

Research Article

A Credit Conflict Detection Model Based on Decision Distance and Probability Matrix

Xiaodong Zhang,¹ Congdong Lv ,¹ and Zhoubao Sun²

¹School of Information Engineering, Nanjing Audit University, Nanjing 211815, China

²School of Engineering Audit, Nanjing Audit University, Nanjing 211815, China

Correspondence should be addressed to Congdong Lv; lvcongdonglv@163.com

Received 17 September 2021; Accepted 25 November 2021; Published 7 January 2022

Academic Editor: Chunguo Li

Copyright © 2022 Xiaodong Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Considering the credit index calculation differences, semantic differences, false data, and other problems between platforms such as Internet finance, e-commerce, and health and elderly care, which lead to the credit deviation from the trusted range of credit subjects and the lack of related information of credit subjects, in this paper, we proposed a crossplatform service credit conflict detection model based on the decision distance to support the migration and application of crossplatform credit information transmission and integration. Firstly, we give a scoring table of influencing factors. Score is the probability of the impact of this factor on credit. Through this probability, the distance matrix between influencing factors is generated. Secondly, the similarity matrix is calculated from the distance matrix. Thirdly, the support vector is calculated through the similarity matrix. Fourth, the credit vector is calculated by the support vector. Finally, the credibility is calculated by the credit vector and probability.

1. Introduction

In recent years, with the development of the Internet, online services in all walks of life came into being. With the advantages of the Internet, users can obtain the desired services through simple processes in various environments, but also, because of the virtualization of the network, fraud is easy to occur. This poses a challenge to the credit evaluation system of each platform. The requirement of accelerating the construction of social credit system is put forward in the 12th Five-Year Plan, which is more clearly explained in the 14th Five-Year Plan. Strengthen the collection, sharing, disclosure, and application of credit information; promote credit products and services that benefit the people and facilitate enterprises; establish a sharing and integration mechanism of public credit information and financial information; cultivate internationally competitive enterprise credit investigation institutions and credit rating institutions; strengthen credit investigation supervision; and promote the healthy development of the credit service market. In the environment where big data technology is widely used, in order to

meet the following challenges, each platform organization uses the data collected by the platform to calculate credit indicators and build its own credit evaluation system. However, there are many problems in this process: for example, the collected information cannot fully evaluate and describe the credit indicators, and the information is collected and entered in the process. Errors and deficiencies and the focus on credit and evaluation models are different. There will be differences in the information and evaluation results of the same object on different platforms, and there is no good coordination mechanism. The data are scattered, heterogeneous, and low-quality, which is difficult to be directly applicable to judge the overall credit level of an object. The outline of the plan for the construction of social credit system issued by the State Council (2014-2020) puts forward that “accelerating the construction of credit information system and improving the recording, integration and application of credit information are the basis and premise for the formation of trustworthy incentive and dishonest punishment mechanism.” From this point of view, to solve the data problem in credit evaluation, it is necessary for all platforms

```

Require:
  The set of credit data,  $y$ ;
Ensure:  $m(y)$ 
1: if  $x_1 > 18 \& x_1 \leq 25$  then
2:    $m_{x_1} = 2/100$ ;
3: end if
4: if  $x_1 > 26 \& x_1 \leq 34$  then
5:    $m_{x_1} = 3/100$ ;
6: end if
7: if  $x_1 > 35 \& x_1 \leq 45$  then
8:    $m_{x_1} = 5/100$ ;
9: end if
10: if  $x_1 > 46 \& x_1 \leq 55$  then
11:    $m_{x_1} = 4/100$ ;
12: end if
13: ...
14: return  $\langle m_{x_1}, m_{x_2}, \dots, m_{x_n} \rangle$ ;

```

ALGORITHM 1: Basic probability generation algorithm.

to establish a perfect information exchange mechanism, gradually form a credit service network with wide coverage and complete categories, and build an objective, fair, reasonable and balanced international credit rating system model.

