BIG DATA MINING AND ANALYTICS
ISSN 2096-0654 03/06 pp242–251
Volume 4, Number 4, December 2021
DOI: 10.26599/BDMA.2021.9020010

Coronavirus Pandemic Analysis Through Tripartite Graph Clustering in Online Social Networks

Xueting Liao, Danyang Zheng*, and Xiaojun Cao

Abstract: The COVID-19 pandemic has hit the world hard. The reaction to the pandemic related issues has been pouring into social platforms, such as Twitter. Many public officials and governments use Twitter to make policy announcements. People keep close track of the related information and express their concerns about the policies on Twitter. It is beneficial yet challenging to derive important information or knowledge out of such Twitter data. In this paper, we propose a Tripartite Graph Clustering for Pandemic Data Analysis (TGC-PDA) framework that builds on the proposed models and analysis: (1) tripartite graph representation, (2) non-negative matrix factorization with regularization, and (3) sentiment analysis. We collect the tweets containing a set of keywords related to coronavirus pandemic as the ground truth data. Our framework can detect the communities of Twitter users and analyze the topics that are discussed in the communities. The extensive experiments show that our TGC-PDA framework can effectively and efficiently identify the topics and correlations within the Twitter data for monitoring and understanding public opinions, which would provide policy makers useful information and statistics for decision making.

Key words: COVID-19; clustering; online social network; Twitter

1 Introduction

The COVID-19 has hit the world and made major impacts in society, economy, medical care, environment, and so on[1,2]. To prevent the virus from spreading all over the world, unnecessary travel has been restricted, quarantines are required, and even Tokyo Olympics was postponed[3–5]. The virus has been spreading to almost the whole world. Since the major pathogen of COVID-19 is a single-stranded RNA, it is intrinsically unstable. The immune system of humans may not be as responsive (even injected with vaccine) if the virus RNA significantly changes its structure[6]. This means the vaccine may need to be constantly updated and there could be a lengthy recovery for many people.

Meanwhile, the pandemic has led to the global economic disruption, the anxiousness for supply shortages, and the fear of disease. A mass of misinformation and conspiracy theories about the coronavirus have spread over the Internet, especially on Online Social Networks (OSNs)[7]. Twitter, as one of the most popular social networking platforms, is widely used to allow people to post and comment the messages, called "tweets". Twitter contains not only the text data, but also the interaction among users. The ever-increasing expansion of the Internet and mobile networks allows real-time propagation of tweets to a large number of people, which makes it a desired environment for the breaking-news discussion and fear contagion. The open source intelligence from the social networks in Twitter can be utilized to analyze and keep track of the attitudes or opinions of people towards coronavirus pandemic events. Important information or knowledge can be

• Xueting Liao and Xiaojun Cao are with the Department of Computer Science, Georgia State University, Atlanta, GA 30302, USA. E-mail: xliao3@student.gsu.edu; cao@gsu.edu.
• Danyang Zheng is with Suzhou Key Laboratory of Advanced Optical Communication Network Technology, School of Electronic and Information Engineering, Soochow University, Suzhou 215006, China. E-mail: drdan940606@gmail.com.
* To whom correspondence should be addressed.
Manuscript received: 2021-02-25; revised: 2021-06-02; accepted: 2021-06-04

derived from such Twitter data, which can provide policy makers better information and statistics for decision making.

In Twitter, the data (such as users and tweets) are linked rather than a bunch of standalone information units. Thus, it is natural to represent such linked data as graphs. The representation of the graph can largely affect the performance of Twitter data analysis. Meanwhile, the scale of the graphs increases explosively from thousands of vertices to billions of vertices, which makes it important to find a proper graph representation for the data. A suitable graph representation can make the entire descriptive or predictive graph analyzing process efficient and effective. For example, multipartite graphs can be used to model networks with different objects, such as documents and terms, movies and preferences, or buyers and sellers. In Refs. [8, 9], unipartite and bipartite graphs have been used to represent Twitter data. Once we have a proper graph presentation of Twitter data, machine learning models can be developed to conduct various data analysis, including sentiment, prediction, trend analysis and so on[10, 11].

Sentiment analysis in Twitter can analyze the tweet texts to identify the opinions or ideas that users express. Much literal work on Twitter sentiment analysis focused on understanding the sentiments of individual tweets and user-level sentiments[12–14]. Some researchers studied both tweet-level and user-level sentiments[15, 16]. Sentiment analysis is challenging because the sentiments of users are correlated with the sentiments expressed in many short tweets, which are intrinsically noisy and labile. In addition, it is difficult to understand and characterize the dynamics in user's sentiments, as different time may lead to contradict opinions towards the same topic. It is not uncommon to see people having a lukewarm and reluctant attitude towards a product at first glance, but later cannot live without it.

In this work, we investigate the issue of pandemic analysis through Twitter data, and propose a framework of Tripartite Graph Clustering for Pandemic Data Analysis (TGC-PDA). In the proposed framework, we first build a tripartite graph, which will take advantage of the characteristics of the Twitter users and tweets network structure to facilitate exploring the community structures. Then we cluster the tweets and users based on the structure of the graph using a clustering approach. Finally, the framework provides the open source intelligence from the clustered communities, which can help keep track of people's feedbacks and opinions towards the Coronavirus pandemic events. To the best of our knowledge, the TGC-PDA framework is the first to effectively analyze the sentiments of users for COVID-19-related topics based on transforming of the tripartite graph. We conduct a set of experiments on COVID-19 related Twitter data, and verify that our approach is effective and efficient.

The rest of the paper is organized as follows. Section 2 introduces the background of our work. Section 3 explains the main notations and problem formulation for our work. Section 4 explains our proposed framework. Section 5 illustrates the computing algorithm. Section 6 elaborates the experimental setting, results, and analysis. Finally, conclusions are drawn in Section 7.

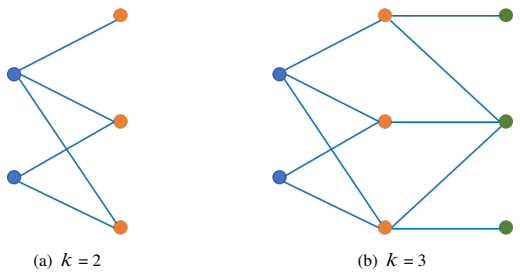
2 Background

In this section, we present some backgrounds on multipartite graph modeling, non-negative matrix factorization, and sentiment analysis.

