Graph Regularized Nonnegative Latent Factor Analysis Model for Temporal Link Prediction in Cryptocurrency Transaction Networks

Yue Zhou
School of Computer Science and Technology
Chongqing University of Posts and Telecommunications
Chongqing, 400065, China
yueyzhou@outlook.com

ZhiGang Liu
School of Computer Science and Technology
Chongqing University of Posts and Telecommunications
Chongqing, 400065, China
liuzhigang@cigit.ac.cn

Ye Yuan
College of Computer and Information Science
Southwest University
Chongqing, 400715, China
yuanyeik@gmail.com

Abstract—With the development of blockchain technology, a cryptocurrency based on blockchain technology is becoming more and more popular. The huge cryptocurrency transaction network has therefore received widespread attention. The link prediction learning structure of the network is supportive to understand the mechanism of networks, so it also has been widely studied in the cryptocurrency network. However, the dynamics of cryptocurrency transaction networks have been neglected in past studies. In this study, therefore, we use a graph-regularized method to link past transaction records with future transactions. Based on this, we propose a single latent factor-dependent, nonnegative, multiplicative, and graph regularized-incorporated update (SLF-NMGRU) algorithm and further propose a graph regularized nonnegative latent factor analysis (GrNLFA) model. Eventually, the experimental results on a real cryptocurrency transaction network show that the proposed method improves both the accuracy and computational efficiency.

Keywords—cryptocurrency transactions network, temporal link prediction, graph regularized method, nonnegative latent factor analysis

I. INTRODUCTION

With the wide application of blockchain technology, cryptocurrency has been broadly considered as a new electronic alternative to exchange currency [7-9]. Being different from traditional currency which needs the central authority to supervise the transaction, cryptocurrency is maintained by the distributed consensus to ensure the efficiency of cryptocurrency transactions [1-3]. More importantly, the transaction data recorded by blockchain technology is verifiable and permanent, which also ensures that the transaction records are unmodified. Therefore, cryptocurrencies provide a great convenience for data mining and analysis in this field, as the wealth information and record of financial transactions are completely contained in a list and stored in the chain which is open access [1-3]. Consequently, the formation of a huge cryptocurrency transaction network has been widely concerned [1-9].

Based on the transaction records of cryptocurrencies, a large transaction network is constructed. Noted that nodes in the networks are abstracted from objects in the cryptocurrency system, such as accounts, smart contracts, and entities, while edges are abstracted from their relationships. The link prediction plays a vital role in the analysis of the cryptocurrency transaction works because it is helpful to the construction and analysis of the whole system [5, 6]. Muzammal et al. [10] decompose the transfer graph into multiple subgraphs and then used probabilistic matrix decomposition (PMF) and the Bayesian PMF model to extract hidden features in the subgraphs. Meo et al. [11] propose a pairwise trust prediction through matrix factorization (PTP-MF) algorithm for pairwise trust prediction, and the algorithm also combines the deviation of the trustor and trustee behavior to predict the strength of trust and distrust relationship of users in cryptocurrency transaction networks. However, they ignore that cryptocurrency transaction data are nonnegative. To ensure nonnegative data, Yu et al. [12] propose a double nonnegative matrix factorization (DouNMF) model, which integrates degree information of nodes into nonnegative matrix factorization processes to obtain a more accurate node transfer matrix. Wang et al. [13] propose a regularized convex nonnegative matrix factorization model (RC-NMF), by introducing graph regularization into convex nonnegative matrix factorization, which could simultaneously constrain nodes with positive links to enter into the same community, besides it also constrains nodes with negative links to enter different communities.

