Journal of Computing and Security

Research Article, 2022, Volume 9, Number 2 (pp. 1–10)

**http://www.jcomsec.org**

# Illegal Miner Detection Based on Pattern Mining: A Practical Approach

Maryam Amiri [a,*], Hesam Askari [b]

[a] *Department of Computer Engineering, Faculty of Engineering, Arak University, Arak, Iran.*
[b] *Power Distribution Company of Markazi Province, Arak, Iran.*

**ARTICLE INFO.**

*Article history:*
**Received:** 16 April 2022
**Revised:** 18 August 2022
**Accepted:** 29 August 2022
**Published Online:** 1 October 2022

*Keywords:*
Miner Detection, Energy Consumption, Data Mining, Behavioral Pattern.

**ABSTRACT**

Since the most critical constituent of the cost of cryptocurrency production is energy bills, the use of illegal electricity in cryptocurrency mining farms is very common. Illegal mining farms have popped up throughout Iran in recent years. They use large collections of computer servers to verify bitcoin transactions, a highly energy-intensive process that can sap hundreds of megawatts from the power grid, which might lead to several large cities facing daily power outages. Therefore, it is essential to detect illegal miners. Although illegal miner detection might seem like a common anomaly detection problem at first glance, the results reported by different power distribution companies in Iran show that the behavior of many normal customers might be very similar to the customers' that have some illegal miners. In addition, power distribution companies prefer models that can recognize useful insights into the behavioral patterns of the customers. To the best of our knowledge, for the first time, this paper proposes a novel classIfier for miNer detection Based On patteRn miNing (INBORN) that considers the correlation between different attributes and extracts the behavioral patterns of costumers explicitly. INBORN consists of two steps: in the first step, the frequent patterns are extracted and the attributes separating miners and non-miners are determined. In the next step, a decision tree is learned based on the frequency of the patterns. Since the Power Distribution Company of Markazi province is a pioneer in the field of illegal miner detection in Iran, the performance of INBORN is evaluated based on real datasets provided by this company. The experimental results show that INBORN improves the classification accuracy compared to the common algorithms and systems used in the Power Distribution Company of Markazi province.

## 1 Introduction

The use of cryptocurrencies is becoming increasingly widespread. Crypto mining is used to earn income from cryptocurrencies. Crypto mining is the process of producing cryptocurrency with a computer system [1, 2]. In the crypto mining process, the investment and

* Corresponding author.
Email addresses: m-amiri@araku.ac.ir (M. Amiri), ictsoft@mpedc.ir (H. Askari)
https://dx.doi.org/10.22108/JCS.2022.133306.1096 ISSN: 2322-4460

operating costs must be lower than the income. The cost of electrical energy is the most important factor determining the income obtained from cryptocurrency production [3].

Bitcoin [4] is a digital currency that was introduced in 2009. It is in widespread use and has the greatest market value. A bitcoin is created by miners, using a complex mathematical procedure based on computing hashes. For each successful attempt, miners get rewards in terms of bitcoin and transaction fees. The miners participate in mining to get this reward as income. Due to the high value of bitcoin, the mining of cryptocurrencies such as bitcoin has become a common interest among miners [5].

As Bitcoin has grown in prominence, energy use has become the latest flashpoint. The energy required for the production of a Bitcoin is almost equal to the one-week energy demand of a house [6]. A study performed in Ireland has shown that the energy spent on Bitcoin can compete with the total consumption of Ireland [7]. According to the Cambridge Center for Alternative Finance (CCAF), Bitcoin currently consumes around 110 Terawatt Hours per year — 0.55% of global electricity production, or roughly equivalent to the annual energy draw of small countries like Malaysia or Sweden [8].

Illegal electricity is often used to increase income obtained from cryptocurrency production. A Bitcoin manufacturer in China generated 150 MWh of illegal electricity in a production facility built using 200 computers [3]. According to blockchain analytics firm Elliptic, around 4.5% of all bitcoin mining globally between January and April 2021 took place in Iran. The price of electricity in Iran is very lower than the average price for households and businesses [9]. Cheap and subsidized electricity causes a majority of the energy consumption from bitcoin mining to come from illegal miners, or those operating without licenses. They consume six or seven times more powerful than those with permits. Therefore, it is essential to detect illegal miners.

Due to the prohibition on cryptocurrency production in some countries such as Algeria and Vietnam or legal regulations in other countries such as Denmark, it is very important how to detect illegal miners. Nowadays, power distribution companies use various methods to detect illegal miners. Some companies use algorithms based on comparing the current energy consumption of the customers with their past. As we will show, although these methods can detect potential miners, they misclassify many customers as miners, which can lead to an increase in side costs.

To the best of our knowledge, for the first time, this paper proposes a novel classIfier for miNer detection Based on patteRn miNing (INBORN). INBORN considers the correlation between different attributes and extracts frequent behavioral patterns of customers explicitly. Figure 1 shows the structure of INBORN. According to the figure, firstly, the dataset is discretized and frequent patterns (note that we use the terms "pattern" and "itemsets" interchangeably in this paper) are extracted. The frequent patterns provide key insights into the behavior of miners [10–12]. In the next phase, based on the frequent patterns and their frequency, some rules are extracted for miner detection in two phases.

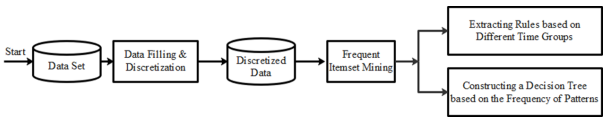


**Figure 1**. The structure of INBORN.

The rest of the paper is organized as follows: Section 2 reviews related works. Primary concepts and definitions are explained in Section 3. Section 4 introduces INBORN in detail. We present the experimental results in Section 5. Finally, the paper is concluded with our future work in Section 6.

