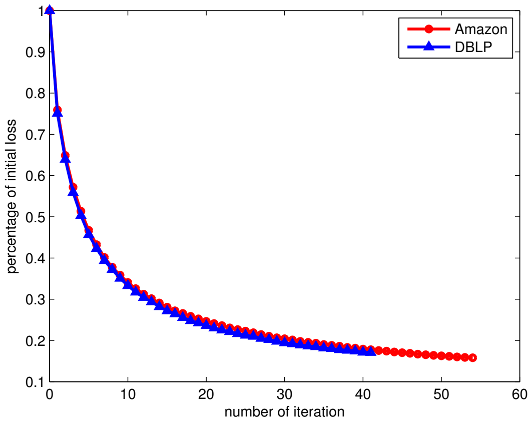

Figure 2: **Convergence speed of learning algorithm**

and

$$\text{recall}(S_i) = \max_j \frac{C_j \bigcap S_i}{|S_i|},$$

where C_j is the node set of a ground-truth community. The average F_1 score for the set of detected communities S is

$$\overline{F_1}(S) = \frac{1}{|S|} \sum_{S_i \in S} F(S_i).$$

Setting the k. Remember that if we set $k = 1$ in k-degree local network, our model will degrade to the *PNMF* model. According to our observation on several datasets, if k is set to be larger than 2, the average number of common communities two nodes in a k-degree local network share is not significantly larger than that two random nodes in the whole network share. Thus, we set k to be 2, which means only a friend's friends are considered as local non-neighbors.

4.3 Results

We set the regularization coefficient to be 0.5 and the convergence parameter ϵ to be 0.001 for all experiments. The sample size t is determined according to data size. For Newman's datasets, we set $t = m$, i.e., the number of links. For SNAP datasets, we set $t = 10\sqrt{n}$ in order to finish one iteration without taking too much time, where n is the number of nodes. The maximum times of iteration is set to 100, though in fact all datasets converge before reaching the limit.

Table 4 shows the performance of our *LNMF* model on Newman's datasets. From the results we find that under the metric of modularity, our *LNMF* model outperforms all baseline methods on all datasets.

Table 5 shows the our experimental results on SNAP datasets. The other baselines methods are not listed here since none of them can finish all three datasets in time. This fact can reflect the scalability of our *LNMF* model to some extend. It can be seen that our model outperforms *BigCLAM* on all datasets and has improvement over $PNMF$ on two of three datasets. For *YouTube*, we find its community formation pattern quite random due to small size of communities and large variety of users. In other words, our model assumption does not fit the community pattern of this dataset so well. This may explain why *LNMF* fails to have improvement on it. The running time of one iteration is about one or two hours for *DBLP* and *Amazon*. For *YouTube*, it takes about four to five hours to finish an iteration.

The convergence speed of our learning algorithm on *Amazon* and *DBLP* is illustrated in Figure 2. A point in the figure represents the ratio of current loss to initial loss after i-th iteration. The results show that our *LNMF* can converge within a fair number of iterations. In fact, if we do not consider the regularization term, the final losses of both datasets are less than 10% of the initial loss.

5 Conclusion And Future Work

In this paper, we propose a *Locality-based Non-negative Matrix Factorization* model to improve the performance of existing work on overlapping community detection. Our *LNMF* model is based on a pairwise preference learning scheme. We exploit local area around a node formally defined as k-degree local network to enhance the previous preference system. In details, we extend a two-level preference system which only distinguish neighbors and non-neighbors to a three-level preference system which split the set of non-neighbors into local non-neighbors and distant non-neighbors. Experiments on several real-world datasets including large ones with ground-truth communities show that this extension can indeed improve the quality of overlapping community detection.

Our model can be further generalized by extending the preference system from three-level to n-level. Mathematically, according to the notations in Section 3.1, now $P_k(u) := L_k(u) \backslash L_{k-1}(u)$ for each $k \geq 1$ is a node set of a particular preference level of node u and u's preference on $P_i(u)$ is larger than its preference on $P_j(u)$ if $i > j$. For model formulation, we only need to modify a few spots to make it a more general one. However, our learning algorithm is not efficient enough. Specifically, a natural extension of our sampling strategy suffers two main problems: (1) for different nodes, the upper limit of k is different; (2) for each node u, pre-processing the whole graph to record node sets of all preference levels is equal to a breath-first search starting from u, which is too expensive for large-scale networks. We plan to first conduct empirical study to see whether this extension makes sense. If it does, we will try to find solutions for the problems mentioned above.

