AD-DCFP: Anomaly Detection Based on the Distance of Closed Frequent Patterns

Yudong Yin
School of Software, Shanxi Agricultural University, China
yudong@sxau.edu.cn

Kun Wang
School of Software, Shanxi Agricultural University, China
wangkun@sxau.edu.cn

Linqiang Deng
School of Software, Shanxi Agricultural University, China
sxaudlq@sxau.edu.cn

Abstract: *Frequent Pattern-based (FP) anomaly detection methods can accurately detect the potential anomalies since they fully consider the appearing frequency as well as the deviating degree of each data sample, which is coincide with the definition of anomalies. Because the Closed Frequent Patterns (CFPs) are the subsets of FPs and its scale is much less, thus, CFP-based Anomaly Detection (AD) methods are more efficient in time. However, the small scale of patterns used in the AD process led to low detection efficiency. That is, the time efficiency and detection accuracy of FP-based anomaly detection are two contradictory individuals. Aimed at this problem, this paper introduces an AD method based on the distance of CFPs, namely Anomaly Detection Based on the Distance of Closed Frequent Patterns (AD-DCFP). AD-DCFP uses the distance of CFPs (the discrepancy between CFPs and data samples) to eliminate the negative impact of patterns with small scale used in the AD, thereby quickly and accurately detecting anomalies. Specifically, the vertical-based mining manner and bit-vector structure are used to mine CFPs for improving mining efficiency; and then, the concept of pattern distance is introduced in the AD phase to calculate the abnormal degree of each data sample; Finally, the data samples with top-k ranked abnormal degree are judged as anomalies. Massive experiments on six datasets show that compared with five state-of-the-arts, the proposed AD-DCFP method can improve the average detection accuracy by about 5% and reduce the time consumption by about 10%, it is a better choice for large-scale or high-dimensional datasets*.

Keywords: *Anomaly detection, closed frequent patterns, pattern distance, vertical-based mining*.

Received May 25, 2024; accepted November 14, 2024
https://doi.org/10.34028/iajit/22/2/6

1. Introduction

With the widespread application of machine learning and deep learning technologies, data has become the most important resource in daily life. Therefore, how to guarantee the quality of collected data is an important issue faced by the industry and academia. As an important method to ensure data security, Anomaly Detection (AD) [13, 14, 20], (AD, aka, outlier detection [4]) is extensively researched in recent years, and it is widely used in credit card fraud detection [8, 18], intrusion detection [3, 12], and other fields. AD aims at seeking for the data samples that have low appearing frequency and differ from most data samples in the datasets. According to the use of different technologies, AD is roughly divided into: distribution-based methods [9], model-based methods [15, 19], learning-based methods [16, 21], clustering-based methods [11, 25], distance-based methods [1, 2], density-based methods [22, 24] and pattern-based methods [5, 6].

Among numerous AD methods [11, 17, 24], pattern-based methods [4, 7, 10] have high detection accuracy since they fully consider the appearing frequency of each data sample as well as the difference between each data sample in the datasets. However, pattern-based AD methods are seriously dependent on the mined patterns (aka, itemsets or features, that is, each feature in the data is a pattern), while the different settings of Minimal Support threshold (abbreviated as *min_sup*) result in mining different patterns, thereby making the detection of anomalies appears large differences. For the pattern-based AD methods, the use of Frequent Patterns (FPs) or Rare Patterns (RPs) [10] would lead to a very long-time consumption due to the extensive patterns could be mined.

To solve this difficulty, the supersets of FPs (such as the Closed Frequent Patterns (CFPs), the Maximal Frequent Patterns (MFPs)) and the subsets of RPs (such as the Minimal Rare Patterns (MRPs)) are often used to reduce the pattern scale, therefore, the CFP-based [5], MFP-based [7], MRP-based [4] AD methods are proposed successively to detect the anomalies. Compared with MFPs and MRPs, CFPs contain the complete information of FPs (while MFPs lose the *support* information belonging to their subsets and MRPs lose the *support* information belonging to their supersets), thus, CFP-based AD methods have obtained more attention. The CFP-based AD methods first mine the CFPs with strong correlation in the data, and then design several deviation indices to measure the abnormal degree of each piece of data to detect anomalies. Essentially, CFP-based AD methods conduct AD operations based on the differences between the contained CFPs and data samples.

Although CFP-based AD methods have competitive detection accuracy, they also face some defects that need to be solved urgently.

1. The number of CFPs that can be mined is very few under the large *min_sup* threshold, which makes their detection efficiency is not so competitive due to less CFPs can be included into the detection process.
2. Although the set of small *min_sup* threshold can solve the problem of low detection accuracy under large *min_sup* threshold, but the time usage on pattern mining process will be very heavy due to more patterns would be determined as FPs and then participate in the pattern mining process, which leads to explosive time usage when the *min_sup* threshold is set very small.

To solve the limitations of CFP-based AD methods mentioned above, with the consideration that the small number of contained CFPs indicates the large difference between the patterns contained in the data and thus leading to the data being more abnormal, this paper proposes an AD method based on the distance of CFPs, namely Detection Based on the Distance of Closed Frequent Patterns (AD-DCFP), to detect the anomalies. Firstly, the *support* value of each 1-pattern in the transaction (composed of data samples) is computed to minimize the impact of infrequent 1-patterns on the mining of CFPs, and the frequent 1-patterns are transformed into the form of vertical representations to compute the *support* value of extended patterns. Then, the *support* value of each extended pattern is calculated using "AND" operation to mine the CFPs. Next, based on the mined CFPs, the deviation index of CFP and pattern distance are designed to measure the abnormal degree of the transactions, and the top-*k* transactions with highest abnormal degree are recognized as anomalies.

The main contributions of this paper can be summarized as follows:

1. The idea of pattern distance (that is, the distance of CFPs) is introduced into AD to calculate the abnormal degree of each transaction, thus solving the problem of a significant decline in the detection accuracy caused by too few CFPs can be mined in the data samples.
2. Based on the pattern distance, a novel anomaly detection method called AD-DCFP is proposed to seek for the anomalies. Although the pattern distance is also a metric used to calculate the abnormal degree of each data instance, but it fully considers the different parts between the mined CFPs and transactions rather than only determining whether any mined CFP contained in the transactions, which can effectively solve the problem of pattern-based AD methods that have low efficiency under large preset *min_sup* threshold.
3. Massive experiments are carried out on one synthetic dataset and five public datasets to test the efficiency of the AD-DCFP method with the comparison of five state-of-the-arts, and the results verify that AD-DCFP can effectively detect potential anomalies with high detection accuracy as well as short time overhead, and it also has better scalability.

The rest can be summarized as follows. In section 2, we first give the backgrounds and definitions, and then review some related works on pattern-based AD methods. In section 3, we first introduce the CFP mining method called CFPM, and then provide the details of the AD method based on the distance of CFP (namely AD-DCFP). In section 4, we carry out massive experiments to test the proposed AD-DCFP method on three views, including detection efficiency, time efficiency and scalability. Finally, we conclude the major works of this paper in section 5.

2. Backgrounds and Related Works

2.1. Backgrounds

In the datasets, the transaction is one piece of data sample, it is abbreviated as *TID*. The transaction consists of some 1-patterns (aka items), and each pattern represents the feature of the monitored data samples. For two different patterns $\{p_i\}$ and $\{p_j\}$, if some items in $\{p_i\}$ have not been appeared in $\{p_j\}$ but all items in $\{p_j\}$ have been existed in $\{p_i\}$, then, $\{p_i\}$ is called the superset of $\{p_j\}$ and $\{p_j\}$ is called the subset of $\{p_i\}$ [5]. For the pattern-based AD methods, the setting of *min_sup* threshold is the foundation, and the suitable *min_sup* threshold can bring high detection accuracy.