## 2 Related Work

This section focuses on anomaly detection in users' electrical energy consumption, specifically in the form of illegal cryptocurrency mining and electricity theft. The following section reviews some of the works employed and proposed in this field.

Nowadays, various methods are used to detect illegal cryptocurrency mining facilities in electricity distribution companies. One of the most common methods, which we call AlgExp and consider a benchmark method, gathers the characteristics of electricity consumption of consumers based on their activities. The gathered data are compared to those of their counterparts or their past consumption [3].

Jiang et al. in [13] propose a game theory-based detection technique for consumers whose consumption cannot be monitored. In this method, network measurements need some equipment. In addition, in the regions where the use of illegal electricity is prevalent, it is not possible to integrate new equipment without the equipment being destructed.

Rahimi et al. in [14] propose a combined model based on statistical approaches and artificial intelligence models to detect power consumption in the smart grid. The authors consider sudden and continuous changes in customers' power usage. Firstly, the changes are detected by filters. Then, users are

clustered based on extracted statistical features. For finding the optimal hyperparameters and adapting the filter of each cluster, the genetic algorithm is used. In [3], the locations of illicit cryptocurrency farms are detected by using unique electrical characteristics, such as current harmonics and the electrical noise of switched power supplies used by the equipment employed in cryptocurrency production. Since the harmonic current is drawn by a huge number of electrical devices in the network, every harmonic distinguished in the network does not demonstrate that there is a cryptocurrency facility in that region. Therefore, to detect a cryptocurrency facility, the harmonic current drawn by this facility should be specified. However, this method had implemented by Markazi Province Electricity Distribution Company (MPEDC) and its results were not satisfactory.

Chahla et al. in [15] detect anomalies in power consumption by an unsupervised approach. For typical behavior learning and the power consumption prediction of the next hour, the authors combine the clustering-based and prediction-based methods. The authors assume that identical daily consumption behavior appears repeatedly. Based on this assumption, the K-means algorithm is applied to 24 different groups of a day to investigate behavior scenarios. To predict the next power consumption, Long Short-Term Memory (LSTM) networks have been used. The predicted value with some earlier data values merges into a vector for comparison with the learned behavior scenarios.

Tehrani et al. in [16] distinguish electricity thefts as anomalies based on three different decision tree-based algorithms, including random forest, gradient boosting methods, and decision trees. The hourly usage of customers is converted to 24-element vectors. K-means clustering is used to find different consumption patterns of users. Finally, some classification algorithms are applied to each cluster.

Ouyang et al. in [17] propose a multi-view stacking ensemble model to detect abnormal energy consumption using different IoT sensors in the industrial environment. The authors consider power consumption data as a time series and use a hierarchical time series feature extraction method to extract different types of features including decompose features, shift features, summary features and transform features. Some three-stage multi-view stacking ensemble machine learning models are trained to explore underlying classes.

Li et al. in [18] propose a framework to detect electricity consumption anomalies for industrial wireless sensor networks. Their mechanism is based on machine learning and blockchain. In the first phase, data is collected from sensors and smart meters. The authors define three classes: outlier class, working day class, and holiday class. Due to the types of electricity consumption anomalies, the outlier class is also divided into three subclasses. The authors use KNN to group data in the classes. An anomaly is detected if the received data is beyond the normal range. In the last phase, the KNN algorithm is used to detect the anomalies which could not be detected in the previous phase.

As was mentioned, most of the papers focus on the energy consumption of customers. In addition, the lack of an explicit, efficient, and practical model for illegal miner detection, motivates us to propose a novel model based on pattern mining for this purpose.

## 3 Foundation and Main Concepts

This section introduces the main concepts and definitions of frequent itemset mining, data discretization, and itemset mining algorithms.

### 3.1 Concepts and Definitions

The following section introduces the most important concepts and definitions for frequent itemset mining [10, 19]:

**Definition 1.** An item represents an object. A set of items is called an item base. In this paper, based on data discretization, there are 35 items $I_i$, i=1...35, and the item base is $\Sigma = \{I_1, ..., I_{35}\}$.

**Definition 2.** A set of items together is called an itemset. Indeed, the itemset is a set of items that co-occur. If an itemset has k items, it is called a $k$-itemset. For example, $I_5I_{12}I_{31}$ is a 3-itemset.

**Definition 3.** Each transaction $T$ is composed of a transaction identifier $tid$ and a discretized instance $DIns$: T= $(tid, DIns)$. A transactional dataset $D$ is a set of all the transactions: $D = (T_1, \ldots, T_m)$.

**Definition 4.** Given the itemset $Y$ and the transaction $T = (tid, DIns)$, $T$ supports $Y$ if $Y \subseteq DIns$.

**Definition 5.** Given the itemset $Y$ and the transactional dataset $D$, a set of the transaction identifiers supporting $Y$ is called the cover of the itemset $Y$ in $D : K_D(Y) = \{i \in \{1, \ldots, m\} | T_i = (tid, DIns), Y \subseteq DIns\}$. For example, if the one transaction $T_2$ supports the itemset $Y$, then $K_D(Y) = \{2\}$.

**Definition 6.** The support of the itemset $Y$ in $D$ is the ratio of transactions in which an itemset appears to the total number of transactions: $Sup_D(Y) = \frac{|K_D(Y)|}{|D|}$.