2.1 Multipartite graph and community detection

A multipartite graph is usually used to model heterogeneous data. In graph theory, a multipartite graph or k-partite graph is a graph whose vertices are or can be partitioned into k different independent sets. Equivalently, it is a graph that can be colored with k colors, while no two endpoints of an edge have the same color. Figure 1 shows two examples of multipartite graphs with different node sets. When $k = 2$, a k-partite graph is called a bipartite graph. When $k = 3$, a k-partite graph is called a tripartite graph, which generally means it is constructed using data from three heterogeneous sources.

Figure 2 shows an example of clustering results for a tripartite graph. The solid lines are the edges connecting different types of nodes. The dashed line polygons cluster the nodes into two different groups, which form two communities. Clustering is an unsupervised approach, no labeled data is required. Compared with the traditional clustering approaches built on just the



(a) $k = 2$ (b) $k = 3$

Fig. 1 Examples of multipartite graph with different k.

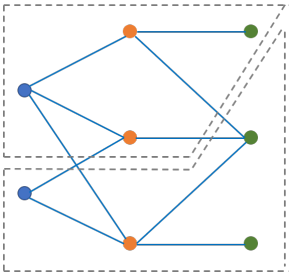


Fig. 2 An example of tripartite graph co-clustering problem.

information from a graph with one type of node, clustering multipartite graph nodes (with different types of nodes) can derive more meaningful and useful statistics or information from the graph.

For unipartite graphs, there are usually two main approaches to detect communities. One is based on modeling the community structure or topology, and the other is based on extracting it from flow calculations. Analyzing multipartite graph as a separate network category to investigate the community structure has become popular in the literature[17, 18]. To handle the multipartite graph cases, one popular way is to utilize the edge features to simplify the multipartite graph, as edges usually are one type. However, such transformation may lose too much information to determine communities accurately. Recently, researchers have proposed approaches based on Non-negative Matrix Factorization (NMF)[19] and ranking[20].

2.2 NMF

NMF is a technique for obtaining low rank representation of matrices that have only non-negative elements[21]. It has many applications, including information retrieval and text mining[22]. The constraints of non-negative basis vectors differ from other rank reduction method, such as Principal Component Analysis (PCA)[23] and Singular Value Decomposition (SVD)[24], which makes it ideal for situations where non-negative numbers are used for data interpretation/representation, for example, pixel intensities in image processing. Generally, NMF aims to factor a data matrix $\boldsymbol{A}$ into two lower dimension matrices ($\boldsymbol{W}$ and $\boldsymbol{H}$), and minimize the square error between the original matrix $\boldsymbol{A}$ and the multiplication of those two matrices,

$$\begin{aligned} &\min ||\boldsymbol{A}-\boldsymbol{W}\boldsymbol{H}||_{\mathrm{F}}^{2}, \\ &\text{subject to } \forall i, j, [\boldsymbol{W}]_{i,j}, [\boldsymbol{H}]_{i,j} \geqslant 0 \end{aligned} \tag{1}$$

where $\boldsymbol{W}$ and $\boldsymbol{H}$ are called dictionary matrix and expansion matrix, respectively. The challenge here is how to effectively identify these two matrices. There are multiple variations to the basic NMF approach wherein additional constraints, such as sparsity and orthogonality, are imposed to limit the solution space for the decomposed result. Decomposing into three matrices has also been proposed. It is called non-negative matrix tri-factorizations and it shows good performance in approximation[25].

The optimization problem in Formula (1) is a non-convex optimization with respect to variables $\boldsymbol{W}$ and $\boldsymbol{H}$. Thus, a local optima is often encountered. To find optimized solutions for the matrices, one common approach to update matrices is the multiplicative updating algorithm[21]. However, it has poor performance[26] and convergence issues[27]. There are some other approaches, such as the block principal pivoting method[28] and the random projections method[29], that can be used to update matrices. These methods try to overcome the shortcomings of the multiplicative updating algorithm, and generally have better performance.

2.3 Sentiment analysis

Sentiment analysis or opinion mining is a natural language processing technique used to extract subjective information, usually in three classes: positive, neutral, and negative. A large number of research in sentiment analysis focused on identifying text polarity[30–34]. There are some other research focusing on feelings and emotions, such as angry or happy, and intentions[35–37]. Sentiment analysis can help gauge public opinion, conduct nuanced research, and monitor extensive data tends effectively.

3 Pandemic Analysis Through Twitter Data

In this section, we discuss how we construct the tripartite graph, the notations, and the problem formulation for pandemic analysis.

3.1 Tripartite graph in twitter

The important information of a tweet includes: (1) user, (2) tweet text, and (3) hashtag/keyword. The relationships among them are straightforward: a user can like or comment or post a tweet, and a tweet might have some topics/hashtags/keywords. In other words, users will perform actions (e.g., like/comment) on tweets, while each tweet is associated with certain topics/hashtags/keywords. As users do not directly perform actions on the topics/keywords/hashtags, we can abstract the relationships among user, tweet contents,

and topics/keywords/hashtags into a tripartite graph. For example, Fig. 3 is an example to model Twitter data as a tripartite graph. The tripartite graph is composed of three types of nodes: user nodes, tweet nodes, and topic nodes. In the tripartite graph, the user nodes only connect with the tweet nodes, while the tweet nodes only connect with the topic nodes. In Fig. 3, the solid lines with red heart icon and message icon represent like or comment relationship between users and tweets, respectively; while the lines without icon represent the containing relationship between tweets and topics.

3.2 Problem definition

We denote the raw data from the Twitter platform with a 3-tuple $RD = \langle U, T, H \rangle$, where U, T, and H represent the set of users, tweets, and topics, respectively. Note that U, T, and H are three mutually disjoint sets (i.e., $(U \cap T) \cup (T \cap H) \cup (H \cap U) = \varnothing$).

Given the raw data, our target is to generate the community's attitude towards COVID-19 events via the following phases: (1) generate tripartite graph representation from the raw data, (2) detect the communities via graph clustering, and (3) infer sentiments for each community. The attitude of each community will be represented by positive or neutral or negative. Table 1 shows the notations used in this work.

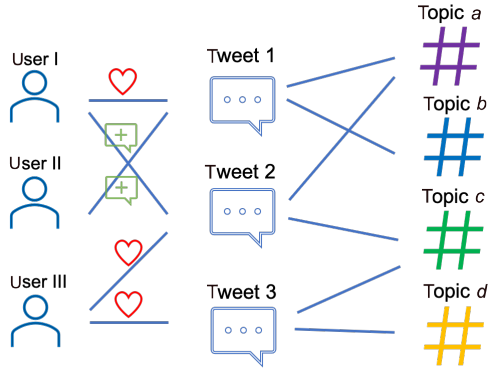


Fig. 3 An example of tripartite graph in Twitter.