- *Definition* 1. *n*-pattern: for a pattern $\{p_i\}$, if its length is n, then, $\{p_i\}$ is called a *n*-pattern.
- *Definition* 2. FP, RP: for a $\{pi\}$, if its *support* value (that is, the appearing times in the dataset) is not less than the preset *min_sup* threshold, then $\{pi\}$ is called a FP, that is, $\{pi\}$ is a FP once $sup(pi) \geq min\_sup$; otherwise, $\{pi\}$ is a RP.
- *Definition* 3. CFP: for a FP $\{p_i\}$, if no any superset of $\{p_i\}$ (denoted as $\{p_j\}$) can make $sup(p_j)=sup(p_i)$, then, $\{p_i\}$ is called a CFP [5].
- *Definition* 4. Pattern extension: for two n-patterns $\{p_1, p_2, \ldots, p_{(n-1)}, p_n\}$ and $\{p_1, p_2, \ldots, p_{(n-1)}, p_{(n+1)}\}$ that have the same prefix with a length of $(n-1)$, the operation of connecting the last item of a n-pattern to the end of other one *n*-pattern to form a $(n+1)$-pattern is called "pattern extension". That is, take the last item $\{p_{(n+1)}\}$ of $\{p_1, p_2, \ldots, p_{(n-1)}, p_{(n+1)}\}$ to connect right with $\{p_1, p_2, \ldots, p_{(n-1)}, p_n\}$ to form $\{p_1, p_2, \ldots, p_{(n-1)}, p_n, p_{(n+1)}\}$.
- *Definition* 5. Abnormal degree: it refers to the deviation between a data sample and other data samples in the dataset, where the large deviation indicates the high abnormal degree.
- *Definition* 6. Anomaly: for a transaction T_i, if its

abnormal degree is in top-k (value k is set in advance) ranked in the datasets, then, T_i is called an anomaly [4].

2.2. Related Works on Pattern-based AD Methods

Pattern-based AD methods first mine the corresponding patterns through pattern mining methods to facilitate the design of deviation indices based on the patterns during the detection phase, and then design some deviation indices to calculate the deviation degree of transactions for AD, thus improving the quality of collected datasets.

As the first pattern-based AD method, Find Frequent Pattern Outlier Factor (Find-FPOF) [10] was proposed in 2005 and it performed AD operation through the mining of FPs, which results in its time efficiency not so competitive due to the huge scale of mined FPs.

In the following years, the compressions of FPs (such as CFPs and MFPs) were used to reduce the scale of FPs applied in the AD phase, thus solving the drawback of the FindFPOF method for its slow detection speed. And the experimental results on Closed Frequent Pattern-based method by considering Anti-monotonic constraints (CFPA) [5] and Maximal Frequent Pattern-based Outlier Detection (MFP-OD) [7] also proved that the use of the compressions of FPs improved the time efficiency of FindFPOF to a great extent.

For the CFPA method, in addition to using the compressions of FPs, it also effectively deals with the Anti-Monotonic Constraints (C_{AM}) preset by the users, thus, its time overhead was shorter than that of other pattern-based AD methods because the transactions in which violate the C_{AM} have been filtered before pattern mining process. Besides the time efficiency, CFPA also could achieve higher detection efficiency than that of FindFPOF because it used complex deviation indices via considering massive influencing factors. MFP-OD used the MFPs to discover anomalies in the uncertain data streams to reduce the time overhead. Similar to CFPA, MFP-OD also designed several deviation indices to enhance the detection efficiency.

Compared with adopting the MFPs, because the CFPs would not lose any information of the patterns, such as the existential probability, thus, more attention has been paid for the CFP-based AD methods. However, for the FP-based, CFP-based and MFP-based AD methods, only a few patterns can be mined from the datasets under large *min_sup* thresholds, which would result in the low detection efficiency.

To solve this problem, Minimal Rare Pattern-based Method by considering Anti-Monotonic Constraints (MRPAC) [4] was proposed to improve the detection accuracy. In the MRPAC method, the C_{AM} were used when mining the MRPs, which could reduce the time overhead.

In comparison with the existing pattern-based AD methods, although the foundation of AD-DCFP is also the mined CFPs, but it calculates the abnormal degree for all transactions in the dataset via the pattern distance rather than deviation indices, which can solve the low detection accuracy for the FP-based AD methods under large *min_sup* values. The differences of the proposed AD-DCFP method and previously proposed pattern-based AD methods are shown in Table 1, where DI indicates the deviation index.

Table 1. Comparisons of pattern-based AD approaches.

Methods	Types of used patterns	Base to measure the abnormal degree
FindFPOF [10]	FPs	Simple DI
CFPA [5]	CFPs	Complex DI
MRPAC [4]	MRPs	Complex DI
MFP-OD [7]	MFPs	Complex DI
AD-DCFP	CFPs	Pattern distance

3. Pattern Distance-based AD Method

In face of heavy time consumption on small *min_sup* threshold, the idea of mining CFP can be introduced to reduce the scale of mined patterns used in the AD stage, which can improve the time efficiency as well as achieve high detection accuracy. In other case, when the *min_sup* threshold is set large, CFP-based AD methods can only achieve low detection accuracy due to the scale of mined CFPs is few. To solve this problem, this paper presents an efficient AD method based on the distance of CFPs called AD-DCFP through introducing the idea of pattern distance of CFPs, to detect the anomalies via pattern mining operation and pattern distance calculation operation. We use the following example with five transactions shown in Table 2 to detailed introduces the specific process of AD-DCFP method.

Table 2. The example of the dataset.

TID	Transactions	*TID*	Transactions
T_1	{*A*, *C*, *D*, *E*, *F*}	T_2	{*A*, *B*, *E*}
T_3	{*C*, *E*, *F*}	T_4	{*A*, *C*, *D*, *F*}
T_5	{*C*, *E*, *F*}	…	……

3.1. The Mining of CFPs

Based on the representation manner of the datasets, the mining of CFPs is roughly concluded into two categories, i.e., horizontal (*TID*×items) and vertical (items×*TID*). It was known from literature [23] that the vertical-based manner outperforms the horizontal-based manner, thus, we adopt the vertical-based way to effectively min the CFPs. Firstly, it is necessary to convert the items (aka 1-patterns) shown in the horizontal form to vertical form, and then the bit-vector structure is used to represent the encoded transactions. However, the traditional CFP mining method converts all items existing in the transactions into bit-vector directly to mine the patterns, which is very memory-consuming as well as seriously affects the mining efficiency. Aimed at this problem, before converting the storage manner, we first calculate the *support* value of each item to discard the RPs, thereby reducing the

number of patterns that can be used to perform "pattern extension" operations, which is very useful for saving the time overhead. Notice that, if we delete n items before performing "pattern extension", then the reduced times on "pattern extension" operations are 2^{m-n} (where m is the number of different items in the transactions). And then, the process of CFP mining is introduced with the example listed in Table 2, where the *min_sup* threshold is set to 2 in this example.

- **Phase** 1. Scan the items in the transactions to calculate the *support* value of each 1-pattern, and then discard the RPs.
- **Example** 1. In this example, because *support*(A)=3, *support*(B)=1, *support*(C)=4, *support*(D)=2, *support*(E)= 4 and *support*(F)=4, thus, $\{B\}$ should be discarded; the other 1-patterns should be regarded as the potential items that can participate in the following operations.
- **Phase** 2. Convert the frequent 1-patterns in the horizontal-based form to vertical form with the representation of bit-vector structure.
- **Example** 2. The frequent 1-patterns $\{A\}$, $\{C\}$, $\{D\}$, $\{E\}$ and $\{F\}$ can convert to the vertical form as shown in Table 3, where "0" represents that this 1-pattern not exists in the transaction and "1" represents that this 1-pattern exists in the transaction.

Table 3. Vertical representation of encoded transaction dataset.

Item	Tidset	Bit-vector
A	T_1, T_2, T_4	11010
C	T_1, T_3, T_4, T_5	10111
D	T_1, T_4	10010
E	T_1, T_2, T_3, T_5	11101
F	T_1, T_3, T_4, T_5	10111

- **Phase** 3. Take out the 1-patterns in the vertical form in turn with the order of increased *support* value to calculate the *support* value of each extended 2-pattern through "AND" operation, and then discard the rare 2-patterns to save the time overhead.
- **Example** 3. For the 1-patterns $\{A\}$, $\{C\}$, $\{D\}$, $\{E\}$ and $\{F\}$, they are arranging by their increased *support* value are $\{D\}$, $\{A\}$, $\{C\}$, $\{E\}$ and $\{F\}$. And then, the *support* value of each extended 2-pattern is calculated as follows:

For 1-patterns $\{D\}$ and $\{A\}$, they can be extended to $\{DA\}$ and its *support* value is calculated as shown in Figure 1.

$$\begin{array}{ll} D & [1\ 0\ 0\ 1\ 0] \\ A & [1\ 1\ 0\ 1\ 0] \\ = & [1\ 0\ 0\ 1\ 0] \end{array}$$

Figure 1. The calculation of *support* value for $\{DA\}$.