**Definition 7.** If the support of an itemset $Y$ is not less than a given support threshold min_sup, $Y$ is a

**Algorithm 1** Apriori

**Input:** $D, min\_sup$ % $D$ is Tranactional data set; $min\_sup$ is the support threshold
**Output:** $L$ % Frequent itemsets in D

1: $L \leftarrow \varnothing$;
2: $L' \leftarrow \varnothing$; % frequent k-itemsets
3: $C' \leftarrow frequent\ 1-itemsets$; % candidate frequent k-itemsets
4: $k \leftarrow 1$; % The length of itemsets
5: **while** $(C' \neq \varnothing)$ **do**
6: $L' \leftarrow \varnothing$;
7: **for each** $(itemset \in C')$ **do**
8: $itemset.sup \leftarrow FindSupport(D, itemset)$;
9: **if** $(itemset.sup \geq min\_sup)$ **then**
10: $L' \leftarrow L' \cup itemset$;
11: $L \leftarrow L \cup itemset$;
12: **end if**
13: **end for**
14: $k \leftarrow k+1$;
15: $C' \leftarrow CandidateGeneration(L', k)$;
16: **end while**
17: return $L$;

**Figure 2**. Apriori Algorithm [19].

**Algorithm 2** FindSupport

**Input:** $D, X$ % $D$ is Tranactional data set; $X$ is an itemset
**Output:** $Sup$ % The support of itemset $X$ in D

1: $Sup \leftarrow 0$;
2: **for each** $(transaction\ T \in D)$ **do**
3: **if** $(X \subseteq T)$ **then**
4: $Sup \leftarrow Sup + 1$;
5: **end if**
6: **end for**
7: return $Sup/|D|$;

**Figure 3**. FindSupport Algorithm [19].

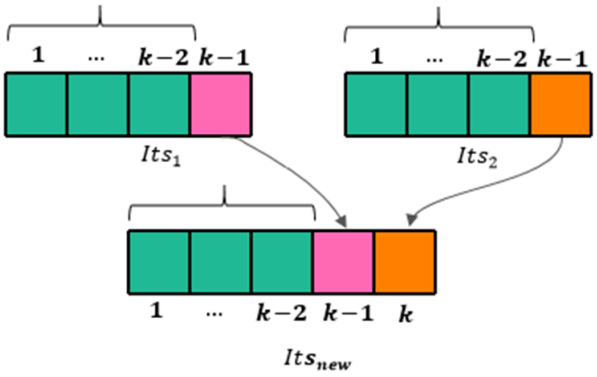


**Figure 4**. Generating the $k$-itemset $Its_{new}$ based on the join of the two frequent $(k-1)$-itemsets $Its_1$ and $Its_2$ [10].

frequent itemset: $Sup_D(Y) \geq min\_sup$.

### 3.2 Frequent Itemset Mining Algorithm

Apriori is one of the well-known algorithms for mining frequent item sets. The algorithm extracts $(k+1)$-itemsets from $k$-itemsets based on an iterative level-wise search technique [19]. Apriori is shown in Figure 2. As the figure shows, the input of the algorithm is a transactional dataset $D$ and a threshold $min\_sup$ and its output is $L$, a set of frequent itemsets in $D$.

Candidate frequent $k$-itemsets and frequent k-itemsets are stored in $C'$ and $L'$ respectively. Firstly, the algorithm finds the frequent 1-itemsets in line 3. The while loop in lines 5 to 16 extracts all the frequent itemsets of different lengths: The FindSupport function computes the support of each candidate frequent $k$-itemset in lines 7 to 13. The candidate itemsets whose support is not less than $min\_sup$, are added to $L'$ and $L$. It means that these itemsets are frequent. The CandidateGeneration function generates candidate frequent $(k+1)$-itemsets from frequent $k$-itemsets in lines 14 and 15.

The FindSupport and CandidateGeneration algorithms are shown in Figures 3 and 4 respectively. According to Figure 4, the FindSupport algorithm computes the support of the frequent itemset $X$ based on definition 6: In lines 2 to 6 of the function, for each transaction $T$ in $D$, where $X \subseteq T$, the support of $X$ increases by +1. As Figure 4 shows, the CandidateGeneration function joins frequent $(k-1)$-itemsets for generating $k$-itemsets: According to lines 4 to 17 of the function, if the first $k-2$ items of the two frequent $(k-1)$-itemsets $Its_1$ and $Its_2$ are identical, $Its_1$ and $Its_2$ join and the new $k$-itemset $Its_{new}$ is generated. Figure 5 shows how $Its_1$ and $Its_2$ join.

**Table 1**. The attributes recorded by AMI and their range.

| Attributes | Range | Abstraction alphabet |
|---|---|---|
| $Time$ | [0,24) | $I_1, I_2, I_3$ |
| $Power$ | $[0, \infty)$ | $I_4, I_5$ |
| $Cos\varphi$ | [0,1] | $I_6, I_7, I_8, I_9$ |
| $ReactivePower$ | $[0, \varphi)$ | $I_{10}, I_{11}$ |
| $Cos\varphi\ L1$ | [0,1] | $I_{12}, I_{13}, I_{14}, I_{15}$ |
| $Cos\varphi\ L2$ | [0,1] | $I_{16}, I_{17}, I_{18}, I_{19}$ |
| $Cos\varphi\ L3$ | [0,1] | $I_20, I_{21}, I_{22}, I_{23}$ |
| $Voltage\ L1$ | $[0, \varphi)$ | $I_{24}, I_{25}, I_{26}, I_{27}$ |
| $Voltage\ L2$ | $[0, \varphi)$ | $I_{28}, I_{29}, I_{30}, I_{31}$ |
| $Voltage\ L3$ | $[0, \varphi)$ | $I_{32}, I_{33}, I_{34}, I_{35}$ |