4 Pandemic Data Analysis Framework

In this section, we propose a framework of TGC-PDA to automatically collect, cluster, and infer the sentiments from the observed tweets.

Figure 4 shows the overview of the TGC-PDA framework. The input of the framework is Twitter raw data. TGC-PDA consists of three main steps: (1) tripartite graph representation, (2) clustering, and (3) sentiment analysis. In the tripartite graph representation step, we find a mathematical model to represent the data with less information loss. Then, the clustering step builds a matrix factorization based on clustering algorithm to find the communities in the graph. Finally,

Table 1 Notation.

Symbol	Definition
n, m, t	Number of users, tweets, and topics
$G(V, E)$	Graph with node set V and edge set E
U, T, H	Node set of users, tweets, and topics
$\boldsymbol{B}$	Matrix representation of a bipartite graph
$P_{i,j}$	Number of paths between node i and node j
$\boldsymbol{L}$	Normalized Laplacian matrix
$\boldsymbol{D}$	Degree matrix: diagonal with $[\boldsymbol{D}]_{i,i} = \text{degree}(v_i)$
$\boldsymbol{F}, \boldsymbol{G}$	Decomposed matrices: $\boldsymbol{F} \in \Psi^{n \times d}$ and $\boldsymbol{G} \in \Psi^{k \times n}$
$\boldsymbol{S}$	Association matrix: $\boldsymbol{S} \in R_{+}^{d \times k}$
Ψ	Set of all cluster indicator matrices
$\text{Tr}(\boldsymbol{X})$	Trace of matrix $\boldsymbol{X}$: $\text{Tr}(\boldsymbol{X}) = \sum_1^n x_{i,i}$
$\|\|\boldsymbol{X}\|\|_{\text{F}}$	Frobenius norm of a matrix $\boldsymbol{X}$

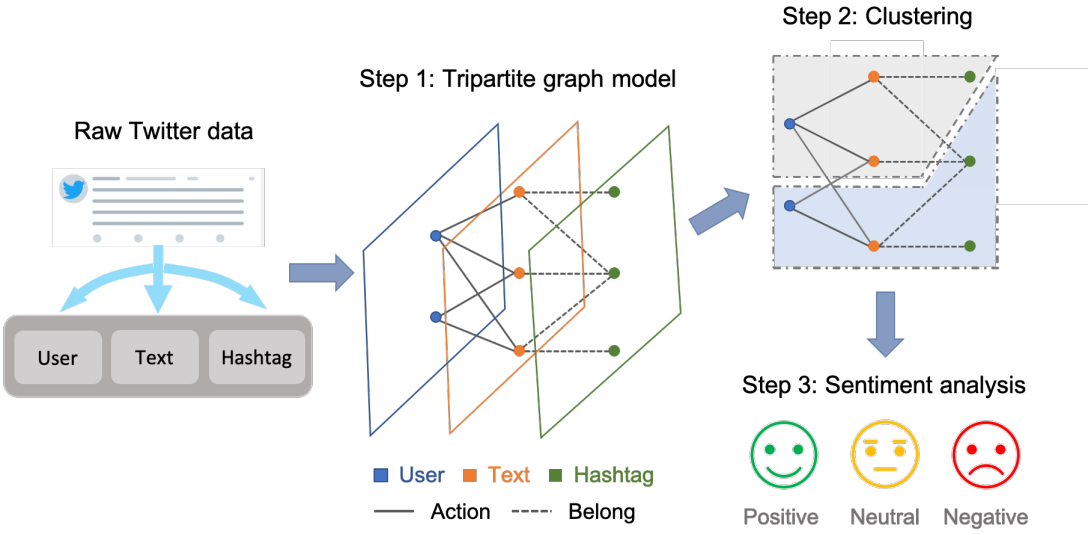


Fig. 4 An overview of the TGC-PDA framework.

the sentiment analysis step extracts attitudes within the communities.

4.1 Tripartite graph representation

A tripartite graph $G(V, E)$ can be constructed from the data to represent the relationships among U, T, and H, where V and E represent the node set and edge set, respectively. In G, we have $V = \{U \cup T \cup H\}$, and $E = \{E_{U,T} \cup E_{T,H}\}$, while $E_{U,T}$ is the edge set between the user nodes and the tweet nodes, and $E_{T,H}$ is the edge set between the tweet nodes and the topic nodes. In graph theory, a tripartite graph is complete if and only if each node in one set of nodes is fully connected with all nodes in the adjacent set. Based on the data we obtained from Twitter, the tripartite graph generated in our case is not complete.

Different from the traditional machine learning techniques or frameworks[38, 39], the TGC-PDA will employ the proposed tripartite graph structure to organize the data. With the tripartite graph, the arising challenge is to identify a proper representation of the graph to embed useful information for further processes, such as clustering and sentiment analyzing. To represent the graph and find a suitable clustering solution for a tripartite graph, one way is to divide the graph into two bipartite graphs. For example, we can build user-tweet bipartite graph and tweet-topic graph, and then find clusters over these two graphs separately. In this case, the connections between users and topics are lost. In addition, the relations between users, which have been proved important in social network analysis, are not considered either.

Based on the analysis above, we propose to build a user-topic bipartite graph and a user-tweet bipartite graph, as shown in Fig. 5. We use tweet-level nodes as the bridges to build the connection between user and topic nodes. In Fig. 5, Node I has three paths (a path is a finite sequence of edges connecting two end nodes) connecting to Node b, which goes through Node 1 and Node 2. Accordingly, we can set the edge weight between Node I and Node b as 3. The user-topic bipartite graph can be denoted by $\boldsymbol{B}_h^{n\times t}$ and the matrix representation for user-tweet bipartite graph is denoted by $\boldsymbol{B}_u^{n\times m}$. Both of the bipartite graph matrices are non-negative and the detailed deduction will be discussed in Section 5.

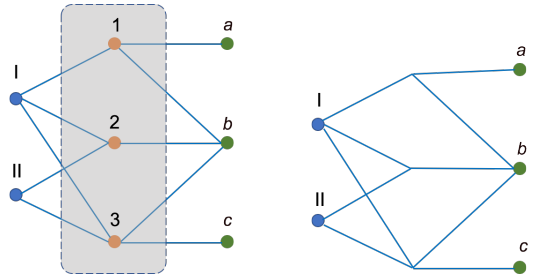


Fig. 5 Build the user-topic bipartite by removing the tweet nodes of the tripartite graph and leveraging the tweet nodes as the connection for user and topic nodes.