However, the above-mentioned methods are inefficient because they need to operate a huge complete matrix in computation and storage. To improve the computational efficiency, Luo et al. [14-16] propose a single latent factor-dependent, nonnegative, multiplicative, and update (SLF-NMU) algorithm, which only depends on the known data in the network. Based on this algorithm, they further proposed a nonnegative latent factor analysis (NLFA) model. The computing cost of NLFA is linear only with the count of known items in its network, while the storage cost is linear only with the sum of its user count and item count. These methods are only applied to static cryptocurrency transaction networks. In the real world, however, cryptocurrency transaction networks are dynamic, which means that the
network structure evolves at different times. Analysis of the
dynamic flow of cryptocurrency is important to the analysis of
the entire network. Being compared with the static
network, the dynamic network is more accurate in the
representation of complex systems. Therefore, temporal link
prediction has been broadly concerned.

This paper is motivated to focus on temporal link prediction in cryptocurrency transaction networks. A graph-
regularized method is applied to link past transaction records
with future transactions. On this basis, we propose a single
latent factor-dependent, nonnegative, multiplicative, and
graph regularized-incorporated update (SLF-NMG RU)
algorithm. Subsequently, we propose a graph-regularized
nonnegative latent factor analysis (GrNLFA) model. The main
contributions of this paper are as follows:

a) A GrNLFA model, which combines the graph regularized
method and NLFA; and

b) Compared with other methods, the current method greatly
improves the computational efficiency without loss of
precision. The experimental results on a real cryptocurrency
prediction has been broadly concerned.

Section II states the preliminaries. Section III presents the
methods. Section IV conducts the empirical studies. Finally, Section V concludes this paper.

II. PRELIMINARIES

A. Problem Formulation

The graph $G$ is used to describe the cryptocurrency transaction
network, which is defined as follows:

Definition 1: Let $G=(U, S, R)_{t (t=1, ..., T)}$ denote the
cryptocurrency transaction network, where $U$ and $S$ represent
the set of sender and receiver nodes in cryptocurrency
transactions at time $t$, respectively. $R$ is the set of edges at time $t$,
and $T$ is the time span. In particular, $G=(G_1, G_2, ..., G_T)$,
where $G_t$ represents the network at time $t$ with sender set $U$,
receiver set $S$ and edge set $R$.

Definition 2: Let edge set $R_{(t) (i,j) \in S}$ be a matrix in which each
entry $r_{ij}$ represents the relationship between $i \in U$, $j \in S$
and $t \in T$. $\Lambda$ and $\Gamma$ be known and unknown entry sets of
$R$, respectively.

Note that, $K$ is the dimension of latent factor (LF) spaces.
$x$ and $y$ are defined as the LF matrices and reflect
characteristics of $U$ and $S$ in $\Lambda$, respectively.

B. Nonnegative Latent Factor Analysis Model

In order to extract nonnegative LFs from the known
datasets of static network, an NLFA model utilizes the
European distance to construct the loss function [14, 15, 18, 19]:

$$O^{NLFA}(P, Q) = \left| Z - PQ \right|^2 = \sum_{i,j \in \Lambda} \left( z_{ij} - \sum_{k=1}^{K} p_{ik} q_{jk} \right)^2$$  \hspace{1cm} (1)

subject to $\forall i \in M, j \in N, k \in \{1, 2, ..., K\}, p_{ik} \geq 0, q_{jk} \geq 0$

The NLFA model applies the SLF-NM algorithm to
guarantee the nonnegativity of the algorithm. The updated rule
is as follows:

$$O^{NLFA}(P, Q) \Rightarrow$$

$$P^{t+1} \rightarrow P^t \left( \sum_{j \in \Lambda} q_{jk} z_{ij} \right) + \sum_{j \in \Lambda} q_{jk} z_{ij}$$

$$Q^{t+1} \rightarrow Q^t \left( \sum_{i \in \Lambda} p_{ik} z_{ij} \right) + \sum_{i \in \Lambda} p_{ik} z_{ij}$$  \hspace{1cm} (2)

where $d$ denotes the number of iterations. With (2), the
original NLFA model is completed, and it can be applied to a
variety of data analysis tasks [28, 29, 32].