Acknowledgement

The work described in this paper was fully supported by the National Grand Fundamental Research 973 Program of China (No. 2014CB340401 and No. 2014CB340405), the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CUHK 413213 and CUHK 415113), and Microsoft Research Asia Regional Seed Fund in Big Data Research (Grant No. FY13-RES-SPONSOR-036).

References

[Ahn *et al.*, 2010] Yong-Yeol Ahn, James P Bagrow, and Sune Lehmann. Link communities reveal multiscale complexity in networks. *Nature*, 466(7307):761–764, 2010.

[Coscia *et al.*, 2012] Michele Coscia, Giulio Rossetti, Fosca Giannotti, and Dino Pedreschi. Demon: a local-first discovery method for overlapping communities. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 615–623. ACM, 2012.

[Fortunato, 2010] Santo Fortunato. Community detection in graphs. *Physics Reports*, 486(3):75–174, 2010.

[Gavin *et al.*, 2002] Anne-Claude Gavin, Markus Bösche, Roland Krause, Paola Grandi, Martina Marzioch, Andreas Bauer, Jörg Schultz, Jens M Rick, Anne-Marie Michon, Cristina-Maria Cruciat, et al. Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature*, 415(6868):141–147, 2002.

[Girvan and Newman, 2002] Michelle Girvan and Mark EJ Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.

[Kumpula *et al.*, 2008] Jussi M Kumpula, Mikko Kivelä, Kimmo Kaski, and Jari Saramäki. Sequential algorithm for fast clique percolation. *Physical Review E*, 78(2):026109, 2008.

[Lin, 2007] Chih-Jen Lin. Projected gradient methods for nonnegative matrix factorization. *Neural computation*, 19(10):2756–2779, 2007.

[McAuley and Leskovec, 2012] Julian McAuley and Jure Leskovec. Learning to discover social circles in ego networks. In *NIPS*, volume 272, pages 548–556, 2012.

[Newman, 2001] Mark EJ Newman. The structure of scientific collaboration networks. *Proceedings of the National Academy of Sciences*, 98(2):404–409, 2001.

[Palla *et al.*, 2005] Gergely Palla, Imre Derényi, Illés Farkas, and Tamás Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814–818, 2005.

[Psorakis *et al.*, 2011] Ioannis Psorakis, Stephen Roberts, Mark Ebden, and Ben Sheldon. Overlapping community detection using bayesian non-negative matrix factorization. *Physical Review E*, 83(6):066114, 2011.

[Rendle *et al.*, 2009] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 452–461. AUAI Press, 2009.

[Wang *et al.*, 2011] Fei Wang, Tao Li, Xin Wang, Shenghuo Zhu, and Chris Ding. Community discovery using nonnegative matrix factorization. *Data Mining and Knowledge Discovery*, 22(3):493–521, 2011.

[Whang *et al.*, 2013] Joyce Jiyoung Whang, David F Gleich, and Inderjit S Dhillon. Overlapping community detection using seed set expansion. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, pages 2099–2108. ACM, 2013.

[Yang and Leskovec, 2012] Jaewon Yang and Jure Leskovec. Defining and evaluating network communities based on ground-truth. In *Proceedings of the ACM SIGKDD Workshop on Mining Data Semantics*, page 3. ACM, 2012.

[Yang and Leskovec, 2013] Jaewon Yang and Jure Leskovec. Overlapping community detection at scale: a nonnegative matrix factorization approach. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 587–596. ACM, 2013.

[Zhang and Yeung, 2012] Yu Zhang and Dit-Yan Yeung. Overlapping community detection via bounded nonnegative matrix tri-factorization. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 606–614. ACM, 2012.

[Zhang *et al.*, 2015] Hongyi Zhang, Irwin King, and Lyu Michael R. Incorporating implicit link preference into overlapping community detection. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*. ACM, 2015.

[Zhao *et al.*, 2014] Tong Zhao, Julian McAuley, and Irwin King. Leveraging social connections to improve personalized ranking for collaborative filtering. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 261–270. ACM, 2014.