DEVELOPMENT OF ADVANCED DATA MINING ALGORITHMS FOR THE ANALYSIS OF DIRECTED NETWORKS

by

ALİ TOSYALI

A dissertation submitted to the

School of Graduate Studies

Rutgers, The State University of New Jersey

In partial fulfillment of the requirements

For the degree of

Doctor of Philosophy

Graduate Program in Industrial and Systems Engineering

Written under the direction of

Myong K. Jeong

And approved by

_____

_____

_____

_____

_____

New Brunswick, New Jersey

MAY, 2019

ABSTRACT OF THE DISSERTATION

Development of Advanced Data Mining Algorithms for the Analysis of Directed Networks

By ALİ TOSYALI

Dissertation Director:

Myong K. Jeong

There are many systems which can be represented as a network, where the parts of the system are nodes and the connections between the parts are the edges. Researchers proposed numerous different network types such as internet networks, citation networks, and transportation networks. Also, numerous analysis tools have been introduced to investigate the structures and pattern of connections of networks. However, existing research is mostly focused on undirected and static networks and analysis of directed dynamic networks, especially citation networks, has received little attention from the researchers.

In this dissertation, we present new methodologies for the analysis of directed networks. We first propose an anomaly (outlier) detection technique based on nonnegative matrix factorization for directed patent citation network (PCN). We have developed a clustering method based on NMF, and an anomaly score function that exploits the clustering result. The proposed outlier ranking method leverages the patent-level analysis as well as group-level analysis in order to measure the graph-based outlierness of a patent. We validate our proposed anomaly ranking methods using small artificial datasets. We then conduct experiments using real-world patent citation network. Results reveal that the proposed outlier ranking and detection method outperforms

existing approaches.

Secondly, we present a regularized asymmetric nonnegative matrix factorization (RANMF) algorithm for clustering in directed networks. The proposed algorithm assumes that if two nodes are similar to each other in the original basis, their representatives in new basis should be close to each other. Therefore, similar nodes appear in the same cluster. The proposed algorithm is for clustering nodes in a given directed network under the guidance of prior similarity information of the network and SVD-based initialization. We also provide proof of the convergence of RANMF algorithm and real-world experiments to show its performance. The experiments reveal that RANMF algorithm is a better solution for clustering in directed networks compared to other clustering algorithms.

Finally, we develop a time-aware ranking method for the identification of important and influential patents in dynamic patent citation network. While the existing ranking methods fail to distinguish the citing and cited patent for the importance of cited patent, the proposed ranking method successfully distinguish them by exploiting the time information of not only citing patent but also the time information of cited patent. We present the performance of our method on real-world patent citation data and compare it to other ranking metrics. The results reveal that our proposed method not only successfully rank the patents in importance but also successfully identifies the influential patents in a dynamic patent citation network.

Acknowledgements

I acknowledge and thank my advisor, Dr. Myong K. Jeong, for his continuous support, guidance, and patience during my PhD. I appreciate his motivation and enthusiasm. He is a great mentor and his encouragement to think critically and smartly made this accomplishment possible.

I would also thank my dissertation committee members, Professors Susan Albin, Hoang Pham, Weihong (Grace) Guo, and Jie Gong for their support and valuable time. I thank other members of my research group for their constructive comments and engaging discussions in this research.

Most importantly, I love and would like to thank my beautiful, strong, patient, and encouraging wife, Sule Tosyali, who always supports me. She always believed in me and encouraged me to achieve my goals. Without her support and love, I could not have this accomplishment. I am also thankful to my son, Mert, who makes me cheer and fills my life with great joy.

I am also grateful to my families in Turkey - The Tosyali and The Tatlilioglu families for their concrete support - you were always with me and your support is so apparent and appreciated.

Dedication

To my loving Family – The Tosyali and The Tatlilioglu Families.

To my beautiful Wife, who fills my life with great joy.

With special gratitute to Mert. You have been a gift from the beginning.

Table of Contents

Abstract ii

Acknowledgements iv

Dedication v

List of Tables ix

List of Figures xi

1. Introduction 1

1.1. Overview 1

1.2. Dissertation outline 4

2. New Node Anomaly Detection Algorithm based on Nonnegative Matrix Factorization for Directed Citation Networks 5

2.1. Introduction 5

2.2. Nonnegative matrix factorization 7

2.3. New scoring method for anomaly detection in directed graph 9

2.3.1. Citation matrix 9

2.3.2. ANMF with citation matrix 11

2.3.3. Proposed scoring method for anomaly detection 12

2.3.4. Initialization based on modified SVD 14

2.3.5. Complexity analysis 15

2.3.6. An illustrative example 16

2.4. Experimental results 18

2.4.1. Artificial dataset: 14-node network 18

2.4.2. Real-world dataset: U.S. patent citation network 22

2.4.3. Parameter sensitivity 23
2.5. Conclusion 23

3. Regularized Asymmetric Nonnegative Matrix Factorization for Clustering in Directed Networks 29
3.1. Introduction 29
3.2. Nonnegative matrix factorization on clustering 31
3.3. Regularized asymmetric nonnegative matrix factorization 33
3.3.1. Optimization problem 34
3.3.2. SVD based initialization 37
3.3.3. An illustrative example 39
3.4. Experiments 41
3.4.1. Patent citation network 42
3.4.2. World wide knowledge base datasets 45
3.4.3. LFR synthetic graphs 55
3.5. Conclusion 57

4. A New Time-aware Ranking Method for Patents in Dynamic Patent Citation Network 58
4.1. Introduction 58
4.2. Patent citation network 60
4.3. Proposed time-aware influence measure 63
4.3.1. Illustrative example 66
4.4. Case study 68
4.4.1. Metrics 70
4.4.2. Performance measures 70
4.4.3. Results 71
4.4.4. Conclusion 74

5. Concluding Remarks and Future Research 75

5.1. Concluding remarks 75
5.2. Future research 77

Appendix A. Proof of Proposition 1 78

Appendix B. Derivation of Regularization Term 79

Appendix C. Derivation of Multiplicative Updating Rules 80

Appendix D. Proof of Theorem 1 82

List of Tables

2.1. Notation summary 8

2.2. Outlier score and rank of 5-node illustrative example 18

2.3. Comparison of the proposed algorithm, OutRank, and OddBall using 14-node patent citation network 22

2.4. Comparison results of the proposed, OutRank and OddBall algorithms on real-world U.S. Patent Citation Network 23

3.1. RANMF clustering results for 13-node sample graph 41

3.2. Comparison results of algorithms on PCN with different r values in terms of distance-based quality function. Rnd and SVD stand for random initialization and SVD-based initialization, respectively. 44

3.3. Comparison results of algorithms on PCN with different r values in terms of DB index. 44

3.4. Comparison results of clustering methods using WebKB datasets. $|V|$ is the number of nodes, $|E|$ is the number of edges, and r is the number of clusters. 52

3.5. Comparison results of clustering methods using WebKB datasets. 52

3.6. Comparison results of clustering methods on LFR graphs. $\mu = 0.1$, $|V| = 1000$, $|E| = 15662$, and $r = 32$ 56

3.7. Comparison results of clustering methods on LFR graphs. $\mu = 0.3$, $|V| = 1000$, $|E| = 15164$, and $r = 31$ 56

3.8. Comparison results of clustering methods on LFR graphs. $\mu = 0.5$, $|V| = 1000$, $|E| = 15249$, and $r = 33$ 56

4.1. Influence scores of citations in sample graph. 67

4.2. Weights of patents based on their age in years. 67

4.3. Ranking result of patents in sample graph. 68
4.5. Comparison results of metrics in terms of Spearman correlation and recommendation intensity scores. 72
4.6. Comparison results of metrics in terms of recommendation intensity for varying k values. 72
4.4. Top 10 patents identified in patent citation data by our proposed ranking method. 74

List of Figures

2.1. Representation of direct and indirect citations. 10

2.2. Proposed node anomaly scoring flowchart 17

2.3. Network structure of artificial dataset 19

2.4. Four clusters obtained based on the matrix factorization results. Each color and shape represent a different cluster. 20

2.5. Sensitivity of our method to varying α values in terms of accuracy on US PCN dataset. 24

2.6. Sensitivity of our method to varying α values in terms of F1 score on US PCN dataset. 25

2.7. Sensitivity of our method to varying β values in terms of accuracy on US PCN dataset. 26

2.8. Sensitivity of our method to varying β values in terms of F1 score on US PCN dataset. 27

3.1. Representatives of nodes in new basis. 10 nodes and 2 clusters. Each shape represents different cluster. 34

3.2. 13-node sample graph 40

3.3. Network structure of US patent citation dataset. 43

3.4. Network structure of Cornell dataset. 47

3.5. Network structure of Texas dataset. 48

3.6. Network structure of Washington dataset. 49

3.7. Network structure of Wisconsin dataset. 50

3.8. Convergence curves of log of the objective function of RANMF algorithm for PCN and WebKB datasets. 51

3.9. Accuracy score of RANMF using λ from 0.1 to 5000 using Cornell dataset. 53

3.10. Accuracy score of RANMF using λ from 0.1 to 5000 using Texas dataset. 53
3.11. Accuracy score of RANMF using λ from 0.1 to 5000 using Washington dataset. 54
3.12. Accuracy score of RANMF using λ from 0.1 to 5000 using Wisconsin dataset. 54
4.1. Raw patent data (partially shown). 61
4.2. Graphical representation of a citation between two patents. 62
4.3. Graphical representation of a citations in raw patent data in Figure 4.1. 62
4.4. Evolution of a sample patent citation network over the time interval $[0, T]$. 63
4.5. Difference in the effect of citation in terms of recency of citing patent. . 65
4.6. 4-node sample graph. 67
4.7. Network structure of patent citation dataset with 4241 patents and 18385 citations. 69
4.8. Network structure of top 10 identified patents (shown partially). 73

Chapter 1

Introduction

1.1 Overview

Network analysis is a collection of tools for investigating systems of interests which can be represented by a network. Researchers introduced many different types of networks (Newman, 2018). Many researchers have proposed methods to analyze undirected networks. However, analysis of directed networks has gained little attention from the researchers. For example, patent citation networks are one of the commonly known directed networks and analysis of directed patent citation networks is very important for decision makers to be able to discover technology opportunities.

One of the crucial tasks in directed network analysis is finding outliers in a given network. Moonesinghe and Tan (2006) propose an algorithm which determines the outlierness of nodes in a given network based on the values of the dominant eigenvector of the transition probability matrix. Transition probabilities are obtained by transforming the edge weights of the underlying graph data. Xu et al. (2007) introduce a cluster-based algorithm to identify the node outliers, which groups the nodes in a network based on their similarities to each other. Akoglu et al. (2010) present the OddBall algorithm to find the outliers in the network, which identifies the outliers by finding the nodes that do not follow the patterns.

Another important task for directed network analysis is partitioning the nodes into some sort of logical groupings, which is called as clustering in data mining and machine learning communities. Researchers proposed numerous methods to identify clusters in a given network (Kernighan and Lin, 1970, Newman, 2006). However, there is little research which has been focused on clustering in directed networks. Recently nonnegative matrix factorization (Lee and Seung, 2001) have become popular as a clustering

technique, since it is fairly easy to interpret and its relationship to k-means has been studied (Ding et al., 2005). Wang et al. (2011) propose a community detection algorithm based on nonnegative matrix factorization techniques for clustering in directed networks. The algorithm, which is called as asymmetric nonnegative matrix factorization (ANMF), identifies the clusters by factorizating the simple adjacency matrix of a given directed network.

In directed network analysis, another crucial task is ranking nodes based on their importance. Therefore, researchers proposed various types of importance measures for networks. Some of these methods are also known as centrality metrics (e.g., degree, eigenvector). These importance measures are for static networks, whose topology do not change over time. However, some of the networks are changing over time such as patent citation networks (PCN). In PCN, new patents and new edges can emerge over time. To identify important nodes in this kind of dynamic networks, researchers present new importance measures. For example, Walker et al. (2007) propose CiteRank algorithm, which is a modified version of PageRank algorithm (Page et al., 1999) to citation networks. CiteRank simulates the dynamics of a large number of researchers and approximates the traffic to an individual node. Lerman et al. (2010) present an algorithm, which counts the number of direct and consecutive indirect edges to a particular node. Ghosh et al. (2011) introduce another importance measure for dynamic networks, which counts the number of direct and indirect edges to a node by giving less weight to older edges.

In this dissertation, we present new methodologies for the analysis of directed networks. We first propose SVD initialized asymmetric nonnegative matrix factorization for node anomaly detection in directed patent citation networks. Outlier detection is a crucial task for network data analysis, which identifies abnormal entities that deviate from the rest of the dataset. Ranking in outlierness is often used for identifying abnormal nodes in directed citation networks containing citation relationship among nodes. A challenging issue in outlier ranking is how to leverage the rich graph data of complex citation networks. To address this challenge, we propose a cluster-based outlier score function to identify outliers in citation networks based on nonnegative

matrix factorization (NMF). We first represent the citation data as a directed graph and cluster the directed graph into logical groupings of nodes using NMF. Based on the clustering results, we obtain the outlier score and ranking for each node using the proposed outlier scoring function. The proposed method leverages the direct and indirect citation links between nodes to measure the graph-based outlierness. We validate the proposed outlier ranking method using small artificial dataset and the real-world U.S. patent data.

Secondly, we present regularized asymmetric nonnegative matrix factorization for clustering in directed networks. There are various methods to cluster nodes in undirected networks, however, little is known about clustering in directed networks. We propose a regularized asymmetric nonnegative matrix factorization (RANMF) algorithm for clustering in directed networks. In a given directed network, the RANMF exploits the pairwise similarity of nodes to make close nodes belong to the same cluster under the guidance of prior information of the network. We also prove the convergence of the RANMF algorithm and provide real-world experiments to show its performance. The experimental results show the superiority of our RANMF algorithm in terms of several clustering validity indices.

Finally, we develop a new dynamic importance measure for dynamic patent citation networks to identify influential patents. Ranking patents is a crucial task in patent analysis as it relates to evaluating the firms' policy regarding R&D processes, assessing the level of technology development in a specific area, and estimating the technological strengths and weaknesses of competitor firms. Existing patent ranking methodologies either do not take advantage of network analysis tools or fail to distinguish the effect of citing and cited patents for the importance of cited patent. The proposed method exploits the time information of both citing and cited patents and dynamic characteristics of patent citation network to distinguish the effects of patents on the importance of a particular patent. The experimental results using a real-world patent citation data show that our proposed ranking method outperforms other metrics.

1.2 Dissertation outline

The rest of the dissertation is structured as follows: we first present a node anomaly detection algorithm based on NMF to identify outlier patents in a directed patent citation network. Second, we introduce a novel NMF-based clustering algorithm for directed networks. We then propose a time-aware ranking methodology for the identification of influential patents in a dynamic patent citation network. Finally, we present the contributions of the dissertation and discuss future research opportunities.

Chapter 2

New Node Anomaly Detection Algorithm based on Nonnegative Matrix Factorization for Directed Citation Networks

2.1 Introduction

Outlier detection aims to identify unusual entities that deviate from the rest of the dataset, which has been researched within diverse areas and application domains (Codetta-Raiteri and Portinale, 2015, Duan et al., 2009, Banker et al., 2017). Recently, there have been numerous researches on graph mining to investigate the patterns in networked systems such as social networks, transportation networks, and patent citation networks (Xu et al., 2007, Holder and Cook, 2009, Zou et al., 2010, Kang et al., 2013, Džamić et al., 2017). Graph mining approaches analyze data represented as a graph, which consist of nodes and edges, to have better understanding of the structure and behaviors in data. The goal of outlier ranking is to score and rank objects to the degree of deviation from the majority of dataset in a graph data. In general, outlier ranking in a graph data corresponds to identifying exceptional nodes, edges, or clusters (or subgraphs). In this chapter, we focus on ranking nodes to identify interesting or exceptional nodes in a citation network.

Patent citation provides an effective explanation for how new technologies are related to other works (Michel and Bettels, 2001, Newman, 2018). These relationships can be presented in a patent citation network (PCN) with nodes as patents and directed edges as citation between patents. In patent data analysis, outlier patent detection is often used as a starting point to investigate possible technological opportunities including the identification of potentially promising trends.

There have been growing attempts to detect outlier nodes in graphs. The OutRank

algorithm transforms the edge weights of the underlying graph into transition probabilities, where weighted edges represent the similarities between node objects (Moonesignhe and Tan, 2006). The OutRank determines the outlierness of each node object based on the values of the dominant eigenvector of the transition probability matrix. Xu et al. (2007) present a cluster-based method (also referred to as community-based method) to find outliers through grouping similar nodes into clusters, which is called the structural clustering algorithm for networks (SCAN) algorithm. The SCAN algorithm is applied to undirected and unweighted graphs and the outliers come from the nodes that do not belong to any clusters in a given network. Akoglu et al. (2010) propose Oddball algorithm for identifying outliers using the subgraphs expanded from a node to the neighboring nodes of the corresponding node. Given a graph, the OddBall identifies outliers by finding the nodes that do not follow the observed patterns in the graph with respect to density, weights, and principal eigenvalues. Sun et al. (2010) apply the SCAN to weighted graph using the weight of edges between common neighbors to measure the similarity between two nodes.

Recently, nonnegative matrix factorization (NMF) has been used for data clustering in various applications (Zhi et al., 2011, Ma et al., 2016). Cao et al. (2013) use nonnegative matrix factorization to find communities in undirected graphs, and then detect hub and outlier nodes in the communities. Tong and Lin (2011) introduce nonnegative residual matrix factorization method to detect link anomalies in a bipartite graph. The residual matrix in their approach shows the deviations of the low rank factorization from the original matrix. Along these lines, the entries of the residual matrix are required to be nonzero in their matrix factorization algorithm to give a meaningful and intuitive view on the significance of the deviation. Aggarwal (2015) introduces a spectral method to complement the matrix factorization method, where the node-link adjacency matrix is augmented into a positive semi-definite matrix and is decomposed using the singular value decomposition methods. The author then follows the residual matrix idea introduced above and finds the anomaly links. To the best knowledge of the authors of this article, this is the first work to employ nonnegative matrix factorization for node outlier ranking on directed citation networks.

A directed graph offers a rich information about the underlying system. This rich information includes (1) pairwise relationship between the components, (2) the direction of relationship between components (i.e., a directed edge between two patents shows citing and cited patents), (3) the intrinsic relationship between the groups of components in the system of interest. Leveraging all this information in a given directed graph to identify the outlier nodes is challenging because it requires identification of group of nodes in the graph and quantification of relationship between these groups. To address this challenge, we propose a new node outlier detection algorithm based on matrix factorization techniques. For this purpose, we utilize the citation structure of a given citation network. Using asymmetric nonnegative matrix factorization (ANMF) algorithm (Wang et al., 2011), we factorize the citation matrix to find the relevant information for clusters, and then, score the nodes with the values of factorized matrices which evaluate the likelihood of node anomalies. The quality of the clusters has significant impact on the quality of anomalies detected. We validate our proposed anomaly ranking method using the real-world U.S. patent citation dataset.

The structure of our work is organized as follows. Section 2.2 provides the background on NMF and asymmetric NMF in graph data. Section 2.3 presents the new node anomaly scoring functions. Section 2.4 presents a data description of the artificial and real-world data used in the experiments along with the experimental results for anomaly ranking. Finally, Section 2.5 concludes this chapter and presents the future research directions.

2.2 Nonnegative matrix factorization

Matrix factorization is a process of decomposing a matrix into two or more matrices, indicating linear combinations of entries in different matrices. Therefore, the multiplications of the factorized matrices are equivalent to the original matrix. In various applications, the matrix factorization is used to explore the latent features through combining different types of entities. Nonnegative matrix factorization is a matrix factorization method that focuses on the analysis of data matrices whose elements are nonnegative. Table 2.1 shows the commonly used symbols of this chapter and their

descriptions.

Table 2.1: Notation summary

Symbol	Description
$\mathbf{C} \in \Re_+^{n \times n}$	Citation matrix with nonnegative components
$\mathbf{A} \in \Re_+^{n \times n}$	Adjacency matrix with nonnegative components
$[\mathbf{A}]_{ij}$	$ij$the value of $\mathbf{A}$
$[\mathbf{A}]_{i*}, [\mathbf{A}]_{*j}$	vectors at ith row and jth column of $\mathbf{A}$
$\mathbf{W}, \mathbf{H}$	Lower rank factorized matrices used in ANMF
$\|\| \cdot \|\|, \|\| \cdot \|\|_F$	Euclidean norm, Frobenious norm
r	Factorization rank
$\mathbf{1}_r$	Column vector whose all r elements are one
$AN(i)$	Anomaly score for node i

Let $\Re_+$ denotes the set of nonnegative real numbers. For a given nonnegative matrix $\mathbf{X} \in \Re_+^{m \times n}$, NMF seeks two lower rank matrices, $\mathbf{W} \in \Re_+^{m \times r}$ and $\mathbf{H} \in \Re_+^{r \times n}$, where $r < \min\{m, n\}$, that approximate $\mathbf{X} \approx \mathbf{WH}$ by solving the optimization problem

$$\min_{\mathbf{W},\mathbf{H} \geq \mathbf{0}} ||\mathbf{X} - \mathbf{WH}||_F^2,$$

where $|| \cdot ||_F$ denotes the Frobenious norm and all elements of both $\mathbf{W}$ and $\mathbf{H}$ are nonnegative. Since the objective function $||\mathbf{X} - \mathbf{WH}||_F^2$ is not convex in both $\mathbf{W}$ and $\mathbf{H}$, NMF solves a non-convex optimization problem. Thus, it is impossible to find the global minimum. **?** suggest an iterative algorithm and prove that their approach could find a local optimal value. Their updating process has been demonstrated to be non-increasing in terms of the Euclidian distance, which often leads to local optimal solutions. When we introduce orthogonal constraints on $\mathbf{H}$, that is $\mathbf{HH}^T = \mathbf{H}^T\mathbf{H} = \mathbf{I}$, the NMF problem can be equivalent to the k-means clustering, except for the nonnegativity constraints (Ding et al., 2005).

Apart from the general expression of the NMF, researchers have found different types of matrix factorization methods for graph and network data based on adjacency matrix. One main purpose of the modification of NMF is to utilize the characteristics of the dataset in order to find useful patterns through matrix factorization. Symmetric

NMF requires that the factorization results should be the multiplication of one matrix and its transpose leading to advantage in factorizing symmetric matrices (Wang et al., 2011). The asymmetric nonnegative matrix factorization (ANMF) method has been developed for analyzing directed graph data by transforming a symmetric NMF into a specific expression that could be applied to an asymmetric square matrix, which shows wider applications in handling the graph problems (Wang et al., 2011). Given a directed adjacency matrix $\mathbf{A} \in \Re^{n \times n}$, ANMF approximates $\mathbf{A}$ with two nonnegative matrices $\mathbf{W} \in \Re^{n \times r}$ and $\mathbf{H} \in \Re^{r \times r}$ as follows:

$$\begin{aligned} &\min_{\mathbf{W},\mathbf{H}} ||\mathbf{A} - \mathbf{W}\mathbf{H}\mathbf{W}^T||_F^2 \\ &\text{s.t. } \mathbf{W} \in \Re_+^{n \times r}, \mathbf{H} \in \Re_+^{r \times r}, \end{aligned} \tag{2.1}$$

where r is the predetermined low-rank matrix size. Wang et al. (2011) develop an iterative algorithm and prove that their approach could find a local optimal value. Accordingly, the multiplicative updating rules for the ANMF are

$$\begin{aligned} [\mathbf{W}]_{ij} &\leftarrow [\mathbf{W}]_{ij} \left(\frac{[\mathbf{A}\mathbf{W}\mathbf{H}^T + \mathbf{A}^T\mathbf{W}\mathbf{H}]_{ij}}{[\mathbf{B}\mathbf{W}\mathbf{H}^T + \mathbf{B}^T\mathbf{W}\mathbf{H}]_{ij}} \right)^{1/4}, \\ [\mathbf{H}]_{ij} &\leftarrow [\mathbf{H}]_{ij} \frac{[\mathbf{B}^T]_{ij}}{[\mathbf{W}^T\mathbf{B}\mathbf{W}]_{ij}}, \end{aligned}$$

where $\mathbf{B} = \mathbf{W}\mathbf{H}\mathbf{W}^T$. The updating rules make the objective function in equation (2.1) to be non-increasing and converge to a local optimal solution.

2.3 New scoring method for anomaly detection in directed graph

In this section, we introduce a new node anomaly scoring method for patent citation network.

2.3.1 Citation matrix

Let $G = (V, E)$ be a directed graph as shown in Fig. 2.1, where V is a set of n nodes and E is a set of m edges, by which nodes in the graph are connected to the other nodes.

