electronics

MDPI

Article

Prompt Update Algorithm Based on the Boolean Vector Inner Product and Ant Colony Algorithm for Fast Target Type Recognition

Quan Zhou [1], Jie Shi [1], Qi Wang [2,*], Bin Kong [3,*], Shang Gao [2] and Weibo Zhong [1]

[1] Ocean College, Jiangsu University of Science and Technology, Zhenjiang 212003, China
[2] School of Computer, Jiangsu University of Science and Technology, Zhenjiang 212003, China
[3] Experimental Centre of Forestry in North China, Chinese Academy of Forestry, Beijing 102300, China
* Correspondence: wangqi@just.edu.cn (Q.W.); konbin@caf.ac.cn (B.K.)

Abstract: In recent years, data mining technology has become increasingly popular, evolving into an independent discipline as research deepens. This study constructs and optimizes an association rule algorithm based on the Boolean vector (BV) inner product and ant colony optimization to enhance data mining efficiency. Frequent itemsets are extracted from the database by establishing BV and performing vector inner product operations. These frequent itemsets form the problem space for the ant colony algorithm, which generates the maximum frequent itemset. Initially, data from the total scores of players during the 2022–2024 regular season was analyzed to obtain the optimal lineup. The results obtained from the Apriori algorithm (AA) were used as a standard for comparison with the Confidence-Debiased Adversarial Fuzzy Apriori Method (CDAFAM), the AA based on deep learning (DL), and the proposed algorithm regarding their results and required time. A dataset of disease symptoms was then used to determine diseases based on symptoms, comparing accuracy and time against the original database as a standard. Finally, simulations were conducted using five batches of radar data from the observation platform to compare the time and accuracy of the four algorithms. The results indicate that both the proposed algorithm and the AA based on DL achieve approximately 10% higher accuracy compared with the traditional AA. Additionally, the proposed algorithm requires only about 25% of the time needed by the traditional AA and the AA based on DL for target recognition. Although the CDAFAM has a similar processing time to the proposed algorithm, its accuracy is lower. These findings demonstrate that the proposed algorithm significantly improves the accuracy and speed of target recognition.

Keywords: target recognition; data mining; ant colony algorithm; vector inner product; apriori algorithm

check for updates

Citation: Zhou, Q.; Shi, J.; Wang, Q.; Kong, B.; Gao, S.; Zhong, W. Prompt Update Algorithm Based on the Boolean Vector Inner Product and Ant Colony Algorithm for Fast Target Type Recognition. *Electronics* **2024**, *13*, 4243. https://doi.org/10.3390/electronics13214243

Academic Editors: Suping Xu, Hengrong Ju, Keyu Liu and Guanfeng Liu

Received: 3 September 2024
Revised: 17 October 2024
Accepted: 26 October 2024
Published: 29 October 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The increasing complexity of the environment poses significant challenges in terms of object detection and necessitates shorter recognition times and greater accuracy for object detection. On specific target recognition platforms, target identification is usually based on detected target signal characteristics, matched with an established feature database for identification, and then, the target type is identified according to the differences in the target types of different platforms. For example, the classification and recognition method of non-cooperative objects based on Deep Learning (DL) is a target recognition method that uses DL to classify and recognize non-cooperative targets based on the micro-doppler effect and the laser coherent detection principle. The drawbacks of this method are as follows: (1) the feature database is generally extensive, leading to low matching and retrieval efficiency, affecting the response time, and (2) detection systems of different combat platform types are partially of the same type, and when multiple combat platforms are used for cooperative identification, contradictory results often occur when identifying

target types. Therefore, to identify target types quickly and accurately, researching and exploring new theories and methods is of significant practical significance.

In response to the problems concerning the target recognition platforms mentioned above, data mining was conducted using the radiation source database. This study introduces an association rule analysis method for data mining technology to improve the speed and accuracy of target recognition. The algorithm proposed in this article extracts frequent itemsets from the feature database of the detection system, identifies various target recognition rules, and forms a target recognition rule database to achieve rapid search for target types. In doing so, the algorithm improves the speed and accuracy of target recognition. Association rule mining [1], as an interdisciplinary field that integrates theories and techniques from databases, artificial intelligence, machine learning, statistics, and other domains, has become a research hotspot in database technology research and applications in recent years because of its ability to help decision-makers in various fields discover potential relationships among large database items. Among the many association rule mining algorithms proposed, the Apriori algorithm (AA) [2], introduced in 1993 by R. Agrawal, Imielinski, and Swami, is the most famous. In recent years, there have been many developments in AA research. In 2022, Meng Chen and Zhi Xiang Yin utilized the algorithm for cardiotocography classification by integrating it with a multi-model ensemble classifier, showcasing its effectiveness in medical data analysis [3]. Around the same time, Jian Zeng and Bao Jia applied the AA to real-time data mining and penalty decision-making in basketball games [4], highlighting its versatility in sports analytics. In 2023, Chen Rumeng and colleagues developed a hypergraph-clustering method based on an improved AA [5], demonstrating its ability to handle complex data structures.