## 4 INBORN

According to data reported by Advanced Meter Infrastructure (AMI), each customer is sampled every 15 min. Table 1 shows the attributes recorded by AMI and their range. The trace of each customer is considered for 20 days. As Figure 1 shows, the empty entries of the dataset are filled first. Then, the dataset is discretized and frequent itemsets are extracted. The frequent itemsets summarize the behavior of each customer. Based on the frequent itemsets and their

**Algorithm 3** CandidateGeneration

**Input:** $L', k$ % $L'$ include frequent $(k-1)$-itemsets;
**Output:** $C'$ % candidate frequent $k$-itemsets

```
C' ← ∅;
Its1 ← ∅, Its2 ← ∅;        % Its1 and Its2 are (k − 1)-itemsets;
Itsnew ← ∅;                % Itsnew is a k-itemset;
for (i = 1 to |L'| − 1) do
    Its1 ← L'[i];
    for (j = i + 1 to |L'|) do
        Its2 ← L'[j];
        if (k = 2 or Its1[1..k − 2] = Its2[1..k − 2]) then
            if (k > 2) then
                Itsnew[1..k − 2] ← Its1[1..k − 2];
            end if
            Itsnew[k − 1] ← Its1[k − 1];
            Itsnew[k] ← Its2[k − 1];
            C' ← C' ∪ {Itsnew}
        end if
    end for
end for
return C';
```

**Figure 5**. CandidateGeneration Algorithm [19].

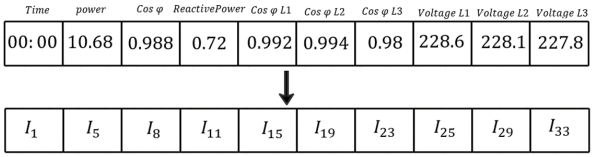


**Figure 6**. An example of converting an instance to the discretized instance.

**Table 2**. The frequent itemsets extracted from a customer with $min\_sup$=0.1.

| No. | Frequent Itemset | Sup |
|---|---|---|
| 1 | $I_5I_9I_{11}I_{15}I_{19}I_{23}I_{25}I_{29}I_{33}$ | 1 |
| 2 | $I_2I_5I_9I_{11}I_{15}I_{19}I_{23}I_{25}I_{29}I_{33}$ | 0.54 |
| 3 | $I_1I_5I_9I_{11}I_{15}I_{19}I_{23}I_{25}I_{29}I_{33}$ | 0.33 |
| 4 | $I_3I_5I_9I_{11}I_{15}I_{19}I_{23}I_{25}I_{29}I_{33}$ | 0.12 |

frequency, some rules are extracted. In the following subsections, the model is explained in more detail.

### 4.1 Data Filling and Discretization

Although AMI records all the attributes, some attributes' values might not be sent or received correctly. So, we encounter an incomplete dataset. According to the experts' viewpoint, the nearest sample after that is used to fill the missing values. After the data filling, the dataset should be discretized.

The discretization of continuous attributes is one of the most fundamental preprocessing methods because many data mining algorithms work on discrete spaces. Discretization divides the range of the attribute into some non-overlapping intervals. Thus, it reduces the number of values for continuous attributes. In other words, discretization helps us to investigate data at a macroscopic level instead of a microscopic level. However, selecting the number of intervals and deciding on their width are two key problems in this regard. In this paper, we divide the range of the attributes into some discrete values according to the experts' viewpoint. Indeed, the experts conceptually discretize the continuous-valued attributes. Then, an abstraction alphabet is defined for each interval. For example, $Voltage\ L1$ is split into the four intervals $I_{24}$:[170,207), $I_{25}$:[207,241.5], $I_{26}$:(241.5,253] and $I_{27}$:(253,+$\infty$). Table 1 shows the abstraction alphabet assigned to each attribute. So, it facilitates finding instances whose behavior is similar. An instance and its corresponding discretized instance are shown in Figure 6. As the figure shows, the numerical values of each attribute are mapped to the corresponding abstraction alphabet $I_i$,1≤i≤35.

### 4.2 Mining Frequent Itemsets

After data discretization, the frequent itemsets of each customer are extracted based on the algorithms explained in Section 3. For this purpose, we set $min\,$sup=0.1 to extract the common behavior. Table 2 shows the frequent itemsets extracted from a customer. To explain the structure of INBORN, we define some concepts as follows.

**Definition 8.** An itemset $X$ is absorbed by $Y$ if $X-Y$=$\varnothing$. For example, in Table 2, itemset 1 is absorbed by itemsets 2, 3, and 4.

**Definition 9.** An itemset $X$ is Maximal if there is no other itemset such as $Y$ that absorbs $X$. For example, itemsets 2, 3, and 4 are Maximal.

**Definition 10.** The set of the abstract alphabet $I_1$,$I_2$,$I_3$ is called Time Alphabet.

**Definition 11.** The behavior of a customer is called stable if removing the Time Alphabet from Maximal itemsets leads to the same itemsets. Table 3 shows that the behavior of the customer in Table 2 is stable.

**Definition 12.** The frequent itemsets whose Time Alphabet is $I_1$ are called TimeLowItemsets. The frequent itemsets whose Time Alphabet is $I_2$ and $I_3$ are called TimeMidItemsets and $TimeHighItemsets$ respectively.

Since in all the miner and non-miner cases, the voltage behavior in different phases is completely similar, it cannot be used to distinguish miners from non-miners. In addition, $Cos\varphi$ is the resultant of the behavior of $Cos\varphi L1$, $Cos\varphi L2$ and $Cos\varphi L3$. Therefore, the three attributes of $Power$, $Cos\varphi$, and $ReactivePower$ and the correlation between them are investigated in INBORN.