4.2 Non-negative Matrix Factorization with Regularization (NMFR)

In the second step of the framework, we need to find the clustering result of the input graph data. Since the matrix representation of the graph is a non-negative matrix, it is straightforward to use the NMF for clustering. In this way, for the user-topic bipartite graph generated in Section 4. At first, we can find an intermediate clustering result of the graph by applying the clustering algorithms. Then, we can feed the intermediate clustering result into clustering process of the user-tweet bipartite graph, and the clusters can be found accordingly.

Because users tend to have consistent preferences, it would be preferable to make tweet nodes close to their user nodes. In other words, node locality needs to be preserved. Thus, standard NMF may not work properly. In fact, as to be described later, our experiment results show that the accuracy of NMF is poor. Hence, we propose the graph regularization technique into NMF to smooth the result[40]. In specific, to cluster users and topics, we use the NMFR modeled as the following formula:

$$\min \ ||\boldsymbol{B}_h\boldsymbol{B}_h^{\mathrm{T}} - \boldsymbol{F}_h\boldsymbol{S}_h\boldsymbol{G}_h||_{\mathrm{F}}^2 + \alpha \operatorname{Tr}(\boldsymbol{F}_h^{\mathrm{T}}\boldsymbol{L}_f\boldsymbol{F}_h) + \beta \operatorname{Tr}(\boldsymbol{G}_h\boldsymbol{L}_g\boldsymbol{G}_h^{\mathrm{T}}) \quad (2)$$

where $||\boldsymbol{X}||_{\mathrm{F}}$ denotes the Frobenius norm of a matrix $\boldsymbol{X}$. The Frobenius norm is a matrix norm of an $m \times n$ matrix $\boldsymbol{X}$, which is defined as the square root of the sum of the absolute squares of all elements, as shown in the following:

$$||\boldsymbol{X}||_{\mathrm{F}} = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|x_{i,j}|^2} \quad (3)$$

In Formula (2), $\boldsymbol{B}_h$ is the matrix representation of the user and topic bipartite graph, which can be defined by

$$[\boldsymbol{B}_h]_{i,j} = \begin{cases} P_{i,j}, & \text{if } i \text{ and } j \text{ are connected;} \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

where $P_{i,j}$ is the number of paths between node i and node j. For example, the matrix representation of Fig. 5 will be

$$\boldsymbol{B}_h = \begin{array}{c} \\ \mathrm{I} \\ \mathrm{II} \end{array} \begin{array}{c} \begin{array}{ccc} a & b & c \end{array} \\ \begin{bmatrix} 1 & 3 & 1 \\ 0 & 2 & 1 \end{bmatrix} \end{array}$$

In Formula (2), $\boldsymbol{B}_h\boldsymbol{B}_h^{\mathrm{T}}$ represents the pairwise similarity matrix, $\boldsymbol{F}_h \in \Psi^{n\times d}$ and $\boldsymbol{G}_h \in \Psi^{k\times n}$ are non-negative matrices and are cluster indicators, respectively, where Ψ is the set of all cluster indicator matrices. $\boldsymbol{S}_h \in R_+^{d\times k}$ is used to increase the degree of freedom, such that the low-rank matrix representation has better approximation[41]. The coefficients α and β are regularization parameters to smooth and balance the reconstruction error. $\boldsymbol{L}_f$ and $\boldsymbol{L}_g$ are the normalized graph Laplacian matrices, with the definition of: $\boldsymbol{L} = \boldsymbol{I} - \boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{W}\boldsymbol{D}^{-\frac{1}{2}}$, where $\boldsymbol{I}$ is identity matrix, $\boldsymbol{D}$ is degree matrix, and $\boldsymbol{W}$ is adjacency matrix. $\boldsymbol{L}_g$ and $\boldsymbol{L}_f$ share the same definition with their specific graph structure, and also the corresponding $\boldsymbol{D}$($\boldsymbol{D}_g$ and $\boldsymbol{D}_f$) and $\boldsymbol{W}$($\boldsymbol{W}_g$ and $\boldsymbol{W}_f$). Tr() represents the trace of a matrix. By optimizing the loss function, we can have the clustering results for the user-topic bipartite graph. To cluster the user-tweet bipartite graph, we utilize the cluster for tweets based on the clustering results of users. If one tweet belongs to different clusters, we use the majority vote strategy to choose a proper placement, which will be the clustering result for user and tweet bipartite graph $\boldsymbol{B}_u$. The detailed optimization algorithm to iteratively update the matrices in Formula (2) will be explained in Section 5.

4.3 Sentiment analysis

As our goal is to extract open source intelligence from each community, we aggregate the tweets based on their cluster labels. Then, we run a sentiment analysis with a mini-batch algorithm when running the full-batch algorithms is intractable. We use the sentiment analysis library, such as Textblob[42], to provide a quantitative result for the polarity in one cluster. Textblob is one of the commonly used libraries for processing textual data[42]. It provides APIs to handle natural language processing tasks, including text cleaning and sentiment analysis. To get the polarity of a cluster, we measure the percentage of positive, neutral, and negative tweets in that cluster. This way we can figure out the overall attitudes of the users in one cluster for the COVID-19 related events.

5 NMFR Updating Algorithm

In this section, we focus on solving the objective function that is formulated as a minimization problem in Formula (2), where $\boldsymbol{F}_h$ and $\boldsymbol{G}_h$ are the cluster indicator matrices. The multiplication result of $\boldsymbol{B}_h\boldsymbol{B}_h^{\mathrm{T}}$ is a symmetric matrix. From Ref. [41], we know that the loss function can be transformed to the following problem:

$$\begin{aligned} &\min\ ||\boldsymbol{B}_h\boldsymbol{B}_h^{\mathrm{T}} - \boldsymbol{F}_h\boldsymbol{S}_h\boldsymbol{G}_h||_{\mathrm{F}}^2 + \\ &\alpha||\boldsymbol{F}_h - \boldsymbol{X}_f\boldsymbol{Y}_f||_{\mathrm{F}}^2 + \beta||\boldsymbol{G}_h^{\mathrm{T}} - \boldsymbol{X}_g\boldsymbol{Y}_g||_{\mathrm{F}}^2, \\ &\text{s.t.,}\quad \boldsymbol{Y}_g^{\mathrm{T}}\boldsymbol{Y}_g = \boldsymbol{I}, \boldsymbol{Y}_f^{\mathrm{T}}\boldsymbol{Y}_f = \boldsymbol{I} \end{aligned} \tag{5}$$

where $\boldsymbol{X}_g = \boldsymbol{W}_f^{-1/2}\boldsymbol{D}_g^{-1/2}$ and $\boldsymbol{X}_f = \boldsymbol{D}_f^{-1/2}\boldsymbol{W}_f^{-1/2}$, $\boldsymbol{Y}_f$ and $\boldsymbol{Y}_g$ are arbitrary orthonormal matrices.