Additionally, Fulin Li and his team used an enhanced version of the algorithm to mine equipment quality information, expanding its application in industrial sensor data analysis [6]. That same year, Lin Wei Li and his team employed the optimized AA for deformation response analysis of landslide hazards [7], making an innovative application in natural disaster research. As of 2024, the algorithm continues to evolve, with Dasgupta Sarbani and Saha Banan integrating it with DL for drug recommendations in big data environments [8–13], significantly contributing to its use in the healthcare sector. Furthermore, Xie R. and his team introduced a cognitively Confidence-Debiased Adversarial Fuzzy Apriori Method (CDAFAM) [14], incorporating fuzzy logic and adversarial learning, bringing new perspectives on the use of the algorithm. Meanwhile, Yan Yiping applied an improved version of the AA to develop psychological crisis behavior models, extending its application to psychology and human–computer interactions. These developments have collectively advanced the AA, demonstrating its adaptability and increasing its relevance across diverse fields by addressing complex data analysis challenges. These latest research achievements enrich the study of the AA in theory and demonstrate its enormous potential and prospects in practical applications.

With the continuing growth of informationization in many industries, massive amounts of data have accumulated. Because of the large scale, inconsistent structure, and diverse sources of data, traditional data analysis methods have challenges in efficiently extracting valuable information. When there is a large amount of data, the complexity of the traditional AA exponentially increases and the running efficiency significantly decreases. The existing improved AAs all have varying degrees of defects, either generating fewer rules or generating too many useless, redundant iterations. Because of the fact that BV only contains 0 and 1, it can significantly improve computational speed, especially on large-scale datasets. The ant colony algorithm has the characteristics of parallel computing and positive feedback and has a fast convergence speed, meaning that it has strong advantages in terms of solving combinatorial optimization problems. This paper proposes a new association rule mining algorithm that introduces BV and the ant colony algorithm into the AA to effectively compensate for the shortcomings of the AA, which requires multiple database scans, and improve the quality and efficiency of association rule mining. In order to target data from different sources, this article adopts NBA professional arena data, disease and

symptom data, and radar data for data mining. The results show that the new algorithm can not only mine more valuable information, such as the best lineup of NBA teams, but it can also identify data types faster and more accurately, such as judging diseases based on symptoms and identifying radiation source types based on radar features.

This study introduces the association rule analysis method for data mining to address the issues concerning the long recognition time and low accuracy in target recognition systems. By discretizing the target feature database to construct a BV and using an improved AA based on the BV and the ant colony algorithm [15–20], frequent itemsets in the target recognition feature database are extracted and used to form a new target recognition database. Constructing the database as a BV and utilizing the inner product of vectors can quickly identify frequent one-itemsets, satisfying the minimum support degree and eliminating miscellaneous items that clearly cannot form frequent itemsets. However, to obtain the maximum frequent itemset, it is still necessary to calculate line by line and continuously search until the end. Therefore, the proposed algorithm is combined with the ant colony algorithm, utilizing the excellent global search ability of the ant colony algorithm to find the most frequent itemset. Compared with previous algorithms, the proposed algorithm first eliminates some miscellaneous items and optimizes the database. Second, it only needs to scan the database once, significantly improving the efficiency and accuracy of the algorithm.

Section 2 of this study relates to the background knowledge needed to improve the algorithm, and Section 3 proposes the fast update algorithm based on the BV inner product and ant colony algorithm. Section 4 relates to the simulations and the analysis of the results, and this paper is finally concluded in Section 5.

2. Background Knowledge

In this section, we explore several key components of association rule mining, including the basic concepts of association rules, the implementation of the Apriori algorithm (AA), the construction of a Boolean matrix (BM), the application of vector inner product, and the ant colony algorithm. The division of these topics into different subsections is intended to clearly articulate the specific methods and functions of each step. First, the definition of association rules and their key metrics provide a theoretical foundation for understanding subsequent algorithms. Next, the AA details how to mine frequent itemsets efficiently, a process that relies on the construction of a BM to represent data relationships effectively. The vector inner product offers an efficient method for performing calculations on the BM, thereby accelerating the discovery of frequent itemsets. Finally, the ant colony algorithm introduces a metaheuristic approach to further enhance the search for optimal frequent itemsets.