That is, *sup*(DA)=2 (the number "1" appears for two times), thus, 2-pattern $\{DA\}$ is a FP.

For the other extended 2-patterns, their *support* value is calculated like mentioned above. Because *sup*(DC)=2, *sup*(DE)=1, *sup*(DF)=2, *sup*(AC)=2, *sup*(AE)=2, *sup*(AF)=2, *sup*(CE)=3, *sup*(CF)=4 and *sup*(EF)=3, thus, 2-patterns $\{DC\}$, $\{DE\}$, $\{DF\}$, $\{AC\}$, $\{AE\}$, $\{AF\}$, $\{CE\}$, $\{CF\}$ and $\{EF\}$ can be further extended.

- **Phase** 4. Extend the frequent 2-patterns with the same suffix to form 3-patterns, and then their *support* value is calculated using the "AND" operation.
- **Example** 4. For two 2-patterns with the same suffix of $\{F\}$, including $\{DF\}$ and $\{AF\}$, $\{DF\}$ and $\{CF\}$, $\{DF\}$ and $\{EF\}$, $\{AF\}$ and $\{CF\}$, $\{AF\}$ and $\{EF\}$, $\{CF\}$ and $\{EF\}$, they can be extended to 3-patterns of $\{DAF\}$, $\{DCF\}$, $\{DEF\}$, $\{ACF\}$, $\{AEF\}$ and $\{CEF\}$; for these three 3-patterns, *sup*(DAF)=2, *sup*(DCF)=2, *sup*(DEF)=1, *sup*(ACF)=2, *sup*(AEF)=1, *sup*(CEF)=3. For two 2-patterns with the same prefix of $\{E\}$, including $\{DE\}$ and $\{CE\}$, $\{DE\}$ and $\{AE\}$, $\{CE\}$ and $\{AE\}$, they can be extended to 3-patterns of $\{DCE\}$, $\{DAE\}$ and $\{CAE\}$; for these three 3-patterns, *sup*(DCE)=1, *sup*(DAE)=1, *sup*(CAE)=1. For the 2-patterns with the same prefix of $\{C\}$, including $\{DC\}$ and $\{AC\}$, they can be extended to 3-patterns of $\{DAC\}$; for this 3-pattern, *sup*(DAC)=2. Thus, 3-patterns $\{DAF\}$, $\{DCF\}$, $\{ACF\}$, $\{CEF\}$ and $\{DAC\}$ can be extended in the next process.
- **Phase** 5. Extend the frequent 3-patterns with the same suffix to form 4-patterns, and then their *support* value is calculated using the "AND" operation.
- **Example** 5. For two 3-patterns with the same suffix of $\{CF\}$, including $\{DCF\}$ and $\{ACF\}$, they can be extended to 4-pattern of $\{DACF\}$; for this 4-pattern, *sup*($DACF$)=2. Because no longer pattern can be extended any more, the "pattern extension" operation is finished.
- **Phase** 6. Verify whether the *support* value of any superset is not equal to its *support* value, thereby seeking for the true CFPs.
- **Example** 6. For the FP $\{A\}$, because the *support* value of any superset of $\{A\}$ is not equal to its *support* value, thus, it is a CFP. For the FP $\{E\}$, because the *support* value of any superset of $\{E\}$ is not equal to its *support* value, thus, it is a CFP. For the FPs $\{AE\}$ and $\{CF\}$, because the *support* value of any superset of $\{AE\}$ and $\{CF\}$ is not equal to them, thus, they are CFPs. For the FP $\{CEF\}$, because the *support* value of any superset of $\{CEF\}$ is not equal to its *support* value, thus, it is a CFP. For the FP $\{DACF\}$, because the *support* value of any superset of $\{DACF\}$ is not equal to its *support* value, thus, it is a CFP.

On the basis of above operations, all CFPs can be correctly mined from the transactions using the proposed CFP mining method called CFPM. Under the

preset *min_sup* threshold (=2), the final mined CFPs are {*A*:3}, {*E*:4}, {*AE*:2}, {*CF*:4}, {*CEF*:3} and {*DACF*:2}. The details of the CFPM method are shown in Algorithm (1).

Algorithm 1: CFPM.

```
Input: Dataset, min_sup threshold
Output: CFPs
foreach (1-pattern {pat} in the transaction)
{
  if (sup(pat)<min_sup)
  {
    delete {pat}
  }
}
convert 1-patterns into vertical form
foreach (frequent 1-patterns {pat_i} and {pat_j})
{
  perform "pattern extension" to form {pat_i,pat_j}
  if (sup(pat_i,pat_j)<min_sup)
  {
    delete {pat_i,pat_j}
  }
}
for (k=2;k++;)
{
    foreach (frequent k-patterns with the same suffix with the
length of (k-1))
    {
    perform "pattern extension" to form (k+1)-pattern {pat_k+1}
    if (sup(pat_k+1)<min_sup)
    {
      delete {pat_k+1}
    }
    }
}
check whether the support value of any superset is equal to that
of the patterns
return CFPs
```

Because the proposed CFPM method consists of four parts, thus, its computing complexity can be summarized in four components.

1. In phase 1, it is required to scan every 1-pattern for one time to calculate their *support* value, its computing complexity is $O(m)$, where m represents the number of 1-patterns in the transaction.
2. In phase 2, it is required to scan every frequent 1-pattern in the transaction to convert them to vertical form, its computing complexity is $O(m)$.
3. In phase 3, the *support* value of every extended patterns are calculated to determine whether they are FPs, where there are $(2^m\text{-}1\text{-}m)$ patterns can be extended, thus, its computing complexity is $O(2^m\text{-}1\text{-}m)$.
4. In phase 4, each extended pattern should be checked whether it is a CFP, its computing complexity is $O(2^m\text{-}1\text{-}m)$.

Overall, in the worst case, the computing complexity of CFPM is $O(m+m+2^m\text{-}1\text{-}m+2^m\text{-}1\text{-}m)$, that is, the final computing complexity of the CFPM method is $O(2^{m+1})$.

3.2. Anomaly Detection

Unlike the traditional *CFP*-based AD methods, once the CFPs are mined from the transactions, the pattern distance between each CFP and transaction needs to be calculated to determine whether the transaction is an anomaly. Therefore, the calculation of pattern distance is very important. For accurately measuring the abnormal degree of each transaction in the dataset, the following important factors should be considered in the design of pattern distance.

1. The difference between transaction and CFP: For two transactions T_1 and T_2, where the difference (that is, the ratio of the number of patterns in a transaction that are different with CFP to the total number of patterns in this transaction) between T_1 and CFP $\{X\}$ is less than that between T_2 and CFP $\{X\}$, then, T_1 is less likely to be determined as anomaly than T_2.
2. The *support* value of contained CFPs in transaction: For two transactions T_1 and T_2, where T_1 contains CFP $\{X\}$, T_2 contains CFP $\{Y\}$ and $sup(X)>sup(Y)$, it indicates that $\{X\}$ appears frequently than $\{Y\}$, which will cause T_1 is less likely to be determined as anomaly than T_2.
3. The length of contained CFPs in transaction: For two transactions T_1 and T_2, where T_1 contains CFP $\{X\}$, T_2 contains CFP $\{Y\}$ and $len(X)>len(Y)$, it indicates that there will be more FPs contained in $\{X\}$ rather than in $\{Y\}$, which will cause T_1 is less likely to be determined as anomaly than T_2.

With the consideration of above factors, the deviation index of CFP and the pattern distance are designed in the AD process, and they are shown as follows:

- *Definition* 7. Deviation Index of CFP (DICFP): For a CFP $\{X\}$, its length is $len(X)$, its *support* value is $sup(X)$, then, $DICFP(X)$ is defined as:

$$DICFP(X) = \frac{len(X)}{2^{len(X)-1}} \times \frac{1}{\sup(X)} \tag{1}$$

- *Definition* 8. Pattern Distance (PD): For each T_i in the dataset, the contained CFP is $\{X\}$, its pattern distance $PD(T_i)$ is defined as:

$$PD(T_i) = \sum_{X \in T_i} \left(1 - \frac{\text{len}(T_i \cap X)}{\text{len}(T_i)}\right) \times DICFP(X) \tag{2}$$

For the designed pattern distance function $PD(T_i)$, in addition to the length of contained CFPs and *support* value of contained CFPs like existing pattern-based AD methods, it also considers the difference between the transaction and contained CFPs, which leads to high detection accuracy under large *min_sup* threshold. The reason for appearing this situation is that the number of mined CFPs becomes less with the gradually increasing of *min_sup* thresholds, and the smaller number of CFPs results in large pattern distance because the difference between CFPs and transactions become large, thus, the

design of pattern distance function is very efficient for detecting anomalies.