To get $\boldsymbol{S}_h$, we can fix $\boldsymbol{F}_h$ and $\boldsymbol{G}_h$. Then, only the first term in Formula (5) can affect the minimization process and the rest two terms are constants. Accordingly, we can calculate $\boldsymbol{S}_h$ by setting the derivative of Formula (5) to zero, and obtain

$$\boldsymbol{S}_h = (\boldsymbol{F}_h^{\mathrm{T}}\boldsymbol{F}_h)^{-1}\boldsymbol{F}_h^{\mathrm{T}}\boldsymbol{B}_h\boldsymbol{B}_h^{\mathrm{T}}\boldsymbol{G}_h^{\mathrm{T}}(\boldsymbol{G}_h\boldsymbol{G}_h^{\mathrm{T}})^{-1} \tag{6}$$

Next, if we fix $\boldsymbol{F}_h$, $\boldsymbol{G}_h$, and $\boldsymbol{S}_h$, we can obtain $\boldsymbol{Y}_f$ and $\boldsymbol{Y}_g$ by doing the singular value decomposition to $\boldsymbol{X}_f^{\mathrm{T}}\boldsymbol{F}_h$ and $\boldsymbol{X}_g^{\mathrm{T}}\boldsymbol{G}_h^{\mathrm{T}}$.

Similarly, we can fix $\boldsymbol{F}_h$, $\boldsymbol{S}_h$, and $\boldsymbol{Y}_g$ and find that the second term in Formula (5) is constant. As a result, we only need to optimize the first and the third terms. Since $\boldsymbol{G}_h$ is the cluster indicator matrix, it can be obtained in the following equation:

$$\mathrm{Tr}(\boldsymbol{M}^{\mathrm{T}}\boldsymbol{Z} - \boldsymbol{M}^{\mathrm{T}}\boldsymbol{M}\boldsymbol{G} - \beta\boldsymbol{G} + \beta\boldsymbol{Y}^{\mathrm{T}}) = 0 \tag{7}$$

where

$$\boldsymbol{M} = \boldsymbol{F}_h\boldsymbol{S}_h, \boldsymbol{Z} = \boldsymbol{B}_h\boldsymbol{B}_h^{\mathrm{T}}, \boldsymbol{G} = \boldsymbol{G}_h, \boldsymbol{Y} = \boldsymbol{X}_g\boldsymbol{Y}_g.$$

To calculate $\boldsymbol{F}_h$, we can fix $\boldsymbol{G}_h$, $\boldsymbol{S}_h$, and $\boldsymbol{Y}_f$. Then, the third term in Formula (5) is constant and we only need to minimize the first and the second term. Now, the cluster indicator matrix $\boldsymbol{F}_h$ can be obtained in the following equation:

$$\mathrm{Tr}(\boldsymbol{Z}\boldsymbol{M}^{\mathrm{T}} - \boldsymbol{F}\boldsymbol{M}\boldsymbol{M}^{\mathrm{T}} - \alpha\boldsymbol{F} + \alpha\boldsymbol{Y}) = 0 \tag{8}$$

where

$$\boldsymbol{M} = \boldsymbol{S}_h\boldsymbol{G}_h, \boldsymbol{Z} = \boldsymbol{B}_h\boldsymbol{B}_h^{\mathrm{T}}, \boldsymbol{F} = \boldsymbol{F}_h, \boldsymbol{Y} = \boldsymbol{X}_f\boldsymbol{Y}_f.$$

Algorithm 1 shows the pseudocode for the proposed NMFR Updating (NMFRU) algorithm, which can cluster the graph in phase two of the TGC-PDA framework. The basic idea of the proposed NMFRU is to fix some factors and update one parameter at a time. Here, we start with one parameter that appears least frequently in the loss function and iteratively update the matrices.

6 Experimental Result

In this section, we analyze our experimental results and the performance of TGC-PDA. We also compare

Algorithm 1 MNFRU algorithm

Input: data matrix $\boldsymbol{B}_h$, parameters α and β
Output: $\boldsymbol{F}_h, \boldsymbol{S}_h$, and $\boldsymbol{G}_h$

1: Initialize $\boldsymbol{F}_h$ and $\boldsymbol{G}_h$ with random class indicator matrices;
2: Calculate $\boldsymbol{X}_f$ and $\boldsymbol{X}_g$;
3: **While** it does not converge
Update $\boldsymbol{S}_h$ according to Eq. (6);
Calculate $\boldsymbol{Y}_f$ by doing singular value decomposition to $\boldsymbol{X}_f^{\mathrm{T}}\boldsymbol{F}_h$;
Calculate $\boldsymbol{Y}_g$ by doing singular value decomposition $\boldsymbol{X}_g^{\mathrm{T}}\boldsymbol{G}_h^{\mathrm{T}}$;
Update $\boldsymbol{G}_h$ according to Eq. (7);
Update $\boldsymbol{F}_h$ according to Eq. (8);
4: **Return** $\boldsymbol{F}_h, \boldsymbol{S}_h$, and $\boldsymbol{G}_h$

NMFRU with the well-known clustering methods, such as Kmeans, NMF, and the commonly used variants, including Semi-NMF (SNMF)[43] and Orthogonal NMTF (ONMTF)[44].

6.1 Dataset

We evaluate the performance of TGC-PDA with real Twitter dataset about "Covid-19" collected between Feb. 15th, 2020 and Sep. 30th 2020. To get the tweet data, we wrote a python program to crawl the tweets and the users who liked them. Multiple hashtag keywords, such as #COVID19, #coronavirus, #covid, covid pandemic, and #COVID20 are used to ensure we can get a large dataset. Since the free Twitter API we use has rate limits and it restricts the number of retrieved tweets during each login access, we have to crawl the data for several months. After removing the duplicate and non-English posts, we obtain 18 327 tweets, with 752 649 users who interacted with the tweets. Some users only interacted with one tweet in our dataset, which are identified as "less interactive" users and excluded. After the data cleanup, we have 301 982 users left.