The core content of building a crossplatform credit index evaluation model is to fuse multisource heterogeneous credit data, and the information conflict caused by data fusion is the focus of the research: there are attribute differences between the same information and different names between the data attributes of each platform, there are numerical differences between the same attributes of different sources, and due to the data collection methods to sum up, the purpose of conflict detection modelling is to match attributes, solve conflict problems, clean up false data, and obtain data with unified standards, reliable sources, and strong authenticity, so as to achieve high efficiency and authenticity in the construction of subsequent credit models.

2. Related Works

In the case of conflicting descriptions of the same attribute in multisource data, pattern matching technology is used to solve problems for different source attributes [1, 2]. Pattern matching technology is divided into two levels—pattern level and instance level. Pattern level method analyses the corresponding relationship between attribute descriptions of different source data, such as attribute name, abbreviation, or attribute storage type [3]. The advantage of similarity analysis with range is simple and intuitive and low time cost, but the amount of attribute description information is small, so it is difficult to use directly [4–6]. It is necessary to build a unified standard for the most effective use. At the instance level, feature extraction and similarity analysis are carried out on data from different sources to obtain the mapping relationship between different attributes. The advantage is that it can reduce the dependence on domain knowledge. However, good data with sufficient scale is needed [7]. In relevant research, pattern-based solutions

are considered from three levels: the more intuitive is the language level, and the research mainly considers the semantics of attribute description [8–10]. For example, coma system determines the relationship between various attributes by constructing attribute language association diagram, inputs attribute pairs in use, and returns a measurement between 0 and 1 to confirm the attribute similarity degree; more studies further consider the attribute constraint level and attribute structure level [11]. For example, the similarity flooding algorithm constructs the attribute description and data type in the graph according to the input information and obtains the mapping between attributes through multiple iterative fixed-point calculation [12]. However, the data description information lacks a unified standard, and it is difficult to obtain, identify, and use the data description information which is very few, which is not conducive to wide application. Now, the solution based on case analysis is mostly used to obtain the mapping relationship between attributes through feature extraction and comparative analysis of data from different sources. SMDD method uses neural network technology to find the element set with similar distribution law according to the data instance information and calculate and return the matching results with high similarity [13]. Mehdi et al. analysed the similarity of different types of data which reduces the scale of similarity matrix, and introduce Google similarity distance to the semantic relationship between character type data in the task [14]. Zhou et al. had established a pattern matching system based on Hungarian algorithm to analyse the characteristics of multisource data and obtain the mapping relationship, which has enhanced the generalization ability of the model. In the research, natural language processing technology is also more and more used for similarity analysis [15, 16]. For example, Nozaki et al. used the word2vec tool to compare and analyse the similarity of attribute semantic relationship of strings in the data set, and Northrop et al. used Bert-based semantic similarity calculation to align the knowledge base index [17, 18].

For the detection of data conflict in multisource data fusion, the abnormal points in conflict are regarded as outliers, and the point outlier detection technology is used to detect and process the conflict [19–22]. In the traditional data mining work, outlier detection is carried out by using statistics, clustering, classification, proximity, and other methods [23–29]. These methods are strong, simple, and direct but need to rely on a certain prior knowledge, and processing effects are directly affected by the level of knowledge. Some studies have also applied machine learning to this field to machine learn the attributes of conflict generation and use the result prediction to replace the conflict. The numerical conflict detection method based on outlier detection is to find data objects that are significantly different from other data distributions. The traditional statistical method is through on the premise of known data distribution, the data that does not conform to the distribution is classified as outliers, but it requires difficult-to-obtain a priori knowledge, which is not conducive to work. In practical work, more outlier detection methods based on clustering or classification are used, and common clustering

TABLE 1: Personal credit table.