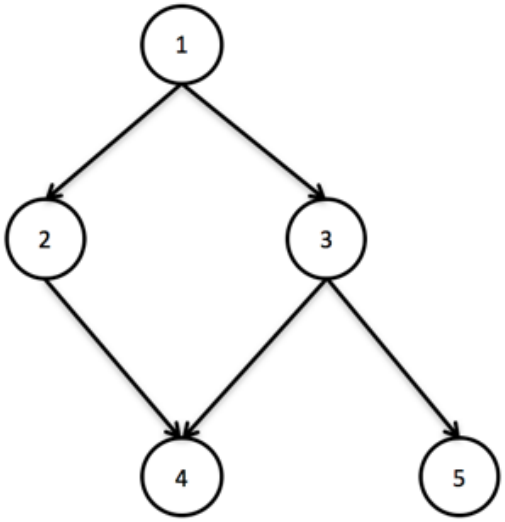

Figure 2.1: Representation of direct and indirect citations.

Graph G can be represented as a directed adjacency matrix $\mathbf{A} \in \Re^{n \times n}$ whose element $[\mathbf{A}]_{ij}$ is 1 if node i is directed to node j. In a directed graph, edges have direction which indicates a one-way relationship between the nodes. The adjacency matrix of the sample graph in Fig. 2.1 is

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

In citation network, a node can cite another node directly or indirectly. For example, as shown in Fig. 2.1, node 4 cites node 2 directly but it cites node 1 indirectly with length 2. We can capture this direct and indirect citation structure of a given citation network by using citation matrix defined as

$$\mathbf{C} = \sum_{l=1}^{\infty} \beta^l \mathbf{A}^l \tag{2.2}$$

where $0 < \beta < 1$ is the discounting factor and $\mathbf{A}^l \in \Re^{n \times n}$ is the matrix product of

l copies of $\mathbf{A}$, where an element $[\mathbf{A}]_{ij}^{l}$ represents the number of paths with length l from node i to node j. Since we assume that two nodes cannot cite each other (i.e., the former published patent cannot cite the later ones), the directed graph is acyclic. Proposition 1 shows that the adjacency matrix $\mathbf{A}$ is nilpotent and the citation network $\mathbf{C}$ is rewritten as $\beta\mathbf{A}(\mathbf{I}-\beta\mathbf{A})^{-1}$. Proof of proposition 1 is in Appendix A.

Proposition 1. If $\mathbf{A}$ is the adjacency matrix of an acyclic directed graph, there exists a constant value $\tau > 0$ such that $\mathbf{A}^{l} = \mathbf{0}, \forall l > \tau$. Then the citation matrix $\mathbf{C}$ is rewritten as $\mathbf{C} = \beta\mathbf{A}(\mathbf{I}-\beta\mathbf{A})^{-1}$.

By using (2.2), we give more weight to direct citations and less weight to indirect citations. A citation with less weight means that the path length between two nodes is long. We can write the $\mathbf{C}$ matrix for the sample graph in Fig. 2.1 with discounting factor $\beta = 0.6$ as

$$\mathbf{C} = \begin{bmatrix} 0 & 0.6 & 0.6 & 0.72 & 0.36 \\ 0 & 0 & 0 & 0.6 & 0 \\ 0 & 0 & 0 & 0.6 & 0.6 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

The non-zero values in $\mathbf{C}$ indicate that there exists at least one direct or indirect citation between corresponding patents. For example, take $[\mathbf{C}]_{12}$ and $[\mathbf{C}]_{15}$ into consideration. These two values are non-zero because there are paths from node 1 to node 2 and node 5. Moreover, $[\mathbf{C}]_{15}$ is less than $[\mathbf{C}]_{12}$ because less weight is given to indirect citations by using discounting factor, which reflects the strength of connectivity between nodes.

2.3.2 ANMF with citation matrix

ANMF is a factorization method that can be used on citation matrix. Applying the algorithm proposed by Wang et al. (2011), we could obtain two factorized matrices, $\mathbf{W}$ and $\mathbf{H}$ such that $\mathbf{C} \approx \mathbf{W}\mathbf{H}\mathbf{W}^{T}$, where $\mathbf{W}, \mathbf{H} \geq \mathbf{0}$. The factorized matrices $\mathbf{W}$ and $\mathbf{H}$ give useful interpretation for the given graph based on clustering. $\mathbf{W}$ denotes the

within-cluster node weight matrix whose element $[\mathbf{W}]_{ik}$ is the weight for inclusion of node i in cluster k for $i = 1, \ldots, n$ and $k = 1, \ldots, r$. $\mathbf{H}$ shows the between-clusters weighted directions, whose element $[\mathbf{H}]_{pq}$ is the weight of the direction from cluster p to cluster q for $1 \leq p, q \leq r$. The diagonal elements of $\mathbf{H}$ are interpreted as the weight of self-direction caused by the nodes directing to the other nodes in the same cluster.

The matrix $\mathbf{W}$ could be normalized by columns and $\mathbf{W}^T$ normalized by row respectively with the following formula:

$$\mathbf{WHT}^T = (\mathbf{WD}^{-1})(\mathbf{DHD}^T)(\mathbf{WD}^{-1})^T \tag{2.3}$$

where $\mathbf{D} = \text{diag}(d_1, \ldots, d_k)$, where $d_j = \mathbf{1}_n^T[\mathbf{W}]_{*j} = \sum_{i=1}^{n}[\mathbf{W}]_{ij}$. A component d_j of the diagonal matrix $\mathbf{D}$ could be considered as the weight of the jth row vector of the matrix $\mathbf{W}$. For simplicity, let $\mathbf{W}^* = \mathbf{WD}^{-1}$ and $\mathbf{H}^* = \mathbf{DHD}^T$, where $\mathbf{W}^*$ is the normalized $\mathbf{W}$ by transferring the diagonal matrix $\mathbf{D}$ to $\mathbf{H}$. The matrix $\mathbf{W}^*$ could provide r clusters of the original graph and the members of each cluster are decided by the non-zero entries in each column of matrix $\mathbf{W}^*$. Since this study mainly deals with anomaly detection, the exact clustering methods and the quality of clustering are not discussed in details. Rather, we group all the nodes with corresponding entries larger than a predetermined value in each column into a cluster.

2.3.3 Proposed scoring method for anomaly detection

The entries in $\mathbf{W}^*$ matrix represent the normalized weight of each node within one cluster, showing the link behavior. For example, if $[\mathbf{W}^*]_{ij}$ is large, then node i links to other nodes within cluster j and such link behavior is stronger compared to that of other nodes in the same cluster. The $\mathbf{W}^{*T}$ matrix, on the other hand, reflects the linked behavior. $\mathbf{H}^*$ matrix shows the nodes' link behavior between two clusters (i.e. connectivity between the clusters). For example, if $[\mathbf{H}^*]_{ij} \neq 0$, then some nodes can be found in cluster i links to some nodes in cluster j. Higher value means that such connectivity is strong, i.e., an edge (1-step path) exits between the nodes of two clusters. Besides, the diagonal terms in $\mathbf{H}^*$ matrix could represent the importance of a cluster

in terms of the whole graph. Therefore, if we only consider such link behavior, the importance of one node in the graph is determined by how that node links to other nodes in the clusters.

Thus, we can obtain the score for each node as $\mathbf{O}_1 = \mathbf{W}^*\mathbf{H}^*$. The row element gives the score value of one node in each community, and we will use the row sum as the anomaly score for the link behavior as

$$AS_1(i) = \sum_{j=1}^{k} [\mathbf{W}^*\mathbf{H}^*]_{ij} \tag{2.4}$$

where $1 \leq i \leq n$. The nodes with less score values in this scenario tend to be anomalous.

Nevertheless, in a directed graph, considering only the link behavior is not sufficient because it describes the behavior of edges that merely point from one node to the other nodes. Due to the possibly asymmetric adjacency matrix and the asymmetric edge directions between two nodes in directed graph, it may be also necessary to consider the linkage of edges pointing from the other nodes to a given node or the linked behavior. Instead of $\mathbf{W}^*$ matrix, $\mathbf{W}^{*T}$ matrix will be used to describe how one node is linked to other nodes. Following the similar idea as above, the expression $\mathbf{O}_2 = \mathbf{H}^*\mathbf{W}^{*T}$ could be used to score the linked behavior, and the row sum could be used to represents the scoring function for each nodes as

$$AS_2(i) = \sum_{j=1}^{k} [\mathbf{W}^*\mathbf{H}^{*T}]_{ij}, \tag{2.5}$$

where $1 \leq i \leq n$. To keep the format as in the $AS_1(i)$, we take the transpose of $\mathbf{O}_2$ matrix, $\mathbf{O}_2 = (\mathbf{H}^*\mathbf{W}^{*T})^T = \mathbf{W}^*\mathbf{H}^{*T}$. The nodes with less score values in this scenario tend to be anomaly.

Based on the discussion above, the connectivity, or the structure of the original graph could be decomposed into link behavior and linked behavior. Both behaviors are important for finding detecting anomaly. The sum of the $AS_1(i)$ and $AS_2(i)$ could give a simple overall score function. However, the importance of these two behaviors may not be the same in different graphs with the actual needs of anomaly types. For

example, in patent citation network, the patents are more likely to be anomalies if they are cited by few other patents so that the linked behavior may be more important in this situation; for email contact network, an email address which keeps sending high volume of emails to other users may be treated as anomalous/spam source so that the link behavior could be more important. To separate the importance, we introduce the weighted scoring function for node anomaly detection as follows:

$$AN(i) = \alpha \sum_{j=1}^{k} [\mathbf{W}^* \mathbf{H}^*]_{ij} + (1-\alpha) \sum_{j=1}^{k} [\mathbf{W}^* \mathbf{H}^{*T}]_{ij} \tag{2.6}$$

where $0 \leq \alpha \leq 1$. Algorithm 1 describes the node anomaly detection procedure using the proposed outlier scoring method and ANMF.

Algorithm 1 Proposed outlier scoring procedure

Given: $\mathbf{A}$, r, α, β

1: Calculate citation matrix using $\mathbf{A}$ and β
2: Use ANMF method to factorize $\mathbf{C}$ matrix and obtain the output matrix $\mathbf{W}$ and $\mathbf{H}$ following the algorithm introduced by Wang et al. (2011).
3: Normalize the $\mathbf{W}$ and $\mathbf{H}$ matrix in terms of columns to obtain $\mathbf{W}^*$ matrix and $\mathbf{H}^*$ matrix.
4: Find the anomaly score for each node using (2.6).
5: Sort the $AN(i)$ score in ascending order.
6: Lock the score values smaller or equal to a predetermined threshold δ and pick out the respected nodes as anomalies.

2.3.4 Initialization based on modified SVD

NMF algorithms needs good initialization strategy as good initialization result in fast convergence and low cost of computational process (Boutsidis and Gallopoulos, 2008). In general, NMF algorithms are initialized using random initialization approach, which requires several instances of the algorithm with different initial matrices and then select the best solution or the average solution in the case of random initialization. Therefore, the overall process can become quite expensive. In order to overcome the issue from the random initialization for general NMF problems, Boutsidis and Gallopoulos (2008) propose a novel initialization algorithm, Nonnegative Double Singular Value Decomposition (NNDSVD), based on the singular value decomposition (SVD), which computes

the approximation of a matrix $\mathbf{C}$ factorized by two nonnegative matrices with rank r. The NNDSVD first computes a decomposition of $\mathbf{C}$ using the SVD, where $\mathbf{C} = \mathbf{USV}^T$, then extracts positive sections of respective matrices and initializes the nonnegative factorization. The NNDSVD leads to rapid error reduction and faster convergence than random initialization.

We modify the SVD-based initialization procedure for the ANMF. When $\mathbf{C}$ is a $n \times n$ matrix, then $\mathbf{U}$, $\mathbf{S}$, and $\mathbf{V}$ are also $n \times n$ matrices, and the SVD can be represented as $\mathbf{C} = \mathbf{U}(\mathbf{SV}^T\mathbf{U})\mathbf{U}^T$, since $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices. Let $[\mathbf{X}]^+_{*j}$ be the nonnegative section of the jth column vector of $\mathbf{X} \in \Re^{n\times n}_+$ and $[\mathbf{X}]^-_{*j}$ be the nonpositive section of $[\mathbf{X}]_{*j}$, then $[\mathbf{X}]_{*j} = [\mathbf{X}]^+_{*j} - [\mathbf{X}]^-_{*j}$. We also define $[\mathbf{X}]_{*,1:r}$ as the first r columns of matrix $\mathbf{X}$ and $[\mathbf{X}]_{1:r,1:r}$ as the first r rows and columns of $\mathbf{X}$. In Algorithm 2, we obtain the initial matrices $\mathbf{W}_0$ and $\mathbf{H}_0$ as $\mathbf{W}_0 = [\tilde{\mathbf{U}}]_{*,1:r}$ and $\mathbf{H}_0 = [\tilde{\mathbf{S}}\tilde{\mathbf{V}}^T\tilde{\mathbf{U}}]_{1:r,1:r}$.

Algorithm 2 SVD-based initialization for ANMF

Given: Citation matrix $\mathbf{C}$ and factorization rank r

Compute SVD of $\mathbf{C}$: $\mathbf{USV}^T = \mathbf{C}$
for $j = 1 : n$ **do**
  Set $u^+ = [\mathbf{U}]^+_{*j}$, $v^+ = [\mathbf{V}]^+_{*j}$, $u^- = [\mathbf{U}]^-_{*j}$, $v^- = [\mathbf{V}]^-_{*j}$.
  if $||u^+|| \cdot ||v^+|| \geq ||u^-|| \cdot ||v^-||$ **then**
    $u = u^+/||u^+||$
    $v = v^+/||v^+||$
    $\sigma = ||u^+|| \cdot ||v^+||$
  else
    $u = u^-/||u^-||$
    $v = v^-/||v^-||$
    $\sigma = ||u^-|| \cdot ||v^-||$.
  end if
  Set $[\tilde{\mathbf{U}}]_{*j} = u$, $[\tilde{\mathbf{V}}]_{*j} = v$, and $[\tilde{\mathbf{S}}]_{jj} = \sigma[\mathbf{S}]_{jj}$
end for
Set $\mathbf{W}_0 = [\tilde{\mathbf{U}}]_{*,1:r}$ and $\mathbf{H}_0 = [\tilde{\mathbf{S}}\tilde{\mathbf{V}}^T\tilde{\mathbf{U}}]_{1:r,1:r}$

2.3.5 Complexity analysis

Matrix multiplication usually takes $O(n^3)$ time for the multiplication of two $n \times n$ matrices. However, since the citation networks are represented by very sparse matrices, it takes $O(s^{0.7}n^{1.2} + n^{2+o(1)})$ time [28] to multiply two very sparse $n \times n$ matrices, where n is the total number of nodes and s is the number of non-zero elements (i.e., edges).

Singular value decomposition of an $n \times n$ matrix takes $O(n^3)$ time. It takes $O(nrt)$ time to factorize an $n \times n$ matrix using ANMF algorithm, where r is the factorization rank and t is the number of iterations for the ANMF algorithm to converge.

2.3.6 An illustrative example

In this section, we present an illustrative example for our proposed algorithm. The specific steps of the proposed method are shown in Fig 2.2.

In this section, we present an illustrative example for our proposed algorithm for better understanding of it. The graph for this illustrative example is shown in Fig. 2.1. Citation matrix has already been calculated in Section 2.3.1. First we obtain the initial matrices $\mathbf{W}_0$ and $\mathbf{H}_0$ by using modified SVD method as:

$$\mathbf{W}_0 = \begin{bmatrix} 0 & 0.7415 \\ 0.3238 & 0.3062 \\ 0.3238 & 0.4751 \\ 0.7302 & 0 \\ 0.4020 & 0 \end{bmatrix}, \mathbf{H}_0 = \begin{bmatrix} 0.0025 & 0 \\ 1.5791 & 0.0001 \end{bmatrix}.$$

Using these initial matrices in ANMF method in order to approximate $\mathbf{C}$ matrix with factorization rank 2, we obtain $\mathbf{W}^*$ and $\mathbf{H}^*$ matrices as:

$$\mathbf{W}^* = \begin{bmatrix} 0 & 0.4869 \\ 0.1819 & 0.2011 \\ 0.1819 & 0.3120 \\ 0.4103 & 0 \\ 0.2253 & 0 \end{bmatrix}, \mathbf{H}^* = \begin{bmatrix} 0.0080 & 0 \\ 4.2798 & 0.0002 \end{bmatrix}.$$

By using these results in our proposed score function; we obtain the outlier scores for nodes in this small illustrative example as shown in Table 2.2. In the results, higher score means less outlier.

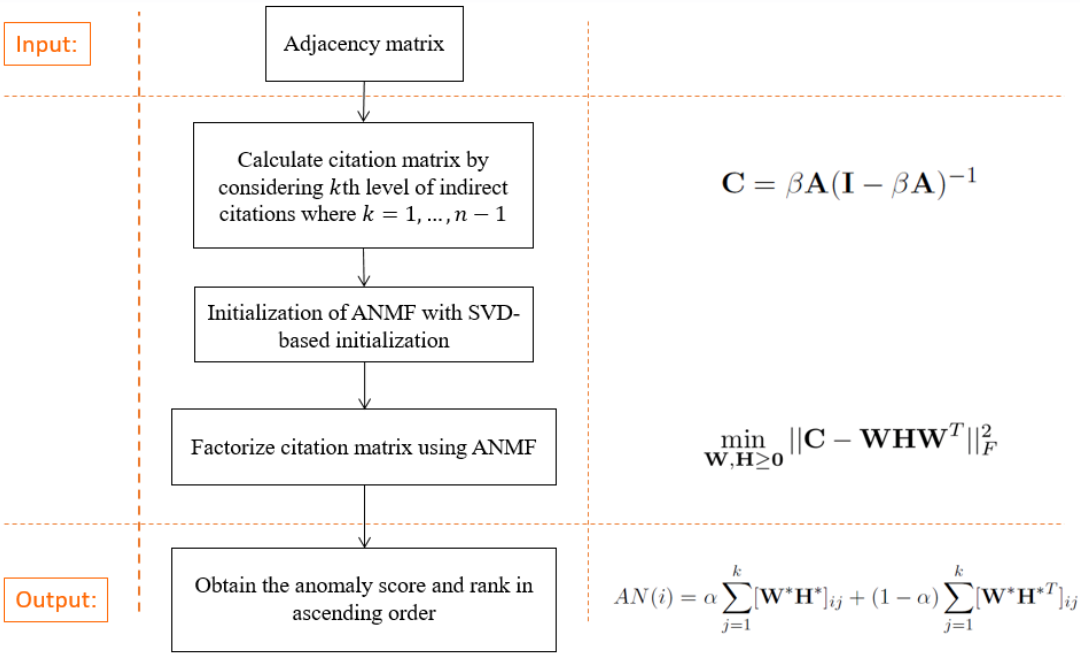

Figure 2.2: Proposed node anomaly scoring flowchart

Table 2.2: Outlier score and rank of 5-node illustrative example

Rank	Node	$AS_1(i)$	$AS_2(i)$	Score
1	5	0.0018	0.9684	0.1468
2	4	0.0033	1.7592	0.2666
3	2	0.8621	0.7801	0.8498
4	3	1.3367	0.7801	1.2532
5	1	2.0840	0.0001	1.7714

The number of citations (i.e., direct and indirect citations) of node 1 is greater than the number of citations of other nodes. Therefore, it is expected to be the least outlier node. In similar way, node 3 is the next least outlier node. Notice that even though node 4 and 5 have no link to them, they obtain different scores, since node 5 cites only node 3, but node 4 cites both node 2 and 3. Our proposed algorithm is able to distinguish leaf nodes by considering both link and linked structure of a given directed graph.

2.4 Experimental results

In this section, experimental results have been presented based on an artificial dataset and a real-world citation network. We first use small artificial dataset to compare our proposed algorithm with well-known anomaly detection algorithms, OutRank and OddBall. Then, we compare our method with OutRank and OddBall using a real-world patent citation data.

2.4.1 Artificial dataset: 14-node network

In this dataset, there are 14 nodes and 16 edges as shown in Fig. 2.3. Note that it is possible for any pair of nodes to have more than one path between them either directly or indirectly.

The ANMF algorithm with $r = 4$ finds four clusters using $\mathbf{W}^*$ matrix, of which each column contains the cluster information. The non-zero elements of each column in $\mathbf{W}^*$ matrix indicate that these corresponding nodes are involved in the same cluster. Using the ANMF, nodes, that are not directly connected, can be contained in the same cluster

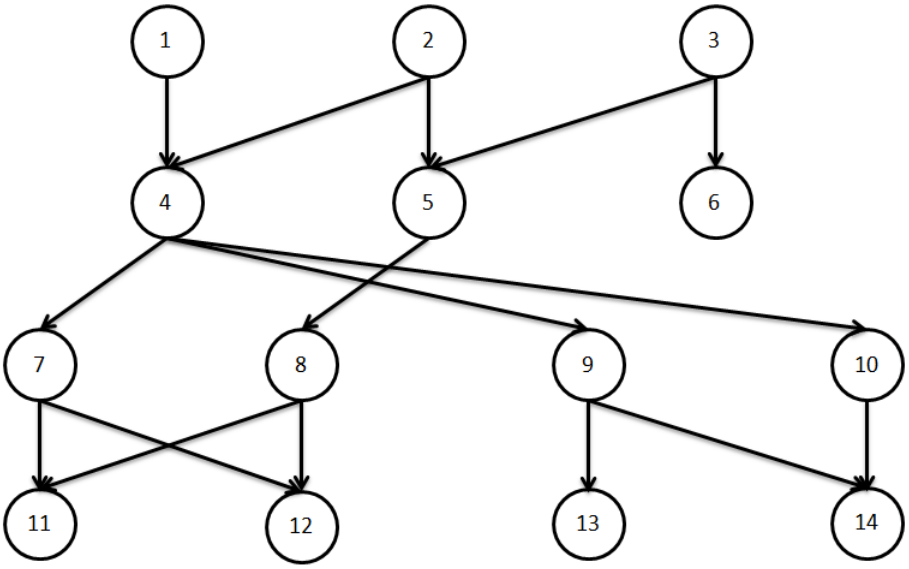

Figure 2.3: Network structure of artificial dataset

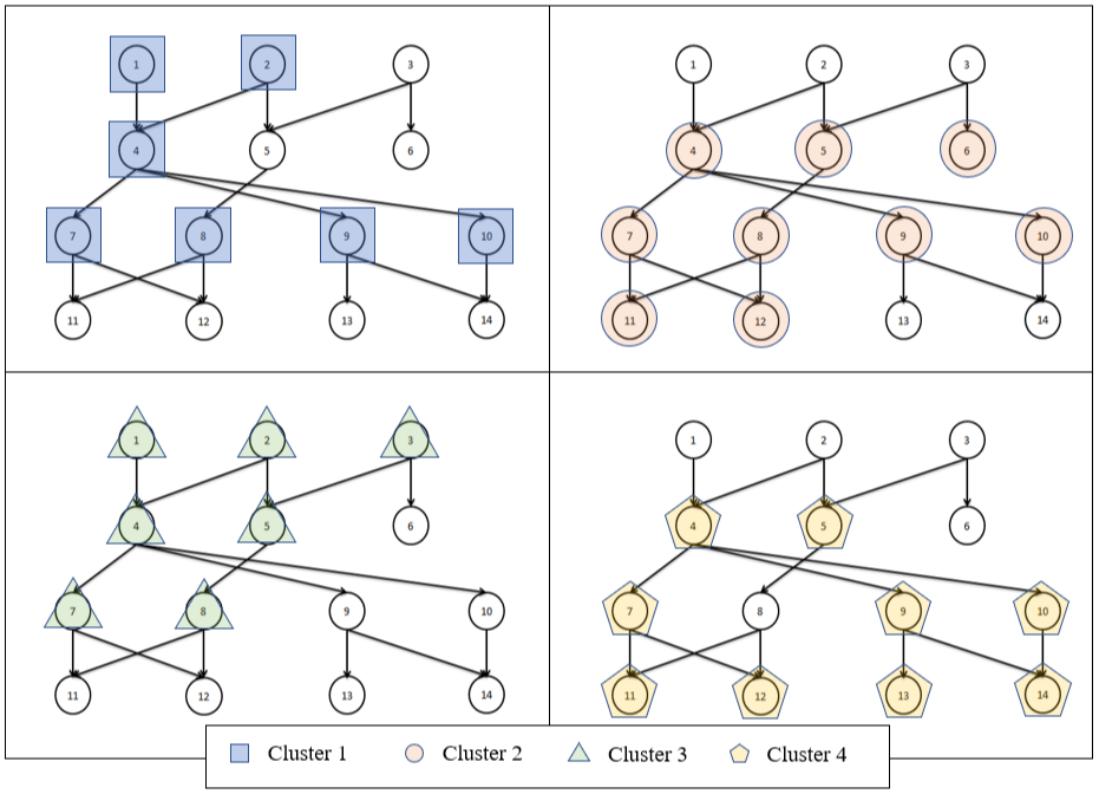

Figure 2.4: Four clusters obtained based on the matrix factorization results. Each color and shape represent a different cluster.

due to the structural information of the original graph. Fig. 2.4 shows the four clusters obtained by the ANMF algorithm, which reveals an interesting relationship between clusters. For example, there exist paths merely from nodes in cluster 3 to nodes in cluster 2, i.e., node 2 in cluster 3 to nodes 4 and node 5 in cluster 2. Similarly, there are paths from nodes in cluster 2 to nodes in cluster 1 when we ignore the overlapped nodes. Such relationship could be explained by the values in $\mathbf{H}^*$ matrix.