III. METHODS

A. Objective Function

The graph regularization method has an effective strategy
[21-24]. Its core idea is that, in a data matrix $R=[R_1, ..., R_T]$, if
the two objects $R_i$ and $R_j$ are closely in the intrinsic geometry
of the data distribution, the feature vectors $Y_i$ and $Y_j$ (the $i$th
and $j$th column of the feature matrix $Y$) should be closely also
for the new basis. Therefore, by combining the (1) with the
graph regularized method, we obtain the regularized term $Gr$
for $Y$ as follows:

$$O^{Gr}(Y) = \alpha \left\| y_i - y_j \right\|^2 w_{ij}$$  \hspace{1cm} (3)

where $w_{ij}$ is the weight matrix that measures the compactness
between two points $y_i$ and $y_j$; $\alpha$ is the regularization
parameter that controls the smoothness of the new
representation. However, cryptocurrencies have multiple
networks, which is ignored in (3).

Based on previous studies [20, 21], it is hypnotized that
the graph regularization from different past transaction
networks is located in the convex hull of a previously given
candidate manifold. This hypothesis implies that the search
space for the potential graph Laplace operator is a linear
combination of $T-1$ regularization terms. Therefore, we
control the relative importance of $\Lambda$ by the parameter $\theta$,
and we obtain the following regularization terms:

$$O^{Gr}(Y, \theta) = \frac{1}{T-1} \sum_{i=1}^{T-1} \left\| y_i - y_{i+1} \right\|^2 w_{ij}$$  \hspace{1cm} (4)

According to (1) and (4), we complete the final objective
function of dynamic network as follows:

$$O^{NLFA}(X, Y) = O^{NLFA}(X, Y) + O^{Gr}(Y, \theta)$$

$$= \sum_{i,j \in \Lambda} \sum_{t=1}^{T-1} \left( r_{ij} - \sum_{k=1}^{K} x_{ik} y_{jk} \right)^2 + \alpha \sum_{t=1}^{T-1} \sum_{i,j \in \Lambda} \left( y_{ij} - y_{i+1,j} \right)^2 w_{ij}$$  \hspace{1cm} (5)

subject to $\forall i \in M, j, l \in N, k \in \{1, 2, ..., K\}, t \in \{1, 2, ..., T\}:

$$x_{ik} \geq 0, y_{jk} \geq 0$$

where $r_{ij}$, $x_{ik}$, and $y_{jk}$ denote the $r_{ij}$, $x_{ik}$ and $y_{jk}$ at time $t$ in
cryptocurrency transaction network, respectively. Based on (5), we achieve the objective function for a GrNLFA.

B. SLF-NMG RU Algorithm

To guarantee the nonnegativity of matrices $X$ and $Y$ in (5),
we use the Lagrange method. To constrain $x_{ik} \geq 0$ and $y_{jk} \geq 0$,
we construct Lagrange multiplier $\psi_{ji \ell}$ and $\phi_{ji}$, the Lagrange $\mathcal{L}$ of (5) as follows:

$$L^{GrNLFA} = O^{GrNLFA} + Tr(\Psi X') + Tr(\Phi Y')$$

$$= O^{GrNLFA} + \sum_{i=1}^{\ell} \sum_{j \in \Lambda(j)} \sum_{k \in \Lambda(j)} \psi_{ji \ell} X_{ji} + \sum_{i=1}^{\ell} \sum_{j \in \Lambda(j)} \sum_{k \in \Lambda(j)} \phi_{ji} Y_{ji}$$

(6)

The partial derivative of $\mathcal{L}$ with respect to $X$ and $Y$ is as follows:

$$\frac{\partial L^{GrNLFA}}{\partial X} = \sum_{i=1}^{\ell} \sum_{j \in \Lambda(j)} \left( -y_{ji} \hat{r}_{ji} + \sum_{k \in \Lambda(j)} \psi_{ji \ell} X_{ji} \right)$$

$$\frac{\partial L^{GrNLFA}}{\partial Y} = \sum_{i=1}^{\ell} \sum_{j \in \Lambda(j)} \left( \alpha X_{ji} \hat{y}_{ji} \right)$$