Table 4 shows these groups. According to the table, the three attributes $Power$,$Cos\varphi$, and

**Table 3**. The stable behavior of the customer in Table 2.

| No. | Frequent Itemset | Sup |
|---|---|---|
| 2 | $I_5I_9I_{11}I_{15}I_{19}I_{23}I_{25}I_{29}I_{33}$ | 0.54 |
| 3 | $I_5I_9I_{11}I_{15}I_{19}I_{23}I_{25}I_{29}I_{33}$ | 0.33 |

**Table 4**. The attributes groups considered in the time slots based on itemsets.

| No. | Attributes Group |
|---|---|
| 1 | $Power$ |
| 2 | $Power, Cos\varphi$ |
| 3 | $Power, Cos\varphi, ReactivePower$ |
| 4 | $Cos\varphi$ |
| 5 | $Cos\varphi, ReactivePower$ |
| 6 | $ReactivePower$ |
| 7 | $Power, ReactivePower$ |

$ReactivePower$ and their possible combinations are considered. For each of the time slot-based itemsets ($TimeLowItemsets$, $TimeMidItemsets$, and $TimeHighItemsets$), the highest support of the itemset including each attribute group is computed. Table 5 shows the attribute groups and their abstraction alphabet and support extracted from $TimeLowItemsets$ for a customer. In the next phase, based on the extracted rules and the support of the attribute groups, potential miners could be detected.

Figure 7 shows the procedure of the miner detection in INBORN in more detail. According to the figure, for the customers whose behavior is stable, the status of $Power$ and $Cos\varphi$ should be checked. According to the stable behavior of all the miners, the corresponding abstraction alphabets of $Power$ and $Cos\varphi$ are $I_5$ and $I_9$ respectively.

The other attributes don't have the same and consistent behavior. The miners whose behavior is not stale should be considered in the next phase. According to Figure 8, the customer's behavior is considered based on rules extracted from the frequent itemsets of the miners and the decision tree in two steps. Firstly, it is checked whether the abstraction alphabet of $Power$ is $I_5$ and its support is greater than the threshold $\alpha$ in $TimeLowItemsets$ and $TimeMidItemsets$.

If not, it means that the customer is not detected as a miner. Then, it is checked whether the highest support of the itemset including $Power$ and Cos $\varphi$ is greater than the threshold $\beta$ in $TimeMidItemsets$ and $TimeHighItemsets$. If not, the customer is not classified as a miner. In the third step, it is checked whether the abstraction alphabet of Cos $\varphi$ is $I_9$ in $TimeLowItemsets$, $TimeMidItemsets$ and $TimeHighItemsets$. If not, the customer is not the miner. Finally, the behavior of the customer is considered based on the decision tree. As the figure shows, in the tree, the highest support of the itemset including $Power$ and Cos $\varphi$ is checked in $TimeHighItemsets$ and $TimeMidItemsets$ respectively. Note that $\gamma$ and $\delta$ are the threshold values. Due to data security, we are not allowed to report the threshold values.

The rules extracted from the frequent itemsets show the abstraction alphabet of the attributes and the support of the itemsets are two significant factors for miner detection. On the contrary, the decision tree is constructed based on the support of the itemsets. In addition, the rules show that the behavior of the customer should be investigated in three-time slots.

**Table 5**. An example of the attribute groups extracted from $TimeLowItemsets$.

| Attributes Group | Abstraction Alphabet | support |
|---|---|---|
| $Power$ | $I_5$ | 1 |
| $Power, Cos\varphi$ | $I_5I_9$ | 1 |
| $Power, Cos\varphi, ReactivePower$ | $I_5I_9I_{10}$ | 0.84 |
| $Cos\varphi$ | $I_9$ | 1 |
| $Cos\varphi, ReactivePower$ | $I_9I_{10}$ | 0.84 |
| $ReactivePower$ | $I_{10}$ | |
| $Power, ReactivePower$ | $I_5I_{10}$ | 0.84 |

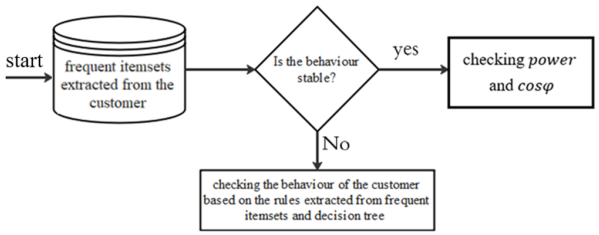


**Figure 7**. The procedure of the miner detection in INBORN.

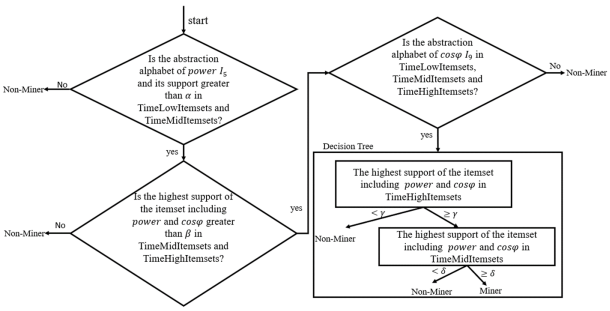


**Figure 8**. The procedure of the miner detection based on rules extracted from the frequent itemsets and the decision tree.