2.1. Association Rule

An association rule [21–25] is a data mining method primarily used to discover interesting and frequent patterns, associations, or causal structures among different items in large datasets.

The main steps in association rule mining include the following:

(1) Generate frequent itemsets: Use algorithms to identify itemsets whose support exceeds the minimum support threshold.

(2) Generate association rules: Extract rules from the frequent itemsets that meet the minimum confidence threshold.

(3) Evaluate and filter Rules: Assess the usefulness of the rules based on metrics like confidence and lift and filter out the most valuable rules.

Association rule analysis focuses on the following key metrics:

Support: The support of a rule is defined as the proportion of data items in the dataset that contain all the items in the rule.

$$Support(A \rightarrow B) = \frac{COUNT(A \cup B)}{COUNT(A)} \quad (1)$$

Confidence: Confidence refers to the probability that B occurs given that A has occurred, used to measure the reliability of the rule.

$$Confidence(A \rightarrow B) = \frac{Support(A \cup B)}{Support(A)} \quad (2)$$

Using these metrics, the strength and reliability of association rules can be evaluated. An item is a frequent item if its support is greater than or equal to a specified minimum support threshold. This indicates that the itemset appears frequently enough in the dataset.

Association rule mining has a wide range of practical applications, including market basket analysis, recommendation systems, medical data mining, and telecommunications fields. In the latest research, data mining has also been applied to control and disaster prevention, such as an intelligent decision support system for groundwater supply management and electromechanical infrastructure controls [26] and enhancing flood risk mitigation by an advanced data-driven approach [27].

2.2. Apriori Algorithm

Association rules can help uncover hidden relationships within data. We must rely on algorithms to automate mining these rules from large datasets effectively. In this regard, the AA plays a crucial role [28–32]. A classic association rule learning algorithm uses an iterative approach to generate frequent itemsets and then extracts applicable association rules from them. Its fundamental idea is to reduce the search space for computing frequent itemsets through joining and pruning.

The entire execution process of the AA mainly includes the following steps:

(1) The algorithm calculates the occurrence frequency of all items and determines the frequent one-itemsets.

(2) The algorithm iteratively generates longer frequent itemsets using the joining and pruning steps based on the currently found frequent itemsets. This process continues until no longer frequent item sets can be found.

The advantage of the AA lies in its simplicity and ease of implementation. However, it also has some drawbacks, such as the need to scan the entire database during each iteration and the potentially time-consuming generation of many candidate itemsets when the dataset is large.

2.3. Boolean Matrix Construction

In data mining, a BM is an important data structure used to represent the presence or absence of relationships between items. Constructing a BM is foundational for association rule mining and frequent itemset mining tasks. The specific steps for constructing a BM are as follows:

Determine the matrix dimensions: First, determine the rows and columns of the BM. Rows typically represent the transactions in the dataset, while columns represent all possible items.

Fill the matrix: Next, iterate through each transaction in the dataset. For each item in the transaction, if the transaction contains the item, enter 1 in the corresponding matrix cell to indicate presence; otherwise, enter 0 to indicate absence.

This article uses disease symptoms and patient data as examples, where the process of constructing a BM is as follows:

Suppose there are three radar data transactions, with metrics including carrier frequency value, pulse width value, and amplitude value.

Transaction 1 (Patient 1): fever, fatigue, headache.

Transaction 2 (Patient 2): fever, fatigue, headache.

Transaction 3 (Patient 3): fever, cough.

Transaction 4 (Patient 4): asymptomatic

The obtained database is shown in Table 1.

Table 1. Patient data.

Number	Fever	Cough	Fatigue	Headache
1	1	0	1	1
2	1	0	1	1
3	0	1	1	0
4	0	0	0	0

Construct the corresponding BM R using transactional database data, as shown in Figure 1.

Number	Symptom
1	Fever, Fatigue, Headache
2	Fever, Fatigue, Headache
3	Cough, Fever
4	N

$$R=\begin{bmatrix}1 & 0 & 1 & 1\\ 1 & 0 & 1 & 1\\ 0 & 1 & 1 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}$$

Figure 1. Generated BM.

2.4. Vector Inner Product

The Boolean vector (BV) inner product [33–37] is calculated by performing a logical AND operation on the corresponding elements of two BVs and then summing the results. It is used to find frequent itemsets mainly because of its high computational efficiency, which accelerates data processing. The vector inner product operation quickly calculates the degree of association between itemsets. Especially in a BM, the vector inner product can quickly determine the cooccurrence count of two itemsets. The basic steps for using the vector inner product to find frequent itemsets are as follow:

Represent transactions: First, convert each transaction into an n-dimensional BV, where n is the number of all distinct items. If the transaction contains a specific item, the corresponding vector element is 1; otherwise, it is 0.