Item	Score	Standard			
(1) Basic information	25	5 + 1 + 1 + 4 + 2 + 6 = 19			
(1.1) Age	5	18~25 2	26~34 3	35~45 5	46~55 4
(1.2) Sex	2	Male 1	Female 2		
(1.3) Marriage	5	Sons and daughters 5	A daughter 4	Unmarried	Others 1
(1.4) Education	5	Postgraduate 5 Others 1	Undergraduate 4	Junior college 3	Senior high school 2
(1.5) Registered	2	Permanent residence 2	Temporary account		
(1.6) House	6	Self 6	Loan 5	Rent 3	Others 1
(2) Occupation	45	4 + 8 + 4 + 6 + 4 + 9 = 35			
(2.1) Company	4	Government 4	Enterprise 3	Individual business 2	Others 1
(2.2) Industry	10	Civil servant 10 Transport 5	Education 9 Media 5	Financial 8 Industrial 4	Registered office 6 Others 2
(2.3) Working years	4	5 4	3~5 3	1~3 2	1 1
(2.4) Post nature	10	Bureau level 10	Division level 8	Section level 6	Others 2
(2.5) Technical title	10	General manager 10	Division manager 8	Staff member 5	Others 2
(2.6) Personal income	5	Senior 5	Intermediate 4	Primary 2	No 1
(2.6) Personal income	12	10K 12 3K~4K 6	8K~10K 10 2K~3K 4	5K~8K 9 1K~2K 2	4K~5K 8 1K 1
(3) Family situation	10	5 + 4 = 9			
(3.1) Family income	6	5K 6 0.5K~1K 2	3K~5K 5 0.5K 1	2K~3K 4	1K~2K 3
(3.2) Fixed charges	4	0.5K 2	0.5~1K 3	1K~3K 4	3K 2

algorithms can be better applied to outlier detection: such as DBSCAN [30, 31] and BIRCH [32, 33]. Clustering algorithms need the authenticity of data sources. Jia et al. used the clustering-based outlier detection algorithm to clean the wrong and missing data in the medical database, showing better performance than the distance-based algorithm. Another idea is the outlier test based on proximity, by calcu-

lating the distance or density of data points that determine outliers [34]. Riahi-Madvar et al. use lof (local outlier factor) as a measure; it represents the local outlier degree of the object, determines the k -distance neighbourhood to calculate the local reachability density of the object, and obtains the outlier degree of data points [35]. The algorithm using lof for measurement also has many applications [36, 37]. Based

Require:
The set of basic probability, m_y ;

Ensure: $D(y)$

- 1: **for** each item $i \in m_y$ **do**
- 2: **for** each item $j \in m_y$ **do**
- 3: $d_{ij} = 2 * \text{abs}((m_y[i] - 1/2) * (m_y[j] - m_s[i]));$
- 4: **end for**
- 5: **end for**
- 6: **return** $D(y)$

ALGORITHM 2: Distance matrix calculation algorithm.

Require:
The distance matrix, $D(y)$;

Ensure: $S(y)$

- 1: **for** each item $i \in D_y$ **do**
- 2: **for** each item $j \in D_y$ **do**
- 3: $s_{ij} = 1 - d_{ij};$
- 4: **end for**
- 5: **end for**
- 6: **return** $S(y)$

ALGORITHM 3: Similarity matrix calculation algorithm.

Require:
The similarity matrix, $S(y)$;

Ensure: $Sup(y)$

- 1: **for** each item $i \in S_y$ **do**
- 2: **for** each item $j \in S_y$ **do**
- 3: $Sup[i] = Sup[i] + s_{ij};$
- 4: **end for**
- 5: **end for**
- 6: **return** $Sup(y)$

ALGORITHM 4: Support vector calculation algorithm.