**Table 6**. The structure of the confusion matrix for binary classification [20, 21].

| Attributes Group | Predicted Negative | Predicted Positive |
|---|---|---|
| *PredictedPositive* | *TN* | *FP* |
| *ActualPositive* | *FN* | *TP* |

## 5 Evaluation

To evaluate the performance of INBORN, we compare it to the performance of the algorithm employed by the experts, called AlgExp. AlgExp compares the past energy consumption of the customer with the current amount's. We apply the two models to one real dataset. The dataset includes 58 customers who have the AMI system and have been collected from MPEDC. The number of miners and non-miners in the dataset is 41 and 17 respectively. In the following subsections, we introduce the metrics used for evaluation firstly. Then, the performance of INBORN and AlgExp are compared based on the metrics.

### 5.1 Evaluation Metrics

To compare the classification results, we use the confusion matrix, which is a common measure to evaluate classification problems. It can be applied to different classification problems (binary or multiclass). Table 6 shows the structure of a confusion matrix for binary classification, where "$TN$" shows the number of negative samples classified correctly. Similarly, "$TP$" is the number of positive samples classified correctly. The terms "$FP$" and "$FN$" are the number of actual negative samples classified as positive and the number of actual positive samples classified as negative respectively [20, 21].

Based on Table 6, some of widely used and popular metrics for classification evaluation are defined as follows [20, 22]:

$$Accuracy = \frac{TP + TN}{TN + TP + FN + FP} \quad (1)$$

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TN}{TN + FN} \quad (3)$$

$$specificity = \frac{TN}{TN + FP} \quad (4)$$

$$Sensitivity = \frac{TP}{TP + FN} \quad (5)$$

**Table 7**. The confusion matrix obtained from AlgExp.

| | predicted non-miner | Predicted miner |
|---|---|---|
| Actual non-miner | 0 | 17 |
| Actual miner | 0 | 41 |

**Table 8**. The confusion matrix obtained from INBORN.

| | predicted non-miner | Predicted miner |
|---|---|---|
| Actual non-miner | 12 | 5 |
| Actual miner | 0 | 41 |

Precision measures how good the model is at assigning positive samples to the positive class. That is, how accurate the miner prediction is. Recall measures how good the model is in detecting positive samples. The measures provide valuable information, but the objective is to improve recall without affecting precision. Sensitivity measures how apt the model is to detecting samples in the positive class. So, given that miners are a positive class, sensitivity quantifies how many of the actual miners are correctly predicted as miners. Specificity measures how exact the assignment to the positive class is, in our case, a miner label assigned to a customer. The classification accuracy is the ratio of the number of correct predictions to the total number of samples [23].

### 5.2 Experiment Results

Since we have few samples, leave-one-out cross-validation (LOOCV) is used to test INBORN. For this purpose, we have trained INBORN based on 57 customers and tested the trained models based on only one customer. We have done this work 58 times based on the different testing customers [24]. Tables 7 and 8 show the confusion matrices obtained from AlgExp and INBORN respectively. As the tables show, AlgExap classifies all the customers as a miner. It is worth mentioning that the customers predicted as a miner should be considered by authorized authorities on-site, which can lead to an increase in different costs. Therefore, in addition to the decrease of FN, FP should decrease for efficiency improvement.

On the contrary, not only does INBORN detect all the miners, but it also decreases FP dramatically. In addition, INBORN identifies the significant attributes and extracts the correlation between them. Thus, INBORN provides understandable results for the experts. The extracted patterns are detailed and could provide more precise results.

In Figure 9, AlgExp and INBORN are compared

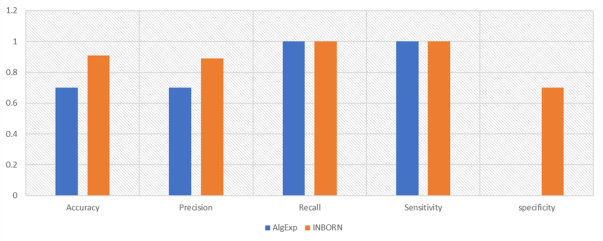


**Figure 9**. The comparison of INBORN with AlgExp in terms of different metrics.

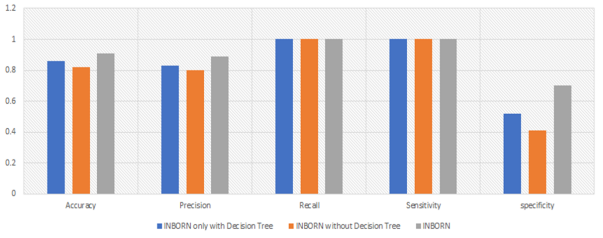


**Figure 10**. The investigation of INBORN with two different structures.

based on the different metrics. As the figure shows, the specificity of AlgExp is 0. It means that AlgExp is eager to classify most of the customers as miners. In other words, it is not good at distinguishing non-miners from miners.

Therefore, in the long term, there will be a lack of trust in the list of potential miners extracted by AlgExp. In addition, the precision and accuracy of INBORN are more than AlgExp's. When INBORN and AlgExp predict that a customer has some miners, it is correct around 89% and 70% of the time respectively.

As the experiment results show, INBORN could identify the customers who have some miners as well as it could distinguish between miners and non-miners very well and decrease the number of FP. Therefore, the results of INBOR are more reliable than AlgExp's. AlgExp models are a "black-box" method and cannot provide any insights into the environment patterns. Not only does INBORN provide accurate prediction results but it also explicitly extracts behavioral patterns independently of the fixed pattern length. The behavioral patterns are easily interpretable and summarize the behavior of similar customers and provide key insights into it.

We investigate the five customers who have been classified as miners by INBORN. The behavior of these customers is very similar to the customers who have some miners. Our studies show that the behavior of agricultural wells, spinning, and weaving factories, and electric equipment producing heat such as electric heaters is very similar to the customers who have some miners. Therefore, it seems that the factors related to energy consumption, alone, may not be sufficient to identify miners. Other factors such as the type of the customer and the history of its energy consumption could help distinguish between miners and non-miners more precisely.