Compute the inner product: To calculate the two itemsets, perform an inner product calculation using their corresponding BV. The result of the vector inner product is the number of times these two itemsets appear together in all transactions. The specific calculation formula is

$$a \times b = \sum a_i b_i \tag{3}$$

Identify frequent itemsets: Compare the co-occurrence count of each itemset with a user-defined minimum support threshold, which can determine the frequent itemsets. If the co-occurrence count of an itemset is greater than or equal to the minimum support threshold, then the itemset is considered frequent.

We use the BM constructed from the previous radar signal data in the following example:

The vector representation for Radar Signal 1 is M = [1,0,1], and the vector representation for Radar Signal 2 is B = [1,0,0]. The inner product of these two vectors is calculated as follows:

$$M \times B = [1,0,1] \times [1,0,0] = 1 \tag{4}$$

If this value is greater than or equal to the minimum support requirement, then the itemset {Radar Signal 1, Radar Signal 2} can be considered frequent.

The advantages of the inner product operation of the BV are mainly reflected in the following aspects:

Simplify complex problems: Through logical operations, complex problems can be decomposed into simple logical judgments, making the problem-solving process more intuitive and easier to understand.

Efficient processing: Because of the binary nature of Boolean operations, they exhibit extremely high efficiency in handling discrete quantities and logical problems, making them suitable for large-scale data processing and computation.

Widely used: Boolean algebra not only has important applications in computer science and electronic technology, such as defining Boolean values with integers in C language, where 0 represents false and non-zero represents true, but it also has extensive applications in fields such as graphics and cryptography, as seen with the use of Boolean operations in graphics processing software.

Logical clarity: The use of Boolean algebra makes logical relationships clearer and more concise, which helps improve problem-solving efficiency and accuracy.

2.5. Ant Colony Algorithm

The ant colony algorithm [38] is a metaheuristic search algorithm inspired by the behavior of ants seeking food. It simulates the process of ants releasing pheromones and choosing paths while searching for food, using cooperation and information sharing to solve combinatorial optimization problems.

The basic idea of the algorithm is that multiple virtual ants randomly search the solution space and release pheromones based on the quality of the paths they find. Other ants prefer to choose paths with higher pheromone concentrations, reinforcing the advantageous paths. Over time, paths with high pheromone concentrations become more attractive to the ants, leading more ants to choose those paths and eventually causing the algorithm to converge to an optimal solution.

The basic steps of the ant colony algorithm are as follows:

(1) Initialization:

Initialization of pheromones: Initialize pheromone values on each path in the search space. In general, the pheromone values on all paths are the same. Pheromones represent the "goodness or badness" of a path, and the concentration of pheromones affects the probability of ants choosing a path.

Heuristic information: For certain problems (such as TSP), heuristic information (such as distance, cost, etc.) can be introduced to guide ants in choosing paths.

(2) Ant construction solution:

Each ant constructs a solution in the search space. Ants will determine their path based on the current concentration of pheromones and heuristic information such as path length. Ants tend to choose paths with higher concentrations of pheromones and better heuristic information.

(3) Calculate path fitness:

After the ant constructs the solution, calculate the fitness of each solution. The higher the fitness, the better the solution. Evaluate the solution (path) constructed by ants based on the objective function (such as the total path length).

(4) Update pheromone:

Local pheromone update: During the process in which ants construct solutions, the number of pheromones along the path gradually increases as the ants pass by. The usual rule for updating pheromones is that the higher the chosen path, the higher the concentration of pheromones.

Global pheromone update: After all ants complete their searches, update pheromones based on their concentrations. At this point, the pheromones on the path will be readjusted according to fitness. A good path will leave more pheromones, while a poor path will evaporate more fragrance.

(5) Volatile pheromones:

Pheromones will gradually evaporate over time. This volatilization process can prevent the search from falling into local optima too early, thereby increasing the global search capability.

(6) Iteration:

Through repeated iterations, ants constantly explore paths, gradually update pheromones, and converge to the optimal solution. After each iteration, the best solution is recorded and used to guide the next round of the search.

(7) Termination conditions:

When the stopping conditions are met (such as reaching the maximum number of iterations, finding a solution that is good enough, or the pheromone changes tend to stabilize), the algorithm terminates.

The AA searches for frequent itemsets as a global search process, while the ant colony algorithm can search for the global optimal path. Therefore, the improved algorithm transforms the association rule mining problem into the ant colony algorithm, solving the traveling salesman problem. Given that the fundamental element of the traveling salesman problem is the target city, all frequent items in the database are used as the target city for mapping, which is the problem space of the ant colony algorithm. In the traveling salesman problem, the optimal solution evaluation criterion for ants to traverse the graph once is to find the "shortest path." In the improved algorithm, the optimal solution evaluation criterion is to find the frequent itemsets with the maximum support.