6.2 Experimental setup

As all the clustering methods (i.e., Kmeans, NMF, SNMF, ONMTF, and our NMFRU) have one or more parameters to be tuned, to make the comparison fair, we run these methods under different parameters and choose the best result for each algorithm. In addition, we set the number of clusters as the true number of classes for all clustering algorithms on the dataset. In specific, for Kmeans and NMF algorithms, the hyperparameter is k_{cluster} (number of clusters). If k_{cluster} is given, no other parameters would be needed. In NMFR, we have two hyperparameters: α and β. To find a proper value for these parameters, we plot a loss-value curve, with value ranging from 0.1 to 1000. Then, the α and β values can be found by scanning the plot. Since our data size is relatively large and cannot be completely labeled manually, we randomly choose 5% of the data to label and use the result tested by sample data as the framework result.

6.3 Evaluation metrics

To evaluate the clustering result, we use the widely used standard metrics, including the clustering accuracy, cluster purity, and Normalized Mutual Information (NMI).

For the clustering accuracy, we compare the outputted clusters $c \in \boldsymbol{C}_{\text{out}}$ with the ground truth labelled data $g \in \boldsymbol{C}_{\text{ground}}$. The accuracy is defined as follows:

$$\text{Accuracy}(\boldsymbol{C}_{\text{out}}, \boldsymbol{C}_{\text{ground}}) = \frac{1}{k_{\text{sample}}} \sum \delta(g_i, \text{map}(c_i)) \tag{9}$$

where k_{sample} is total number of data samples, $\delta(a, b)$ is the delta function, in which the value equals one if $a = b$, and equals zero otherwise. The $\text{map}(c_i)$ is a mapping function that maps each cluster label c_i to the same label in the ground truth data. We can use Kuhn-Munkres algorithm to find the best mappings[45].

For the cluster purity, we compare the cluster output $c \in \boldsymbol{C}_{\text{out}}$ with the ground truth labelled data $g \in \boldsymbol{C}_{\text{ground}}$. The purity of the cluster result is calculated,

$$\text{Purity}(\boldsymbol{C}_{\text{out}}, \boldsymbol{C}_{\text{ground}}) = \frac{1}{k_{\text{sample}}} \sum_{c \in \boldsymbol{C}_{\text{out}}} \max_{g \in \boldsymbol{C}_{\text{ground}}} (c \cap g) \tag{10}$$

where k_{sample} is the number of data samples. A perfect clustering result has a purity of one, and a bad one has the purity value close to zero.

For the NMI, we compare the cluster output $c \in \boldsymbol{C}_{\text{out}}$ with the ground truth labelled data $g \in \boldsymbol{C}_{\text{ground}}$. The NMI is defined as

$$\text{NMI}(\boldsymbol{C}_{\text{out}}, \boldsymbol{C}_{\text{ground}}) = \frac{2 \times I(\boldsymbol{C}_{\text{out}}, \boldsymbol{C}_{\text{ground}})}{[H(\boldsymbol{C}_{\text{out}}) + H(\boldsymbol{C}_{\text{ground}})]} \tag{11}$$

where $H()$ represents the entropy, and $I(\boldsymbol{C}_{\text{out}}, \boldsymbol{C}_{\text{ground}})$ denotes the mutual information between $\boldsymbol{C}_{\text{out}}$ and $\boldsymbol{C}_{\text{ground}}$. A higher NMI value means the better clustering result.

To obtain a less biased estimation of the framework, we run NMFRU algorithms twenty times and take the average result for each model.

6.4 Results and discussion

Table 2 shows the comparison between NMFRU and several baseline models, such as Kmeans, NMF, SNMF, and ONMTF. When applying these baseline models to

Table 2 Performance results of classifiers.

Method	Accuracy	Purity	NMI
Kmeans	0.613	0.549	0.513
NMF	0.583	0.536	0.493
SNMF	0.627	0.562	0.534
ONMTF	0.674	0.578	0.557
NMFRU	0.706	0.617	0.621

our data, we do not embed the topic nodes to user nodes. Instead, we use the user and tweet bipartite graph to calculate the clustering result. The matrix form of the bipartite graph is that the columns and rows correspond to the two sets of vertices, with each entry corresponding to an edge between a column and a row. From Table 2, we can see that NMFRU achieves the best performance in terms of accuracy, purity, and NMI. This is because our bipartite graph is created based on our tripartite graph model, and it embedded more information than the plain bipartite graph. We also utilize the tri-factorization and locality preserved schemes, which can further improve the performance.

We study the average convergence time of our framework in Fig. 6. When the number of iterations is around 23, our framework tends to converge with a total loss of 2, which shows that the calculation of NMFRU is fast. Meanwhile, when comparing the convergence time by the different baseline methods in Fig. 7, we can see that NMFRU is slower than Kmeans but faster than other baseline clustering methods. It is because we do fewer matrix multiplication operations in NMFRU, hence saving some running time. Therefore, TGC-PDA that utilizes NMFRU as the core clustering algorithm can be used for a large dataset.

As for the polarity of the communities, Table 3 shows the largest ten communities with its polarity ratio. From Table 3, we find that the neutral ratio is quite high among all topics. In order to figure out the rationale here, we

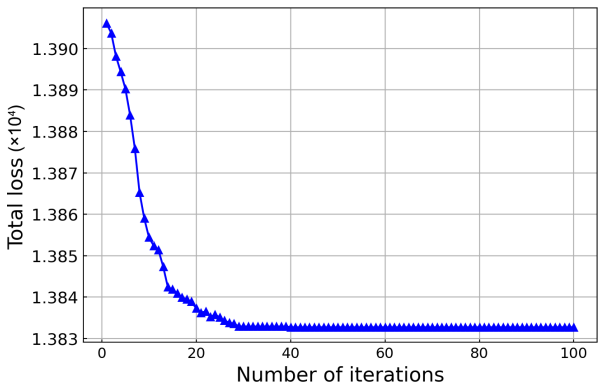


Fig. 6 Total loss with different numbers of iterations.

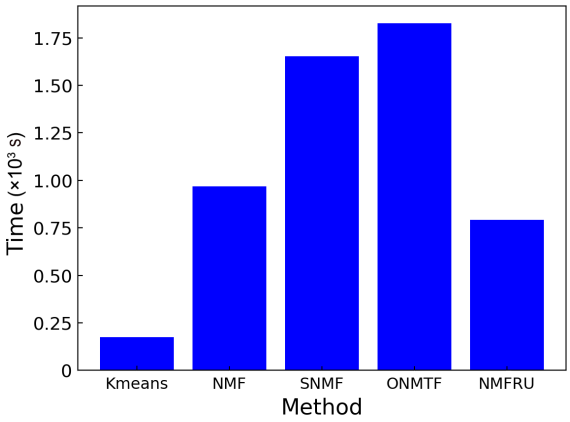


Fig. 7 Convergence time of methods.