**Table 9**. The confusion matrix obtained from INBORN only with the decision tree.

| Attributes Group | predicted non-miner | Predicted miner |
|---|---|---|
| Actual non-miner | 9 | 8 |
| Actual miner | 0 | 41 |

**Table 10**. The confusion matrix obtained from INBORN without the decision tree.

| | predicted non-miner | Predicted miner |
|---|---|---|
| Actual non-miner | 7 | 10 |
| Actual miner | 0 | 41 |

### 5.3 The investigation of INBORN with two different structures

As it was mentioned, INBORN uses a decision tree in the last phase of miner detection. To investigate the impact of the decision tree on the classification result, we employ INBORN on the dataset in three different situations: 1) only with the rules extracted by the decision tree, 2) without the rules extracted by the decision tree, and 3) both of the rules extracted by the frequent itemset mining and the decision tree. Tables 9 and 10 show the confusion matrices obtained from INBORN with different structures. As the tables show, although both of them could classify all of the miners successfully, none of them could distinguish the non-miners from the miners as INBORN does.

Figure 10 considers the importance of the patterns extracted by the different phases. The figure shows the classification results of INBORN with the different structures in terms of the metrics. As the figure shows, the rules extracted by the decision tree improve the performance of INBORN. The decision tree focuses on the frequency of the patterns in the different time slots. In addition, the results show that the rules extracted by the decision tree, alone, could not distinguish non-miners from miners. Therefore, not only are the occurring patterns very important, but their frequency is also a significant factor in distinguishing miners and non-miners.

### 5.4 INBORN in MPEDC

In MPEDC, except AlgExp, another system is used for miner detection. This system considers a linear

combination of some attributes. The weights of the attributes are empirically determined by the experts and the designers of the system. Indeed, no outstanding data mining techniques are used in this system. The system extracts a list of potential miners from the customers who have the AMI system every three days. The system assigns a probability to each customer, which is the probability of having some miners. The list might include the repeated customers who had been identified as a miner in previous periods. A large number of customers are usually detected as a miner by this system.

For example, in a period, the system identified 114 customers as a miner. It is clear that considering all of the 114 customers is very time-consuming. So, the experts selected 10 potential miners from the customers with a high probability randomly. These 10 customers were considered by authorized authorities on-site and none of them had any miners. To improve the efficiency and identify a more confident list of the potential number, INBORN was employed on the 114 customers. Among the 114 customers, INBORN only identified 13 customers. Considering the 13 customers showed that their behavior is very similar to the miners' behavior.

The experiment results show that INBORN is a practical model that can be used for miner detection successfully. It only considers the customer's behavior in recent weeks, extracts the behavioral patterns of the customers, and based on the occurring patterns in different time slots and their frequency decides whether to classify the customer as a miner or not.

It is worth mentioning that the current trend of the customers who have miners is similar to the trend of the potential miners predicted by INBORN. It shows that not only could INBORN predict the potential miners reliably, but it also provides the experts with useful information in a way that they focus on some specific types of customers.

## 6 Conclusions

The use of cryptocurrencies is becoming increasingly widespread. Due to the increase of illegal miners in some countries such as Iran, England, and China in recent years, illegal miner detection is essential to restrict them. For this purpose, for the first time, this paper proposes INBORN based on machine learning techniques to detect illegal miners. INBORN extracts all the behavioral patterns of the customers independently of the fixed pattern length. It focuses on unearthing the interesting trends or patterns of the energy consumption of the customers explicitly. INBORN investigates the correlation between different attributes and extracts the corresponding patterns in the different time slots. In addition, it investigates the frequency of the patterns and constructs a decision tree based on it. Thus, it considers and extracts the behavioral patterns of the energy consumption of the customers comprehensively. The patterns are readily interpretable by the experts. The experiment results reported on a real dataset show that methods used by power distribution companies are inadequate to detect illegal cryptocurrency mining farms. On the contrary, INBORN outperforms the common systems and algorithms used in power distribution companies and provides more reliable results.

In future work, we focus on proposing a new approach to adapt according to the energy consumption variations of the customers. For this purpose, we plan to investigate the capabilities of online learning and decrease the prediction error of INBORN. In addition, it seems that other significant factors are not related to energy consumption but can provide useful information for the improvement of the performance of INBORN.

## References

[1] J. Yli-Huumo, D. Ko, S. Choi, S. Park, and K. Smolander. Where is current research on blockchain technology?—a systematic review. *PLoS ONE*, 11(10), 2016. doi:10.1371/journal.pone.0163477.

[2] K. Christidis and M. Devetsikiotis. Blockchains and Smart Contracts for the Internet of Things. *IEEE Access*, 4:2292–2303, 2016. doi:10.1109/ACCESS.2016.2566339.

[3] B. Dindar and Ö. Gül. The detection of illicit cryptocurrency mining farms with innovative approaches for the prevention of electricity theft. *Energy & Environment*, 2021. doi:10.1177/0958305X211045066.

[4] S. Nakamoto. Bitcoin mining pools: A cooperative game theoretic analysis. *Decentralized Business Review*, 2008.

[5] Y. Lewenberg, Y. Bachrach, Y. Sompolinsky, A. ohar, and J. S. Rosenschein. Bitcoin mining pools: A cooperative game theoretic analysis. In *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems*, pages 919–927. Citeseer, 2015.

[6] C. Malmo. One Bitcoin Transaction Consumes As Much Energy As Your House Uses in a Week. *Vice (blog). November*, 1, 2017.