3. Proposed Algorithm

The Apriori algorithm (AA) adopts a layer-by-layer search iteration method, and its complexity is mainly manifested in the first step of accessing the transaction item set. The running efficiency is significantly reduced when the number of items is large. The existing improved AAs all have varying degrees of defects, generating fewer rules or too many useless, redundant iterations. Therefore, the proposed algorithm utilizes the construction of Boolean matrices (BMs) and the results of Boolean inner product operations to continuously search for the maximum frequent itemset of each row vector, reducing the number of scans. The ant colony algorithm is introduced into the new proposed algorithm. It has substantial advantages in solving combinatorial optimization problems because of its parallel computing, positive feedback characteristics, and fast convergence speed. The proposed algorithm can effectively compensate for the shortcomings of the AA and improve the quality and efficiency of association rule mining.

The AA searches for frequent itemsets as a global search process, while the ant colony algorithm can search for the global optimal path. Therefore, the proposed algorithm transforms the association rule mining problem into an ant colony algorithm, solving the TSP problem. Given that the fundamental element of the TSP problem is the target city, all frequent items in the database are used as the target city to establish a complete graph, which is the problem space of the ant colony algorithm. In the TSP problem, the optimal solution evaluation criterion for ants to traverse the complete graph once is to find a "shortest path", while the optimal solution evaluation criterion in the proposed algorithm is to find the frequent itemset with "maximum support".

3.1. The Apriori Algorithm Improved with Vector Inner Product Generates Frequent 1-Itemsets

One-itemsets are a part of frequent itemsets [39–41] in data mining. In association rule mining, one-itemsets are frequent itemsets that only contain a single item. The proposed algorithm reconstructs the BM and adds rows to store intermediate computation results. Applying each row vector's inner product with itself identifies frequent one-itemsets that meet the minimum support threshold. The algorithm is described as follows:

Step 1: Scan the transaction database to construct the BM $D = \left\{ I_1^n, I_2^n, I_3^n \ldots, I_n^n, I_{n+1}^n \right\}$, and arrange it according to the number of 1s in each row vector. The last row I_{n+1}^n of matrix

D is an all-zero row used to store the results of the previous step's computations. Set the minimum support threshold to min_support.

Step 2: Calculate the inner products for the first n row vectors of the BM *D*. Delete the rows in *D* where the sum of the inner product is less than the min_support and identify the frequent one-itemsets.

Step 3: Output all frequent one-itemsets that meet the minimum support threshold.

3.2. Association Rule Mining Algorithm Based on the Ant Colony Algorithm

Next, an undirected graph with all frequent items in the database as locations is established, which represents the problem space for the ant colony algorithm. An ant traversing the graph once to find the optimal path is equivalent to finding the frequent itemset with the highest support. The steps include the following:

(1) Construct an undirected graph.

Step 1: Calculate the number n of all frequent items based on the proposed AA using Boolean vectors (BVs) and construct an undirected graph G with n vertices.

Step 2: Use all frequent one-itemsets as the vertices of this undirected graph G.

Step 3: Calculate the support of the itemsets formed by any two vertices in graph G and use the reciprocal of the support value as the distance between the two points.

Step 4: Use the constructed graph G as the problem space for the ant colony algorithm to mine association rules, as shown in Figure 2. *A* to *F* in the figure represent vertices.

(2) Generate Frequent Itemsets.

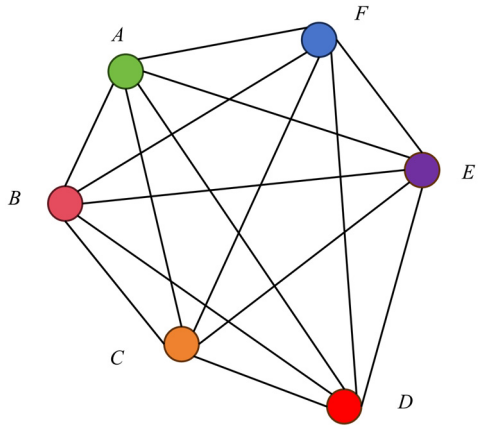


Figure 2. Problem space.