Require:
The support vector, $Sup(y)$;

Ensure: $Crd(y)$

- 1: **for** each item $i \in Sup_y$ **do**
- 2: $SumSup = SumSup + Sup[i]$
- 3: **end for**
- 4: **for** each item $i \in Sup_y$ **do**
- 5: $Crd[i] = Sup[i]/SumSup;$
- 6: **end for**
- 7: **return** $Crd(y)$

ALGORITHM 5: Credibility vector calculation algorithm.

on the work, Liu et al., based on lof mining outliers, a local outlier degree measurement method is constructed, which reduces the complexity of the algorithm and avoids misjud-

Require:
The support vector, $Crd(y)$;
The set of basic probability, m_y ;

Ensure: $Crd(y)$

- 1: **for** each item $i \in Sup_y$ **do**
- 2: $m_{avg}(y) = m_{avg}(y) + Crd[i] * m(x_i)$
- 3: **end for**
- 4: **return** $m_{avg}(y)$;

ALGORITHM 6: Average trust value calculation algorithm.

gement to a certain extent depending on the local characteristics [38, 39]. Because of not considering the characteristics of the overall data distribution, it is very suitable for anomaly detection of data with different density distributions [40].

2.1. The Credit Conflict Detection Model. In order to effectively apply decision distance measurement and probability distance matrix to deal with multisource conflict evidence, it is necessary to convert multisource conflict evidence into decision distance measurement value and probability distance matrix. On this basis, the similarity matrix, support vector, reliability, and average trust function are obtained. The treatment process is as follows.

Each record of credit data is represented by a company attribute or personal information attribute, such as age. We use a multisource group to represent

$$Y = \langle X_1, X_2, \dots, X_n \rangle. \quad (1)$$

For any record $y = \langle x_1, x_2, \dots, x_n \rangle$, we need to calculate its basic probability:

$$m(y) = \langle m(x_1), m(x_2), \dots, m(x_n) \rangle. \quad (2)$$

The algorithm is shown in Algorithm 1, and the judgment conditions are based on Table 1.

After obtaining the basic probability vector, we need to calculate the distance matrix:

$$D_y = \begin{pmatrix} d_{11} & \cdots & d_{1n} \\ \vdots & \ddots & \vdots \\ d_{n1} & \cdots & d_{nn} \end{pmatrix}. \quad (3)$$

d_{ij} is calculated as follows:

$$d_{ij} = 2 \left| m_i - \frac{1}{2} \right| (m_j - m_i). \quad (4)$$

The algorithm is shown in Algorithm 2.

Require:
The average trust value, $m_{avg}(y)$;

Ensure: *Conflict*

1: **if** $m_{avg}(y) > \Theta$ **then**

2: return *True*;

3: **end if**

4: **return** *False*;

ALGORITHM 7: Credit conflict detection algorithm.

Similarity matrix is as follows:

$$S_y = \begin{pmatrix} 1 & \cdots & s_{1n} \\ \vdots & \ddots & \vdots \\ s_{n1} & \cdots & 1 \end{pmatrix}, \quad (5)$$

$$s_{ij} = 1 - d_{ij} (0 < s_{ij} < 1).$$

The algorithm is shown in Algorithm 3.

The column cells of the similarity matrix S_s are summed to obtain the support vector $\text{Sup}_j(y)$. The calculation formula is as follows:

$$\text{Sup}_j(y) = \sum_{i=1}^n s_{ij}, \quad i = 1, 2, 3, \dots, n. \quad (6)$$

The algorithm is shown in Algorithm 4.

The credibility vector $\text{Crd}_j(y)$ can be obtained by regularization of support vector $\text{Sup}_j(y)$. The calculation formula is as follows:

$$\text{Crd}_j(y) = \frac{\text{Sup}_j(y)}{\sum_{j=1}^n \text{Sup}_j(y)}. \quad (7)$$

The algorithm is shown in Algorithm 5.

The average trust value of n evidences is obtained. The calculation formula is as follows:

$$\overline{m}(y) = \sum_{i=1}^n \text{Crd}_i(y) * m(x_i). \quad (8)$$

The algorithm is shown in Algorithm 6.