Based on the designed metric, the pattern distance of each transaction is calculated to measure its abnormal degree, thus accurately discovering potential anomalies. For the transactions, the larger pattern distance indicates the lower similar degree between the contained *CFPs* and the transaction. A large number of contained *CFPs* in the transaction indicate the patterns appearing more frequently, that is, the transactions are less like anomalies since one feature of anomaly is appearing rarely. With the above analysis, the transactions in the datasets are arranged with their decreasing value of pattern distance, while the transactions with large pattern distance are determined as anomalies. Until now, AD-DCFP algorithm finished the AD operation. The pseudo-code of the proposed AD-DCFP method is shown in Algorithm (2).

Algorithm 2: AD-DCFP.

```
Input: Dataset, min_sup threshold, k
Output: Anomalies
mine the CFPs using Algorithm 1
PD(Ti)=0
foreach (Ti)
{
  foreach (CFP {X})
  {
    calculate DICFP(X)
    calculate PD(X)
  }
}
sort the transactions via their decreasing PD(Ti) value
Anomalies ← top k transactions
return Anomalies
```

The set of parameter k in the proposed AD-DCFP method is decided by the users themselves. In fact, in the AD-DCFP method, the transactions are sorted according to their decreased value of pattern distance, that is, AD-DCFP method provides the information about which transactions having a bigger probability to be the anomalies, which leads to the set of value k is not so important.

The computing complexity of AD-DCFP method is made up of the mining of CFPs, the calculation of DICFP and PD values, and the sorting of transactions.

1. The computing complexity of the mining of CFPs is $O(2^{m+1})$.
2. When calculating the DICFP and PD values, it is required to scan each CFP, thus, in the worst case, its computing complexity is $O(k \times p)$, where k is the number of CFPs and p is the number of transactions.
3. The operation of sorting the transactions needs to use the quick sort manner, its computing complexity is $O(plog_2p)$. In general, the computing of AD-DCFP method is $O(2^{m+1}+(k+log_2p)\times p)$.

And then, we use the example shown in Table 2 to illustrate the specific process of the AD-DCFP method, where the *min_sup* threshold and k are all set to 2. As reported in section 3.1, the mined CFPs and their *support* value are {*A*:3}, {*E*:4}, {*AE*:2}, {*CF*:4}, {*CEF*:3} and {*DACF*:2}.

- *Step* 1: Calculate the *DICFP*(*X*) value for each CFP {*X*}.

  For CFP {*A*:3}, its *DICFP* value is $DICFP(\{A\})=1/(2^1-1)\times(1/1)=1$.
  For CFP {*E*:4}, its *DICFP* value is $DICFP(\{E\})=1/(2^1-1)\times(1/4)=1/4$.
  For CFP {*AE*:2}, its *DICFP* value is $DICFP(\{AE\})=2/(2^2-1)\times(1/2)=1/3$.
  For CFP {*CF*:4}, its *DICFP* value is $DICFP(\{CF\})=2/(2^2-1)\times(1/4)=1/6$.
  For CFP {*CEF*:3}, its *DICFP* value is $DICFP(\{CEF\})=3/(2^3-1)\times(1/3)=1/7$.
  For CFP {*DACF*:2}, its *DICFP* value is $DICFP(\{DACF\})=4/(2^4-1)\times(1/2)=2/15$.

- *Step* 2: Determine the contained CFPs and calculate the $PD(T_i)$ value.

  In transaction T_1, the contained CFPs are {*DACF*}, {*A*}, {*AE*}, {*CF*}, {*CEF*} and {*E*}, thus, $PD(T_1)=(1-1/5)\times2/15+(1-1/5)\times1+(1-2/5)\times1/3+(1-2/5)\times1/6+(1-3/5)\times1/7+(1-1/5)\times1/4=1453/1050$.
  In transaction T_2, the contained CFPs are {*A*}, {*AE*} and {*E*}, thus, $PD(T_2)=(1-1/3)\times1+(1-2/3)\times1/3+(1-1/3)\times1/4=17/18$.
  In transaction T_3, the contained CFPs are {*CF*}, {*CEF*} and {*E*}, thus, $PD(T_3)=(1-2/3)\times1/6+(1-3/3)\times1/7+(1-1/3)\times1/4=2/9$.
  In transaction T_4, the contained CFPs are {*DACF*}, {*A*} and {*CF*}, thus, $PD(T_4)=(1-4/4)\times2/15+(1-1/4)\times1+(1-2/4)\times1/6=5/6$.
  In transaction T_5, the contained CFPs are {*CF*}, {*CEF*} and {*E*}, thus, $PD(T_5)=(1-2/3)\times1/6+(1-3/3)\times1/7+(1-1/3)\times1/4=2/9$.

- *Step* 3: Sort the transactions according to their decreasing $PD(T_i)$ values, and then the top k transactions with largest $PD(T_i)$ are judged as anomalies.

In this example, because $PD(T_1)> PD(T_2)> PD(T_4)> PD(T_3)=PD(T_5)$, thus, the anomalies judged by AD-DCFP method are T_1 and T_2.

4. Experiments and Analysis

To test the efficiency of AD-DCFP method, we conduct extensive experiments to answer the following questions (RQs):

- **RQ1:** Can the proposed AD-DCFP method more accurately detect the potential anomalies than state-of-the-art AD methods?
- **RQ2:** Whether the proposed AD-DCFP method can detect anomalies using less time overhead?
- **RQ3:** Whether the proposed AD-DCFP method can

be applied in large-scale or large-dimensional datasets?

Massive experiments are conducted on one synthetic dataset [7] and five public datasets, where the information of used datasets is shown in Table 4. The anomalies on synthetic dataset have been marked. The transactions in minority class of public datasets are regarded as anomalies, therefore, the transactions that belong to classes 2 and 4 on Lymphography dataset are considered as anomalies, the transactions that belong to class 2 on Satimage-2 dataset are considered as anomalies, the transactions that belong to class 0 on Heart dataset are considered as anomalies, the transactions that belong to class 1 on KDDCUP99 dataset are considered as anomalies, and the transactions that belong to class 4 on ForestCover dataset are considered as anomalies.

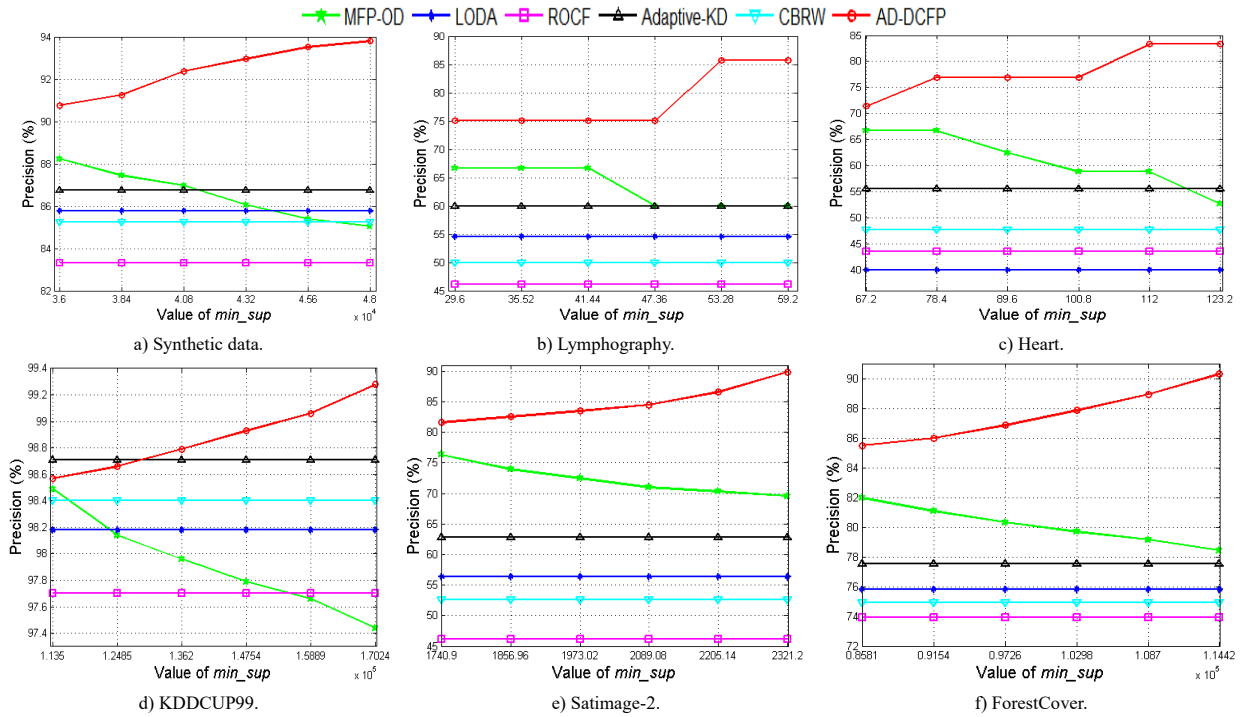
Table 4. The information of used datasets.