Randomly select m ants to start from different vertices and create parameters. Y_1 and $Y_1 \cdot Y_2$ represent the number of iterations and Y_2 represents the ant's path. The initial value of Y_1 is 0, and the initial value of Y_2 is 1. For each ant, k, calculate the next item j to reach based on the following transition probability formula:

$$P_{ij}^{k}(t) = \begin{cases} \dfrac{\tau_{ij}^{\alpha}\eta_{ij}^{\beta}}{\sum\limits_{s=allowed_k} \tau_{ij}^{\alpha}(t)\eta_{ij}^{\beta}(t)}, j \in allowed_k \\ 0, other \end{cases} \tag{5}$$

where $\tau_{ij}(t)$ represents the pheromone concentration on the path from point i to point j at time t. The value of τ_{ij} on each path is within the range $[\tau_{\min}, \tau_{\max}]$, and the initial value is set as $\tau_{ij}(0) = \tau_{\max}$. η_{ij} represents the heuristic value from point i to point j, which equals the reciprocal of the support value between the two vertices. α and β represent the influence of pheromone and heuristic information on the ant's decision, respectively. $allowed_k$ represents the next city that ant k can choose.

List the selected city j in the ant's tabu list and determine whether the path, after adding j, contains frequent itemsets, including item j. If it does, extract the frequent itemset; otherwise, remove j from the path. When $Y_2 < n$, increment Y_2 by 1, and continue to select nodes to make the ant traverse the entire path. When $Y_2 \geq n$, the ant completes traversing all paths and obtains the frequent item set with the maximum support using this cycle. The pheromone update rule is as follows:

$$\tau_{ij}(t+1) = \rho \cdot \tau_{ij}(t) + \tau_{ij}^{best},\ \Delta\tau_{ij}^{best} = \frac{1}{L^{best}} \tag{6}$$

where $\Delta\tau_{ij}^{best}$ represents the pheromone increment on the path from point i to point j. If the path from i to j does not belong to the optimal solution, its value is 0. ρ represents the pheromone residual coefficient.

If Y_1 < maximum number of iterations, increment Y_1 by 1, and the ants continue to traverse. When Y_1 > maximum number of iterations, output all frequent itemsets obtained from this cycle. The flowchart of this algorithm is shown in Figure 3.

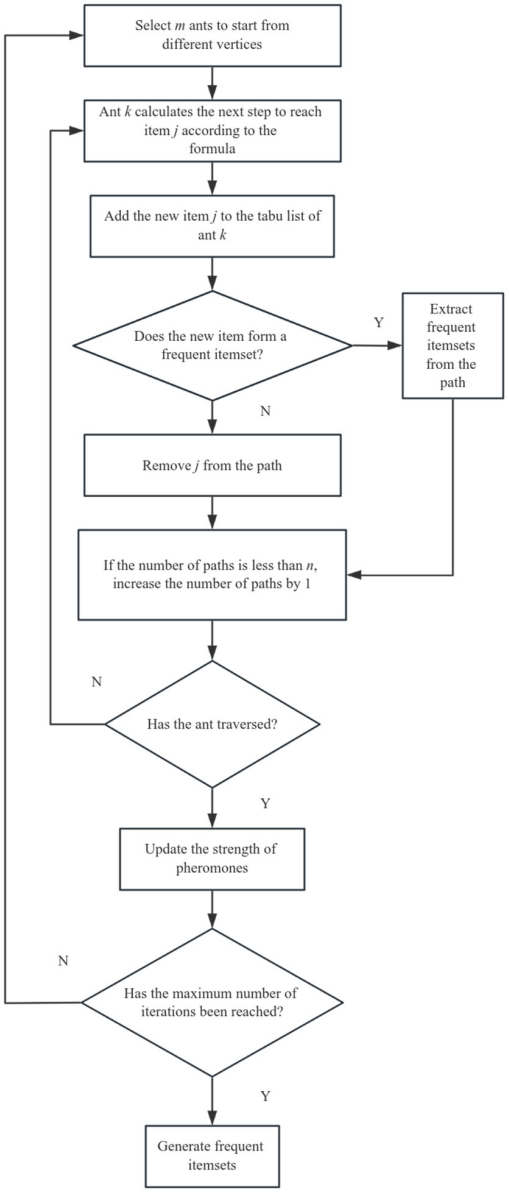


Figure 3. Proposed algorithm flowchart.