Finally, we have our credit conflict detection model. Given a threshold θ , if $\overline{m}(y) \geq \theta$, the record has no conflict. Otherwise, the record has conflict. The algorithm is shown in Algorithm 7.

3. Conclusions

In today's life, a large number of new data are generated every day. In the direction of credit research, the credit evaluation of each credit subject also changes dynamically with the new data. In the future research, conflict detection and processing of dynamic data and improving the timeliness

and accuracy of model operation will become the key issues of conflict model construction. Because of the diversity of detection methods, no general model can be applied to all data, so the expansion of the use of new technologies will also be one of the focuses of conflict detection.

In this paper, we analyse cluster, outlier detection, machine learning, and other technologies and design a crossplatform, multilevel, multidimensional, and multigranularity service credit conflict detection model. The model is used to deal with the semantic differences, calculation differences, false data, and other problems of credit indicators among platforms such as Internet finance, e-commerce, and health pension. It is expected to effectively detect and deal with credit information conflicts such as credit deviation and lack of credit subject information.

In the future, on the premise of certain identification samples, artificial intelligence can play a better role. This is a very worthy research content.

Data Availability

The data cannot be fully disclosed for the time being because it contains private data. The data structure and its implementation code have been uploaded to GitHub; please visit <https://github.com/juckylv/Credit-data>.

Conflicts of Interest

There is no conflict of interest regarding the publication of this paper.

Acknowledgments

This study was supported by the National Key R&D Program (Grant No. 2019YFB1404602) and Natural Science Research Project of Jiangsu Province Universities and Colleges (No. 21KJB520022).

References

- [1] M. Lee, "Plastic pollution mitigation - net plastic circularity through a standardized credit system in Asia," *Ocean & Coastal Management*, vol. 210, no. 1, article 105733, 2021.
- [2] R. Iqbal, F. Doctor, B. More, S. Mahmud, and U. Yousuf, "Big data analytics and computational intelligence for cyber-physical systems: recent trends and state of the art applications," *Future Generation Computer Systems*, vol. 105, pp. 766–778, 2020.
- [3] J. Trávníček, J. Janoušek, B. Melichar, and L. Cleophas, "On modification of Boyer-Moore-Horspool's algorithm for tree pattern matching in linearised trees," *Theoretical Computer Science*, vol. 830–831, pp. 60–90, 2020.
- [4] S. Cruz and A. Aguiar, "MagLand: magnetic landmarks for road vehicle localization," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 4, pp. 3654–3667, 2020.
- [5] G. Ding, S. Sun, and G. Wang, "Schema matching based on SQL statements," *Distributed and Parallel Databases*, vol. 38, no. 1, pp. 193–226, 2020.
- [6] M. C. Freund, J. A. Etzel, and T. S. Braver, "Neural coding of cognitive control: the representational similarity analysis