Name	Num. Trans	Dimensions	Num. anomalies
Synthetic data	120000	8	2800
Lymphography	148	18	6
Heart	224	44	10
KDDCUP99	567479	3	2211
Satimage-2	5803	36	71
ForestCover	286048	10	2747

In the experiment, the compared methods include pattern-based method MFP-OD [7], model-based method Local Outlier Detection Algorithm (LODA) [19], Relative Outlier Clustering-based Factor method (ROCF) [11], density-based method Adaptive Kernel Density-based anomaly detection (Adaptive-KD) [24] as well as other categorical AD method Coupled Biased Random Walks (CBRW) [17]. All methods are running on a computer with an Intel dual core I7-10700 2.90 GHz processor.

4.1. Answer to RQ1

This subsection aims to verify the detection accuracy of AD-DCFP under different *min_sup* thresholds, where the evaluation indices of Precision, Recall and F1-measure are applied to measure the efficiency. Different from the traditional indices of Precision and Recall, the Precision in this experiment represents the ratio of the true anomalies to the retrieved transactions when all anomalies are identified, and the Recall represents the ratio of the retrieved true anomalies to all true anomalies as the number of retrieved transactions is equal to that of true anomalies. The experimental results are shown in Figures 2 to 4.



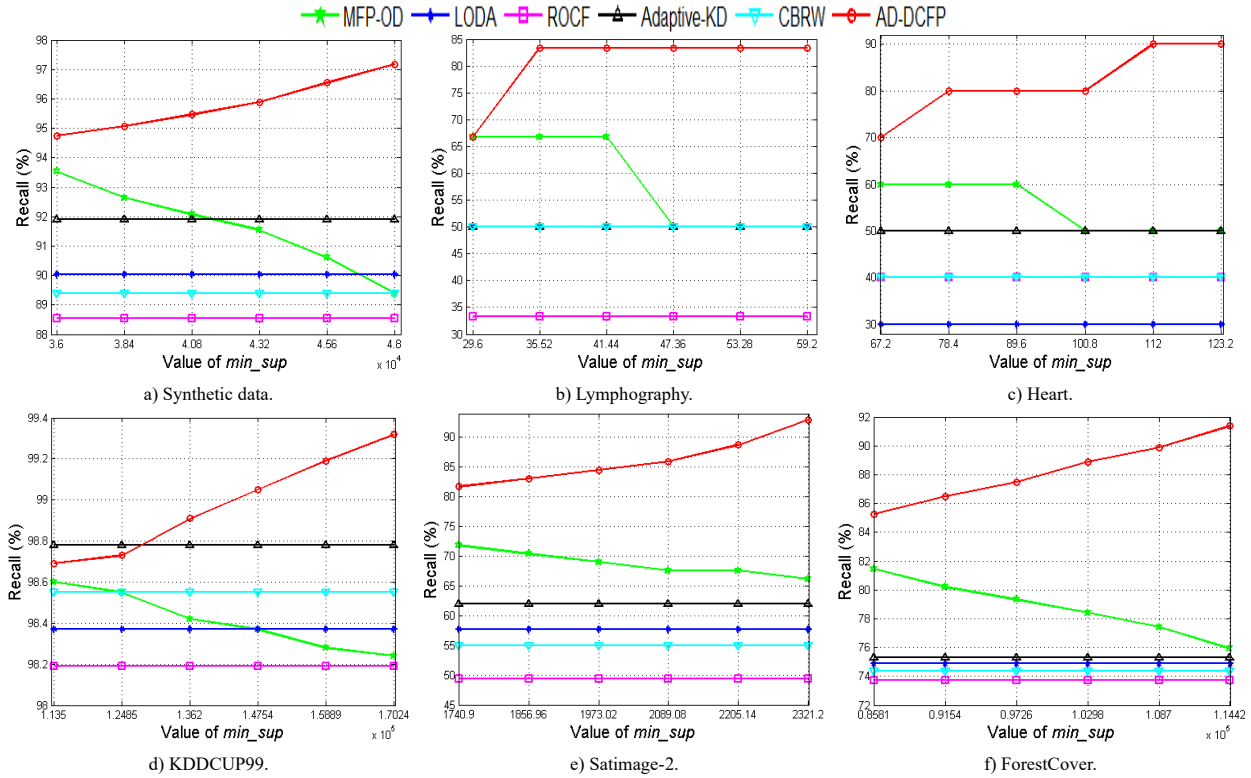
a) Synthetic data. b) Lymphography. c) Heart.

d) KDDCUP99. e) Satimage-2. f) ForestCover.

Figure 2. The Precision of compared AD methods.

As is shown in Figure 2, except for the dataset KDDCUP99, the Precision of AD-DCFP is always the highest under relatively large *min_sup* thresholds, and its Precision presents an increasing trend with the increase of *min_sup* thresholds, while the Precision of MFP-based AD method MFP-OD shows a decreasing trend; the Precision of other four other kinds of AD methods keep constant under different *min_sup* thresholds. The reason for appearing the increasing trend of Precision of AD-DCFP method is that the scale of mined CFPs presents much smaller as the gradually increasing of *min_sup* thresholds, while the small scale of CFPs results in the big pattern distance because the different parts between CFPs and transactions are much larger; on the opposite, the foundation of MFP-OD is the MFPs and deviation indices, because the scale of mined MFPs is very less under large *min_sup* thresholds, which causes only less patterns can be participated in

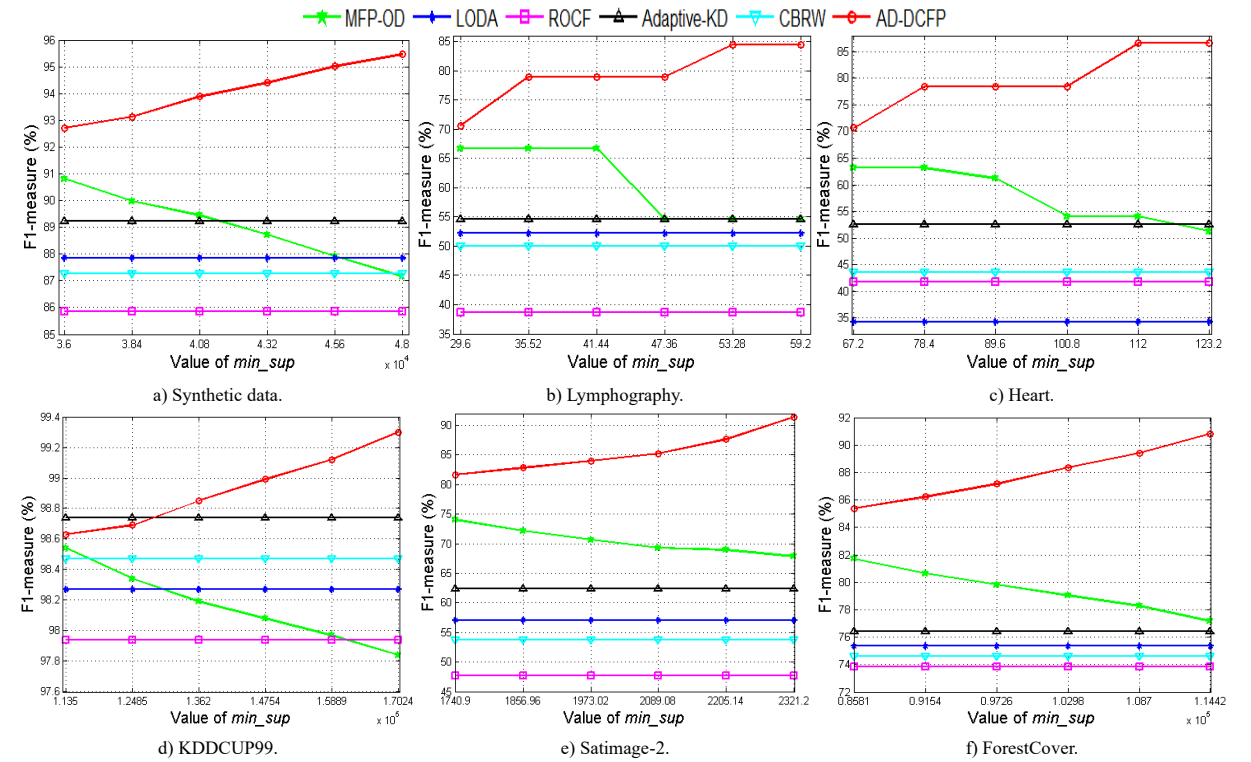
the follow-up AD phase, therefore, the designed deviation indices only consider less influencing factors and thus decreasing the detection accuracy of MFP-OD. For the LODA, ROCF, Adaptive-KD and CBRW methods, the determination of anomalies will not be influenced by the mined patterns, thus, the different settings of *min_sup* thresholds do no influence the determining of anomalies. On dataset KDDCUP99, the Precision of Adaptive-KD is slightly higher than that of AD-DCFP under small *min_sup* thresholds, which is caused by Adaptive-KD using adaptive kernel width to calculate the local density and this strategy makes it adapting different properties of dataset KDDCUP99, while the extensive CFPs make the pattern distance much less.