Using the factorization results, we compare our proposed algorithm with existing outlier detection algorithms. For the experiment, we set $\alpha = 0.8$ and $\beta = 0.6$ in the proposed algorithm. Table 2.3 shows the comparison results for the artificial dataset. The proposed algorithm finds that the top two outliers are nodes 6 and 13, since both nodes are not cited by the others and both cite only one node. Node 6 has higher score than node 13, since node 6 has a single path only from node 3, but node 13 has several paths from the other nodes. The experimental result demonstrates that our algorithm performs well and identifies the desired anomalies successfully. For a fair comparison, we transform the adjacency matrix of the directed network to that of the undirected network by ignoring the directions for the best use OutRank and OddBall algorithms as they work best on undirected networks. The OddBall cannot distinguish the nodes based on anomaly scores, since all scores are equal to zero. The Outrank provides rank in outlierness, but the result is different from what we have expected, since node 3 has the second highest score in outlierness. We cannot say node 3 is more outlier than node 13, since node 3 is cited by two other nodes, but node 13 is not cited by others. Hence, results suggest that our method is advantageous in detecting anomalies in artificial dataset.

Table 2.3: Comparison of the proposed algorithm, OutRank, and OddBall using 14-node patent citation network

	Proposed		OutRank		OddBall	
Rank	Node ID	Score	Node ID	Score	Node ID	Score
1	6	0.1212	6	0.0263	1	0
2	13	0.2577	3	0.0356	2	0
3	14	0.4900	13	0.0441	3	0
4	11	0.5652	1	0.0623	4	0
5	12	0.5652	14	0.0631	5	0
6	10	1.0683	8	0.0646	6	0
7	3	1.1495	9	0.0764	7	0
8	5	1.4087	10	0.0764	8	0
9	9	1.4642	11	0.0806	9	0
10	8	1.5601	12	0.0806	10	0
11	7	1.6014	5	0.0851	11	0
12	1	2.1966	7	0.0917	12	0
13	2	3.1407	2	0.0930	13	0
14	4	3.328	4	0.1202	14	0

2.4.2 Real-world dataset: U.S. patent citation network

In this section, we evaluate the performance of our proposed method on the US Patent Citation Network. The dataset is a directed and unweighted graph consisting of 750 nodes and 1376 directed edges. Performance evaluation is a difficult task for the outlier detection algorithms because of the inexistence of ground truth anomaly labels. One way to handle this issue is to inject synthetic anomalies to the real-world dataset. In this work, we inject 79 synthetic anomalies into our PCN data and show the performance evaluation of our algorithm based on how well it detects the anomalies.

As for the performance metric, we employ the $F1$ score and accuracy index. $F1$ score is defined as

$$F1 = 2 \cdot \frac{\frac{TP}{TP+FP} \cdot \frac{TP}{TP+FN}}{\frac{TP}{TP+FP} + \frac{TP}{TP+FN}},$$

where TP is the number of correctly labeled positive nodes, FP is the number of positive labeled nodes while the true labels are negative, TN is the number of correctly labeled negative nodes, and FN is the number of negative labeled nodes while the true labels are positive. The highest and the lowest values that $F1$ score can get are 1 and

0, respectively. Higher the score, better the results. Accuracy is defined as

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}.$$

We first **A**. Using **A**, we calculate the citation matrix, **C**, as shown in Proposition 1 with $\beta = 0.9$. After SVD based initialization of **W** and **H** matrices, we factorize the **C** using ANMF algorithm with factorization rank $r = 5$. Then, anomaly scores of each node are obtained as shown in (2.6) with $\alpha = 0.3$. Finally, ground truth anomaly labels and predicted anomaly labels are compared for each algorithm. Table 2.4 shows the performance evaluations of outlier detection algorithms. The results reveal that our proposed algorithm outperforms Oddball and Outrank algorithms in terms of both performance metrics.

Table 2.4: Comparison results of the proposed, OutRank and OddBall algorithms on real-world U.S. Patent Citation Network

	F1 score	Accuracy
OutRank	0.8960	0.8118
Oddball	0.8947	0.8094
Proposed	0.9107	0.8384

2.4.3 Parameter sensitivity

Our proposed node anomaly detection algorithm has two parameters: β and α. In this section, we present the sensitivity of the proposed method to these two parameters. Figures 2.5-2.8 show the performance of the proposed method on PCN dataset with varying β and α values. The results show that the performance of our method is stable with respect to the parameter values and it outperforms the other methods with varying parameter values.

2.5 Conclusion

The present study proposes a novel node anomaly detection algorithm using the non-negative matrix factorization technique. Experimental results with the artificial data

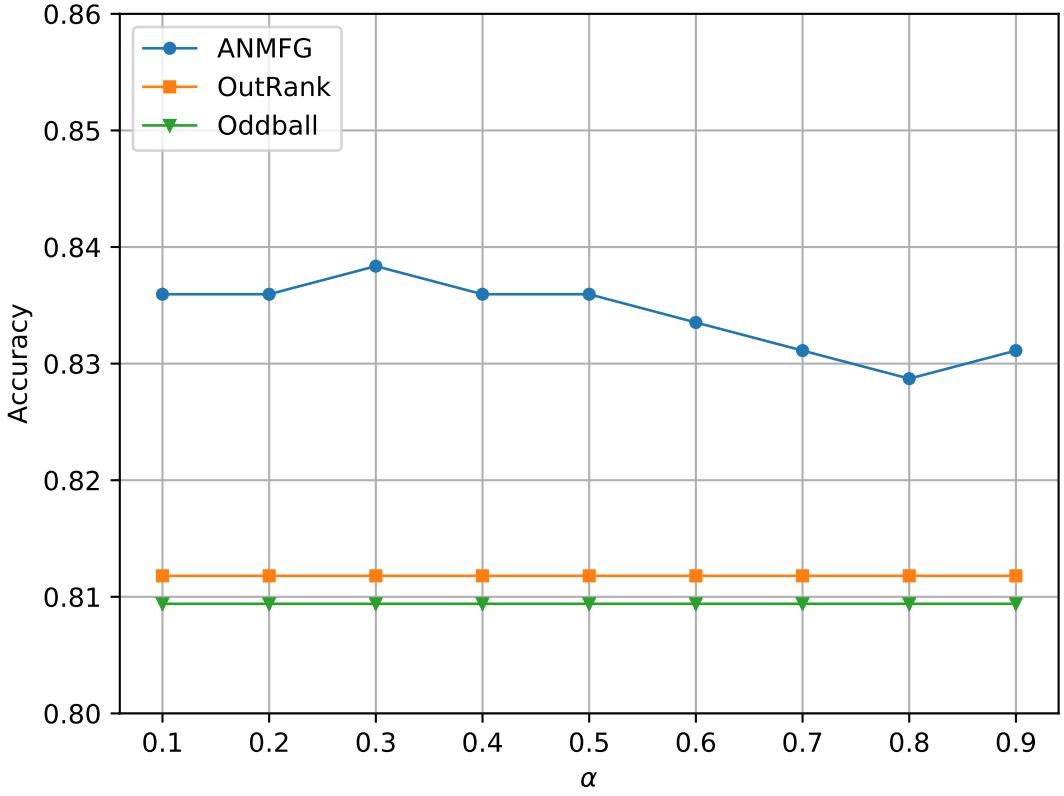

Figure 2.5: Sensitivity of our method to varying α values in terms of accuracy on US PCN dataset.

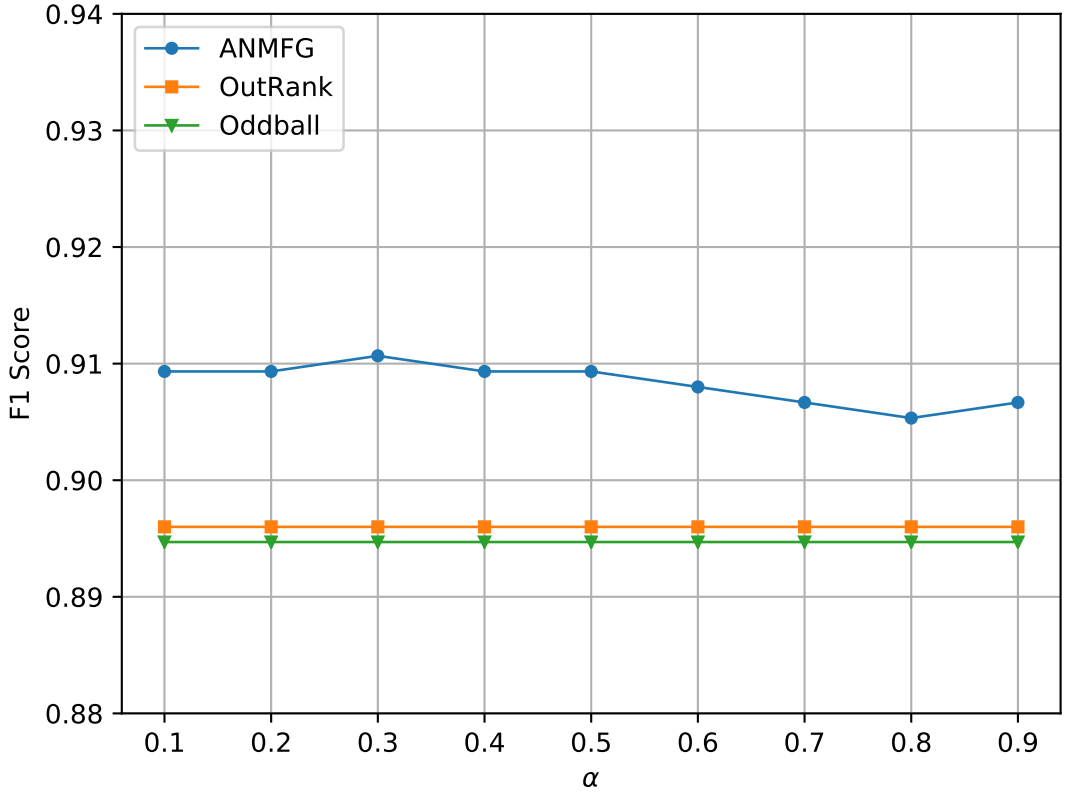

Figure 2.6: Sensitivity of our method to varying α values in terms of F1 score on US PCN dataset.

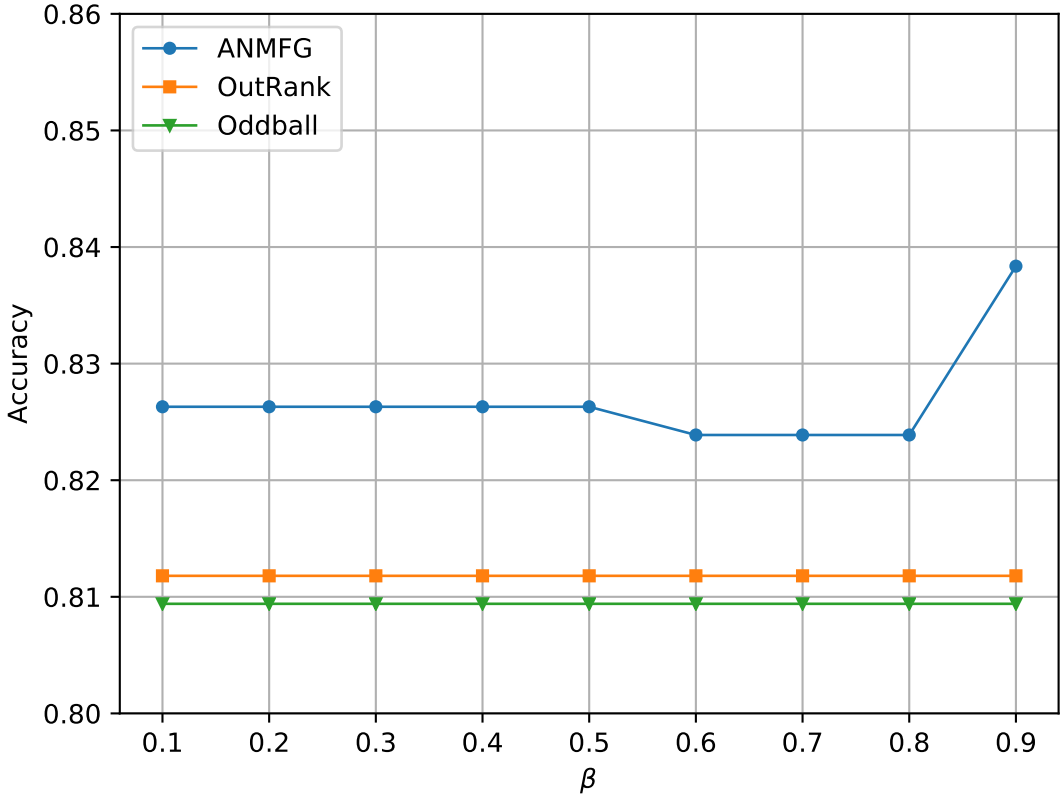

Figure 2.7: Sensitivity of our method to varying β values in terms of accuracy on US PCN dataset.

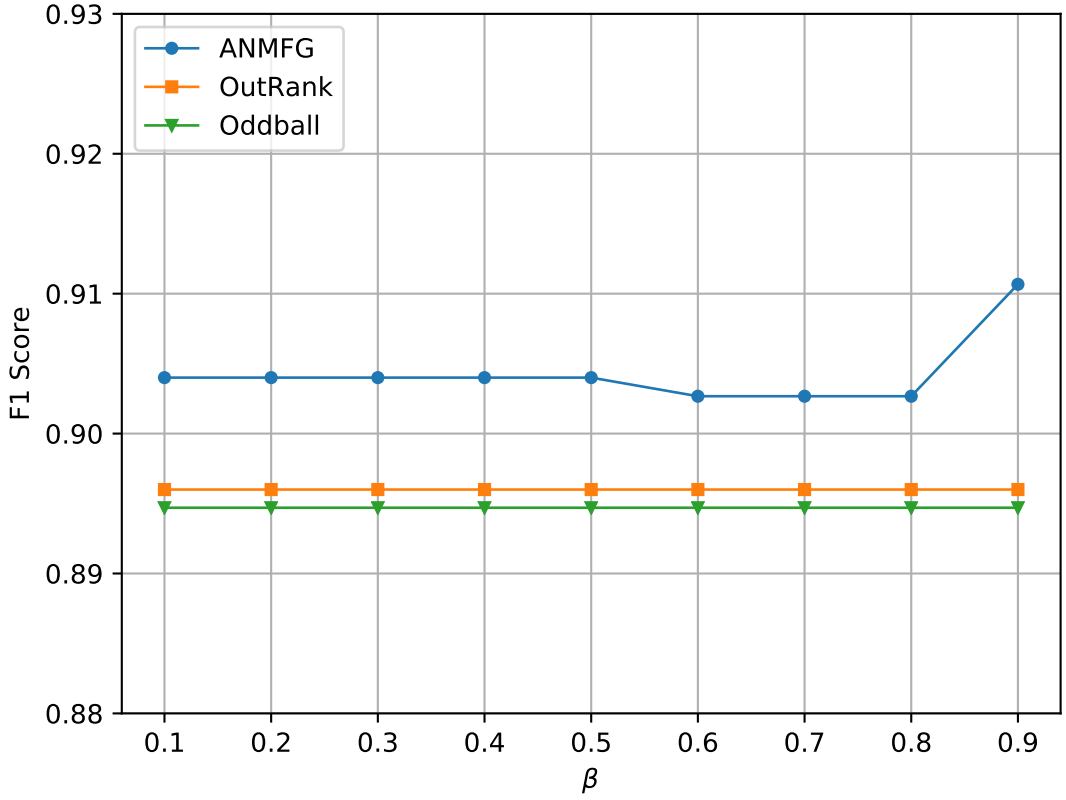

Figure 2.8: Sensitivity of our method to varying β values in terms of F1 score on US PCN dataset.

and real-world patent citation network show that the proposed algorithm detects outlier nodes successfully. Moreover, we present meaningful interpretations by studying the graphs with anomalies which have provided real meaning on the results.

As for the future research, the performance of our proposed algorithm needs to be tested with more real-world datasets. This may help not only compare the performance of different algorithms with different cases but also accumulate more experience on the selection of factorization rank r. Furthermore, we consider the possible known attributes of the nodes and edges in graphs when designing the algorithms. There are possible ways to make use of the node attributes like the anomaly detection approaches for continuous or discrete data. Therefore, future study can be made on the performance of our algorithm and the updated version by considering the node attributes.

Chapter 3

Regularized Asymmetric Nonnegative Matrix Factorization for Clustering in Directed Networks

3.1 Introduction

Over the past few years, numerous network models have been developed and network analysis has become a crucial work for a better understanding of various types of networks (Newman, 2018). These networks include social networks, neural networks, citation networks, transportation networks, protein-protein interaction networks, etc.

Among various research areas in network analysis, clustering is a key task of partitioning entities into logical groupings of components. In general, there are two clustering approaches in networks (Schaeffer, 2007). One is clustering the nodes based on their similarities and the other is clustering the set of subgraphs by considering each of them as a separate object (Schaeffer, 2007). In this chapter, we focus on the former approach. In this sense, we define the set of similar nodes in the context of network as a cluster and clustering is grouping nodes based on their pairwise similarities in the context of networks. Efficiently identifying clusters helps us understand the nature of a given network. The vast amounts of algorithms have been proposed by the researchers in order to cluster nodes in a given network (Liao et al., 2013, Gómez et al., 2015).

In most cases, networks are directed such as citation networks, hyperlinked structure of the web, lateral gene transfer networks, etc. and there has been little research conducted on partitioning nodes in a directed network (Malliaros and Vazirgiannis, 2013). Clustering in directed networks is a complicated process compared to clustering in undirected networks. Characterization is based on the asymmetrical matrices contrast to undirected cases. As a result, this makes spectral analysis much more difficult (Fortunato, 2010). For clustering in directed networks, several different approaches

have been proposed such as naive graph transformation approaches, transformation approaches that maintain directionality, and approaches that extend objective functions of clustering problem in undirected networks to directed networks (Malliaros and Vazirgiannis, 2013). One common way for clustering in directed networks is simply ignoring the directionality of edges and then applying algorithms designed specifically for undirected networks (Malliaros and Vazirgiannis, 2013). On one hand, this approach may lose unique and useful information that the network structure provides. On the other hand, since this approach results in eigenvalue decomposition problem, it is hard to tell the physical meaning of resultant eigenvectors, which are used to identify clusters, in real-world applications (Wang et al., 2011).

More recently, nonnegative matrix factorization (NMF) draws attention as a powerful tool for data representation and interpretation (Wang and Zhang, 2013). It has been applied to various research areas successfully such as image processing (Lee and Seung, 1999, Cai et al., 2011), acoustic signal analysis (Virtanen, 2007), document clustering (Shahnaz et al., 2006), music analysis (Févotte et al., 2009), community discovery (Cao et al., 2013, Wang et al., 2011), etc. Since it is very easy to interpret and it has similarity with k-means clustering (Ding et al., 2005), various versions of NMF have been proposed for clustering (Guan et al., 2011, Shiga and Mamitsuka, 2015). Wang et al. (2011) propose asymmetric nonnegative matrix factorization (ANMF) to find the communities (clusters) of nodes in directed networks. The ANMF algorithm, in particular, is designed for clustering in directed networks based on simple adjacency matrix. However, due to simple structure of adjacency matrix, it fails to capture critical information of the data such as similarity and connectivity between nodes. Consequently, some similar nodes are often located in different clusters. In addition, the ANMF algorithm is computationally expensive when the dataset is large due to random initialization.

In this study, we propose regularized asymmetric nonnegative matrix factorization (RANMF) algorithm for clustering in directed networks. The aim of the present study is to cluster set of nodes in a given directed network by taking advantage of prior similarity information of nodes. To achieve this, we incorporate the prior similarity information into ANMF algorithm as an additional regularization term and design a new

objective function for matrix factorization. We also develop multiplicative updating rules to solve the proposed objective function. Convergence proof of updating rules is presented.

The main contributions of this study are as follows:

- While ANMF algorithm considers only simple adjacency matrix of a given directed network, the proposed RANMF algorithm exploits the pairwise similarity of nodes. Thus, if two nodes are similar to each other in the original space, their representatives in new basis should be close to each other. Consequently, they belong to the same cluster.
- Existing NMF algorithms take much time to run algorithms repeatedly and to obtain a stable solution since they use random initialization. In order to overcome this issue, the RANMF algorithm is developed with singular value decomposition (SVD) based initialization approach (Boutsidis and Gallopoulos, 2008), which is computationally inexpensive and results in stable solution.
- With the proposed framework, one can leverage other types of prior information besides similarity information (e.g. class label information for patent citation networks)

The overall structure of this chapter is as follows. Section 3.2 summarizes the NMF on clustering. Section 3.3 introduces the proposed RANMF algorithm. In Section 3.4, we apply the proposed algorithm to real-world and synthetic datasets, and then compare its performance with other methods. We also provide convergence analysis of the RANMF algorithm. Finally, Section 3.5 concludes the chapter and presents future work.

3.2 Nonnegative matrix factorization on clustering

For a given nonnegative matrix $\mathbf{B} \in \Re_+^{n \times m}$, nonnegative matrix factorization seeks an approximation given by nonnegative matrices $\mathbf{W} \in \Re_+^{n \times r}$ and $\mathbf{X} \in \Re_+^{r \times m}$ by solving the optimization problem

$$\min_{\mathbf{W},\mathbf{X} \geq \mathbf{0}} ||\mathbf{B} - \mathbf{W}\mathbf{X}||_F^2. \tag{3.1}$$

where $r < \min\{n, m\}$ is a predetermined rank for the factorized matrices and $||\cdot||_F$ is the Frobenious norm. Although the problem in (3.1) is nonconvex in both $\mathbf{W}$ and $\mathbf{X}$, it becomes convex in $\mathbf{W}$ given $\mathbf{X}$, and vice versa. To obtain the approximated $\mathbf{B}$, the multiplicative updating rules (Lee and Seung, 2001) optimize (3.1) with respect to $\mathbf{W}$ and $\mathbf{X}$ by iteratively updating each of the matrices as

$$[\mathbf{W}]_{ik} \leftarrow [\mathbf{W}]_{ik} \frac{[\mathbf{B}\mathbf{X}^T]_{ik}}{[\mathbf{W}\mathbf{X}\mathbf{X}^T]_{ik}}, \quad [\mathbf{X}]_{kj} \leftarrow [\mathbf{X}]_{kj} \frac{[\mathbf{W}^T\mathbf{B}]_{kj}}{[\mathbf{W}^T\mathbf{W}\mathbf{X}]_{kj}}. \tag{3.2}$$

In (3.2), each element of the matrices is updated following a rescaled gradient descent scheme and a local minimum of the objective function in (3.1) can be found by the multiplicative updating rules (Lee and Seung, 2001).

Given n observations represented as the rows of $\mathbf{B}$, the NMF groups the observations into r clusters. The ith row in $\mathbf{W}$ denotes the assignment of the ith observation to the clusters in terms of the cluster centroids shown as the columns of $\mathbf{X}$. In particular, the clustering result from the NMF is closely related to k-means clustering (Kim and Park, 2008), and, in the case of the symmetric NMF (i.e., $\mathbf{X} = \mathbf{W}^T$), the clustering from the NMF even becomes equivalent to kernel k-means clustering (Ding et al., 2005).

A directed network with n nodes can be represented by its adjacency matrix, $\mathbf{A} \in \Re_+^{n \times n}$, where $[\mathbf{A}]_{ij}$ is 1 if there is a directed edge connecting node i to node j, 0 otherwise. The NMF is computed with an adjacency matrix $\mathbf{A}$ for the clustering in a network. It is worth noting that the additivity of the NMF facilitates understanding the shape of the clusters in the graph. That is, the entire graph is made up of the summation of r clusters where the partial networks for the kth cluster for $k = 1, 2, \ldots, r$ are illustrated by the outer products between the kth column in $\mathbf{W}$ and the kth row in $\mathbf{X}$.