(7)

According to the Karush-Kuhn-Tucker conditions $\psi_{ji \ell} x_{ji} = 0$ and $\phi_{ji} y_{ji} = 0$, the equations for $X$ and $Y$ are obtained as follows:

$$\sum_{i=1}^{\ell} x_{ji} \sum_{j \in \Lambda(j)} \left( y_{ji} \hat{r}_{ji} - y_{ji} r_{ji} \right) + \sum_{i=1}^{\ell} \sum_{j \in \Lambda(j)} \psi_{ji \ell} x_{ji} = 0$$

$$\sum_{i=1}^{\ell} \sum_{j \in \Lambda(j)} \left( \hat{r}_{ji} x_{ji} - r_{ji} x_{ji} \right) + \sum_{i=1}^{\ell} \sum_{j \in \Lambda(j)} \phi_{ji} y_{ji} = 0$$

(8)

According to (8), we get the final update rule as follows:

$$O^{GrNLFA}(X, Y) \xrightarrow{SLF--NMGRU} \left\{ \begin{array}{l}
x_{ji}^{t+1} = \sum_{i=1}^{\ell} \sum_{j \in \Lambda(j)} \left( \sum_{i=1}^{\ell} \sum_{j \in \Lambda(j)} y_{ji} \hat{r}_{ji} \right) \\
y_{ji}^{t+1} = \sum_{i=1}^{\ell} \sum_{j \in \Lambda(j)} \left( \sum_{i=1}^{\ell} \sum_{j \in \Lambda(j)} r_{ji} x_{ji} + \alpha \sum_{i=1}^{\ell} \sum_{j \in \Lambda(j)} |\Lambda(j)| \hat{y}_{ji} y_{ji} \right)
\end{array} \right.$$ (9)

According to (9), we achieve the SLF-NMGRU algorithm for the GrNLFA model. Subsequently, we predict that the temporal link as $r_{ji}^{t+1} x_{ji}^{t+1} y_{ji}^{t+1}$.

C. Algorithm Design and Analysis

Algorithm GrNLFA

Input: $G$: cryptocurrency transaction networks $A$: the dimension of LF spaces $\alpha$: graph regularization parameter $\theta$: time slice parameter

Operation | Cost
---|---
Initialize $X^{GrNLFA}, Y^{GrNLFA}$ | $\Theta(|\Lambda| + |S|) \times d$ \\[\text{initialize} \]Initialize $\ell$, Max-training-round=N | $\Theta(1)$ \\[\text{initialize} \]
while not converge and $\ell \neq N$ | $\times n$ \\[\text{while} \]
for each $r_{ji}^{t+1}$ in $\Lambda^{t+1}$ | $\times |\Lambda|$ \\[\text{for} \]
Update $x_{ji}^{t+1}$ and $y_{ji}^{t+1}$ with (9) | $\Theta(d)$ \\[\text{Update} \]
end for | $\Theta(1)$ \\[\text{end} \]
$n = n+1$ | $\Theta(1)$ \\[\text{end while} \]
\[\text{Predict the temporal link as } r_{ji}^{t+1} x_{ji}^{t+1} y_{ji}^{t+1} \]

Output: $X, Y$

On the basis of Section III (A) and (B), we design the algorithm of the GrNLFA model, as shown in the algorithm GrNLFA. Note that the algorithm GrNLFA summarizes the time cost of each step. According to these summaries, the total time cost is:

$$T_{GrNLFA} = \Theta(n \times |\Lambda| \times d)$$ (10)

where the condition of (10) is that $|\Lambda| \geq \max \{|U|, |S|\}$, used in a variety of applications to drop lower-order-complexity terms and constants.