a) Synthetic data. b) Lymphography. c) Heart.

d) KDDCUP99. e) Satimage-2. f) ForestCover.

Figure 3. The Recall of compared AD methods.



a) Synthetic data. b) Lymphography. c) Heart.

d) KDDCUP99. e) Satimage-2. f) ForestCover.

Figure 4. The F1-measure of compared AD methods.

Similar to Precision, the Recall of AD-DCFP method (shown in Figure 3) on the synthetic dataset and four public datasets (except for KDDCUP99) is highest in all six compared AD methods, but it is slightly lower than Adaptive-KD under small *min_sup* threshold on dataset KDDCUP99. In addition, the Recall of AD-DCFP presents an increasing trend accompanied when the *min_sup* thresholds gradually becoming larger. Except for dataset KDDCUP99, the Recall of MFP-OD is higher than other four AD methods, but its Recall presents a decreasing trend as the *min_sup* thresholds becoming larger. In other four AD methods, the Recall of Adaptive-KD is slightly higher than LODA, ROCF and CBRW methods (the Recall of Adaptive-KD is the same as LODA and CBRW on dataset Lymphography), while the Recall of ROCF is lowest in most cases.

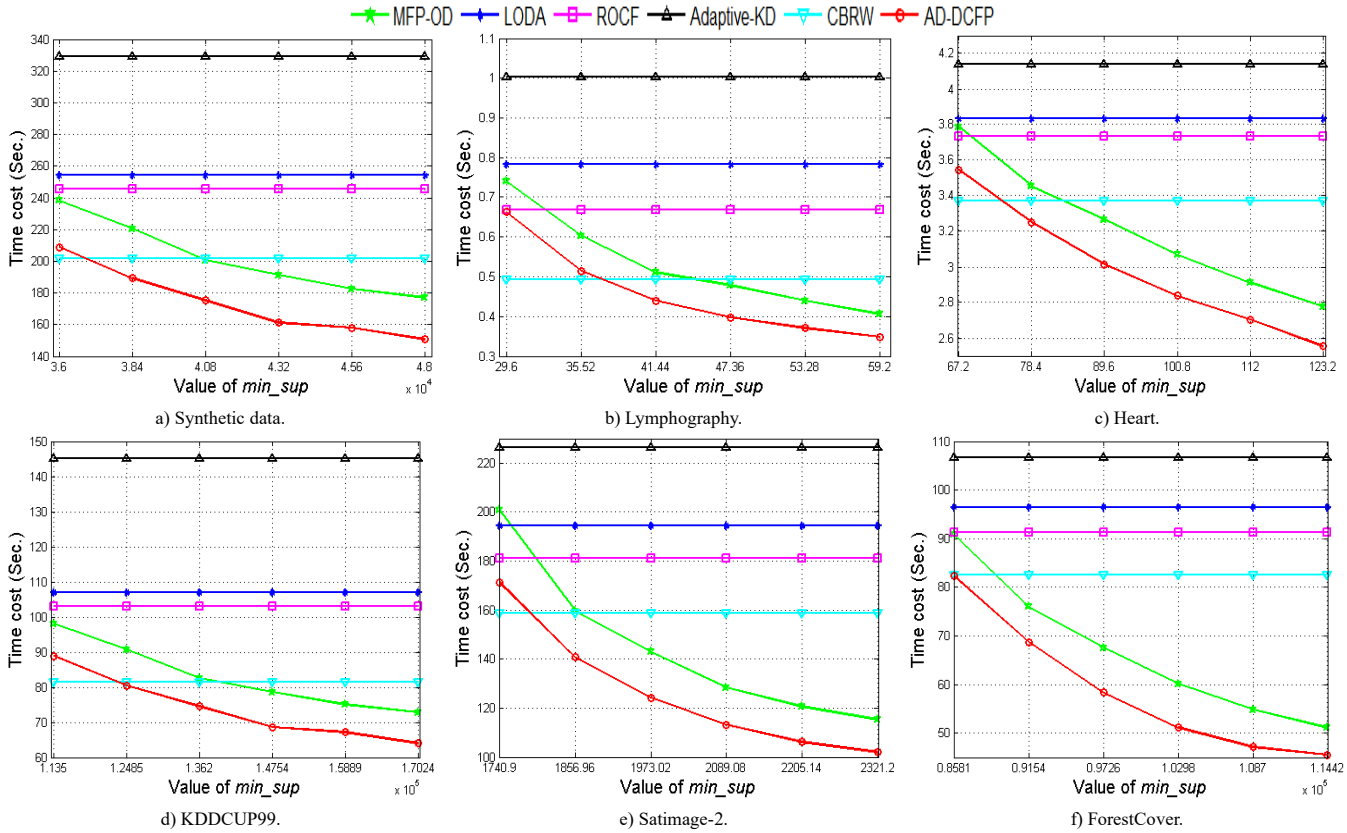
As can be seen from Figure 4 that except on the KDDCUP99 dataset, the proposed AD-DCFP method achieves the best F1-measure on other five datasets at different *min_sup* threshold; and the F1-measure of the AD-DCFP method exhibits an increasing trend as the *min_sup* keeps increasing, which is due to the fact that the number of mined CFPs is less at larger *min_sup* threshold, which leads to the gap between CFPs and transactions becoming more pronounced, and thus more capable of detecting anomalies. On the KDDCUP99 dataset, the F1-measure metric of the proposed AD-DCFP method is lower than that of Adaptive-KD method at smaller *min_sup*, which is mainly due to the fact that large scale of CFPs make the distance between the transaction and contained CFPs becoming small, thus making it less easy to detect anomalies; however, as the *min_sup* continues to increase, the advantage of AD-DCFP method is realized to a greater extent, which makes its F1-measure becoming higher. Since four compared methods of LODA, RPCF, Adaptive-KD and CBRW are not pattern-based AD methods, their detection efficiency does not fluctuate with the change of *min_sup* thresholds. Overall, the experimental result shows that the AD-DCFP method has a good ability to detect anomalies.

4.1.1. Answer to RQ1

Extensive experiments show that the proposed AD-DCFP method can more accurately detect anomalies than compared five state-of-the-art AD methods, which indicates that the designed deviation index of CFP and pattern distance can promote the detection of anomalies. In addition, with the increase of *min_sup* threshold, the AD-DCFP method can obtain a higher detection accuracy than FP-based and RP-based AD methods.

4.2. Answer to RQ2

This subsection aims to test the time efficiency of the proposed AD-DCFP on six datasets under different *min_sup* thresholds. In order to eliminate the contingency, each experiment is run for fifty times, and then the average time cost is output as the final experimental result, which are shown in Figures 5-a) and (f).



a) Synthetic data. b) Lymphography. c) Heart.

d) KDDCUP99. e) Satimage-2. f) ForestCover.

Figure 5. The time cost of compared AD methods.