[7] C. Malmo. Bitcoin Mining and its Energy Footprint. In *25th IET Irish Signals & Systems Conference 2014 and 2014 China-Ireland International Conference on Information and Communications*

*Technologies*, page 280 – 285. IET, 2014. ISBN 978-1-84919-924-7. doi:978-1-84919-924-7.

[8] Cambridge Centre for Alternative Finance. Cambridge bitcoin electricity consumption index. `https://ccaf.io/cbeci/index`, Date Accessed: September 10, 2022.

[9] GlobalPetrolPrices. Iran electricity prices. `https://www.globalpetrolprices.com/Iran/electricity_prices/`, Date Accessed: September 10, 2022.

[10] M. Amiri, M. Hasanipanah, and H. Bakhshandeh Amnieh. Predicting ground vibration induced by rock blasting using a novel hybrid of neural network and itemset mining. *Neural Computing and Applications*, 32:14681–14699, 2020. doi:10.1007/s00521-020-04822-w.

[11] M. Amiri, L. Mohammad-Khanli, and R. Mirandola. A sequential pattern mining model for application workload prediction in cloud environment. *Journal of Network and Computer Applications*, 105:21–62, 2018. doi:10.1016/j.jnca.2017.12.015.

[12] M. Amiri, L. Mohammad-Khanli, and R. Mirandola. A new efficient approach for extracting the closed episodes for workload prediction in cloud. *Computing*, 102:141–200, 2020. doi:10.1007/s00607-019-00734-3.

[13] R. Jiang, R. Lu, Y. Wang, J. Luo, C. Shen, and X. Shen. Energy-theft detection issues for advanced metering infrastructure in smart grid. *Tsinghua Science and Technology*, 19(2):105 – 120, 2014. ISSN 1007-0214. doi:10.1109/TST.2014.6787363.

[14] A. Rahimi, A. Shahrestani, S. Ramezani, P. Zamani, S. O. Tehrani, and M. H. Y. Moghaddam. Filter Based Time-Series Anomaly Detection in AMI using AI Approaches. In *2021 5th International Conference on Internet of Things and Applications (IoT)*, pages 1–6. IEEE, 2021. ISBN 978-1-6654-4448-4. doi:10.1109/IoT52625.2021.9469717.

[15] C. Chahla, H. Snoussi, L. Merghem, and M. Esseghir. A deep learning approach for anomaly detection and prediction in power consumption data. *Energy Efficiency*, 13(8):1633–1651, 2020. doi:10.1007/s12053-020-09884-2.

[16] S. O. Tehrani, M. H. Y. Moghaddam, and M. Asadi. Decision Tree based Electricity Theft Detection in Smart Grid. In *2020 4th International conference on smart city, internet of things and applications (SCIOT)*, pages 46–51. IEEE, 2020. ISBN 978-1-7281-9611-4. doi:10.1109/SCIOT50840.2020.9250194.

[17] Z. Ouyang, X. Sun, J. Chen, D. Yue, and T. Zhang. Multi-View Stacking Ensemble for Power Consumption Anomaly Detection in the Context of Industrial Internet of Things. *IEEE Access*, 6:9623 – 9631, 2018. ISSN 2169-3536. doi:10.1109/ACCESS.2018.2805908.

[18] M. Li, K. Zhang, J. Liu, H. Gong, and Z.Zhang. Blockchain-based anomaly detection of electricity consumption in smart grids. *Pattern Recognition Letters*, 138:476–482, 2020. doi:10.1016/j.patrec.2020.07.020.

[19] R. Agrawal and R. Srikant. Fast Algorithms for Mining Association Rules in Large Databases. In *Proceedings of the 20th International Conference on Very Large Data Bases*, page 487–499. ACM, 1994. doi:10.1109/SCIOT50840.2020.925019.

[20] A. Kulkarni, D. Chong, and F. A. Batarseh. Foundations of data imbalance and solutions for a data democracy. *Data Democracy*, pages 83–106, 2020. doi:10.1016/B978-0-12-818366-3.00005-8.

[21] S. Visa, B. Ramsay, A. L. Ralescu, and E. Van Der Knaap. Confusion Matrix-Based Feature Selection. In *Proceedings of The 22nd Midwest Artificial Intelligence and Cognitive Science Conference*, pages 120–127, 2011.

[22] M. O'Reilly, J. Duffin, T. Ward, and B. Caulfield. Mobile App to Streamline the Development of Wearable Sensor-Based Exercise Biofeedback Systems: System Development and Evaluation. *JMIR rehabilitation and assistive technologies*, 4 (2):83–106, 2017. doi:10.2196/rehab.7259.

[23] Peter Bruce, Andrew Bruce, and Peter Gedeck. *Practical statistics for data scientists: 50+ essential concepts using R and Python*. O'Reilly Media, 2020.

[24] T. Wong. Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation. *Pattern Recognition*, 48(9):2839–2846, 2015. doi:10.1016/j.patcog.2015.03.009.

**Maryam Amiri** received the BS degree in computer engineering from Arak University, Arak, Iran, in 2009; the MS degree in computer engineering from the Bu-Ali Sina University, Hamedan, Iran, in 2012; the Ph.D. degree in computer engineering from the University of Tabriz, Tabriz, Iran, in 2018. She is currently an assistant professor in the Department of Computer Engineering, the faculty of engineering, Arak University, Arak, Iran. Her research interests include cloud computing, machine learning, and data mining.

**Hesam Askari** received a B.S. degree in computer engineering from Islamic Azad University of Arak, Arak, Iran. He is currently the head of the software group and the director of the Smart Monitoring Center in the Power Distribution Company of Markazi province. He is also an adviser of Tavanir company in the field of cryptocurrency.