- approach[J],” *Trends in Cognitive Sciences*, vol. 25, no. 7, pp. 115–136, 2021.
- [7] X. Zhang, R. Li, B. Zhang, Y. Yang, J. Guo, and X. Ji, “An instance-based learning recommendation algorithm of imbalance handling methods,” *Applied Mathematics and Computation*, vol. 351, pp. 204–218, 2019.
 - [8] S. Song, G. Gu, C. Ryu, S. Faro, T. Lecroq, and K. Park, “Fast algorithms for single and multiple pattern Cartesian tree matching,” *Theoretical Computer Science*, vol. 849, pp. 47–63, 2021.
 - [9] S. Munir, F. Khan, and M. A. Riaz, “An instance-based schema matching between opaque database schemas,” in *Proceedings of the 4th International Conference on Engineering Technology and Technopreneurship*, pp. 177–182, Kuala Lumpur, 2014.
 - [10] H. Zhao and S. Ram, “Combining schema and instance information for integrating heterogeneous data sources,” *Data & Knowledge Engineering*, vol. 61, no. 2, pp. 281–303, 2007.
 - [11] P. A. Bernstein, J. Madhavan, and E. Rahm, “Generic schema matching, ten years later,” in *Proceedings of the 37th International Conference on Very Large Data Bases*, pp. 695–701, Seattle, 2011.
 - [12] H. Yang, L. Shen, X. Dong, Q. Ding, P. An, and G. Jiang, “Low complexity CTU partition structure decision and fast intra mode decision for versatile video coding,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 4, pp. 1–18, 2019.
 - [13] M. Shrestha, T. X. Tran, B. Bhattarai, M. L. Pusey, and R. S. Aygun, “Schema matching and data integration with consistent naming on protein crystallization screens,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 15, no. 39, pp. 1–1, 2019.
 - [14] Z. Gao, Y. Wu, M. Harandi, and Y. Jia, “A robust distance measure for similarity-based classification on the SPD manifold,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 16, no. 29, pp. 1–15, 2019.
 - [15] T. Zhou, M. Chen, and J. Zou, “Reinforcement learning based data fusion method for multi-sensors,” *IEEE/CAA Journal of Automatica Sinica*, vol. 6, pp. 128–149, 2020.
 - [16] F. Y. Rao, J. Cao, E. Bertino, and M. Kantarcioglu, “Hybrid private record linkage,” *ACM Transaction on Information and System Security*, vol. 22, no. 3, pp. 1–36, 2019.
 - [17] K. Nozaki, T. Hochin, and H. Nomiya, “Semantic schema matching for string attribute with word vectors and its evaluation,” *International Journal of Networked and Distributed Computing*, vol. 7, pp. 100–106, 2019.
 - [18] J. A. Northrop and A. Papandreou-Suppappola, “Computationally efficient estimation of compound K-distribution sea clutter in thermal noise and its application to sea echo reflectivity observations,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 56, no. 3, pp. 2340–2350, 2020.
 - [19] J. Li, Z. Zhang, X. Li, and H. Chen, “Kernel-based learning for biomedical relation extraction,” *Journal of the American Society for Information Science and Technology*, vol. 59, no. 5, pp. 756–769, 2008.
 - [20] W. Liao, B. Zeng, X. Yin, and P. Wei, “An improved aspect-category sentiment analysis model for text sentiment analysis based on RoBERTa,” *Applied Intelligence*, vol. 4, pp. 1–12, 2020.
 - [21] S. Yang and C. Tan, “Detection of conflicts between resource authorization rules in extensible access control markup language based on dynamic description logic,” *Ingénierie des Systèmes d’Information*, vol. 25, no. 3, pp. 178–201, 2020.
 - [22] L. Li, F. Zhu, H. Sun, Y. Hu, Y. Yang, and D. Jin, “Multi-source information fusion and deep-learning-based characteristics measurement for exploring the effects of peer engagement on stock price synchronicity,” *Information Fusion*, vol. 69, no. 3, pp. 1–21, 2021.
 - [23] L. Ying, L. Junting, and G. Fuxiang, “Adaptive conflict detection algorithm based on Rochester Software Transactional Memory,” *Journal of Physics: Conference Series*, vol. 1746, no. 1, p. 012050, 2021.