Furthermore, replacing $\mathbf{X}$ with $\mathbf{H}\mathbf{W}^T$ in (3.1) where $\mathbf{H} \in \Re_+^{r \times r}$, the optimization problem can be written as

$$\min_{\mathbf{W},\mathbf{H} \geq \mathbf{0}} ||\mathbf{A} - \mathbf{W}\mathbf{H}\mathbf{W}^T||_F^2. \tag{3.3}$$

With this formulation, we can induce not only entry-level grouping information from $\mathbf{W}$ but also cluster-level information from $\mathbf{H}$. To be specific, in the case of an undirected

network represented by a symmetric $\mathbf{A}$, $\mathbf{H}$ is obtained as a diagonal matrix whose diagonal elements show the connectivity within the clusters (Wang et al., 2011). On the other hand, in the case of a directed network with an asymmetric $\mathbf{A}$, the elements in $\mathbf{H}$ shows the inter-cluster directness. For example, $[\mathbf{H}]_{pq}$ has a non-zero value if the pth cluster (and its nodes) is directed to the qth cluster (and its nodes). For the problem in (3.3) with a directed network, ANMF method (Wang et al., 2011) is proposed optimizing (3.3) with the following multiplicative updating rules:

$$\begin{aligned} [\mathbf{W}]_{ik} &\leftarrow [\mathbf{W}]_{ik} \left(\frac{[\mathbf{AWH}^T + \mathbf{A}^T\mathbf{WH}]_{ik}}{[\mathbf{WHW}^T\mathbf{WH}^T + \mathbf{WH}^T\mathbf{W}^T\mathbf{WH}]_{ik}} \right)^{\frac{1}{4}} \\ [\mathbf{H}]_{kj} &\leftarrow [\mathbf{H}]_{kj} \frac{[\mathbf{W}^T\mathbf{AW}]_{kj}}{[\mathbf{W}^T\mathbf{WHW}^T\mathbf{W}]_{kj}}. \end{aligned} \tag{3.4}$$

Although a directed network (i.e., its adjacency matrix) is successfully approximated by the ANMF method in (3.3), the intrinsic attribute of directed networks is overlooked for clustering. To be specific, the approximated $\mathbf{A}$ is represented by the combination of the additive subnetworks, and each subnetwork consists of the nodes involved in the group corresponding to the column in $\mathbf{W}$. However, the connection between the nodes in each subnetwork is not guaranteed wherein the optimization with (3.4) is only for the representation of $\mathbf{A}$ without consideration for the connectivity within the resulted group despite the original clustering task. Thus, some groups identified by the ANMF method may consist of disconnected nodes within the subnetworks. In addition, the computation of most NMF models, including the ANMF, is based on random initialization, and such randomness causes unstable clustering results in that multiplicative updating rules converge on local minima (Wang et al., 2011). This may lead to heavy computation as discussed in Section 3.2.

3.3 Regularized asymmetric nonnegative matrix factorization

This section proposes RANMF algorithm. The proposed RANMF algorithm considers the pairwise similarities between nodes in a given directed network and adds a

regularization term to ANMF in order to cluster nodes by exploiting the prior similarity information. We then propose the multiplicative updating rules of the RANMF algorithm. The convergence of our proposed updating rules is also proved in Appendix.

3.3.1 Optimization problem

Let $\mathbf{E} \in \Re^{r \times r}$ be a diagonal matrix whose kth element is the sum of kth column of $\mathbf{W}$. Then factorized matrix $\mathbf{W}$ can be normalized as $\mathbf{W}^* = \mathbf{W}\mathbf{E}^{-1}$, where $\mathbf{W}^*$ is the normalized $\mathbf{W}$ matrix and $\sum_i [\mathbf{W}^*]_{ik} = 1$ for $k = 1, 2, \ldots, r$ (Wang et al., 2011). $\mathbf{W}^*$ gives useful interpretation for clustering in a given network (Cao et al., 2013). It denotes the within-cluster node weight whose element $[\mathbf{W}^*]_{ik}$ is the weight for inclusion of node i in cluster k and ith row of $\mathbf{W}$, $\mathbf{w}_i$, is the representation of node i in new basis.

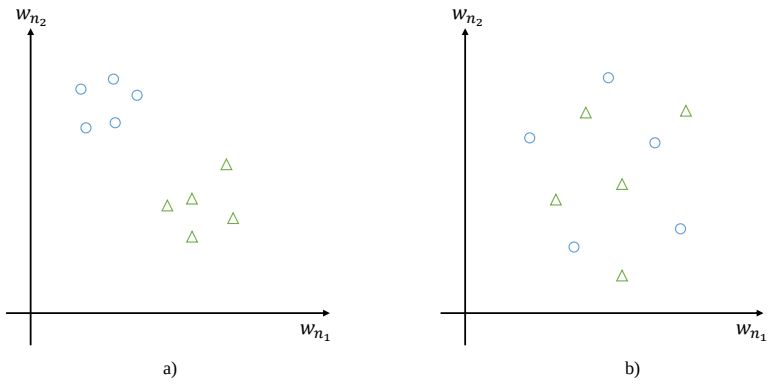

Figure 3.1: Representatives of nodes in new basis. 10 nodes and 2 clusters. Each shape represents different cluster.

One can naturally assume that the representatives of nodes in the same cluster should be close to each other and far from the nodes in the other clusters, which means that the higher similarity between nodes should lead to smaller distance between corresponding representatives as shown Fig. 3.1-a) rather than 3.1-b). Existing ANMF algorithm is not able to capture this prior similarity information of a given network. Thus, it might happen that similar nodes are assigned to the different clusters or vice versa. To adapt this prior similarity information of a given network, we suggest adding

a regularization term to the objective function of asymmetric nonnegative matrix factorization as

$$f(\mathbf{W},\mathbf{H}) = \min_{\mathbf{W},\mathbf{H}\geq\mathbf{0}} ||\mathbf{A} - \mathbf{W}\mathbf{H}\mathbf{W}^T||_F^2 + \frac{\lambda}{2}\omega(\mathbf{W}), \tag{3.5}$$

where ω is a penalty that varies based on the information being considered. Since the similarity between two nodes in the original space is considered as prior information to our clustering algorithm, $\omega(\mathbf{W})$ can be written as follows:

$$\omega(\mathbf{W}) = \sum_{i=1}^{n}\sum_{j=1}^{n} d(\mathbf{w}_i,\mathbf{w}_j)[\mathbf{S}]_{ij}, \tag{3.6}$$

where $\mathbf{S}$ is the similarity matrix whose i,j entry, $[\mathbf{S}]_{ij}$, denotes the similarity of nodes i and j and $d(\mathbf{w}_i,\mathbf{w}_j)$ is the distance between representatives of nodes i and j in new basis.

The distance between two nodes in a graph indicates how close the nodes are based on the structural information of the graph. Nodes in the same cluster are supposed to be closer to each other than the nodes in different clusters. A larger closeness between two nodes in a graph means a shorter distance between them. The typical method to measure the closeness of two data points is Euclidean distance. We use Euclidean distance to obtain the closeness of the representations of two nodes in new basis as

$$d(\mathbf{w}_i,\mathbf{w}_j) = ||\mathbf{w}_i - \mathbf{w}_j||^2. \tag{3.7}$$

There are several similarity measures to obtain the similarity matrix $\mathbf{S}$. The adjacency matrix is the simplest way to present the similarity between nodes. Simply, $[\mathbf{S}]_{ij}$ is 1 if there is an edge directed from node i to node j, and 0 otherwise. This is commonly known as the unweighted adjacency matrix. One can consider weighted adjacency matrix such that $[\mathbf{S}]_{ij}$ is the number of edges directed from node i to node j.

Katz centrality is also one of the commonly used similarity measures to compute similarity between two nodes, which counts every path in a given graph with a weighting

scheme and it can be shown as

$$\mathbf{S} = \sum_{l=0}^{\infty} (\beta \mathbf{A})^l = (\mathbf{I} - \beta \mathbf{A})^{-1}, \tag{3.8}$$

where $0 \leq \beta \leq 1$ is a weight parameter, which guarantees that longer paths have less value than shorter ones (Newman, 2018). Using Katz centrality, nodes i and j are similar when they are connected either by a few short paths or by many long paths.

Another way of measuring similarity between two nodes is counting the number of common neighbors. Cosine similarity does it with the following formulation:

$$[\mathbf{S}]_{ij} = \frac{c_{ij}}{\sqrt{o_i o_j}}, \tag{3.9}$$

where o_i is the number of edges node i has and c_{ij} is the number of common neighbors of nodes i and j. Cosine similarity of nodes i and j is the ratio of the number of common neighbors of these two nodes to the geometric mean of their degrees (Newman, 2018).

Different similarity measures can be used for different situations. Since $[\mathbf{S}]_{ij}$ is for only measuring the pairwise similarity between nodes i and j, we do not treat similarity measures differently in the following description.

Considering similarity measures and given an adjacency matrix $\mathbf{A} \in \Re^{n \times n}$, our RANMF algorithm aims to solve the following optimization problem.

$$f(\mathbf{W}, \mathbf{H}) = \min_{\mathbf{W}, \mathbf{H} \geq \mathbf{0}} ||\mathbf{A} - \mathbf{W}\mathbf{H}\mathbf{W}^T||_F^2 + \frac{\lambda}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} ||\mathbf{w}_i - \mathbf{w}_j||^2 [\mathbf{S}]_{ij}. \tag{3.10}$$

By adding the second term on the right hand side of (3.10), we consider both similarity of two nodes in original space and closeness of their representations in new basis. Similarity of nodes i and j in the original space leads their representatives to be close to each other. The regularization term in (3.10) can be rewritten as:

$$\frac{\lambda}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} ||\mathbf{w}_i - \mathbf{w}_j||^2 [\mathbf{S}]_{ij} = \lambda \mathrm{Tr}(\mathbf{W}^T \mathbf{D} \mathbf{W}) - \lambda \mathrm{Tr}(\mathbf{W}^T \mathbf{S} \mathbf{W}),$$

where $\mathrm{Tr}(\cdot)$ is the trace of a matrix (for the detailed derivation of regularization term,

please see the Appendix B). Thus, (3.10) can be rewritten as:

$$\min_{\mathbf{W},\mathbf{H}\geq\mathbf{0}} ||\mathbf{A}-\mathbf{WHW}^T||_F^2 + \lambda \mathrm{Tr}(\mathbf{W}^T\mathbf{DW}) - \lambda \mathrm{Tr}(\mathbf{W}^T\mathbf{SW}), \tag{3.11}$$

where $\mathbf{D}$ is diagonal matrix with diagonal entry d_i, which is the sum of ith row of $\mathbf{S}$. Then $\mathbf{W}$ and $\mathbf{H}$ can be solved by the following iterative multiplicative updating rules:

$$\begin{aligned}
[\mathbf{W}]_{ik} &\leftarrow [\mathbf{W}]_{ik}\left(\frac{[\mathbf{AWH}^T + \mathbf{A}^T\mathbf{WH} + \lambda\mathbf{S}^T\mathbf{W}]_{ik}}{[\mathbf{WHW}^T\mathbf{WH}^T + \mathbf{WH}^T\mathbf{W}^T\mathbf{WH} + 2\lambda\mathbf{D}^T\mathbf{W}]_{ik}}\right)^{\frac{1}{4}} \\
[\mathbf{H}]_{kj} &\leftarrow [\mathbf{H}]_{kj}\frac{[\mathbf{W}^T\mathbf{AW}]_{kj}}{[\mathbf{W}^T\mathbf{WHW}^T\mathbf{W}]_{kj}}.
\end{aligned} \tag{3.12}$$

For detailed derivations of multiplicative update rules, please see the Appendix C.

Theorem 1. The objective function in (3.11) is nonincreasing under the updating rules in (3.12).

For the detailed proof of Theorem 1, please see the Appendix D. Theorem 1 shows that multiplicative updating rules in (3.12) will converge to a stationary point.

3.3.2 SVD based initialization

Most NMF based algorithms use random initialization to set the values of factorized matrices. In general, random initialization requires running the algorithm several times to obtain a stable solution, since single run can generate bad initial matrices. This approach may be computationally inefficient when the dataset is large. In order to overcome this issue for general NMF problems, Boutsidis and Gallopoulos (2008) propose an initialization algorithm, Nonnegative Double Singular Value Decomposition (NNDSVD), based on the SVD, which computes the approximation of a matrix $\mathbf{A}$ factorized by two nonnegative matrices with rank r. The NNDSVD first computes a decomposition of $\mathbf{A}$ using the SVD, where $\mathbf{A} = \mathbf{USV}^T$, then extracts positive sections of respective matrices and initializes the NMF. The NNDSVD leads to rapid error reduction and faster convergence than random initialization. In this study, we modify the NNDSVD procedure for the RANMF algorithm. When $\mathbf{A}$ is an $n \times n$ matrix and $\mathbf{U}$, $\mathbf{S}$, and $\mathbf{V}$ are also $n \times n$ matrices. Since $\mathbf{U}$ and $\mathbf{V}$ orthogonal matrices, the SVD

can be represented as $\mathbf{A} = \mathbf{U}(\mathbf{S}\mathbf{V}^T\mathbf{U})\mathbf{U}^T$. Let $[\mathbf{Z}]^+_{*j}$ be the nonnegative section of the jth column vector of $\mathbf{Z} \in \Re^{n\times n}_+$ and $[\mathbf{Z}]^-_{*j}$ be the nonpositive section of $[\mathbf{Z}]_{*j}$, then $[\mathbf{Z}]_{*j} = [\mathbf{Z}]^+_{*j} - [\mathbf{Z}]^-_{*j}$. We also define $[\mathbf{Z}]_{*,1:r}$ as the first r columns of matrix $\mathbf{Z}$ and $[\mathbf{Z}]_{1:r,1:r}$ as the first r rows and columns of $\mathbf{Z}$. Then, we obtain the initial matrices $\mathbf{W}_0$ and $\mathbf{H}_0$ as shown in Algorithm 3.

Algorithm 3 SVD-based initialization for RANMF

Given: Adjacency matrix $\mathbf{A}$ and factorization rank r

Compute SVD of $\mathbf{A}$: $\mathbf{U}\mathbf{S}\mathbf{V}^T = \mathbf{A}$
for $j = 1 : n$ **do**
  Set $u^+ = [\mathbf{U}]^+_{*j}$, $v^+ = [\mathbf{V}]^+_{*j}$, $u^- = [\mathbf{U}]^-_{*j}$, $v^- = [\mathbf{V}]^-_{*j}$.
  if $||u^+|| \cdot ||v^+|| \geq ||u^-|| \cdot ||v^-||$ **then**
    $u = u^+/||u^+||$
    $v = v^+/||v^+||$
    $\sigma = ||u^+|| \cdot ||v^+||$
  else
    $u = u^-/||u^-||$
    $v = v^-/||v^-||$
    $\sigma = ||u^-|| \cdot ||v^-||$.
  end if
  Set $[\tilde{\mathbf{U}}]_{*j} = u$, $[\tilde{\mathbf{V}}]_{*j} = v$, and $[\tilde{\mathbf{S}}]_{jj} = \sigma[\mathbf{S}]_{jj}$
end for
Set $\mathbf{W}_0 = [\tilde{\mathbf{U}}]_{*,1:r}$ and $\mathbf{H}_0 = [\tilde{\mathbf{S}}\tilde{\mathbf{V}}^T\tilde{\mathbf{U}}]_{1:r,1:r}$

Algorithm 4 provides the overall procedure of our proposed algorithm. Given a directed adjacency matrix $\mathbf{A}$, similarity matrix $\mathbf{S}$, factorization rank r and a stopping criteria, our proposed RANMF algorithm first obtains the initial matrices $\mathbf{W}_0$ and $\mathbf{H}_0$ using SVD based initialization. Then updates the $\mathbf{W}$ and $\mathbf{H}$ matrices using multiplicative updating rules in (3.12) until a predetermined stopping criteria is met. Finally, it returns the $\mathbf{W}$ and $\mathbf{H}$ matrices.

Algorithm 4 Regularized asymmetric nonnegative matrix factorization

1: **procedure** RANMF($\mathbf{A} \in \Re^{n \times n}$, $\mathbf{S} \in \Re^{n \times n}$, $1 \leq r \leq n$, ϵ)
2: SVD-based initialization of $\mathbf{W}_0$, $\mathbf{H}_0$
3: **repeat**
4: Update $\mathbf{W}$ as

$$[\mathbf{W}]_{ik} \leftarrow [\mathbf{W}]_{ik} \left(\frac{[\mathbf{AWH}^T + \mathbf{A}^T\mathbf{WH} + \lambda \mathbf{S}^T\mathbf{W}]_{ik}}{[\mathbf{WHW}^T\mathbf{WH}^T + \mathbf{WH}^T\mathbf{W}^T\mathbf{WH} + 2\lambda \mathbf{D}^T\mathbf{W}]_{ik}} \right)^{\frac{1}{4}}$$

5: Update $\mathbf{H}$ as

$$[\mathbf{H}]_{kj} \leftarrow [\mathbf{H}]_{kj} \frac{[\mathbf{W}^T\mathbf{AW}]_{kj}}{[\mathbf{W}^T\mathbf{WHW}^T\mathbf{W}]_{kj}}$$

6: **until** $f(\mathbf{W},\mathbf{H}) < \epsilon$
7: **return** $\mathbf{W}$ and $\mathbf{H}$
8: **end procedure**

3.3.3 An illustrative example

In this section, we present an illustrative example for better understanding of our RANMF algorithm. Figure 3.2 shows 13-node sample graph used in this illustration. This example is suitable for a manual detection of expected clusters in a systematical way.

It is attempting to group this sample graph into two clusters, C_1 and C_2. When we group the graph into two clusters, one can expect that node 1 belongs to either C_1 or C_2 since it is the root node and linked by all the other nodes directly or indirectly; nodes 2, 4, 5, 8, 9, and 11 belong to C_1; nodes 3, 6, 7, 10, 12, and 13 belong to C_2.

We first obtain initial matrices $\mathbf{W}_0$ and $\mathbf{H}_0$ using modified SVD method. Then we used initial matrices in our RANMF method to approximate $\mathbf{A}$ with factorization rank 2 and regularization parameter $\lambda = 100$. The results shown in Table 3.1 demonstrates that our RANMF algorithm performs well to meet the expectations for this sample graph.

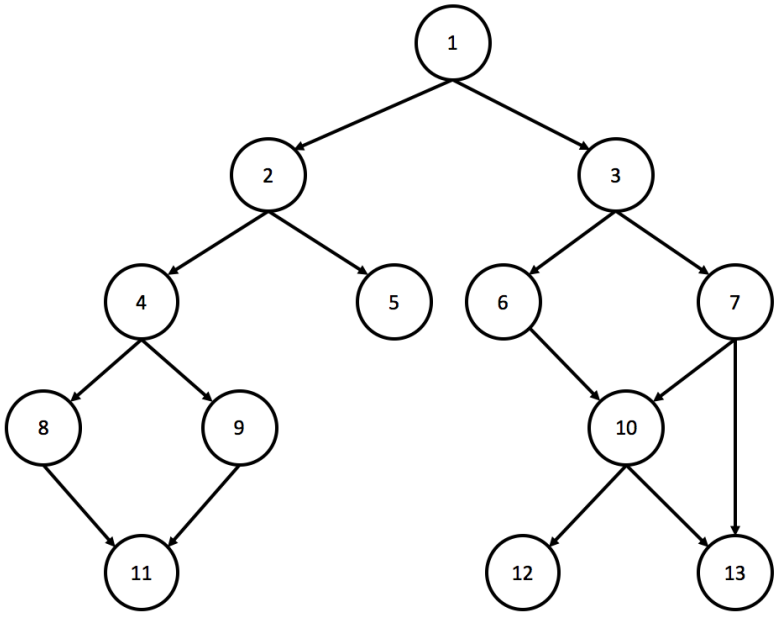

Figure 3.2: 13-node sample graph

Table 3.1: RANMF clustering results for 13-node sample graph

Node	Expected Cluster	RANMF Cluster
1	C_1 or C_2	C_1
2	C_1	C_1
3	C_2	C_2
4	C_1	C_1
5	C_1	C_1
6	C_2	C_2
7	C_2	C_2
8	C_1	C_1
9	C_1	C_1
10	C_2	C_2
11	C_1	C_1
12	C_2	C_2
13	C_2	C_2

3.4 Experiments

In this section, we present experimental results based on real-world and synthetic datasets. We compare our proposed RANMF algorithm with ANMF, community detection by spectral clustering (Hespanha, 2004), and NCut (Shi and Malik, 2000). We also compare it with random prediction. Sensitivity and convergence analyses for the proposed algorithm is provided.

Different performance measures are employed to evaluate the clustering results of the algorithms. We use distance-based quality function (Pitsoulis) and Davies-Bouldin (DB) index (Davies and Bouldin, 1979) for datasets for which true labels are not available. Distance-based quality function measures the average distance between clusters of an algorithm. Higher the score, the more separated the clusters. Higher scores mean better clusters. DB index is the within cluster scatterness divided by the between cluster separation. Therefore, a lower DB score means better clustering. As for the datasets for which the true labels are available, we use Jaccard similarity, Normalized Mutual Information (NMI) (Danon et al., 2005), and accuracy. Jaccard similarity is a class-specific measure. For a given cluster, it is defined as the ratio of the predicted and true labels intersection to their union. Accuracy is defined as the ratio of the number

of correctly labeled nodes to the total number of nodes. NMI is open to information-theoretically interpretation. It is defined as the amount of information that we get if we know the cluster labels. Higher scores of Jaccard, NMI, and accuracy mean better clusters.

As for the implementation of the proposed method, we first construct an $n \times n$ directed adjacency matrix, where n is the total number of nodes in a given dataset. Then, we obtain the similarity matrix $\mathbf{S}$ using one of the similarity measures explained in Section 3.3.1. After initializing $\mathbf{W}$ and $\mathbf{H}$ with SVD based initialization, we iteratively update them until convergence using the updating rules in (3.12). As for the distance measure, we used the Euclidean distance measure.

3.4.1 Patent citation network

Patents are needed to be cited like any other resources such as books, journal articles, etc. when referenced in a document. This citation contains useful information for readers to understand the relationship between corresponding patent and other patents. Citation between two patents implies that citing patent is related to cited patent in some way (Rodriguez et al., 2016). We show the performance of our algorithm using patent citation network (PCN) with 149 nodes (patents) and 215 directed edges (citations). PCN is an example of directed acyclic networks in which there is no way to loop back to a node i if we start at node i. Figure 3.3 shows the network structure of the PCN dataset.

A citation to a previously published patent indicates an extension of the previous technology or art. In a PCN, a direct citation of a patent is manifested as the use of its neighbor information, and an indirect citation is as the use of information from a non-immediate neighbor connected through one or more intermediate patents. Using only immediate neighbor information is not enough to capture the similarities between patents. Because direct citations indicated extension of recent technologies and indirect citations indicate technological change over time. Therefore, we use Katz similarity measure with $\beta = 0.1$ to obtain the similarity matrix $\mathbf{S}$ in order to capture the indirect citations.

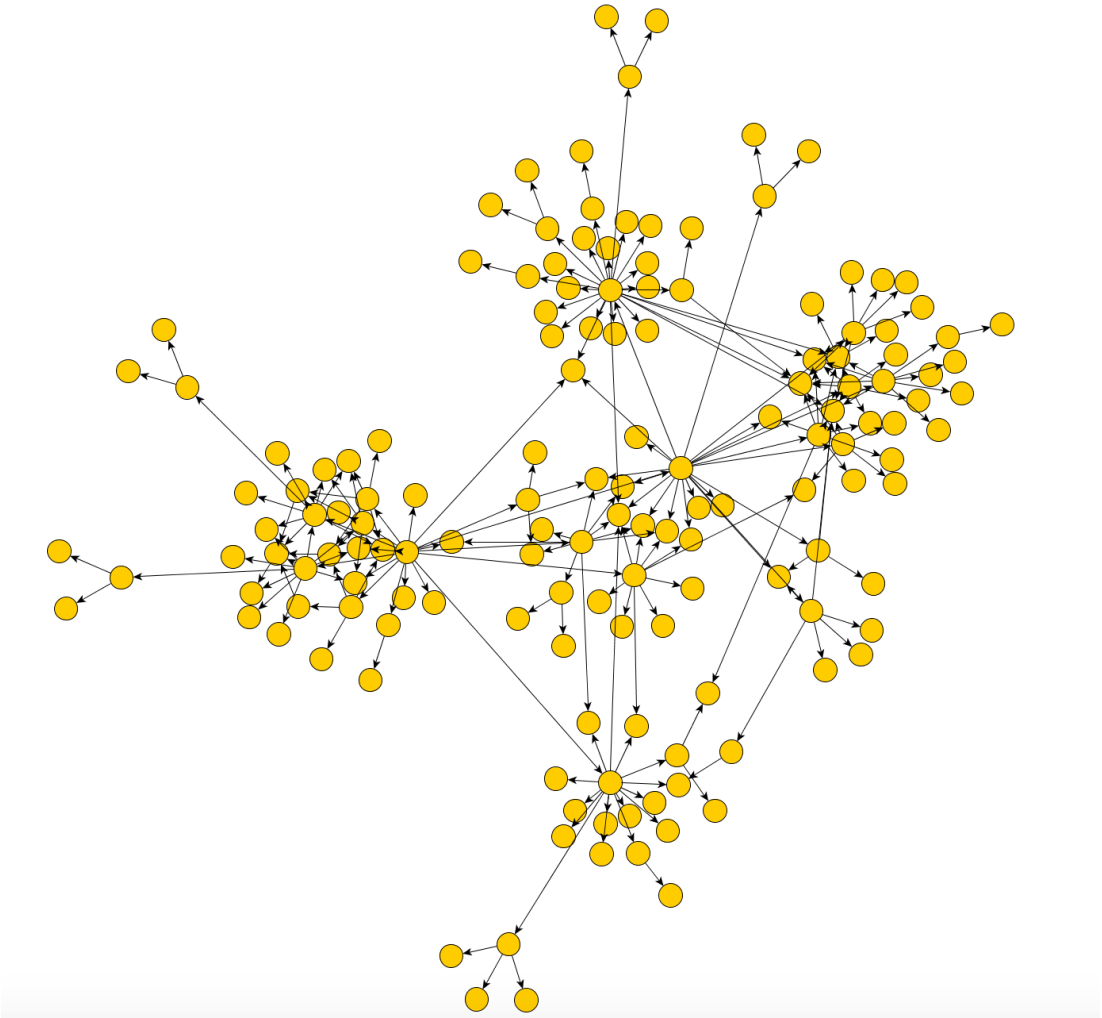

Figure 3.3: Network structure of US patent citation dataset.