The storage cost of GrNLFA relies on $\Lambda$, $U$, $S$, and $d$-related auxiliary arrays, yielding:

$$S_{GrNLFA} = \Theta(|\Lambda| + |U| + |S|) \times d.$$ (11)

which is linear with the involved entity count.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

A. General Setting

Evaluation Metrics. In this study, the root mean squared error (RMSE) and mean absolute error (MAE) are picked up as an evaluation metric [18, 19, 30-35, 42-44]:

$$RMSE = \sqrt{\frac{\sum_{k=1}^{K} |u,v| a_{u,v,k} (r_{u,v,k} - \hat{r}_{u,v,k})}{K}},$$

$$MAE = \frac{\sum_{k=1}^{K} |u,v| a_{u,v,k} |r_{u,v,k} - \hat{r}_{u,v,k}|}{K};$$

where $K$ is the validation set disjoint with the training set $\Lambda$ and the testing set $\Gamma$, and $\hat{r}_{u,v}$ denotes the estimation generated by a tested model for $\forall u,v \in K$, respectively.

Note that all experiments are carried out on the same PC with a 3.2 GHz i5 CPU and 8 GB RAM. All tested models are implemented in JAVA SE 7U60.

Datasets. As shown in Table I, two real cryptocurrency transaction networks were used in our experiment. For temporal link prediction, we divide 1 to T-2 as a training set, T-1 as the verification set, and the last time slice as the test set.

| Dataset Details | Dataset |
|------------------|---------|
| **Dataset** | | |
| Ethereum transaction network [25] (D1) | 8,626 | 2,904 | 1,014 | 203,921 |
| Bitcoin transaction network [5] (D2) | 6,351 | 7,784 | 237 | 50,178 |

Model Settings. Table II summarizes the details of compared models. In addition, the learning process of a tested model terminates if i) it converges, i.e., its error difference between two consecutive epochs becomes smaller than $10^{-5}$, or ii) its iteration count reaches a given threshold, i.e., 1000.

In order to obtain an objective result, our experiment follows these rules: a) For the M1 and the M2 model, we make the regularization coefficient $\alpha = 0.1$; b) We set the dimension of LF space uniformly, i.e., $k = 20$ uniformly, and c) For the M2 model dependent on time slices $\theta$, we make $\theta = 0.2$ [21].
TABLE II. COMPARED MODELS

| Model       | Description                                                                 |
|-------------|-----------------------------------------------------------------------------|
| NMF (M1)    | The classical NMF model approximates the original matrix by decomposing the product of two nonnegative matrices [36]. |
| GrNMF (M2)  | A GrNMF model utilizes graph regularization to regularize the historical network [21]. |
| GrNLFA (M3) | A GrNLFA model relying on SLF-NMGRU proposed in Section III (B). |

B. Parameter Sensitivity Tests

As discussed in Section III, the GrNLFA depends on the time slice parameter $\theta$. The effect of $\theta$ on GrNLFA data at D1 and D2 has been presented in Fig. 1. The following findings are achieved:

a) For SLF-NMGRU as a learning algorithm, both RMSE and MAE of M3 are sensitive to $\theta$. For instance, as shown in Fig. 1(a), RMSE of M3 with $\theta=2^{-1}$, $2^{-2}$, $2^{-3}$, $2^{-4}$, and $2^{-5}$ is 0.6452, 0.6408, 0.6355, 0.6457, 0.6481, respectively. RMAE of $\theta=2^{-3}$ increases 3.64%, 0.83%, 1.57% and 1.94%, respectively. Similar findings are found in the remaining data, as shown in Fig. 1(b)-(d).

b) Small $\theta$ affects the prediction accuracy of M3. For example, as depicted in Fig. 1(c), compared with $\theta=2^{-5}$ and $\theta=2^{-2}$, the prediction precision lost 6.03%. Similar findings are found in the remaining data, as shown in Fig. 1(a), (b), and (d).