The proposed algorithm pseudocode is shown in Figures 4 and 5:

```
Step 1 : Calculate frequent itemsets
// Calculate all frequent 1-itemsets and record their support
frequent_itemsets = Apriori_Algorithm(D, min_support)
n = number of frequent 1 - itemsets  // Calculate the number of frequent 1-itemsets

// Construct an undirected graph G
G = EmptyGraph(n)  // Build an empty graph with n vertices, each representing a frequent 1-itemset

Step 2: Use frequent 1 - itemsets as the vertices of graph G
// Each frequent 1-itemset becomes a vertex in graph G
for each frequent_itemset x ∈ frequent_itemsets :
G.addVertex(x)

Step 3 : Calculate support for itemsets formed by any two vertices in graph G
// For every pair of vertices in the graph G, calculate the support for the itemset formed by them
// Use the reciprocal of the support value as the distance between the two vertices
for each pair of vertices(v1, v2) in G :
    itemset = v1 ∪ v2  // Combine two vertices into an itemset
    support = calculate_support(itemset, D)  // Calculate the support of the itemset
    distance = 1 / support  // Use the reciprocal of the support as the distance between the vertices
    G.addEdge(v1, v2, distance)  // Add an edge between v1 and v2 with weight as distance (reciprocal support)

    Step 4: Use graph G as the problem space for the ant colony algorithm to mine association rules
    pheromone = InitializePheromone(G)  // Initialize pheromone on the graph G
    for iteration = 1 to max_iterations :
for ant = 1 to num_ants :
// Each ant constructs a path in the graph G and computes the fitness of the path
path = ConstructPath(G, pheromone, alpha, beta)
fitness = CalculateFitness(path, D)  // Calculate the fitness of the path, typically strength of the association rules

// Update pheromone based on the path taken by the ant and its fitness
pheromone = UpdatePheromone(pheromone, path, fitness, rho)

// Return the best association rules
return BestRules(G, pheromone)
```

Figure 4. Proposed algorithm pseudocode part 1.

```
Initialize:minsup, minconf, Nc_max, m, α, β, ρ
L_1 = find_frequent_1_itemsets(D)
f(Item_i, Itemj) = CheckSup(dataset, {i, j}), ∀i, j∈L_1
Create matrix TSP consisting off(Item_i, Item_j)

Start:
    m = random(TSP), m < n
    for (k = 0; k < m; k++) {
        j = next item(k)//Select next item based on probability P_ij^k(t)
            add j to tabu_k

        if (fqset ⊆ path and j ∈ fqset) {
                add fqset to L_k
        }
        else {
                add j to forbid
        }

        if ({ i,j } ∈ L_best) {
            update τ_ij = τ_ij + 1
        }

        if (iteration_count ≥ NC_max) {
            goto Exit
        }
    }

Exit:
    for each L_i ∈ L_k{
        if (support(L_k) / support(L_i)≥ minconf) {
            print L_i => (L_k - L_i)
            }
    }
```

Figure 5. Proposed algorithm pseudocode part 2.

The pseudocode shown in Figures 4 and 5 combines the AA for frequent itemset minimization with the ant colony optimization algorithm to solve an optimization problem. It first uses the BV inner product to remove frequent terms that cannot form a frequent itemset, identifies frequent itemsets from the dataset, and then uses a probabilistic approach to build potential solutions (paths) iteratively by selecting items and updating pheromones to guide future selections. The algorithm continues until a stopping condition (maximum iterations) is met, at which point, it evaluates the support and confidence of the discovered itemsets to generate association rules.

The proposed algorithm utilizes the BV to convert data into 0 and 1, reducing computational complexity. Through the use of the inner product, frequent items in the database are reduced, effectively reducing the vertices in the problem space used by the ant colony algorithm and improving its efficiency. Utilizing the advantage of only scanning the database once with an ant colony algorithm significantly reduces the time required for data mining.

4. Simulation Results

This section introduces the Apriori algorithm (AA), the Confidence-Debiased Adversarial Fuzzy Apriori Method (CDAFAM), the AA based on Deep Learning (DL), and the proposed algorithm, along with experimental evaluations on three different datasets. The experiments focus on assessing the accuracy and runtime of the algorithm results. The results are analyzed to determine the reasons for the performance differences among the four algorithms, demonstrating the superiority of the proposed algorithm.