Table 3.2: Comparison results of algorithms on PCN with different r values in terms of distance-based quality function. Rnd and SVD stand for random initialization and SVD-based initialization, respectively.

	$r = 2$	$r = 3$	$r = 4$	$r = 5$	$r = 6$	$r = 10$	$r = 15$
Random Prediction	0.502	0.340	0.259	0.210	0.178	0.113	0.080
ANMF Rnd	0.763	0.490	0.390	0.317	0.271	0.181	0.134
ANMF SVD	0.772	0.499	0.426	0.310	0.267	**0.234**	0.167
NCut	0.583	0.346	0.261	0.243	0.221	0.152	0.107
Spect	0.503	0.358	0.298	0.211	0.168	0.164	0.103
RANMF Rnd	0.761	0.441	0.329	0.271	0.230	0.159	0.125
RANMF SVD	**0.802**	**0.558**	**0.439**	**0.319**	**0.295**	0.177	**0.187**

Table 3.3: Comparison results of algorithms on PCN with different r values in terms of DB index.

	r=2	r=3	r=4	r=5	r=6	r=10	r=15
Random Prediction	3.413	2.058	1.421	1.061	0.846	0.444	0.289
ANMF Rnd	1.468	1.500	1.194	0.935	0.751	1.123	8.042
ANMF SVD	1.431	**1.127**	1.123	0.902	0.735	0.825	0.303
Spect	2.488	1.488	1.242	0.917	0.783	0.426	0.255
Ncut	2.864	2.122	1.273	1.298	0.794	0.541	0.346
RANMF Rnd	1.458	1.621	1.189	0.890	0.721	0.410	2.397
RANMF SVD	**1.418**	1.264	**0.880**	**0.867**	**0.713**	**0.341**	**0.243**

Tables 3.2 and 3.3 show the comparison results of clustering methods on PCN dataset with varying number of clusters in terms of distance-based quality score and DB index, respectively. For RANMF-SVD, we used $\lambda = 0.1$ and Katz similarity measure with $\beta = 0.1$. For RANMF-Rnd, we ran 100 instances of the algorithm with the same parameters used for RANMF-SVD and different initial matrices at each time. For ANMF-Rnd, we ran 100 instances of the algorithm with different initial matrices at each time. For random prediction, we randomly assigned each node to a cluster and repeated this process 100 times. Results show that the clusters produced by our proposed method are well separated and better than the clusters produced by the other methods and random guessing in terms of within cluster scatter. Overall, NMF based methods outperforms NCut and Spec clustering algorithms. To show the effect of initialization approaches for NMF algorithms, we also ran the ANMF and RANMF algorithms using SVD-based initialization and random initialization. Results reveal that SVD-based initialization improves the performances of each method in terms of distance-based quality score and DB index. In Table 3.2 when $r = 10$ and in Table 3.3 when $r = 3$, ANMF with SVD-based initialization has better results than of RANMF with SVD-based initialization. This might be the effect of using Katz similarity as considering non-immediate neighbors increases the between-cluster distances and within-cluster scatterness.

3.4.2 World wide knowledge base datasets

World Wide Knowledge Base (WebKB) datasets contain web pages collected from 4 universities (Cornell, Wisconsin, Texas, and Washington). Nodes and directed edges represent web pages and link information between web pages, respectively. Web pages are classified into 5 categories including student, course, project, faculty and staff. Figures 3.4-3.7 show the network structure of WebKB datasets.

A good cluster is the one with less inter-cluster connectivity and more intra-cluster connectivity. However, inter-cluster connectivity is very high in WebKB datasets, which means that nodes have more connectivity between clusters than the within clusters. This situation makes it difficult to detect the clusters for the clustering algorithms. In

this kind of structures, using m-level links might be misleading to obtain the similarities between nodes. Therefore, we use cosine similarity measure to obtain the similarity matrix. The convergences of our RANMF algorithm for 5 datasets are shown in Fig 3.8. For Fig. 3.8, we ran RANMF algorithm with SVD-based initialization (with aforementioned parameters for each dataset) until iteration number 200. At each iteration, we observed the logarithm of the objective function in (3.11). Overall, the objective value curves of RANMF converge on the stationary values are very fast.

Table 3.4 and 3.5 show the comparison results of clustering algorithms for WebKB datasets. For RANMF-SVD, we used Cosine similarity measure on all WebKB datasets; $\lambda = 10$ on Cornell; $\lambda = 1250$ on Wisconsin; $\lambda = 1$ on Washington; and $\lambda = 3220$ on Texas. Overall, our proposed method outperforms other methods with varying values but we chose best values between 0 and 5000 for each dataset. For RANMF-Rnd, we ran 100 instances of the algorithm with the same parameters used for RANMF-SVD on WebKB datasets and different initial matrices at each time. For ANMF-Rnd, we ran 100 instances of the algorithm with different initial matrices at each time. For random prediction, we randomly assigned each node to a cluster and repeated this process 100 times. The results show that RANMF is better than the other clustering algorithms and random guessing in terms of Jaccard similarity, NMI, and accuracy indices. The results reveal that SVD-based initialization improves the performance of the proposed method in all datasets except the Cornell dataset. The Cornell dataset is more mixed compared to the other datasets, which means that inter-cluster connectivity is very high. In this kind of highly mixed structures, random initialization might perform better because in some of the multiple runs, the algorithm might find a better local minimum. Results also reveal that SVD-based initialization improves the performance of ANMF algorithm in most cases.

We also present the accuracy of RANMF method with different λ values for Cornell, Washington, Texas, and Wisconsin datasets. Figures 3.9-3.12 show that our proposed method is stable with respect to λ and outperforms the other methods with varying λ from 0.1 through 5000. For Figures 3.9-3.12, we ran the RANMF algorithm with

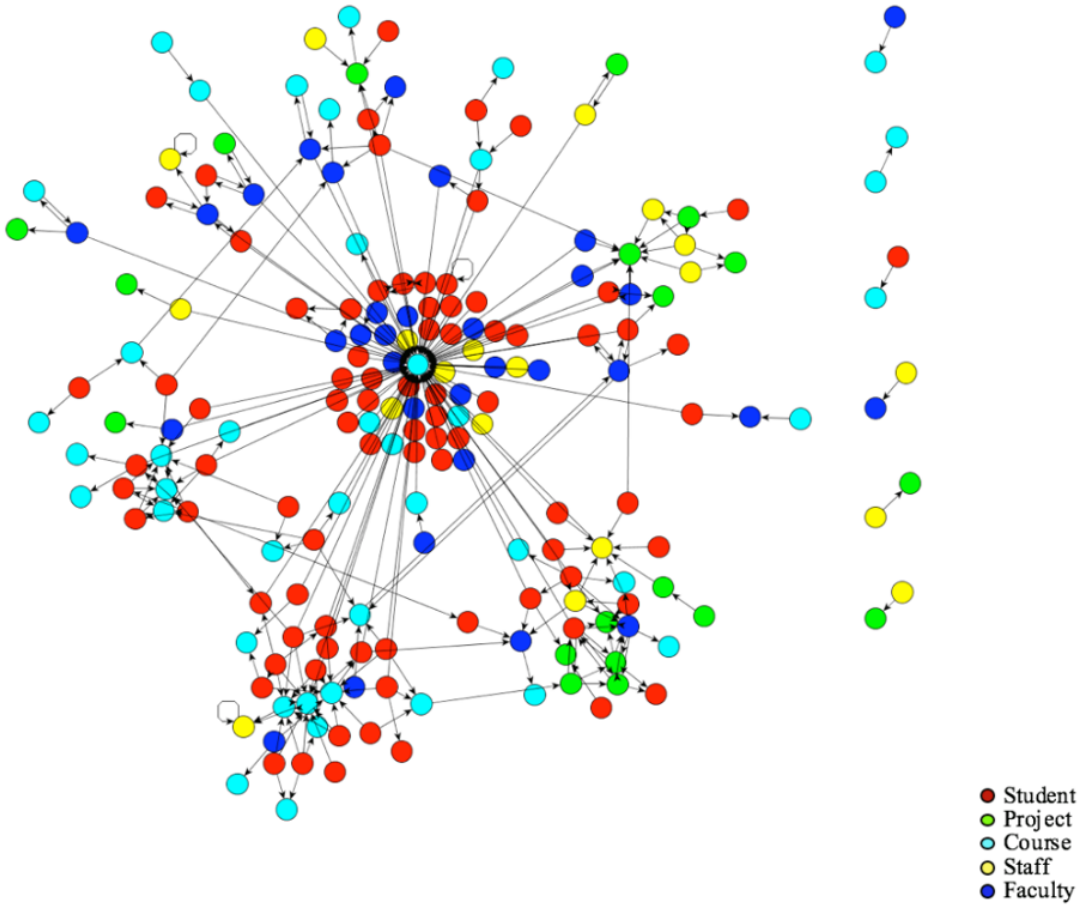

Figure 3.4: Network structure of Cornell dataset.

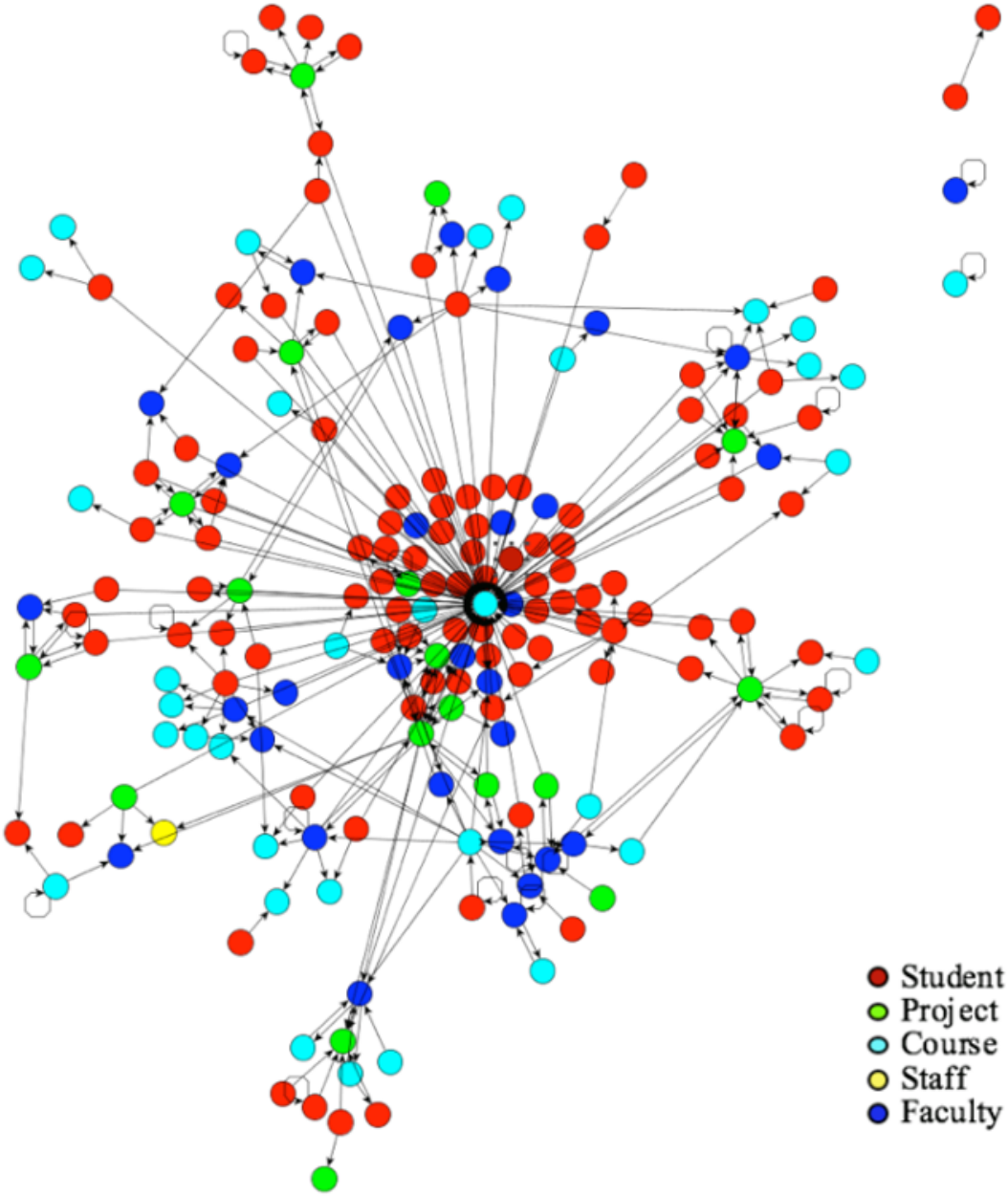

Figure 3.5: Network structure of Texas dataset.

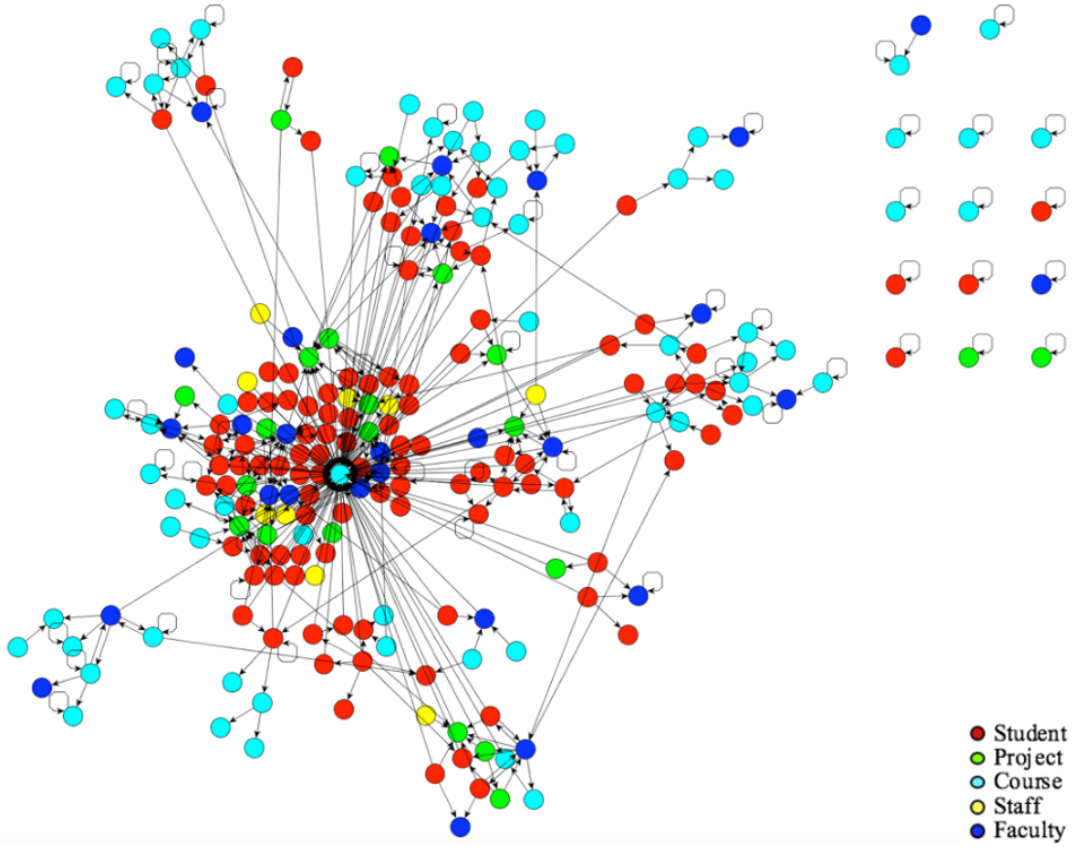

Figure 3.6: Network structure of Washington dataset.

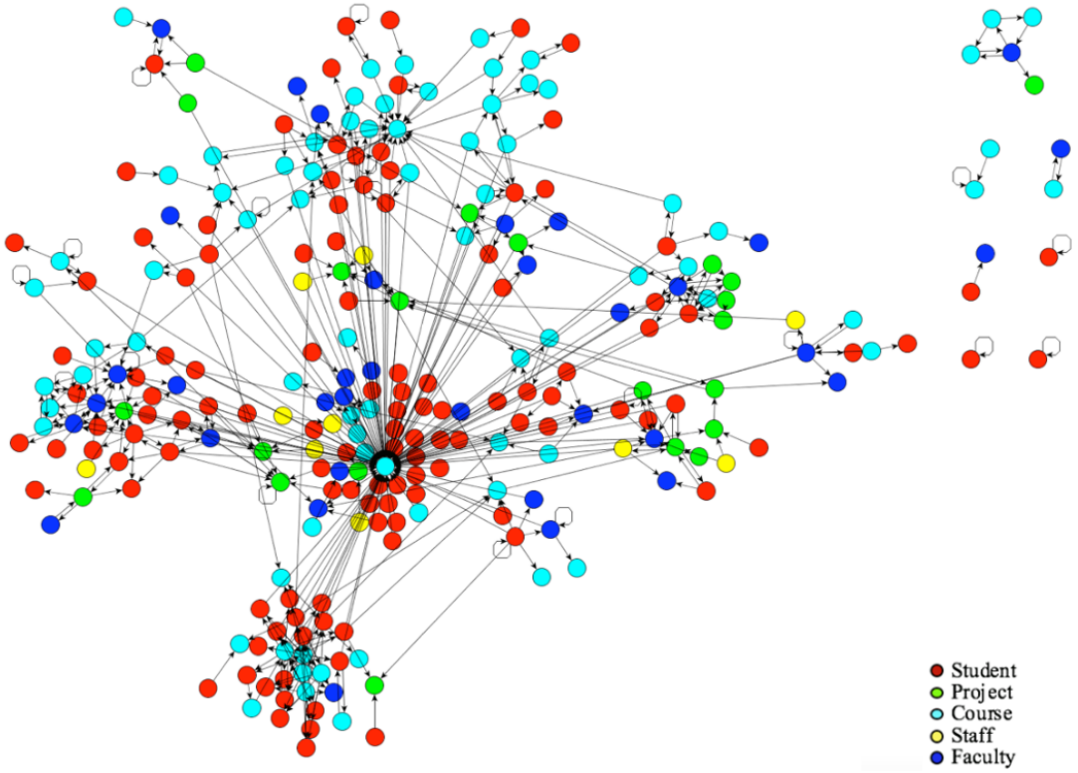

Figure 3.7: Network structure of Wisconsin dataset.

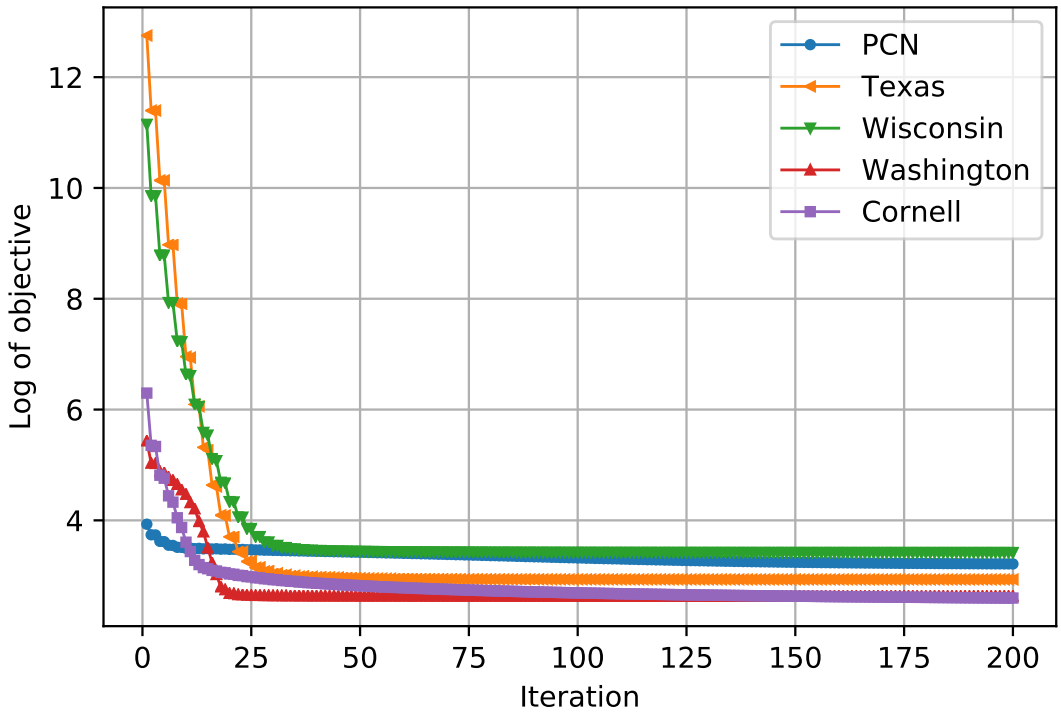

Figure 3.8: Convergence curves of log of the objective function of RANMF algorithm for PCN and WebKB datasets.

Table 3.4: Comparison results of clustering methods using WebKB datasets. $|V|$ is the number of nodes, $|E|$ is the number of edges, and r is the number of clusters.

	Cornell $\|V\| = 195$, $\|E\| = 304$, $r = 5$			Texas $\|V\| = 187$, $\|E\| = 328$, $r = 5$		
	Jaccard	NMI	Accuracy	Jaccard	NMI	Accuracy
Random Prediction	0.129	0.029	0.268	0.148	0.029	0.266
ANMF Rnd	0.183	0.143	0.363	0.213	0.147	0.367
ANMF SVD	0.187	0.097	0.353	0.347	**0.224**	0.550
Spect	0.189	0.042	0.323	0.218	0.024	0.342
NCut	0.132	0.016	0.277	0.149	0.018	0.262
RANMF Rnd	**0.282**	**0.167**	**0.455**	0.346	0.177	0.529
RANMF SVD	0.203	0.127	0.379	**0.416**	0.194	**0.594**

Table 3.5: Comparison results of clustering methods using WebKB datasets.