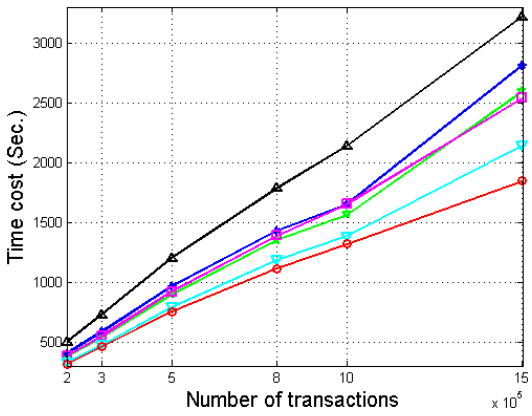
As is presented in Figures 5-a) to (f) that under relatively large *min_sup* thresholds, the time cost of AD-DCFP is shorter than that of compared MFP-OD, LODA, ROCF, Adaptive-KD and CBRW methods on all datasets. The reason is that only a small scale of extensible FPs is existing under large *min_sup* thresholds and thus reducing the time cost on time-consuming "pattern extension" operations; in addition, the small scale of mined CFPs leads to less time overhead on the calculation of DICFP and pattern distance. However, when the *min_sup* threshold is set small, the time overhead of AD-DCFP is slightly longer than that of CBRW method due to AD-DCFP method needs to perform time-consuming "pattern extension" operations on FPs. Accompanied with the addition of *min_sup* thresholds, the time overhead of two pattern-based AD methods (including AD-DCFP and MFP-OD) becomes shorter, it is owing to that the number of extensible FPs is much smaller when the *min_sup* threshold is set large, which results in the time cost on the time-consuming "pattern extension" operation reducing to a great extent. Compared with it, the time cost of LODA, ROCF, Adaptive-KD and CBRW methods keeps constant under different *min_sup* thresholds, which is caused by the foundation of these AD methods is the distance or density of transactions rather than mined patterns, thus, the time cost of these methods will not be influenced by the *min_sup* thresholds. Except for the pattern-based AD methods, the time overhead of Adaptive-KD is longer than LODA, ROCF and CBRW methods, it is owing to that all items in the transactions need to calculate their density to determine whether the transactions are anomalies. Extensive experiments also verify that the proposed AD-DCFP method has high time efficiency.

4.2.1. Answer to RQ2

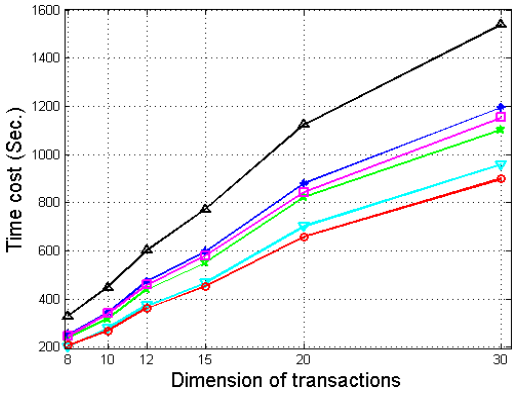
The experiments show that the proposed AD-DCFP method can detect anomalies from the datasets with less time overhead than the compared state-of-the-art AD methods, which indicates that the use of CFPs in the AD process can reduce the computing resource computation compared with FPs and RPs.

4.3. Answer to RQ3

For the proposed AD-DCFP method, its time overhead is much shorter than six compared methods under these preset *min_sup* thresholds. However, it is not clear whether the AD-DCFP method can be effectively used in large datasets or high-dimensional datasets. To verify this question, we use a synthetic dataset to test the scalability of AD-DCFP, where the number of transactions is extended to 200000, 300000, 500000, 800000, 1000000, and 1500000 (the dimension is kept on 8) firstly, and then the average dimension of transactions is extended to 8, 10, 12, 15, 20 and 30 (the number of transactions is kept to 1200000). In the experiment, they also performed for fifty times, and then we calculate the average time, they are shown in Figures 6-a) and (b).



a) Data size.



b) Data dimension.

Figure 6. Scalability of the compared AD methods on a synthetic data.

As is shown in Figure 6 that with the addition of transactions, the time overhead of six compared methods presents an increasing trend, and the increasing trend of all compared methods is very close to linear; similarly, with the increase of the dimension of transactions, the time overhead of all methods also presents an increasing trend, and the increasing trend is also close to linear. In particular, Figure 6-a) shows that the time overhead of AD-DCFP is shorter than other five methods when processing a large number of transactions, and the slope of the growth of AD-DCFP method is lowest in the six methods, which indicates that AD-DCFP is more effective for detecting the anomalies from large scale datasets than other five compared methods. Because the Adaptive-KD method consumes longer time overhead, thus, it is not suitable for the large-scale datasets. As is shown in Figure 6-b) that the time overhead of AD-DCFP is also the shortest no matter the dimension of the datasets is set low or high, and the time overhead of AD-DCFP is close to increasing one time when the dimension is added for every five. In the five compared AD approaches, the time overhead of Adaptive-KD is also longest, and the time overhead of MFP-OD, LODA and ROCF is relatively same, while the time overhead of CBRW is closest to that of AD-DCFP method, which is very

similar to that under a different number of transactions. Although the time overhead of AD-DCFP is becoming even longer when the number of datasets is becoming larger and the dimensions of datasets are becoming higher, its time cost is shorter than that of five compared AD methods. Thus, the AD-DCFP method is an ideal choice when it is required to process large-scale datasets or the datasets with high-dimension.

4.3.1. Answer to RQ3

The experiments show that the proposed AD-DCFP method can detect anomalies from the datasets with less time overhead than the compared state-of-the-art AD methods, which indicates that the use of CFPs in the AD process can reduce the computing resource computation compared with FPs and RPs.

4.4. Discussion

Extensive experimental results on six datasets show that the proposed AD-DCFP method has best detection performance than five state-of-the-art AD methods, especially with the increase of *min_sup* threshold. The reason for appearing this better detection performance of AD-DCFP method is that the number of mined CFPs is less at larger *min_sup* threshold, which leads to the gap between CFPs and transactions becoming more pronounced, and thus more capable of detecting anomalies. With the increase of *min_sup* threshold, the proposed AD-DCFP method consumes less time, it is owing to that only a small scale of extensible FPs is existing under large *min_sup* thresholds and thus reducing the time cost on time-consuming "pattern extension" operations; in addition, the small scale of mined CFPs leads to less time overhead on the calculation of DICFP and pattern distance.

Although the proposed AD-DCFP method can achieve good detection results, it still has the following problems:

1. When the dataset to be processed is very large or high-dimensional, the time consumption of AD-DCFP method will show a rapid growth trend, which poses a great challenge for real-time AD. To solve this problem, we would like to introduce parallel computing in the future for CFP mining and pattern distance calculation, to reduce the time required for AD through collaborative work of multiple computers, thereby improving the algorithm's ability to handle high-dimensional and large-scale data.
2. In real life, uncertain data has become a common type of data, and the existence of uncertainty requires full consideration of the probability of each feature in the data during pattern mining. However, the proposed AD-DCFP method does not have the ability to handle uncertain data, resulting in very low detection accuracy. Therefore, we would like to introduce an uncertainty processing module into AD-DCFP in the future, enabling it to effectively detect anomalies from uncertain data through considering the probability of each feature.

5. Conclusions

To solve the low detection accuracy of CFP-based AD method when processing the large *min_sup* thresholds, based on the mining of CFPs and the idea of pattern distance, this paper proposes an efficient AD method called AD-DCFP to detect the anomalies through two phases. In the CFP mining phase, instead of using a horizontal-based manner, the vertical-based manner as well as the bit-vector are used to represent the 1-patterns whose *support* value is equal or larger than predefined *min_sup* threshold, to further enhance the efficiency of CFP mining. In the AD phase, based on the mined CFPs, the deviation index of CFP and pattern distance are designed instead of using deviation indices used in traditional CFP-based AD methods to compute the abnormal degree, thereby overcoming the shortcomings of low efficiency of traditional pattern-based methods. Finally, the transactions have top-k ranked pattern distance value are determined as anomalies.

Massive experiments on six datasets verify that compared with five state-of-the-art AD methods, the proposed AD-DCFP method has better detection efficiency when processing the large *min_sup* thresholds, which is benefited by the large distance between CFPs and each transaction (the large distance is caused by the small number of mined CFPs) under large *min_sup* thresholds. In addition, the experimental results also show that the time overhead of AD-DCFP is shorter than six compared methods, it is owing to that the abnormal degree of each transaction in the AD-DCFP method is calculated only through pattern distance rather than several deviation indices, which can reduce the scanning times of transactions.