Table 3 Largest ten communities with its polarity ratio. (%)

Keyword	Positive	Neutral	Negative
marketcrash2021	18.2	48.7	33.1
maskshortage	14.1	41.6	44.3
death	4.1	73.1	22.8
NYbreak	12.2	57.5	30.3
antibody	30.5	41.3	28.2
stimulus	32.6	41.7	25.7
testing	32.7	38.9	28.4
vaccine	20.4	61.2	18.4
symptoms	26.3	48.9	24.8
stayathome	23.6	51.8	24.6

manually examined 1000 posts and found that there are lots of media or government agencies (e.g., CNN and CDC) that use Twitter to publish real-time news and the latest policy. These tweets tend to be retweeted many times. Obviously, such tweets are more likely to be considered neutral.

7 Conclusion

The outbreak of COVID-19 makes the whole world chaotic. People often search for real-time news and ventilate their emotions through the Internet. OSNs are widely used for opinions sharing, news publishing, and information spreading. The large useful data from OSNs can be leveraged to help public officials and governments make better decisions. In this paper, we build a framework of TGC-PDA to utilize Twitter data to monitor and automatically collect the voice of the people during COVID-19 pandemic. The TGC-PDA framework takes advantage of the characteristics of the Twitter users and tweets network structure to effectively analyze the community structures and sentiments. It enables us to extract the open source intelligence from

each community, which could be utilized to track people's feedbacks and opinions towards the coronavirus pandemic events. Our work currently is a pioneering work and it only focused on English-language tweets. It would be feasible to extend our work to handle tweets in other languages. Similar techniques can be applied to other online and publicly available social media platforms, such as Reddit. Since a tweet may contain not only text, but also embedded hyperlinks, images, or even videos, it would be interesting and challenging to explore more information from them. Moreover, some events during COVID-19 are time-sensitive, it would be also interesting to study the tweets from the perspective of time-series analysis.

References

[1] Everyone included: Social impact of COVID-19, https://www.un.org/development/desa/dspd/everyone-included-covid-19.html, 2020.

[2] Wikipedia, COVID-19 pandemic, https://en.wikipedia.org/wiki/COVID-19pandemic, 2021.

[3] Domestic travel during the COVID-19 pandemic, https://www.cdc.gov/coronavirus/2019-ncov/travelers/travel-during-covid19.html, 2020.

[4] Travelers prohibited from entry to the United States, https:// www.cdc.gov/coronavirus/2019-ncov/travelers/from-other- countries.html, 2020.

[5] K. Cohen, Tokyo 2020 Olympics officially postponed until 2021, https://tv5.espn.com/olympics/story/_/id/28946033/tokyo-olympics-officially-postponed-2021, 2020.

[6] Wikipedia, RNA virus, https://en.wikipedia.org/wiki/RNAvirus, 2021.

[7] How does fake news of 5G and COVID-19 spread worldwide?, https://www.medicalnewstoday.com/articles/5g-doesnt-cause-covid-19-but-the-rumor-it-does-spread-like-a-virus, 2021.

[8] L. J. Chang, W. Li, L. Qin, W. J. Zhang, and S. Y. Yang, pSCAN: Fast and exact structural graph clustering, *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 2, pp. 387–401, 2017.

[9] R. El Bacha and T. T. Zin, Ranking of influential users based on user-tweet bipartite graph, in *Proc. of 2018 IEEE Int. Conf. Service Operations and Logistics, and Informatics* (*SOLI*), Singapore, 2018, pp. 97–101.

[10] A. Rodríguez, C. Argueta, and Y. L. Chen, Automatic detection of hate speech on facebook using sentiment and emotion analysis, in *Proc. of 2019 Int. Conf. Artificial Intelligence in Information and Communication* (*ICAIIC*), Okinawa, Japan, 2019, pp. 169–174.

[11] J. Zhou and C. Kwan, Missing link prediction in social networks, in *Proc. 15th Int. Symp. Neural Networks*, Minsk, Belarus, 2018, pp. 346–354.

[12] A. Reyes-Menendez, J. R. Saura, and C. Alvarez-Alonso, Understanding #worldEnvironmentDay user opinions in twitter: A topic-based sentiment analysis approach, *Int. J. Environ. Res. Public Health*, vol. 15, no. 11, p. 2537, 2018.

[13] C. H. Tan, L. L. Lee, J. Tang, L. Jiang, M. Zhou, and P. Li, User-level sentiment analysis incorporating social networks, in *Proc. 17th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, New York, NY, USA, 2011, pp. 1397–1405.

[14] A. Giachanou and F. Crestani, Like it or not: A survey of twitter sentiment analysis methods, *ACM Comput. Surv.*, vol. 49, no. 2, p. 28, 2016.

[15] R. R. Iyer, J. Chen, H. N. Sun, and K. Y. Xu, A heterogeneous graphical model to understand user-level sentiments in social media, arXiv preprint arXiv: 1912.07911, 2019.

[16] H. B. Deng, J. W. Han, H. Li, H. Ji, H. N. Wang, and Y. Lu, Exploring and inferring user–user pseudo-friendship for sentiment analysis with heterogeneous networks, *Stat. Anal. Data Min.*, vol. 7, no. 4, pp. 308–321, 2014.

[17] C. A. Phillips, Multipartite graph algorithms for the analysis of heterogeneous data, PhD dissertation, Univ. Tennessee, Knoxville, TN, USA, 2015.

[18] D. W. Zhou, S. Zhang, M. Y. Yildirim, S. Alcorn, H. H. Tong, H. Davulcu, and J. R. He, A local algorithm for structure-preserving graph cut, in *Proc. 23rd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, Halifax, Canada, 2017, pp. 655–664.

[19] P. M. Comar, P. N. Tan, and A. K. Jain, A framework for joint community detection across multiple related networks, *Neurocomputing*, vol. 76, no. 1, pp. 93–104, 2012.

[20] Y. Z. Sun, Y. T. Yu, and J. W. Han, Ranking-based clustering of heterogeneous information networks with star network schema, in *Proc. 15th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, Paris, France, 2009, pp. 797–806.

[21] D. D. Lee and H. S. Seung, Algorithms for non-negative matrix factorization, in *Proc. 13th Int. Conf. Neural Information Proc. Systems*, Cambridge, MA, USA, 2001, pp. 535–541.

[22] N. Gillis, The why and how of nonnegative matrix factorization, arXiv preprint arXiv: 1401.5226v2, 2014.

[23] H. Abdi and L. J. Williams, Principal component analysis, *WIRs Comput. Stat.*, vol. 2, no. 4, pp. 433–459, 2010.