	Wisconsin $\|V\| = 265$, $\|E\| = 530$, $r = 5$			Washington $\|V\| = 230$, $\|E\| = 446$, $r = 5$		
	Jaccard	NMI	Accuracy	Jaccard	NMI	Accuracy
Random Prediction	0.140	0.023	0.257	0.140	0.024	0.259
ANMF Rnd	0.165	0.068	0.315	0.195	0.139	0.355
ANMF SVD	0.225	0.076	0.422	0.240	0.100	0.452
Spect	0.208	0.058	0.404	0.28	0.076	0.457
NCut	0.157	0.034	0.294	0.155	0.04	0.304
RANMF Rnd	0.239	0.078	0.457	0.286	0.169	0.475
RANMF SVD	**0.270**	**0.085**	**0.502**	**0.375**	**0.209**	**0.543**

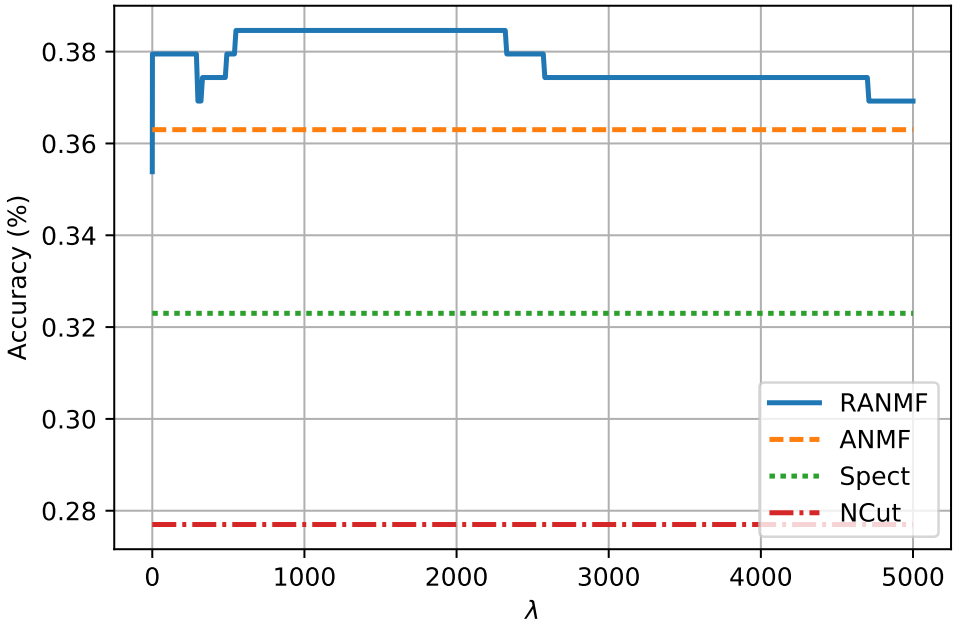

Figure 3.9: Accuracy score of RANMF using λ from 0.1 to 5000 using Cornell dataset.

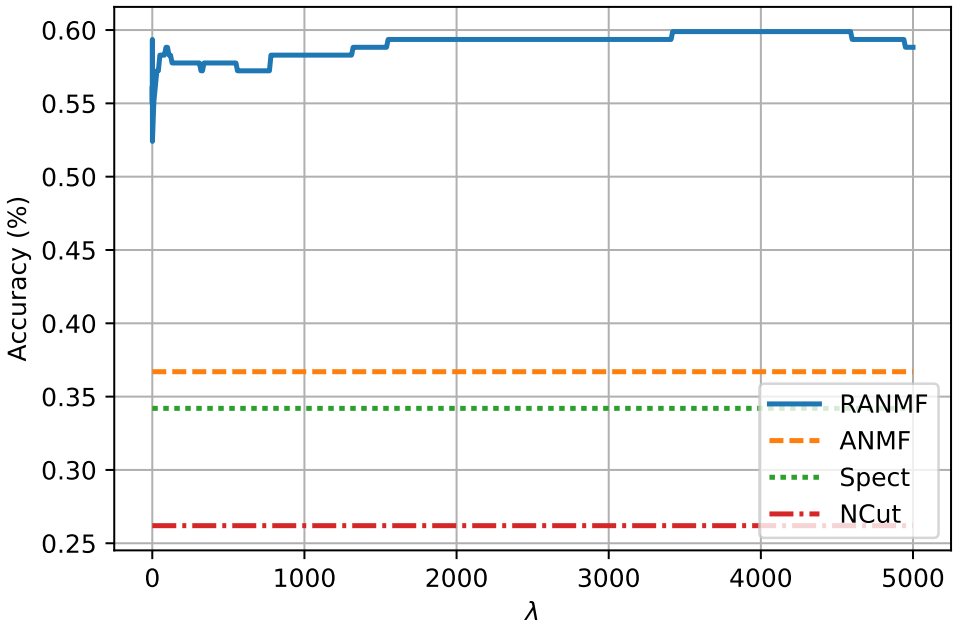

Figure 3.10: Accuracy score of RANMF using λ from 0.1 to 5000 using Texas dataset.

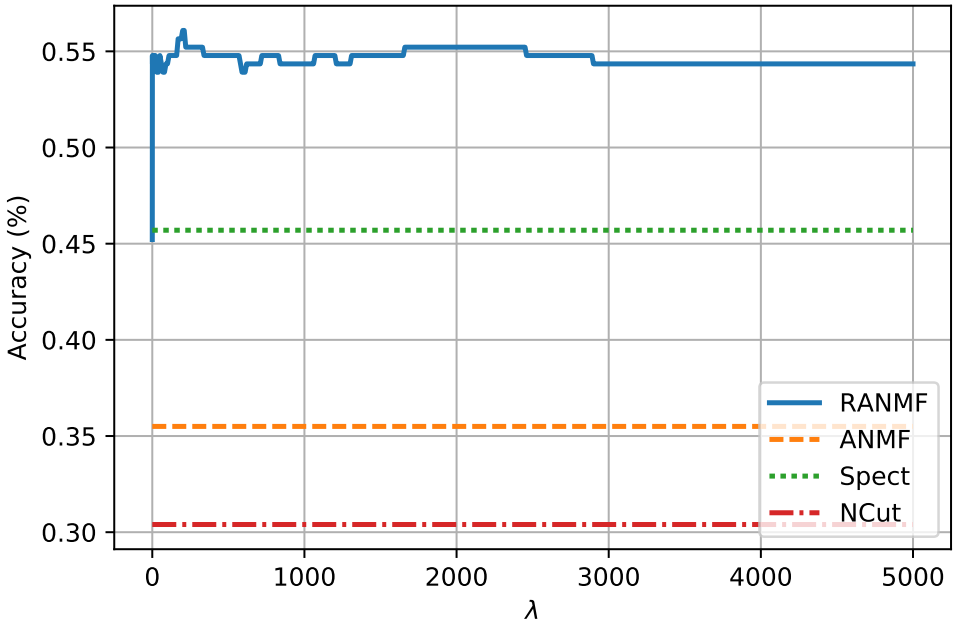

Figure 3.11: Accuracy score of RANMF using λ from 0.1 to 5000 using Washington dataset.

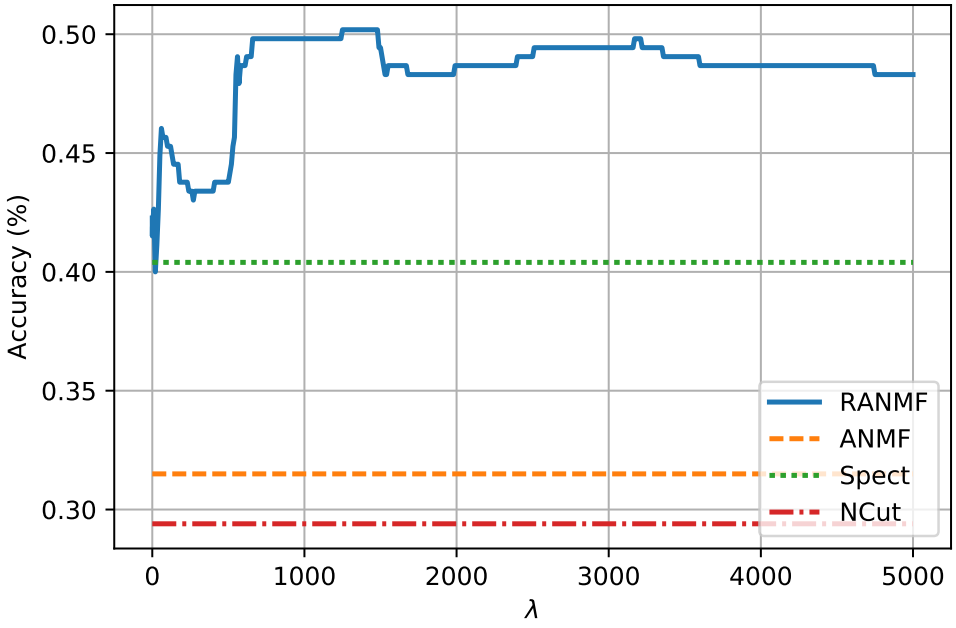

Figure 3.12: Accuracy score of RANMF using λ from 0.1 to 5000 using Wisconsin dataset.

SVD-based initialization (with aforementioned parameters for each dataset) with varying λ values between 0 and 5000. When λ is high, the performance of our proposed method is still good. Because the penalty term in (3.11) tends to revise the columns of SVD-initialized $\mathbf{W}$ matrix only for the high values of $\mathbf{S}$. If two nodes are highly similar to each other in a network, then the penalty term will force the representations of those nodes to be close to each other.

3.4.3 LFR synthetic graphs

In this section, we compared the clustering algorithms on LFR synthetic benchmark graphs which are generated as described in (Lancichinetti and Fortunato, 2009). The LFR benchmark graphs mimic the real-world networks by accounting for the heterogeneity of degree and cluster size. Therefore, like the real-world networks, the LFR graphs have a skewed node degree distribution and a broad distribution of cluster sizes with a tail that can be approximated by a power law, which results in clusters with very different sizes (Lancichinetti and Fortunato, 2009). In LFR graphs, it is possible to control the structure of the synthetic graph by mixing parameter μ. Mixing parameter controls the inter-cluster connectivity. Larger mixing paratemer means more inter-cluster connectivity, which makes it more difficult to detect clusters for clustering algorithm. We created three different network structures using three different mixing parameters ($\mu = 0.1, 0.3, 0.5$).

Tables 3.6-3.8 show the comparison results of algorithms on synthetic graphs. For RANMF-SVD, we used $\lambda = 0.1$. For RANMF-Rnd, we ran 20 instances of the algorithm with the same parameters used for RANMF-SVD on LFR graphs and different initial matrices at each time. For ANMF-Rnd, we ran 20 instances of the algorithm with different initial matrices at each time. For random prediction, we randomly assigned each node to a cluster and repeated this process 100 times. Since ANMF-SVD was not able to provide 33 clusters, we couln't calculate the NMI for it using LFR graph with $\mu = 0.5$. The results show the superiority of our proposed method over the other methods and random guessing in terms of Jaccard similarity, NMI, and accuracy indices. As for the calculation of $\mathbf{S}$ matrix, we employed different similarity measures. In the case

Table 3.6: Comparison results of clustering methods on LFR graphs. $\mu = 0.1$, $|V| = 1000$, $|E| = 15662$, and $r = 32$

	similarity	Jaccard	NMI	Accuracy
Random Prediction		0.016	0.158	0.102
ANMF Rnd		0.074	0.322	0.183
ANMF SVD		**1.000**	**1.000**	**1.000**
Ncut		0.394	0.886	0.621
Spec		0.619	0.880	0.797
RANMF Rnd		0.192	0.594	0.417
RANMF SVD	cos	**1.000**	**1.000**	**1.000**
RANMF SVD	katz	**1.000**	**1.000**	**1.000**
RANMF SVD	adj	**1.000**	**1.000**	**1.000**

Table 3.7: Comparison results of clustering methods on LFR graphs. $\mu = 0.3$, $|V| = 1000$, $|E| = 15164$, and $r = 31$

	similarity	Jaccard	NMI	Accuracy
Random Prediction		0.016	0.152	0.103
ANMF Rnd		0.105	0.450	0.278
ANMF SVD		0.867	0.976	0.940
Ncut		0.773	0.963	0.86
Spec		0.535	0.828	0.748
RANMF Rnd		0.095	0.432	0.284
RANMF SVD	cos	**0.925**	**0.982**	**0.955**
RANMF SVD	katz	**0.925**	**0.982**	**0.955**
RANMF SVD	adj	**0.925**	**0.982**	**0.955**

Table 3.8: Comparison results of clustering methods on LFR graphs. $\mu = 0.5$, $|V| = 1000$, $|E| = 15249$, and $r = 33$

	similarity	Jaccard	NMI	Accuracy
Random Prediction		0.015	0.167	0.103
ANMF Rnd		0.193	0.582	0.376
ANMF SVD		**0.843**	-	0.912
Ncut		0.367	0.895	0.674
Spec		0.642	0.865	0.815
RANMF Rnd		0.040	0.287	0.176
RANMF SVD	cos	0.768	0.935	0.872
RANMF SVD	katz	0.837	**0.959**	**0.915**
RANMF SVD	adj	0.797	0.949	0.888

compared to the random initialization.

3.5 Conclusion

In this chapter, we propose the RANMF algorithm for clustering in directed networks. The proposed algorithm exploits the prior similarity information and incorporates it as an additional regularization term into ANMF algorithm, which achieves the goal of putting similar nodes in the same cluster and dissimilar nodes in different clusters. In addition, we utilize SVD based initialization rather than random initialization since random initialization is ming. Clustering results using real-world datasets and synthetic datasets demonstrate that our proposed RANMF algorithm outperforms other clustering algorithms in terms of several clustering validity indices.

Despite apparent outperformance of the proposed algorithm, here are some rooms for further research. For example, a network can change dynamically over time. In PCN, new patents appear continuously and new citations are added over time in practice. To address this issue, future work needs to investigate the dynamic characteristics of networks and develop new algorithms.

Chapter 4

A New Time-aware Ranking Method for Patents in Dynamic Patent Citation Network

4.1 Introduction

With the rapid improvements of patent analysis tools, patent citation data has been used for various purposes such as following the evolution of technology innovation. Following the evolution on technology is crucial for firms and significant number of decision makers started to use patent citation data in order to give better decisions comparing to their competitors. Therefore, patent analysis has started to be considered as a significant management tool for firms in order to assess diverse aspects of technological change. It has been used by numerous studies for various purposes such as understanding the relationship between technological growth and economic growth or to evaluate and analyze the firms R&D process, etc.

One of the major problems in patent analysis is to measure the importance of patents in PCN. Ranking patents in importance and identifying the influential ones is an important yet challenging task for understanding the current technological trends and identifying the promising technological activities. Being able to know the influential patents may give some advantages to firms such as helping the firm to evaluate its policy regarding R&D process, helping the firm to assess the level of technology development in a specific area or helping the firm to estimate technological strengths and weaknesses of its competitors.

Patents are needed to be cited like any other resources such as books, journal articles, etc. when referenced in a document. This citation contains useful information for readers to understand the relationship between corresponding patent and other patents. Citation between two patents implies that citing patent is related to cited patent in

some way. In the past, patent citation counts, i.e. the number of citations that a patent receives, have been one of the most important and widely used indicator for patent importance and influence (Hall et al., 2005, Oh et al., 2012). Recently, since patents are highly interdependent, network analysis tools have become popular for patent analysis and have introduced new perspectives. Patent citations can be represented as a network where nodes represent patents and directed edges represent the citations. Thus, the problem of ranking and identifying influential patents in a given patent citation network (PCN) can be solved by centrality metrics concept in network analysis.

Many studies have applied centrality metrics such as degree, closeness, betweenness, and PageRank to rank the patents in influence and importance in a patent citation network (Lukach and Lukach, 2007, Oh et al., 2012). While these methods provide a systematic approach to ranking patents, they do not consider the dynamic characteristic of patent citation networks. However, the patent citation network is an evolving graph, which means that new patents and citations between patents appear over time. Researchers have proposed centrality metrics which are extension of aforementioned centrality metrics for ranking nodes in a dynamic network (Baeza-Yates et al., 2002, Walker et al., 2007). While these metrics consider the dynamic characteristic of network, since they are not designed specifically for patent citation networks, they still fail to distinguish the citing and cited patents in terms of importance which could have useful implications for the value of the cited patent.

In this study, we propose a new time-aware measure for ranking patents in influence and importance. The proposed method is designed for dynamic patent citaion network with directed unweighted edges (citations) between nodes (patents). Instead of using simple adjacency matrix, we defined a new weighted adjacency matrix. Proposed citation weighting scheme exploits the time information of citations and distinguishes them for the importance of cited patent. We did not exploit the time information of citing patent only but also the time information of patents. To this extend, we also introduce a weighting scheme to distinguish the patents based on their ages. To show the performance evaluation of our method, we first form a dynamic patent citation network using the real-world patent citation data from USPTO. We then rank the patents using

our proposed method and other commonly used centrality metrics. Results reveal that our proposed method outperforms other centrality metrics in terms of two performance measures Spearman correlation of rankings and recommendation intensity.

This chapter is presented as follows: we first summarize the patent citation networks. We then introduce the proposed time-aware influence measure and evaluate its performance using a real-world patent citation data from USPTO. Finally, we present the concluding remarks and discuss future work.

4.2 Patent citation network

A patent is a representative of an invention in a specific area and patent analysis is an important task as it relates to managing the relationships between patents and search complexities (Abbas et al., 2014). The rapid growth of patent information has made the patent analysis a vital task for both managerial and legal parties. Thus, patent data has been analyzed for various purposes such as understanding patent trends, technology opportunity discovery, identification of promising technologies, and competitor identification (Abbas et al., 2014).

A patent contains two types of data: structured data and unstructured data. Figure 4.1 partially shows the typical data available in a patent.[1] Unstructured data includes text such as title, description, abstract of the patent. The structured data includes citation information, inventors, application number, family ID, etc.

Recently, network analysis tools received much attention in the area of patent analysis. As shown in Figure 4.1, when a patent is published, it cites previously published related patents and this citation reflects the innovative relationship between citing and cited patents. If there is a citation between two patents, it means that the citing patent is related to cited patent in terms of technological innovation. Expending this idea, citations between patents can be represented by an evolving patent citation network which provides useful information for the innovation process. For example, Figure 4.2 shows a graphical representation of citation between two patents and Figure 4.3 shows

[1]The screenshot has been taken from the website of USPTO.

United States Patent **7,696,027**
Cho , et al. **April 13, 2010**

Method of fabricating display substrate and method of fabricating display panel using the same

Abstract

Disclosed is a method of fabricating a display substrate. A black matrix and a color filter layer are formed on a base substrate, and then a transparent electrode and a photoresist layer pattern are sequentially formed. The transparent electrode is patterned using the photoresist layer pattern as a mask to form a common electrode, and a spacer is formed using the photoresist layer pattern.

Inventors: **Cho; Woo-Sik** (Seoul, **KR**), **Lee; Yun-Seok** (Cheonan-si, **KR**), **Woo; Dong-Won** (Busan-si, **KR**), **Son; Ji-Hyeon** (Seongnam-si, **KR**)
Assignee: ***Samsung*** **Electronics Co., Ltd.** (Suwon-si, **KR**)
Family ID: **39189127**
Appl. No.: **11/856,450**
Filed: **September 17, 2007**

Prior Publication Data

Document Identifier	Publication Date
US 20080070332 A1	Mar 20, 2008

Foreign Application Priority Data

Sep 18, 2006 [KR]	10-2006-0090257

Current U.S. Class: **438/157**; 257/57; 257/72; 257/E21.314; 257/E27.111; 438/149; 438/151; 438/48
Current CPC Class: G02F 1/13394 (20130101); G02F 2001/134318 (20130101); G02F 2001/136236 (20130101)
Current International Class: H01L 29/74 (20060101)
Field of Search: ;438/151,149,609,155-158,479,517 ;257/E27.116,E29.117,E29.147,57-59,72,347

References Cited [Referenced By]

U.S. Patent Documents

2006/0181665	August 2006	Hirota
2007/0042136	February 2007	Ju et al.
2007/0093005	April 2007	Kim et al.
2007/0165179	July 2007	Jang

Figure 4.1: Raw patent data (partially shown).

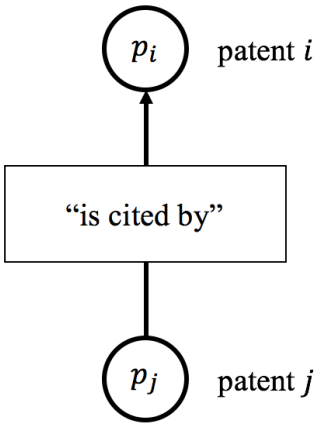

Figure 4.2: Graphical representation of a citation between two patents.

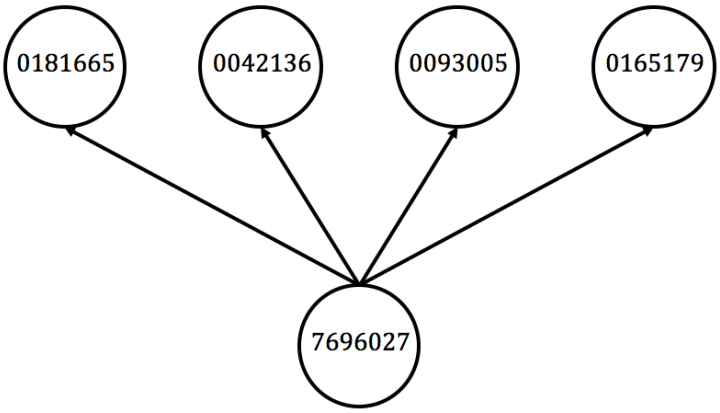

Figure 4.3: Graphical representation of a citations in raw patent data in Figure 4.1.

the citation information depicted in raw patent data in Figure 4.1.

4.3 Proposed time-aware influence measure

Let $G_t = (V_t, E_t)$ represent a dynamic patent citation network, which consists of a set of nodes (patents) V_t and a set of directed edges (citations) E_t at time t. Figure 4.4 shows the evolution of a sample patent citation network over the observation time interval $[0, T]$ as a sequence of non-overlapping time windows $\{[0, 0 + \Delta t_1], [t_1, t_1 + \Delta t_2], \ldots, [t_{M-1}, t_{M-1} + \Delta t_M]\}$, where $T = t_M$, M is the number of time windows, and Δt_m is the length of the mth time window, where $m = 1, 2, \ldots, M$.

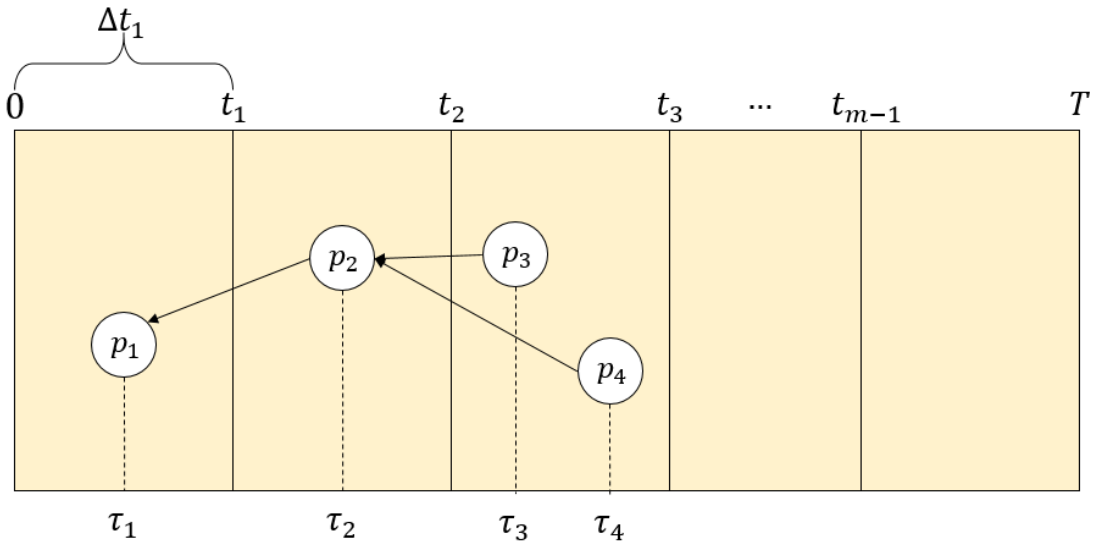

Figure 4.4: Evolution of a sample patent citation network over the time interval $[0, T]$.

Each time window in the evolving graph can be represented by its own time-dependent adjacency matrix $\mathbf{A}_m, m = 1, 2, \ldots, M$. Unlike majority of other dynamic networks, the edges are persistent in the case of patent citation network. Once a citation occurs between two patents, it never disappears. Therefore, adjacency matrix for time window m, $\mathbf{A}_m$, contains the citations occurred in previous time windows. For example, sample PCN in Figure 4.4 can be represented by the following adjacency

matrices as:

$$\mathbf{A}_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \mathbf{A}_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \mathbf{A}_3 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

The number of direct and indirect citations is a strong indicator of a patent's importance. The more citations (giving less weight to indirect citations) a patent has, the more important that patent is. In the case of static network approach, $\mathbf{A}^r$ is used to obtain the number of r-level paths that a patent has. In the case of dynamic network approach, r-level paths of a patent can be obtained by

$$\prod_{m=1}^{r} \mathbf{A}_{M-m+1} \tag{4.1}$$

Since $\mathbf{A}_M$ is strictly lower triangular (all the entries on the main diagonal are 0), Equation (4.1) reduces to $\mathbf{A}_M^r$.

Since it assigns the same value to all citations, the simple adjacency matrix is not enough to distinguish the effect of a citation to a patent's importance. For example, taking the small graph shown in Figure 4.5 into account, if we use the simple adjacency matrix, it is not possible to distinguish effects of citations (p_2, p_1) and (p_3, p_1) to the importance of patent 1. However, the citation from patent 3 to patent 1 should be more important than the citation from patent 2 to patent 1 because patent 3 is published more recently compared to patent 2.