C. Comparison Results

In this part, we aim to clarify their performance by comparing the proposed M3 model with other models. Figs. 2 and 3 depict the training process from M1 to M3. Table III summarizes the lowest RMAE and MAE, the round of iterations, and the total training time. From these results, we acquire the listed below findings:

a) The GrNLFA model has a higher prediction accuracy. As shown in Fig. 2(a), the RMSE of M1-M3 is 0.3003, 0.2894, and 0.2608, respectively. The M3 RMSE is 3.97% and 3.26% higher than M1’s RMSE and M2’RMSE, respectively. The results are similar for other datasets, as shown in figs. 2 (b) and 3.

b) The GrNLFA model greatly improves computational efficiency. Due to M3 only depending on the known value of the historical transaction data, it avoids the full matrix operation and greatly improves the storage efficiency and the computational efficiency of the model. Thus, as shown in Table III, on D2, the time consumption for M1 to M3 is increased by 61.49% and 94.37% for 19,923s, 136,389s, and 7,672s, and M2, respectively. Similarly, on D1, the M3 model’s 211,610 s is 80.66% faster than the M2 model’s 1,093,892s.

V. CONCLUSIONS

In this paper, the SLF-NMGRU algorithm and GrNLFA model have been proposed. Experimental results in real cryptocurrency transaction networks show the superiority of the proposed model. In addition, taking into account future research, the following directions deserve further discussion:

a) The tensor method can be used to express the temporal characteristics of networks [12, 17, 20, 27, 37, 38].

b) The performance of the model can be further optimized, and effective optimization methods such as the second-order method [26, 40, 41] and momentum method [28, 39] can be studied.

REFERENCES

[1] Y. Liu, F. R. Yu, X. Li, H. Ji and V. C. M. Leung, “Blockchain and Machine Learning for Communications and Networking Systems;”
IEEE Communications Surveys Tutorials, vol. 22, no. 2, pp. 1392-1431, 2020.

[2] Y. Lu, “The blockchain: State-of-the-art and research challenges,” Journal of Industrial Information Integration, vol. 15, pp. 80–90, Sep. 2019.

[3] M. H. u. Rehman, K. Salah, E. Damiani and D. Svetinovic, “Trust in Blockchain Cryptocurrency Ecosystem,” IEEE Trans. on Engineering Management, vol. 67, no. 4, pp. 1196-1212, Nov. 2020.

[4] E. Livieris, N. Kiriakidou, S. Stavroyiannis, and P. Pintelas, “An advanced CNN-LSTM model for cryptocurrency forecasting,” Electronics, vol. 10, no. 3, p. 287, Jan. 2021.

[5] W. Chen, J. Wu, Z. Zheng, C. Chen, and Y. Zhou, “Market manipulation of bitcoin: Evidence from mining the Mt. Gox transaction network,” in Proc. IEEE Conf. Computer Communications (INFOCOM), pp. 964–972, Apr. 2019.

[6] Q. Yuan, B. Huang, J. Zhang, J. Wu, H. Zhang and X. Zhang, “Detecting Phishing Scams on Ethereum Based on Transaction Records,” 2020 IEEE Int. Symposium on Circuits and Systems (ISCAS), pp. 1-5, Sept. 2020.

[7] M. C. K. Khalilov and A. Levi, “A survey on anonymity and privacy in Bitcoin-like digital cash systems,” IEEE Communications Surveys Tutorials, vol. 20, no. 3, pp. 2543–2555, Mar. 2018.

[8] F. Tschorsch and B. Scheuermann, “Bitcoin and beyond: A technical survey on decentralized digital currencies,” IEEE Communications Surveys Tutorials, vol. 18, no. 3, pp. 2084–2123, Mar. 2016.

[9] M. Conti, E. S. Kumar, C. Lal, and S. Ruj, “A survey on security and privacy issues of bitcoin,” IEEE Communications Surveys Tutorials, vol. 20, no. 4, pp. 3416–3452, Apr. 2018.

[10] M. Muzammal, F. Abbasi, Q. Qu, R. Talat and J. P. Fan, “A decentralised approach for link inference in large signed graphs,” Future Generation Computer Systems, vol. 102, pp. 827-837, Jan. 2020.

[11] P. Meo, “Trust Prediction via Matrix Factorisation,” ACM Trans. on Internet Technology, vol. 19, no. 4, pp. 1-20, Nov. 2019.