[24] M. E. Wall, A. Rechtsteiner, and L. M. Rocha, Singular value decomposition and principal component analysis, in *A Practical Approach to Microarray Data Analysis*, D. P. Berrar, W. Dubitzky, M. Granzow, eds. Norwell, MA, USA: Springer, 2003, pp. 91–109.

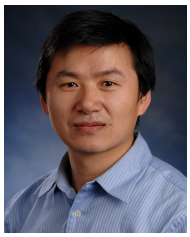
[25] C. Ding, T. Li, W. Peng, and H. Park, Orthogonal nonnegative matrix t-factorizations for clustering, in *Proc. 12th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, Philadelphia, PA, USA, 2006, pp. 126–135.

[26] D. Kim, S. Sra, and I. S. Dhillon, Fast newton-type methods for the least squares nonnegative matrix approximation problem, in *Proc. 2007 SIAM Int. Conf. Data Mining*, Minneapolis, MN, USA, 2007, pp. 343–354.

[27] C. J. Lin, On the convergence of multiplicative update algorithms for nonnegative matrix factorization, *IEEE Trans. Neural Netw.*, vol. 18, no. 6, pp. 1589–1596, 2007.

[28] J. Kim and H. Park, Toward faster nonnegative matrix factorization: A new algorithm and comparisons, in *Proc. o*f 2008 Eighth IEEE Int. Conf. Data Mining, Pisa, Italy,

2008, pp. 353–362.
[29] F. Wang and P. Li, Efficient nonnegative matrix factorization with random projections, in *Proc. 2010 SIAM Int. Conf. Data Mining*, Columbus, OH, USA, 2010, pp. 281–292.
[30] M. Annett and G. Kondrak, A comparison of sentiment analysis techniques: Polarizing movie blogs, in *Proc. 21st Conference of the Canadian Society for Computational Studies of Intelligence*, Windsor, Canada, 2008, pp. 25–35.
[31] R. Hillmann and M. Trier, Sentiment polarization and balance among users in online social networks, http://aisel.aisnet.org/amcis2012/proceedings/VirtualCommunities/10, 2021.
[32] M. Del Vicario, G. Vivaldo, A. Bessi, F. Zollo, A. Scala, G. Caldarelli, and W. Quattrociocchi, Echo chambers: Emotional contagion and group polarization on facebook, *Sci. Rep.*, vol. 6, p. 37825, 2016.
[33] S. M. Mohammad, X. D. Zhu, S. Kiritchenko, and J. Martin, Sentiment, emotion, purpose, and style in electoral tweets, *Informat. Proc. Manag.*, vol. 51, no. 4, pp. 480–499, 2015.
[34] K. Chakraborty, S. Bhattacharyya, R. Bag, and A. Hassanien, Sentiment analysis on a set of movie reviews using deep learning techniques, in *Social Network Analytics Computational Research Methods and Techniques*, Cambridge, MA, USA, 2019, pp. 127–147.
[35] K. Sailunaz and R. Alhajj, Emotion and sentiment analysis from twitter text, *J. Comput. Sci.*, vol. 36, p. 101003, 2019.
[36] H. Meisheri, K. Ranjan, and L. Dey, Sentiment extraction from consumer-generated noisy short texts, in *Proc. of 2017 IEEE Int. Conf. Data Mining Workshops* (*ICDMW*), New Orleans, LA, USA, 2017, pp. 399–406.
[37] A. S. M. Alharbi and E. de Doncker, Twitter sentiment analysis with a deep neural network: An enhanced approach using user behavioral information, *Cogn. Syst. Res.*, vol. 54, pp. 50–61, 2019.
[38] M. E. J. Newman, Modularity and community structure in networks, *Proc. Natl. Acad. Sci. USA*, vol. 103, no. 23, pp. 8577–8582, 2006.
[39] M. Wang, C. K. Wang, J. X. Yu, and J. Zhang, Community detection in social networks: An in-depth benchmarking study with a procedure-oriented framework, *Proc. VLDB Endow.*, vol. 8, no. 10, pp. 998–1009, 2015.
[40] D. Cai, X. F. He, X. Y. Wu, and J. W. Han, Non-negative matrix factorization on manifold, in *Proc. 2008 8th IEEE Int. Conf. Data Mining*, Pisa, Italy, 2008, pp. 63–72.
[41] H. Wang, F. P. Nie, H. Huang, and F. Makedon, Fast nonnegative matrix tri-factorization for large-scale data co-clustering, in *Proc. 22nd Int. Joint Conf. Artificial Intelligence*, Barcelona, Spain, 2011, pp. 1553–1558.
[42] TextBlob: Simplified text processing, https://textblob.readthedocs.io/en/dev/, 2020.
[43] C. H. Q. Ding, T. Li, and M. I. Jordan, Convex and semi-nonnegative matrix factorizations, *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 32, no. 1, pp. 45–55, 2010.
[44] H. Abe and H. Yadohisa, Orthogonal nonnegative matrix tri-factorization based on tweedie distributions, *Adv. Data Anal. Classi.*, vol. 13, no. 4, pp. 825–853, 2019.
[45] P. K. Shivaswamy and T. Jebara, Permutation invariant SVMs, in *Proc. 23rd Int. Conf. Machine Learning*, Pittsburgh, PA, USA, 2006, pp. 817–824.



Xiaojun Cao received the BEng degree from Tsinghua University, China in 1996, the MEng degree from Chinese Academy of Sciences, China in 1999, and the PhD degree in computer science from the State University of New York at Buffalo, USA in 2004. He is currently a professor at the Department of Computer Science, Georgia State University, where he leads the Advanced Network Research Group (aNet). Prior to joining Georgia State University, he was an assistant professor at the College of Computing and Information Sciences, Rochester Institute of Technology. He and his group are working on modeling, analysis, protocols/algorithms design, as well as data processing for networks and cyber physical systems. He was a distinguished lecturer of the IEEE ComSoc (2019–2020) and served as the secretary/vice chair/chair for IEEE ComSoc Optical Networking Technical Committee (ONTC). His research has been sponsored by U.S. National Science Foundation (NSF), Centers for Disease Control and Prevention (CDC), IBM, and Cisco's University Research Program. He is a recipient of NSF Career Award, 2006–2011.

Xueting Liao received the MEng degree from Rutgers University, USA in 2013. She is currently a PhD candidate in computer science at Georgia State University, USA. Her research interests include applications of data mining and graph mining algorithms, social network related analysis, and network related algorithms.

Danyang Zheng received the PhD degree in computer science from Georgia State University, USA in 2021. He is currently an assistant professor at Soochow University, China. His research interests include network function virtualization, software-defined networks, optical networks, networking performance optimization, and combinational optimization.