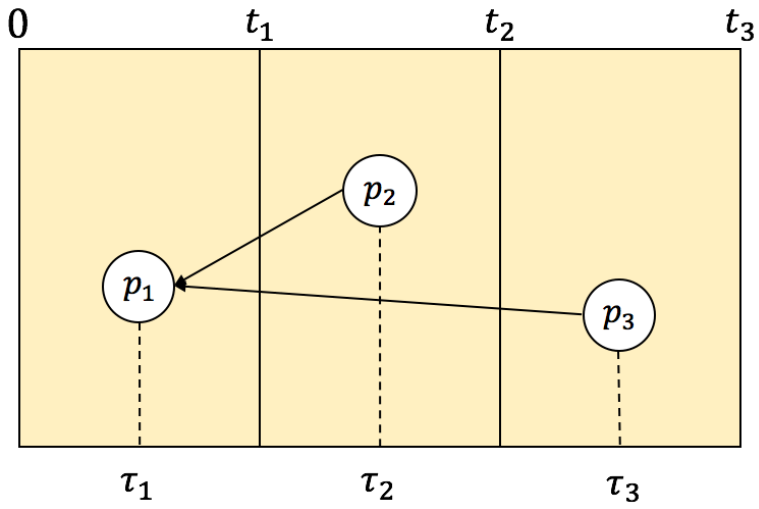

Figure 4.5: Difference in the effect of citation in terms of recency of citing patent.

In order to leverage citations from recently published patents, we propose a weighted influence matrix, which assigns a value to a citation based on a decreasing function. Thus, an element of the weighted influence matrix can be defined as:

$$[\mathbf{W}]_{ji} = \begin{cases} f(\tau_j, T), & \text{if there is a citation from } p_j \text{ to } p_i \\ 0, & \text{otherwise,} \end{cases} \tag{4.2}$$

where $f(\tau_j, T) = e^{-b(T-\tau_j)}$, b is the smoothing parameter, and τ_j is the time that patent j is published. When $b = 0$, time information of citing patent is not considered.

A patent's importance is proportional to the sum of the importance of its citing patents. For instance, if a patent is being cited by many important patents, then it also should be considered as an important patent. With this in mind, we propose a new scoring scheme to identify the important and influential patents in a time-evolving PCN, which can be defined as follows:

$$C_i = \alpha \sum_j [\mathbf{W}]_{ji} C_j + \beta, \tag{4.3}$$

where C_i is the score of patent i, and α and β are constants between 0 and 1. With this scoring scheme, patents with many citations from recently published patents will be considered important.

Older patents naturally might have many direct/indirect citations and Equation (4.3) tends to give more weight to older patents. However, a patent that is relatively new and has many citations should be considered important. Therefore, we add a decreasing function to Equation (4.3) as follows:

$$C_i = \alpha \cdot f(\tau_i, T) \sum_j [\mathbf{W}]_{ji} C_j + \beta, \tag{4.4}$$

where $f(\tau_i, T) = e^{-a \cdot (T - \tau_i)}$, a is the smoothing parameter. When $a = 0$, time information of patent will not be considered. Equation (4.4) can be shown in matrix form as follows:

$$\mathbf{c} = \alpha \mathbf{F} \mathbf{W} \mathbf{c} + \beta \mathbf{1}, \tag{4.5}$$

where $\mathbf{c}$ is n-dimensional vector with elements $C_i, i = 1, 2, \ldots, n$, n is the total number of patents, $\mathbf{F}$ is $n \times n$ diagonal matrix with $[\mathbf{F}]_{ii} = f(\tau_i, T)$, and $\mathbf{1}$ is n-dimensional vector of ones. With this new scoring scheme, recently published patents with many citations will be considered important and will have high centrality score.

4.3.1 Illustrative example

In this section, we use a small graph to illustrate the proposed centrality metric for better understanding of it. Figure 4.6 shows the small graph that is used for illustration. In this graph, time values are in years and $T = 3$.

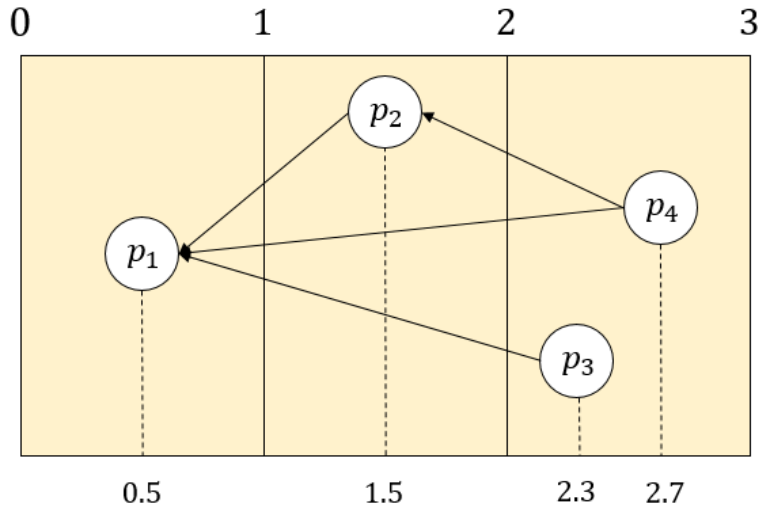

Figure 4.6: 4-node sample graph.

We first obtain the influence score of each citation and weight of patents in our sample PCN as shown in Equations (4.2) and (4.4). Tables 4.1 and 4.2 show the influence scores of citations and weights of patents, respectively.

Table 4.1: Influence scores of citations in sample graph.

Citing	Cited	Influence ($b = 0.5$)
p_2	p_1	0.472
p_3	p_1	0.704
p_4	p_1	0.860
p_4	p_2	0.860

Table 4.2: Weights of patents based on their age in years.

Patent	Weight ($a = 0.3$)
p_1	0.472
p_2	0.637
p_3	0.810
p_4	0.913

Based on the influence scores and weights shown in Tables 4.1 and 4.2, we obtain

the weighted influence matrix, $\mathbf{W}$, and $\mathbf{F}$ matrix as follows:

$$\mathbf{F} = \begin{bmatrix} 0.472 & 0 & 0 & 0 \\ 0 & 0.637 & 0 & 0 \\ 0 & 0 & 0.810 & 0 \\ 0 & 0 & 0 & 0.913 \end{bmatrix} \quad \mathbf{W} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0.472 & 0 & 0 & 0 \\ 0.704 & 0 & 0 & 0 \\ 0.860 & 0.860 & 0 & 0 \end{bmatrix}$$

We then solve Equation (4.4) iteratively with initial vector $\mathbf{c}(0) = [1, 1, 1, 1]^T, \alpha = 0.6$, and $\beta = 0.1$ until convergence. The centrality scores of patents after convergence are shown in Table 4.3. From the results, one can see that our proposed method is capable of distinguishing patents and citations for the calculation of cited patent's importance. For example, patent 1 is cited by patent 2, patent 3, and patent 4. However, since patent 4 is recently published compared to patent 2 and patent 3, its citation has more effect on the importance of patent 1.

Table 4.3: Ranking result of patents in sample graph.

Rank	Patent	Score
1	p_1	0.162
2	p_2	0.132
3 (tie)	p_3	0.100
3 (tie)	p_4	0.100

4.4 Case study

In this section, we evaluate the performance of our proposed ranking method and compare it to the other existing ranking schemes using a real-world patent citation dataset from USPTO (Rodriguez et al., 2016). The dataset consists of 4,241 patents and 18,385 citations among them. We form a citation network using the dataset, which has a single connected tree structure and unweighted directed edges. Figure 4.7 shows the network structure of the dataset.

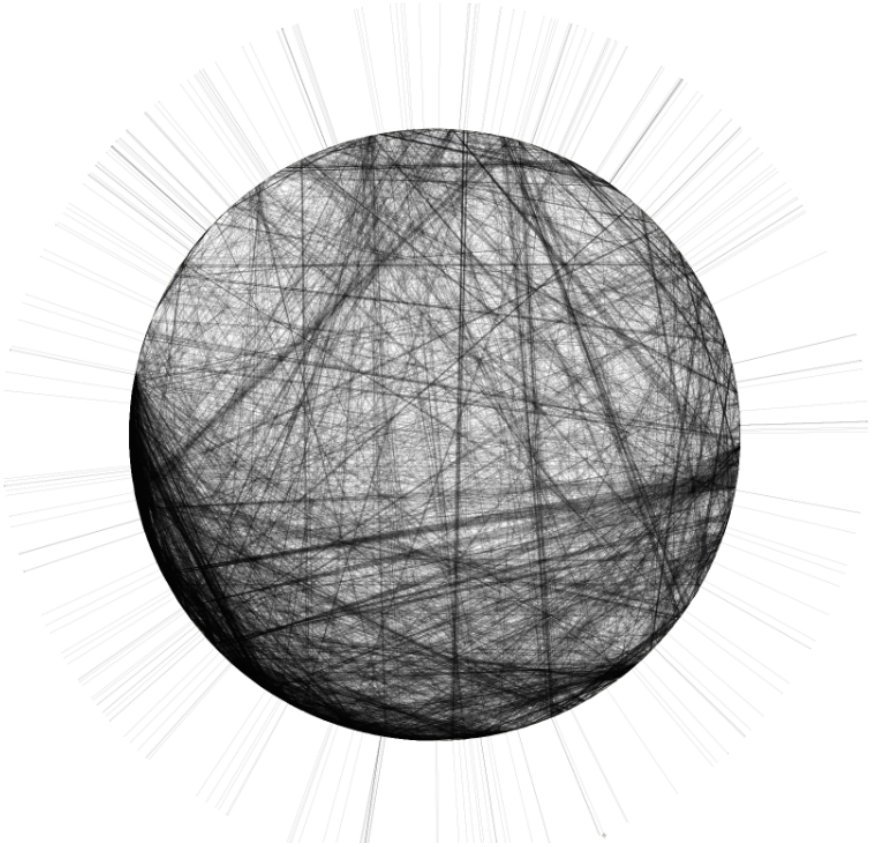

Figure 4.7: Network structure of patent citation dataset with 4241 patents and 18385 citations.

4.4.1 Metrics

Comparison metrics that we used include degree centrality (Newman, 2018), closeness centrality (Newman, 2018), betweenness centrality (Newman, 2018), PageRank (Page et al., 1999), age-based PageRank (Baeza-Yates et al., 2002), and CiteRank (Walker et al., 2007). Degree centrality counts the number of edges upon a node and can be identified as in-degree and out-degree centrality. In-degree centrality is the number of edges directed to the corresponding node, and out-degree centrality is the number of edges from the corresponding node pointing to other nodes in the network. The closeness centrality of a node is defined as the reciprocal of its farness (the sum of the shortest path distances to nodes in the network). The betweenness centrality of a node is the number of the shortest paths in the network that pass through the node of interest. PageRank is developed to rank the webpages in terms of their importance and it models the centrality of a node as a recursive function of its neighbors' centralities. Age-based PageRank is extension of PageRank algorithm. It considers the age of the page and gives less weight to the older pages. CiteRank is developed for ranking nodes in citation networks. The idea of CiteRank is similar to Age-based PageRank and it assigns less weight to older publications based on a decreasing function of age.

4.4.2 Performance measures

As for the performance measures, we use Spearman rank-order correlation coefficient and recommendation intensity (Jiang et al., 2012, Wang et al., 2014). The Spearman correlation measures the monotonicity of the relationship between the ground truth rankings and the returned rankings of a ranking method. Values of correlation varies from -1 to 1. Higher the value, the better the ranking results. Recommendation intensity assigns a score to each patent based on the rank of the patent in top-k returned patents of a ranking method and the list of top-k ground truth patents. Thus, recommendation intensity of a list of top-k patents of a ranking method is summation of recommendation intensities of all patents in the returned list.

Since the ground truth rankings of patents are not available, we use the following

procedure to evaluate the performance of each ranking method. We first divided the dataset into training and testing datasets. Training dataset consists of 80% of older patents and testing dataset consists of the remaining 20% of patents. As for the ground truth rankings of patents, we rank the patents in training dataset using the citations only from the patents in testing dataset. We then rank the patents in training dataset based only on the citations from the patents in training dataset using each ranking method. Finally, we computed the Spearman correlation and recommendation intensity for the returned list of patent ranks of each ranking method using the ground truth rankings of patents. This way, we aim to see if the methods could rank the patents by their potential to attract new citations from recent patents.

4.4.3 Results

This section presents the comparison results of ranking methods using the real-world patent citation data from USPTO in terms of aforementioned performance measures. We also present the top 10 identified patents in our real-world PCN.

We first obtain the $\mathbf{W}$ and $\mathbf{F}$ matrices of the PCN data. We then solve the Equation (4.4) iteratively with initial scores $\mathbf{c}(0) = [1, 1, \ldots, 1]^T$ until convergence. Finally, we obtain the Spearman correlation and recommendation intensity scores of the proposed ranking method using the ranks of the patents after convergence. Table 4.5 shows the comparison results of ranking methods in terms of Spearman correlation coefficient and recommendation intensity score. As for the parameters of the proposed method, we tried different set of values for each parameter and chose the values which provide the best performance. As for the other methods, we used the parameter values as suggested in the corresponding reference. As shown in Table 4.5, our proposed method outperforms all of the other ranking methods in terms of Spearman correlation coefficient and recommendation intensity. The results reveal that our proposed method is capable of both effectively ranking the entire patents and detecting the highly influential patents in a given PCN. To show the sensitivity of our proposed method to k in recommendation intensity score, we compared the ranking methods using varying k values. Table 4.6 shows the comparison results of ranking methods in terms of recommendation intensity

Table 4.5: Comparison results of metrics in terms of Spearman correlation and recommendation intensity scores.

	Parameters	Spearman Correlation	Recommendation Intensity (top-10)
Degree	N/A	0.339	10.5
In-degree	N/A	0.447	10.5
Out-degree	N/A	-0.14	0
Closeness	N/A	0.442	7.7
Betweenness	N/A	0.387	3.6
PageRank	$\alpha = 0.85$	0.399	3.6
Age-based PageRank	$\alpha = 0.85, A = 0.3, B = 0.005$	0.422	3.6
CiteRankR	$\alpha = 0.5, \tau_{dir} = 2.6$	0.145	7.6
Proposed	$\alpha = 0.3, a = 0.06, b = 0.84$	0.467	11.5

Table 4.6: Comparison results of metrics in terms of recommendation intensity for varying k values.

	$k = 10$	$k = 20$	$k = 30$	$k = 40$	$k = 50$	$k = 100$	$k = 150$	$k = 200$
Degree	10.500	25.150	37.70	51.450	60.420	96.280	123.00	137.50
In-degree	10.5	23.3	39.03	50.375	60.42	101.81	133.35	179.73
Out-degree	0	0	0	0	0	1.63	10.57	20.29
Closeness	7.7	21.5	32.93	44	56.28	95.45	129.64	154.95
Betweenness	3.6	8.25	15.66	25.275	28.92	65.3	101.82	128.09
PageRank	3.6	13.2	19.06	24.475	30.14	65.06	87.45	128.57
Age-based PageRank	3.6	13.2	19.06	24.475	30.14	65.06	89.18	130.42
CiteRank	7.6	10.4	14.13	16.15	20.46	33.75	45.84	64.83
Proposed	11.5	26.8	39.03	52.85	61.88	106.99	157.51	195.69

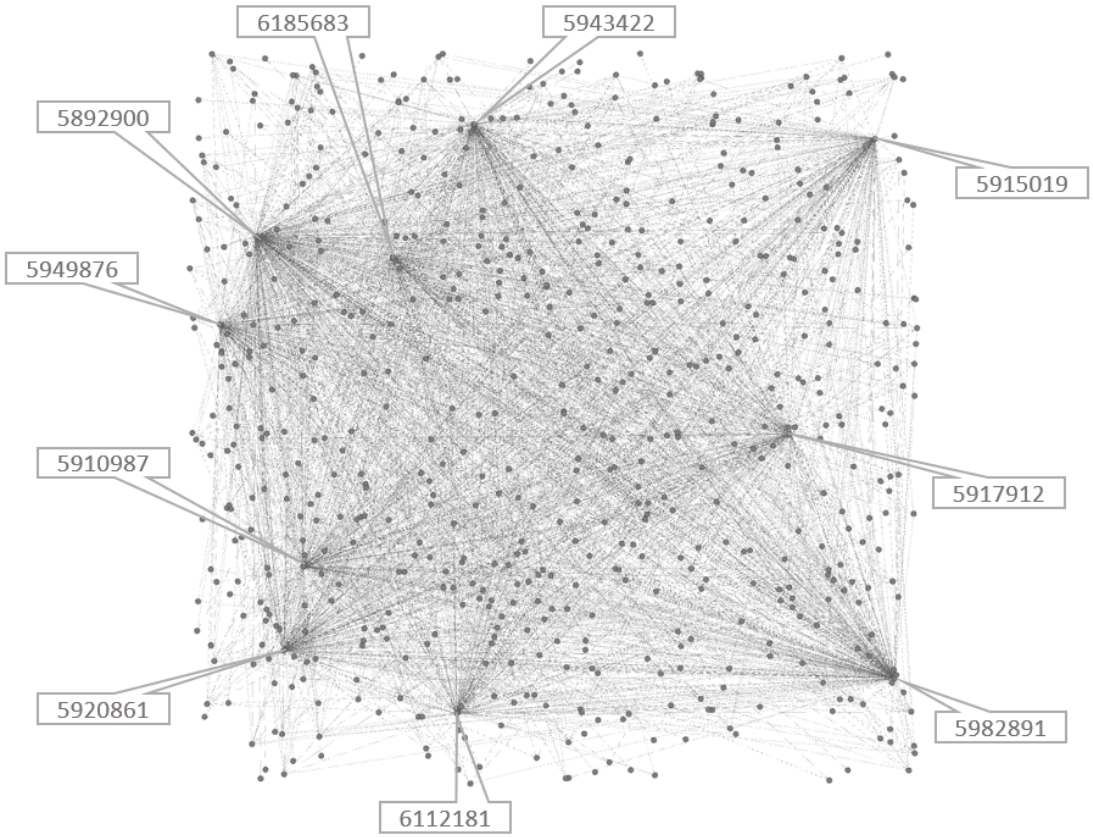

Figure 4.8: Network structure of top 10 identified patents (shown partially).

score with varying k values. The results reveal that our proposed method is robust to the changes in k and still outperforms the other ranking methods. We also show the top-10 identified patents using our proposed method. Table4.4 shows the top 10 patents identified in USPTO patent citation data by our proposed method and Figure 4.8 shows the network structure of these top 10 patents along with their immediate neighbors. One can see from the results that, top 10 patents are highly connected in their neighborhood. One can also see that except the patent with patent ID 5892900, all other top patents have similar scores, which means that patent 5892900 dominates this area.

Table 4.4: Top 10 patents identified in patent citation data by our proposed ranking method.

Rank	Patent ID	Importance Score
1	5892900	0.883
2	5943422	0.679
3	5982891	0.658
4	6185683	0.638
5	5920861	0.617
6	6112181	0.606
7	5915019	0.582
8	5910987	0.579
9	5917912	0.569
10	5949876	0.543

4.4.4 Conclusion

In this chapter, we proposed a new time-aware ranking scheme to identify the important and influential patents in a time-dependent PCN. The proposed method exploits the time information of both citing and cited patents and successfully distinguishes the effect of each citation to the importance of a cited patent. To show the superiority of our proposed method over the well-known ranking methods, we compared our ranking scheme with other metrics in terms of Spearman correlation coefficient of rankings and recommendation intensity score using a real-world patent citation dataset from USPTO in the area of digital information and security. The results reveal that our proposed ranking method outperforms other metrics.

Chapter 5

Concluding Remarks and Future Research

5.1 Concluding remarks

In this dissertation, we proposed and develop advanced data mining methodologies for the analysis of directed networks. To this extent, in Chapter 2, we propose a node anomaly detection algorithm based on nonnegative matrix factorization to rank the patents in outlierness in a patent citation network. The proposed method first clusters the patents using all types of citations that a patent has with asymmetric nonnegative matrix factorization - a clustering method specifically designed for directed networks. To do so, we introduce a citation matrix which is the extension of tbe adjacency matrix to exploit the information of direct and indirect citations. Then, the results of the clustering algorithm are used as input to the proposed scoring function. The proposed scoring function considers not only the individual patent relationships but also the link and linked information of clusters. To show the implementation of the proposed method in detail, we used an illustrative example with a small patent citation network. Then, we show the performance evaluation of our method using a real-world patent citation data. As for the performance measure, we injected synthetic outliers to the real-world PCN and calculate the accuracy and F1 scores of the proposed method. We also compare the proposed method with other outlier detection algorithms and results reveal that our proposed method outperforms others in detecting anomalous patents.

In Chapter 3, we introduced a regularized asymmetric nonnegative matrix factorization for clustering in directed networks. Asymmetric nonnegative matrix factorization is designed specifically for clustering in directed networks. However, ANMF cannot capture the intrinsic information hidden in the structure of the network. To address this issue, we proposed to add a regularization term to ANMF. The regularization term

aims to force the representatives of the nodes in new basis to be closer to each other if the nodes are similar to each other in structural similarity and force the representatives to be far from each other if the nodes are not similar to each other in the network structure. Then, we propose updating rules for the new optimization problem. In addition, ANMF algorithm is designed based on random initialization, which requires the algorithm to be run several times to obtain a stable solution. However, running several instances of the algorithm is very time-consuming in many cases. Therefore, we propose to initialize our proposed RANMF algorithm using SVD-based initialization. Since SVD-based initialization is specifically designed for NMF algorithms, we modify it for the RANMF algorithm. To evaluate the performance of our proposed clustering algorithm, we used real-world datasets and synthetic datasets. To capture the different aspects of the proposed algorithm, we evaluate it using several popular clustering validity indices such as DB index, Jaccard index, and NMI. In most of the experiments, our proposed clustering algorithm outperforms other clustering algorithms in terms of all validity indices. We also prove the convergence of the multiplicative updating rules numerically and theoretically. Sensitivity analysis of the proposed method is also presented. Our proposed method is robust to the changes in the parameter values.

Finally, in Chapter 4 we developed a time-aware importance and influence measure for ranking patents in dynamic patent citation networks. There is currently no work that considers the time information of citing and cited patents at the same time. The proposed method is capable of distinguishing not only the citing patent but also cited patent for the importance of cited patent in time-evolving patent citation network. To show the effectiveness and performance of the proposed ranking scheme, we used a real-world patent citation network. As for the evaluation of performance, we used the Spearman correlation coefficient of rankings and recommendation intensity scores. Spearman correlation coefficient calculates the correlation using entire dataset and recommendation intensity considers only top patents identified by a ranking method. Experimental results show that our proposed ranking method outperforms other ranking metrics in terms of both ranking the entire patents in importance and identifying the influential ones.

5.2 Future research

Despite the satisfactory performance of the proposed methodologies in this dissertation, there is still some room for further improvements. For the node anomaly detection research, we didn't consider the time information of patents. However, patent citation networks are evolving graphs and new patents and new citations appear over time. Future study should devise a new node anomaly detection algorithm which incorporates the time information of patents.

For clustering research in directed networks, one can investigate the effect of the different similarity measures on the performance of the proposed method. In addition, the (1/4) term in the multiplicative updating rule makes the algorithm converge a little slower. So, future research can devise new updating rules which converge faster.

As for the influential patent identification research, one can incorporate the attribute information of the patents and devise a new ranking method. Currently, we only consider the network structure of the patent citation data. However, patents have rich information such as who owns the patent, class information of the patent, and citations to/from non-patent sources.