 - [24] Y. Jin, W. Cao, M. Wu, and Y. Yuan, “Simplified outlier detection for improving the robustness of a fuzzy model,” *Science China*, vol. 63, no. 4, 2020.
 - [25] X. Qin, J. Li, W. Hu, and J. Yang, “Machine learning K-means clustering algorithm for interpolative separable density fitting to accelerate hybrid functional calculations with numerical atomic orbitals,” *The Journal of Physical Chemistry A*, vol. 124, no. 48, pp. 10066–10074, 2020.
 - [26] H. H. Pajouh, R. Javidan, R. Khayami, A. Dehghantanha, and K. K. R. Choo, “A two-layer dimension reduction and two-tier classification model for anomaly-based intrusion detection in IoT backbone networks,” *IEEE Transactions on Emerging Topics in Computing*, vol. 7, no. 2, pp. 314–323, 2019.
 - [27] L. Zhongbao and Z. Wenjuan, “Study on stellar spectral outlier mining based on fuzzy large margin and minimum ball classification model,” *Spectroscopy and Spectral Analysis*, vol. 36, no. 4, pp. 1245–1248, 2016.
 - [28] S. Cai, R. Sun, S. Hao, S. Li, and G. Yuan, “An efficient outlier detection approach on weighted data stream based on minimal rare pattern mining,” *China Communications*, vol. 16, no. 10, pp. 83–99, 2019.
 - [29] Q. Pu, K. Y. Ng, M. Zhou, and J. Wang, “A joint rogue access point localization and outlier detection scheme leveraging sparse recovery technique,” *IEEE Transactions on Vehicular Technology*, vol. 70, no. 2, pp. 1866–1877, 2021.
 - [30] L. Zhang, Y. Xie, X. Luan, and X. Zhang, “Multi-source heterogeneous data fusion,” in *2018 International Conference on Artificial Intelligence and Big Data (ICAIBD)*, IEEE, 2018.
 - [31] J. Li, I. Tobore, Y. Liu, A. Kandwal, L. Wang, and Z. Nie, “Non-invasive monitoring of three glucose ranges based on ECG by using DBSCAN-CNN,” *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 9, pp. 3340–3350, 2021.
 - [32] S. Lee, J. Jung, I. Park, K. Park, and D. S. Kim, “A deep learning and similarity-based hierarchical clustering approach for pathological stage prediction of papillary renal cell carcinoma,” *Computational and Structural Biotechnology Journal*, vol. 18, no. 2, pp. 2639–2646, 2020.
 - [33] L. Yufeng, J. H. Tian, L. Jiyong, L. Xiaozhong, S. Zhiwei, and L. Min, “MR-BIRCH: a scalable MapReduce-based BIRCH clustering algorithm,” *Journal of Intelligent & Fuzzy Systems*, vol. 40, no. 3, pp. 1432–1451, 2021.
 - [34] P. Jia, X. Wang, and K. Zheng, “Distributed clock synchronization based on intelligent clustering in local area industrial IoT systems,” *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 3697–3707, 2020.
 - [35] M. Riahi-Madvar, A. Akbari Azirani, B. NaserSharif, and B. Raahemi, “A new density-based subspace selection method using mutual information for high dimensional outlier detection,” *Knowledge-Based Systems*, vol. 216, no. 2, pp. 106733–106733, 2021.
 - [36] O. Alghushairy, R. Alsini, X. Ma, and T. Soule, “A genetic-based incremental local outlier factor algorithm for efficient

- data stream processing,” in *International Conference On Compute And Data Analysis (Iccda 2020)*, Silicon Valley, CA, USA, 2020.
- [37] M. Yang and D. Ergu, “Anomaly detection of vehicle data based on LOF algorithm,” *Frontiers in Signal Processing*, vol. 4, no. 1, pp. 678–694, 2020.
- [38] F. Liu, Y. Yu, P. Song, Y. Fan, and X. Tong, “Scalable KDE-based top-n local outlier detection over large-scale data streams,” *Knowledge-Based Systems*, vol. 204, no. 9, pp. 106186–106186, 2020.
- [39] J. Jiang, Q. Ma, X. Jiang, and J. Ma, “Ranking list preservation for feature matching,” *Pattern Recognition*, vol. 111, no. 8, pp. 107665–107665, 2021.
- [40] L. Chen, W. Wang, and Y. Yang, “CELOF: effective and fast memory efficient local outlier detection in high-dimensional data streams,” *Applied Soft Computing*, vol. 102, no. 12, pp. 107–129, 2021.