[12] W. Yu, R. Mu, Y. Sun, X. Chen, W. Wang and H. Wu, “A Double Non-Negative Matrix Factorization Model for Signed Network Analysis,” in 2019 IEEE Int. Conf. on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPAB/BDCloud/SocialCom/SustainCom), pp. 936-943, Dec. 2019.

[13] J. Wang and R. J. Mu, “A Regularized Convex Nonnegative Matrix Factorization Model for signed network analysis,” Social Network Analysis and Mining, vol. 11, no. 7, Jan. 2021.

[14] X. Luo, Z. Liu, L. Jin, Y. Zhou and M. Zhou, “Symmetric Nonnegative Matrix Factorization Based Community Detection Models and Their Convergence Analysis,” IEEE Trans. on Neural Networks and Learning Systems, vol. 33, no. 3, pp. 1203-1215, March 2022, doi: 10.1109/TNNLS.2020.3041360.

[15] X. Luo, M.-C. Zhou, Y.-N. Xia, and Q.-S. Zhu, “An efficient nonnegative matrix-factorization-based approach to collaborative filtering for recommender systems,” IEEE Trans. on Industrial Informatics, vol. 10, no. 2, pp. 1273-1284, 2014.

[16] Z. Liu, X. Luo, and Z. Wang, “Convergence Analysis of Single Latent Factor-dependent, Non-negative and Multiplicative Update-based Non-negative Latent Factor Models,” IEEE Trans. on Neural Networks and Learning Systems, DOI: 10.1109/TNNLS.2020.2990090.

[17] H. Wu, X. Luo and M. Zhou, “Advancing non-negative latent factorization of tensors with diversified regularizations,” IEEE Trans. on Services Computing, DOI: 10.1109/TSC.2020.2988760.

[18] X. Luo, M. Zhou, S. Li, and M. Shang, “An inherently non-negative latent factor model for high-dimensional and sparse matrices from industrial applications,” IEEE Trans. Ind. Informat., vol. 14, no. 5, pp. 2011-2022, May 2018.

[19] X. Luo, M.-C. Zhou, S. Li, Z.-H. You, Y.-N. Xia, and Q.-S. Zhu, “A nonnegative latent factor model for large-scale sparse matrices in recommender systems via alternating direction method,” IEEE Trans. Neural Netw. Learn. Syst., vol. 27, no. 3, pp. 579–592, Mar. 2016.

[20] D. M. Dunlavy, T. G. Kolda, and E. Acar, “Temporal link prediction using matrix and tensor factorizations,” ACM Trans. on Knowledge Discovery from Data, vol. 5, no. 2, p. 27, 2011.

[21] X. Ma, P. Sun, and Y. Wang, “Graph regularized nonnegative matrix factorization for temporal link prediction in dynamic networks,” Physica A Statistical Mechanics & Its Applications, vol. 496, 2018.
[41] Z. Li, S. Li and X. Luo, “An overview of calibration technology of industrial robots,” IEEE/CAA Journal of Automatica Sinica, vol. 8, no. 1, pp. 23-36, January 2021.

[42] X. Luo, Z. Ming, Z. You, S. Li, Y. Xia, and H. Leung, “Improving network topology-based protein interactome mapping via collaborative filtering,” Knowledge-Based Systems, vol. 90, pp. 23–32, 2015.

[43] D. Wu, Y. He, X. Luo and M. Zhou, “A Latent Factor Analysis-Based Approach to Online Sparse Streaming Feature Selection,” IEEE Trans. on Systems, Man, and Cybernetics: Systems, doi: 10.1109/TSMC.2021.3096065.

[44] X. Luo, Z. Wang and M. Shang, “An Instance-Frequency-Weighted Regularization Scheme for Non-Negative Latent Factor Analysis on High-Dimensional and Sparse Data,” IEEE Trans. on Systems, Man, and Cybernetics: Systems, vol. 51, no. 6, pp. 3522-3532, Jun. 2021.