Appendix A

Proof of Proposition 1

Suppose that a graph G is acyclic directed graph with n nodes, then the largest value of the length of the longest path in the graph can be $n-1$. Let τ be the longest path in the graph G. For $l > \tau$ and for all $1 \leq i, j \leq n$, $[\mathbf{A}]_{ij}^{(l)} = 0$ since there is no path with length l from node i to node j. Therefore $\mathbf{A}^l = \mathbf{0}$, for all $l > \tau$. Let $\mathbf{C} = \sum_{l=1}^{\infty} \beta^l \mathbf{A}^l$, then $\mathbf{C} = \sum_{l=1}^{\tau} \beta^l \mathbf{A}^l$. With $0 < \beta < 1$, $\mathbf{C}$ can be rewritten as

$$
\begin{aligned}
\mathbf{C} &= \beta\mathbf{A} + \beta^2\mathbf{A}^2 + \ldots + \beta^{\tau+1}\mathbf{A}^{\tau+1} \\
&= \beta\mathbf{A} + \beta\mathbf{A}(\beta\mathbf{A} + \ldots + \beta^{\tau}\mathbf{A}^{\tau}) \\
&= \beta\mathbf{A} + \beta\mathbf{A}\mathbf{C} \\
\Rightarrow &(\mathbf{I} - \beta\mathbf{A})\mathbf{C} = \beta\mathbf{A} \\
\Rightarrow &\mathbf{C} = \beta\mathbf{A}(\mathbf{I} - \beta\mathbf{A})^{-1}.
\end{aligned}
$$

Appendix B

Derivation of Regularization Term

$$\begin{aligned}
\frac{\lambda}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}||\mathbf{w}_i-\mathbf{w}_j||^2[\mathbf{S}]_{ij} &= \frac{\lambda}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}(\mathbf{w}_i-\mathbf{w}_j)^T(\mathbf{w}_i-\mathbf{w}_j)[\mathbf{S}]_{ij} \\
&= \frac{\lambda}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}(\mathbf{w}_i^T\mathbf{w}_i[\mathbf{S}]_{ij}+\mathbf{w}_j^T\mathbf{w}_j[\mathbf{S}]_{ij}-2\mathbf{w}_i^T\mathbf{w}_j[\mathbf{S}]_{ij}) \\
&= \lambda\sum_{i=1}^{n}(\mathbf{w}_i^T\mathbf{w}_i)[\mathbf{D}]_{ii}-\lambda\sum_{i=1}^{n}\sum_{j=1}^{n}(\mathbf{w}_i^T\mathbf{w}_j)[\mathbf{S}]_{ij} \\
&= \lambda\mathrm{Tr}(\mathbf{W}^T\mathbf{D}\mathbf{W})-\lambda\mathrm{Tr}(\mathbf{W}^T\mathbf{S}\mathbf{W}),
\end{aligned}$$

Appendix C

Derivation of Multiplicative Updating Rules

We introduce multiplicative update rules to minimize the objective function in (3.11). Regularization term in objective function in (3.11) can be rewritten as

$$\lambda \mathrm{Tr}(\mathbf{W}^T\mathbf{D}\mathbf{W}) - \lambda \mathrm{Tr}(\mathbf{W}^T\mathbf{S}\mathbf{W}) = \lambda \mathrm{Tr}(\mathbf{W}^T\mathbf{L}\mathbf{W}) \tag{C.1}$$

where $\mathbf{L} = \mathbf{D} - \mathbf{S}$ is the graph Laplacian (Chung, 1997). Thus, (3.11) can be rewritten as

$$\begin{aligned} f(\mathbf{W},\mathbf{H}) &= \min_{\mathbf{W},\mathbf{H}\geq\mathbf{0}} ||\mathbf{A} - \mathbf{W}\mathbf{H}\mathbf{W}^T||_F^2 + \lambda \mathrm{Tr}(\mathbf{W}^T\mathbf{L}\mathbf{W}) \\ &= \mathrm{Tr}((\mathbf{A} - \mathbf{W}\mathbf{H}\mathbf{W}^T)(\mathbf{A} - \mathbf{W}\mathbf{H}\mathbf{W}^T)^T) \\ &+ \lambda \mathrm{Tr}(\mathbf{W}^T\mathbf{L}\mathbf{W}) \\ &= \mathrm{Tr}(\mathbf{A}\mathbf{A}^T - \mathbf{A}\mathbf{W}\mathbf{H}^T\mathbf{W}^T - \mathbf{W}\mathbf{H}\mathbf{W}^T\mathbf{A}^T + \mathbf{W}\mathbf{H}\mathbf{W}^T\mathbf{W}\mathbf{H}^T\mathbf{W}^T) \\ &+ \lambda \mathrm{Tr}(\mathbf{W}^T\mathbf{L}\mathbf{W}) \\ &= \mathrm{Tr}(\mathbf{A}\mathbf{A}^T) - \mathrm{Tr}(\mathbf{A}\mathbf{W}\mathbf{H}^T\mathbf{W}^T) - \mathrm{Tr}(\mathbf{W}\mathbf{H}\mathbf{W}^T\mathbf{A}^T) \\ &+ \mathrm{Tr}(\mathbf{W}\mathbf{H}\mathbf{W}^T\mathbf{W}\mathbf{H}^T\mathbf{W}^T) + \lambda \mathrm{Tr}(\mathbf{W}^T\ \mathbf{L}\mathbf{W}) \end{aligned} \tag{C.2}$$

We obtain the Lagrangian of (3.11) with Lagrangian multipliers $\mathbf{\Psi}_1$ and $\mathbf{\Psi}_2$ for the nonnegativity of $\mathbf{W}$ and $\mathbf{H}$ as

$$\begin{aligned} \mathcal{L} &= \mathrm{Tr}(\mathbf{A}\mathbf{A}^T) - \mathrm{Tr}(\mathbf{A}\mathbf{W}\mathbf{H}^T\mathbf{W}^T) - \mathrm{Tr}(\mathbf{W}\mathbf{H}\mathbf{W}^T\mathbf{A}^T) \\ &+ \mathrm{Tr}(\mathbf{W}\mathbf{H}\mathbf{W}^T\ \mathbf{W}\mathbf{H}^T\mathbf{W}^T) \\ &+ \lambda \mathrm{Tr}(\mathbf{W}^T\ \mathbf{L}\mathbf{W}) - \mathrm{Tr}(\mathbf{\Psi}_1\mathbf{W}^T) - \mathrm{Tr}(\mathbf{\Psi}_2\mathbf{H}^T) \end{aligned} \tag{C.3}$$

Taking partial derivatives of (C.3) w.r.t. $\mathbf{W}$ and $\mathbf{H}$ leads to

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \mathbf{W}} &= -2\mathbf{A}\mathbf{W}\mathbf{H}^T - 2\mathbf{A}^T\mathbf{W}\mathbf{H} \\ &\quad + 2\mathbf{W}\mathbf{H}\mathbf{W}^T\mathbf{W}\mathbf{H}^T + 2\mathbf{W}\mathbf{H}^T\mathbf{W}^T\mathbf{W}\mathbf{H} + 2\lambda\mathbf{L}^T\mathbf{W} - \mathbf{\Psi}_1 \\ \frac{\partial \mathcal{L}}{\partial \mathbf{H}} &= -2\mathbf{W}^T\mathbf{A}\mathbf{W} + 2\mathbf{W}^T\mathbf{W}\mathbf{W}^T\mathbf{H}\mathbf{W} - \mathbf{\Psi}_2\end{aligned} \tag{C.4}$$

Let partial derivatives in (C.4) equal to 0. Using the KKT complementary slackness conditions we obtain

$$\begin{aligned}&[-2\mathbf{A}\mathbf{W}\mathbf{H}^T - 2\mathbf{A}^T\mathbf{W}\mathbf{H} + 2\mathbf{W}\mathbf{H}\mathbf{W}^T\mathbf{W}\mathbf{H}^T + 2\mathbf{W}\mathbf{H}^T\mathbf{W}^T\mathbf{W}\mathbf{H} + 2\lambda\mathbf{L}^T\mathbf{W}]_{ik}[\mathbf{W}]_{ik} = 0 \\ &[-2\mathbf{W}^T\mathbf{A}\mathbf{W} + 2\mathbf{W}^T\mathbf{W}\mathbf{W}^T\mathbf{H}\mathbf{W}]_{kj}[\mathbf{H}]_{kj} = 0\end{aligned} \tag{C.5}$$

Equations in (C.5) lead us to following multiplicative update rules

$$\begin{aligned}[\mathbf{W}]_{ik} &\leftarrow [\mathbf{W}]_{ik}\left(\frac{[\mathbf{A}\mathbf{W}\mathbf{H}^T + \mathbf{A}^T\mathbf{W}\mathbf{H} + \lambda\mathbf{S}^T\mathbf{W}]_{ik}}{[\mathbf{W}\mathbf{H}\mathbf{W}^T\mathbf{W}\mathbf{H}^T + \mathbf{W}\mathbf{H}^T\mathbf{W}^T\mathbf{W}\mathbf{H} + 2\lambda\mathbf{D}^T\mathbf{W}]_{ik}}\right)^{\frac{1}{4}} \\ [\mathbf{H}]_{kj} &\leftarrow [\mathbf{H}]_{kj}\frac{[\mathbf{W}^T\mathbf{A}\mathbf{W}]_{kj}}{[\mathbf{W}^T\mathbf{W}\mathbf{H}\mathbf{W}^T\mathbf{W}]_{kj}}\end{aligned} \tag{C.6}$$

Appendix D

Proof of Theorem 1

We prove the theorem that objective function in (3.11) is nonincreasing under the updating rules in (3.12) by using auxiliary funcion approach. We first write our objective function in (3.11) with the help of simple linear algebra as

$$f(\mathbf{W},\mathbf{H}) = \mathrm{Tr}(\mathbf{A}\mathbf{A}^T) - 2\mathrm{Tr}(\mathbf{A}^T\mathbf{W}\mathbf{H}\mathbf{W}^T) + \mathrm{Tr}(\mathbf{W}\mathbf{H}\mathbf{W}^T\mathbf{W}\mathbf{H}^T\mathbf{W}^T) + \lambda\mathrm{Tr}(\mathbf{W}^T\mathbf{D}\mathbf{W}) - \lambda\mathrm{Tr}(\mathbf{W}^T\mathbf{S}\mathbf{W}).$$

Based on the results from (Wang et al., 2011), we can obtain the following three inequalities as

$$\mathrm{Tr}(\mathbf{B}\mathbf{W}^T\mathbf{A}\mathbf{W}) \leq \frac{1}{2}\mathrm{Tr}(\mathbf{B}\mathbf{Y}^T\mathbf{A}\tilde{\mathbf{W}}) + \frac{1}{2}\mathrm{Tr}(\mathbf{B}\tilde{\mathbf{W}}^T\mathbf{A}\mathbf{Y})$$

$$\mathrm{Tr}(\mathbf{P}\mathbf{A}) \leq \mathrm{Tr}(\mathbf{R}\mathbf{A}\tilde{\mathbf{W}}^T)$$

$$-\mathrm{Tr}(\mathbf{B}\mathbf{W}^T\mathbf{A}\mathbf{W}) \leq -\mathrm{Tr}(\mathbf{B}\tilde{\mathbf{W}}^T\mathbf{A}\mathbf{Z}) - \mathrm{Tr}(\mathbf{B}\mathbf{Z}^T\mathbf{A}\tilde{\mathbf{W}}) - \mathrm{Tr}(\mathbf{B}\tilde{\mathbf{W}}^T\mathbf{A}\tilde{\mathbf{W}})$$

where $[\mathbf{Y}]_{ij} = [\mathbf{W}]_{ij}^2/[\tilde{\mathbf{W}}]_{ij}$; $[\mathbf{R}]_{ik} = [\mathbf{W}]_{ik}^4/[\tilde{\mathbf{W}}]_{ik}^3$; $\mathbf{P}_{kl} = [\mathbf{W}^T\mathbf{W}]_{kl}^2/[\mathbf{W}^T\mathbf{W}]_{kl}$; and $[\mathbf{Z}]_{ij} = [\tilde{\mathbf{W}}]_{ij}.\ln([\mathbf{W}]_{ij}/[\tilde{\mathbf{W}}]_{ij})$. Using above inequalities,

$$\begin{aligned} f(\mathbf{W},\mathbf{H}) \leq\ & \frac{1}{2}\mathrm{Tr}(\mathbf{R}\mathbf{H}\tilde{\mathbf{W}}^T\tilde{\mathbf{W}}\mathbf{H}^T\tilde{\mathbf{W}}^T + \mathbf{R}\mathbf{H}^T\tilde{\mathbf{W}}^T\tilde{\mathbf{W}}\mathbf{H}\tilde{\mathbf{W}}^T) \\ & + \frac{\lambda}{2}\mathrm{Tr}(\mathbf{R}^T\mathbf{D}\tilde{\mathbf{W}} + \mathbf{R}^T\mathbf{D}\tilde{\mathbf{W}}) \\ & - 2\mathrm{Tr}(\mathbf{A}^T\tilde{\mathbf{W}}\mathbf{H}\mathbf{Z}^T) - 2\mathrm{Tr}(\mathbf{A}^T\mathbf{Z}\mathbf{H}\tilde{\mathbf{W}}^T) \\ & - 2\mathrm{Tr}(\mathbf{A}^T\tilde{\mathbf{W}}\mathbf{H}\tilde{\mathbf{W}}^T) - \lambda\mathrm{Tr}(\tilde{\mathbf{W}}^T\mathbf{S}\mathbf{Z}) \\ & - \lambda\mathrm{Tr}(\mathbf{Z}^T\mathbf{S}\tilde{\mathbf{W}}) - \lambda\mathrm{Tr}(\tilde{\mathbf{W}}^T\mathbf{S}\tilde{\mathbf{W}}) + \mathrm{Tr}(\mathbf{A}\mathbf{A}^T) \\ \stackrel{def}{=}\ & G(\mathbf{W},\tilde{\mathbf{W}}) \end{aligned}$$

where $G(\mathbf{W}, \tilde{\mathbf{W}})$ is an auxiliary function for $\mathbf{W}$. Now let's define as in (Lee and Seung, 2001)

$$\mathbf{W}^{(t+1)} = \arg\min_{\mathbf{W}} G(\mathbf{W}, \mathbf{W}^{(t)})$$

where t stands for iteration number. Then we have

$$G(\mathbf{W}^{(t)}, \mathbf{W}^{(t)}) \geq G(\mathbf{W}^{(t+1)}, \mathbf{W}^{(t)}) \geq G(\mathbf{W}^{(t+1)}, \mathbf{W}^{(t+1)}).$$

Thus, the objective function $f(\mathbf{W}^{(t)}, \mathbf{H}) = G(\mathbf{W}^{(t)}, \mathbf{W}^{(t)})$ is monotonically decreasing. Let $\mathcal{L}(\mathbf{W}) = G(\mathbf{W}, \tilde{\mathbf{W}})$ and by the below KKT condition

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial [\mathbf{W}]_{ik}} = 2\frac{[\mathbf{W}]_{ik}^3}{[\tilde{\mathbf{W}}]_{ik}^3}[&\tilde{\mathbf{W}}\mathbf{H}\tilde{\mathbf{W}}^T\tilde{\mathbf{W}}\mathbf{H}^T + \tilde{\mathbf{W}}\mathbf{H}^T\tilde{\mathbf{W}}^T\tilde{\mathbf{W}}\mathbf{H} \\ &+ 2\lambda\mathbf{D}^T\tilde{\mathbf{W}}]_{ik} - 2\frac{[\tilde{\mathbf{W}}]_{ik}}{[\mathbf{W}]_{ik}}[\mathbf{A}^T\tilde{\mathbf{W}}\mathbf{H} \\ &+ \mathbf{A}\tilde{\mathbf{W}}\mathbf{H}^T + \lambda\mathbf{S}^T\tilde{\mathbf{W}}]_{ik}\end{aligned}$$

which leads us to updating rules for $\mathbf{W}$ as in (3.12).

Since our regularization term is only related to $\mathbf{W}$, we have the exact same updating rule for $\mathbf{H}$ as it is in the original ANMF algorithm. Therefore, we proved the convergence of updating rule, which shows that objective function in (3.11) is nonincreasing under the updating rules in (3.12), only for $\mathbf{W}$.

References

Assad Abbas, Limin Zhang, and Samee U Khan. A literature review on the state-of-the-art in patent analysis. *World Patent Information*, 37:3–13, 2014.

Charu C Aggarwal. Outlier analysis. In *Data Mining*, pages 237–263. Springer, 2015.

Leman Akoglu, Mary McGlohon, and Christos Faloutsos. Oddball: Spotting anomalies in weighted graphs. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 410–421. Springer, 2010.

Ricardo Baeza-Yates, Felipe Saint-Jean, and Carlos Castillo. Web structure, dynamics and page quality. In *International Symposium on String Processing and Information Retrieval*, pages 117–130. Springer, 2002.

Rajiv D Banker, Hsihui Chang, and Zhiqiang Zheng. On the use of super-efficiency procedures for ranking efficient units and identifying outliers. *Annals of Operations Research*, 250(1):21–35, 2017.

Christos Boutsidis and Efstratios Gallopoulos. Svd based initialization: A head start for nonnegative matrix factorization. *Pattern Recognition*, 41(4):1350–1362, 2008.

Deng Cai, Xiaofei He, Jiawei Han, and Thomas S Huang. Graph regularized nonnegative matrix factorization for data representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1548–1560, 2011.

Xiaochun Cao, Xiao Wang, Di Jin, Yixin Cao, and Dongxiao He. Identifying overlapping communities as well as hubs and outliers via nonnegative matrix factorization. *Scientific Reports*, 3:2993, 2013.

Fan RK Chung. *Spectral graph theory*, volume 92. American Mathematical Soc., 1997.

Daniele Codetta-Raiteri and Luigi Portinale. Dynamic bayesian networks for fault detection, identification, and recovery in autonomous spacecraft. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(1):13–24, 2015.

Leon Danon, Albert Diaz-Guilera, Jordi Duch, and Alex Arenas. Comparing community structure identification. *Journal of Statistical Mechanics: Theory and Experiment*, 2005(09):P09008, 2005.

David L Davies and Donald W Bouldin. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (2):224–227, 1979.

Chris HQ Ding, Xiaofeng He, and Horst D Simon. On the equivalence of nonnegative matrix factorization and spectral clustering. In *SDM*, volume 5, pages 606–610. SIAM, 2005.

Lian Duan, Lida Xu, Ying Liu, and Jun Lee. Cluster-based outlier detection. *Annals of Operations Research*, 168(1):151–168, 2009.

Dušan Džamić, Daniel Aloise, and Nenad Mladenović. Ascent–descent variable neighborhood decomposition search for community detection by modularity maximization. *Annals of Operations Research*, pages 1–15, 2017.

Cédric Févotte, Nancy Bertin, and Jean-Louis Durrieu. Nonnegative matrix factorization with the itakura-saito divergence: With application to music analysis. *Neural Computation*, 21(3):793–830, 2009.

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

Rumi Ghosh, Tsung-Ting Kuo, Chun-Nan Hsu, Shou-De Lin, and Kristina Lerman. Time-aware ranking in dynamic citation networks. In *2011 IEEE 11th International Conference on Data Mining Workshops*, pages 373–380. IEEE, 2011.

Daniel Gómez, Edwin Zarrazola, Javier Yáñez, and Javier Montero. A divide-and-link algorithm for hierarchical clustering in networks. *Information Sciences*, 316:308–328, 2015.

Naiyang Guan, Dacheng Tao, Zhigang Luo, and Bo Yuan. Manifold regularized discriminative nonnegative matrix factorization with fast gradient descent. *IEEE Transactions on Image Processing*, 20(7):2030–2048, 2011.

Bronwyn H Hall, Adam Jaffe, and Manuel Trajtenberg. Market value and patent citations. *RAND Journal of Economics*, pages 16–38, 2005.

Joao P Hespanha. An efficient matlab algorithm for graph partitioning. *University of California*, pages 1–8, 2004.

Lawrence B Holder and Diane J Cook. Graph-based data mining. *Encyclopedia of Data Warehousing and Mining*, 2:943–949, 2009.

Xiaorui Jiang, Xiaoping Sun, and Hai Zhuge. Towards an effective and unbiased ranking of scientific literature through mutual reinforcement. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, pages 714–723. ACM, 2012.

U Kang, Leman Akoglu, and Duen Horng Polo Chau. Big graph mining: algorithms, anomaly detection, and applications. *Proceedings of the ACM ASONAM*, 13:25–28, 2013.

Brian W Kernighan and Shen Lin. An efficient heuristic procedure for partitioning graphs. *Bell System Technical Journal*, 49(2):291–307, 1970.

Jingu Kim and Haesun Park. Sparse nonnegative matrix factorization for clustering. Technical report, Georgia Institute of Technology, 2008.

Andrea Lancichinetti and Santo Fortunato. Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *Physical Review E*, 80(1):016118, 2009.

Daniel D Lee and H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

Daniel D Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems*, pages 556–562, 2001.

Kristina Lerman, Rumi Ghosh, and Jeon Hyung Kang. Centrality metric for dynamic networks. In *Proceedings of the Eighth Workshop on Mining and Learning with Graphs*, pages 70–77. ACM, 2010.

Ying Liao, Huan Qi, and Weiqun Li. Load-balanced clustering algorithm with distributed self-organization for wireless sensor networks. *IEEE Sensors Journal*, 13(5): 1498–1506, 2013.

Ruslan Lukach and Maryna Lukach. Ranking uspto patent documents by importance using random surfer method (pagerank). *Available at SSRN 996595*, 2007.

Yuanyuan Ma, Xiaohua Hu, Tingting He, and Xingpeng Jiang. Hessian regularization based symmetric nonnegative matrix factorization for clustering gene expression and microbiome data. *Methods*, 2016.

Fragkiskos D Malliaros and Michalis Vazirgiannis. Clustering and community detection in directed networks: A survey. *Physics Reports*, 533(4):95–142, 2013.

Jacques Michel and Bernd Bettels. Patent citation analysis. a closer look at the basic input data from patent search reports. *Scientometrics*, 51(1):185–201, 2001.

HDK Moonesignhe and Pang-Ning Tan. Outlier detection using random walks. In *2006 18th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'06)*, pages 532–539. IEEE, 2006.

HDK Moonesinghe and Pang-Ning Tan. Outlier detection using random walks. In *Tools with Artificial Intelligence, 2006. ICTAI'06. 18th IEEE International Conference on*, pages 532–539. IEEE, 2006.

Mark Newman. *Networks*. Oxford university press, 2018.

Mark EJ Newman. Finding community structure in networks using the eigenvectors of matrices. *Physical Review E*, 74(3):036104, 2006.

Sooyoung Oh, Zhen Lei, Prasenjit Mitra, and John Yen. Evaluating and ranking patents using weighted citations. In *Proceedings of the 12th ACM/IEEE-CS joint conference on Digital Libraries*, pages 281–284. ACM, 2012.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

Leonidas Pitsoulis. Quality functions in graph clustering. URL `https://nnov.hse.ru/data/2014/11/10/1102862238/presentation%20Leonidas.pdf`.

Andrew Rodriguez, Ali Tosyali, Byunghoon Kim, Jeongsub Choi, Jae-Min Lee, Byoung-Youl Coh, and Myong K Jeong. Patent clustering and outlier ranking methodologies for attributed patent citation networks for technology opportunity discovery. *IEEE Transactions on Engineering Management*, 63(4):426–437, 2016.

Satu Elisa Schaeffer. Graph clustering. *Computer Science Review*, 1(1):27–64, 2007.

Farial Shahnaz, Michael W Berry, V Paul Pauca, and Robert J Plemmons. Document clustering using nonnegative matrix factorization. *Information Processing & Management*, 42(2):373–386, 2006.

Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

Motoki Shiga and Hiroshi Mamitsuka. Non-negative matrix factorization with auxiliary information on overlapping groups. *IEEE Transactions on Knowledge and Data Engineering*, 27(6):1615–1628, 2015.

Heli Sun, Jianbin Huang, Jiawei Han, Hongbo Deng, Peixiang Zhao, and Boqin Feng. gskeletonclu: Density-based network clustering via structure-connected tree division or agglomeration. In *2010 IEEE International Conference on Data Mining*, pages 481–490. IEEE, 2010.

Hanghang Tong and Ching-Yung Lin. Non-negative residual matrix factorization with application to graph anomaly detection. In *SDM*, pages 143–153. SIAM, 2011.

Tuomas Virtanen. Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(3):1066–1074, 2007.

Dylan Walker, Huafeng Xie, Koon-Kiu Yan, and Sergei Maslov. Ranking scientific publications using a model of network traffic. *Journal of Statistical Mechanics: Theory and Experiment*, 2007(06):P06010, 2007.

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.

Senzhang Wang, Sihong Xie, Xiaoming Zhang, Zhoujun Li, Philip S Yu, and Xinyu Shu. Future influence ranking of scientific literature. In *Proceedings of the 2014 SIAM International Conference on Data Mining*, pages 749–757. SIAM, 2014.

Yu-Xiong Wang and Yu-Jin Zhang. Nonnegative matrix factorization: A comprehensive review. *IEEE Transactions on Knowledge and Data Engineering*, 25(6):1336–1353, 2013.

Xiaowei Xu, Nurcan Yuruk, Zhidan Feng, and Thomas AJ Schweiger. Scan: a structural clustering algorithm for networks. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 824–833. ACM, 2007.

Ruicong Zhi, Markus Flierl, Qiuqi Ruan, and W Bastiaan Kleijn. Graph-preserving sparse nonnegative matrix factorization with application to facial expression recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 41(1):38–52, 2011.

Zhaonian Zou, Jianzhong Li, Hong Gao, and Shuo Zhang. Mining frequent subgraph patterns from uncertain graph data. *IEEE Transactions on Knowledge and Data Engineering*, 22(9):1203–1218, 2010.