In the future, we would like to verify the efficiency of AD-DCFP in some real applications, such as track detection, intrusion detection and so on.

References

[1] Angiulli F. and Fassetti F., "Uncertain Distance-Based Outlier Detection with Arbitrarily Shaped Data Objects," *Journal of Intelligent Information Systems*, vol. 57, no. 1, pp. 1-24, 2021. https://link.springer.com/article/10.1007/s10844-020-00624-7

[2] Arias L., Oosterlee C., and Cirillo P., "AIDA: Analytic Isolation and Distance-based Anomaly Detection Algorithm," *Pattern Recognition*, vol. 141, pp. 109607, 2023. https://doi.org/10.1016/j.patcog.2023.109607

[3] Boahen E., Bouya-Moko B., and Wang C., "Network Anomaly Detection in a Controlled Environment Based on an Enhanced

PSOGSARFC," *Computers and Security*, vol. 104, pp. 102225, 2021. https://doi.org/10.1016/j.cose.2021.102225

[4] Cai S., Chen J., Chen H., Zhang C., Li Q., Sosu R., and Yin S., "An Efficient Anomaly Detection Method for Uncertain Data Based on Minimal Rare Patterns with the Consideration of Anti-Monotonic Constraints," *Information Sciences*, vol. 580, pp. 620-642, 2021. https://doi.org/10.1016/j.ins.2021.08.097

[5] Cai S., Huang R., Chen J., Zhang C., Liu B., Yin S., and Geng Y., "An Efficient Outlier Detection Method for Data Streams Based on Closed Frequent Patterns by Considering Anti-Monotonic Constraints," *Information Sciences*, vol. 555, pp. 125-146, 2021. https://doi.org/10.1016/j.ins.2020.12.050

[6] Cai S., Li L., Chen J., Zhao K., Yuan G., Sun R., Sosu R., and Huang L., "MWFP-Outlier: Maximal Weighted Frequent-Pattern-Based Approach for Detecting Outliers from Uncertain Weighted Data Streams," *Information Sciences*, vol. 591, pp. 195-225, 2023. https://doi.org/10.1016/j.ins.2022.01.028

[7] Cai S., Li L., Li S., Sun R., and Yuan G., "An Efficient Approach for Outlier Detection from Uncertain Data Streams Based on Maximal Frequent Patterns," *Expert Systems with Applications*, vol. 160, pp. 113646, 2020. https://doi.org/10.1016/j.eswa.2020.113646

[8] Carcillo F., Borgne Y., Caelen O., Kessaci Y., Oble F., and Bontempi G., "Combining Unsupervised and Supervised Learning in Credit Card Fraud Detection," *Information Sciences*, vol. 557, pp. 317-331, 2021. https://doi.org/10.1016/j.ins.2019.05.042

[9] Ghafoori Z., Erfani S., Bezdek J., Karunasekera S., and Leckie C., "LN-SNE: Log-Normal Distributed Stochastic Neighbor Embedding for Anomaly Detection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 4, pp. 815-820, 2020. DOI:10.1109/TKDE.2019.2934450

[10] He Z., Xu X., Huang J., and Deng S., "FP-Outlier: Frequent Pattern Based Outlier Detection," *Computer Science and Information Systems*, vol. 2, no. 1, pp. 103-118, 2005. DOI:10.2298/CSIS0501103H

[11] Huang J., Zhu Q., Yang L., Cheng D., and Wu Q., "A Novel Outlier Cluster Detection Algorithm Without Top-N Parameter," *Knowledge-Based Systems*, vol. 121, pp. 32-40, 2017. https://doi.org/10.1016/j.knosys.2017.01.013

[12] Idrissi M., Alami H., Mahdaouy A, Mekki A., Oualil S., Yartaoui Z., and Berrada I., "Fed-Anids: Federated Learning for Anomaly-based Network Intrusion Detection Systems," *Expert Systems with Applications*, vol. 234, pp. 121000, 2023. https://doi.org/10.1016/j.eswa.2023.121000

[13] Li J. and Wang R., "An Anomaly Detection Method for Weighted Data Based on Feature Association Analysis," *The International Arab Journal of Information Technology*, vol. 21, no. 1, pp. 117-127, 2024. DOI: 10.34028//iajit/21/1/11

[14] Li Z., Zhu Y., and Leeuwen M., "A Survey on Explainable Anomaly Detection," *ACM Transactions on Knowledge Discovery from Data*, vol. 18, no. 1, pp. 1-54, 2023. https://doi.org/10.1145/3609333

[15] Lin W., Wang S., Wu W., Li D., and Zomaya A., "HybridAD: A Hybrid Model-Driven Anomaly Detection Approach for Multivariate Time Series," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 51, pp. 3290027, 2023. DOI:10.1109/TETCI.2023.3290027

[16] Liu B., Li X., Xiao Y., Sun P., Zhao S., Peng T., Zheng Z., and Huang Y., "Adaboost-Based SVDD for Anomaly Detection with Dictionary Learning," *Expert Systems with Applications*, vol. 238, pp. 121770, 2024. https://doi.org/10.1016/j.eswa.2023.121770

[17] Pang G., Cao L., and Chen L., "Outlier Detection in Complex Categorical Data by Modelling the Feature Value Couplings," *in Proceedings of the 25th International Joint Conference on Artificial Intelligence*, New York, pp. 1902-1908, 2016. https://ink.library.smu.edu.sg/sis_research/7146/

[18] Peng H., Zhao J., Li L., Ren Y., and Zhao S., "One-Class Adversarial Fraud Detection Nets with Class Specific Representations," *IEEE Transactions on Network Science and Engineering*, vol. 10, no. 6, pp. 3793-3803, 2023. DOI:10.1109/TNSE.2023.3273543

[19] Safaei M, Ismail A., Chizari H., Driss M., Boulila W., Asadi S., and Safaei M., "Standalone Noise and Anomaly Detection in Wireless Sensor Networks: A Novel Time-Series and Adaptive Bayesian-Network-based Approach," *Software Practice Experience*, vol. 50, no. 4, pp. 428-446, 2020. DOI:10.1002/spe.2785

[20] Xu M., Zhou X., Gao X., He W., and Niu S., "Discriminative Feature Learning Framework with Gradient Preference for Anomaly Detection," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1-10, 2023. DOI:10.1109/TIM.2022.3228007

[21] Yang X. and Li X., "ATDAD: One-Class Adversarial Learning for Tabular Data Anomaly Detection," *Computers and Security*, vol. 134, pp. 103449, 2023. https://doi.org/10.1016/j.cose.2023.103449

[22] Yuan Z., Chen B., Liu J., Chen H., Peng D., and Li P., "Anomaly Detection Based on Weighted Fuzzy-Rough Density," *Applied Soft Computing*, vol. 134, pp. 109995, 2023. https://doi.org/10.1016/j.asoc.2023.109995

[23] Zaki M., "Scalable Algorithms for Association Mining," *IEEE Transactions on Knowledge and Data Engineering*, vol. 12, no. 3, pp. 372-390, 2000. DOI:10.1109/69.846291

[24] Zhang L., Lin J., and Karim R., "Adaptive Kernel Density-Based Anomaly Detection for Nonlinear Systems," *Knowledge-Based Systems*, vol. 139, pp. 50-63, 2018. https://doi.org/10.1016/j.knosys.2017.10.009

[25] Zou B., Yang K., Kui X., Liu J., Liao S., and Zhao W., "Anomaly Detection for Streaming Data Based on Grid-Clustering and Gaussian Distribution," *Information Sciences*, vol. 638, pp. 118989, 2023. https://doi.org/10.1016/j.ins.2023.118989

Yudong Yin received the M.S. degree in Control Engineering from Nanchang Hangkong University in 2016. Since 2018, He worked as the Experimenter at the School of Software Shanxi Agricultural University. His current research interests include Smart Agriculture and 3D Modeling.

Kun Wang received the M.S. degree in Software Engineering from Xi'an Jiaotong University in 2018. He worked as the Lecturer at Shanxi Agricultural University (2018-now). His current research interests include Smart Agriculture and 3D Modeling.

Linqiang Deng received his Master degree from Nankai University in 2016. He is currently working at the School of Software, Shanxi Agricultural University (SXAU). He is the head of the Planning and Cooperation Department in the School of Software, SXAU, where he also a lecturer. His research interest is Natural Language Processing.