4.1. Experimental Setup and Dataset Preparation

- The hardware deployed in this study was an NVIDIA GeForce RTX 3070 GPU and 16 GB of VRAM. The hardware configuration of the detection platform was as follows: Loongson 3A6000 CPU, DDR5 6000 CL30 16GBx2 RAM, and 2 TB of storage space. The software environment included the Windows operating system, Pytorch framework (PyTorch 1.6.0), Python 3.7, Anaconda3-2021.05-Windows-x86_64, and MATLAB 2022.
- For the algorithms used in the experiment, we employed the AA, CDAFAM, AA based on DL, and an optimized fuzzy-based FP-growth algorithm (OFBFPGA) [42]; however, because all data types are 0 and 1, the data were already clear and there was no ambiguity. Therefore, we only compared the experimental results of the AA, CDAFAM, and AA based on DL algorithms.
- Three publicly available datasets and one non-public dataset were selected. Data on the coordinated scoring of ten regular rotation players of the NBA Lakers in the 2002–2023 [43] and 2023–2024 [44] seasons were selected as pick and roll, assist, cover, rebound, and other forms. The data source was the well-known website, www.basketballreference.com, provided by the sports data analysis company Sports Radar. The third dataset was the Disease Symptoms and Patient Profile Dataset, sourced from Kaggle and supplied by the Google Database [45]. The fourth dataset was a radar simulation database, simulated from real radar signals; this part was subject to a data usage agreement signed with the provider and thus cannot be disclosed currently. The reason for choosing the NBA dataset as the experimental object is that data mining techniques are rarely applied in the sports industry. By mining and analyzing a large amount of game data, some less intuitive information can be obtained to assist coaches in arranging game lineups and making scientific decisions on lineups and tactics during specific game stages. The reason why symptom and disease datasets were selected as experimental objects is that internet medicine has become a new direction in the medical field, and many AAs proposed in recent years have used medical industry datasets as experimental objects. Therefore, selecting symptom and disease datasets as experimental subjects can more intuitively demonstrate the advantages of the proposed algorithm. The reason for choosing the radar dataset is that data mining is a completely new field in the field of radar recognition. By discovering hidden patterns, rules, and knowledge from a large amount of radar

data, it is possible to predict, classify, cluster, and perform other operations on radar signals. Therefore, this algorithm has broad application potential and practical effects in target recognition.

4.2. Data Analysis

4.2.1. Processing the Lakers Player Data from the 2022–2024 NBA Season

The regular season data on the Lakers for the 2022–2024 season are shown in Tables 2 and 3.

Table 2. Player data for the 2022–2023 season.

Player	Age	G	GS	MP	FG	FGA	FG%	3P	...
LeBron James	38	55	54	35.5	11.1	22.2	0.5	2.2	...
Anthony Davis	29	56	54	34	9.7	17.2	0.563	0.3	...
D'Angelo Russell	26	17	17	30.9	6.3	13	0.484	2.7	...
Dennis Schröder	29	66	50	30.1	4.1	9.8	0.415	1.1	...
...	...	...	...	...	...	...	...	...	...
Sterling Brown	27	4	0	6	0	1	0	0	...
Cole Swider	23	7	0	5.9	0.4	1.3	0.333	0.4	...
Scotty Pippen Jr.	22	6	0	5.3	0.7	2	0.333	0.2	...
Davon Reed	27	8	0	3.4	0.4	0.5	0.75	0.1	...

Table 3. Player data for the 2023–2024 season.

Player	Age	G	GS	MP	FG	FGA	FG%	3P	...
Anthony Davis	30	76	76	35.5	9.4	16.9	0.556	0.4	...
LeBron James	39	71	71	35.3	9.6	17.9	0.54	2.1	...
D'Angelo Russell	27	76	69	32.7	6.5	14.2	0.456	3	...
Austin Reaves	25	82	57	32.1	5.6	11.5	0.486	1.9	...
...	...	...	...	...	...	...	...	...	...
Alex Fudge	20	4	0	3.5	0.3	1.5	0.167	0	...
Dylan Windler	27	8	0	3.5	0.5	1.1	0.444	0.5	...
Maxwell Lewis	21	34	0	3	0.1	0.6	0.19	0	...
Harry Giles	25	7	0	2.7	0.1	0.9	0.167	0	...

The AA was compared with the proposed algorithm. The association analysis results were validated and compared to demonstrate the feasibility and superiority of the proposed algorithm for data mining.

The data were obtained via processing, as shown in Tables 4 and 5; N represents a vacancy.

Table 4. Player cooperation score data for the 2022–2023 season.

	Play1	Play2	Play3	Play4	Play5
1	Reaves	James	Westbrook	Beverley	N
2	Davies	James	Walker	Westbrook	N
3	Westbrook	James	Davies	N	N
4	Reaves	Westbrook	N	N	N
...	...	...	...	...	...
3300	James	Hachimura	N	N	N
3301	James	Davies	N	N	N
3302	Hachimura	James	Davies	Russel	N

Table 5. Player cooperation score data for the 2023–2024 season.

2023–2024	Play1	Play2	Play3	Play4	Play5
1	Davies	Russel	N	N	N
2	Prince	James	N	N	N
3	Prince	James	N	N	N
4	James	N	N	N	N
...	...	...	...	...	...
3571	James	Davies	Russel	N	N
3572	James	Davies	Reaves	N	N
3573	Reaves	James	Russel	N	N

The minimum support was set to 0.1, and the minimum confidence was set to 0.6. The results of data mining on the dataset are shown in Tables 6 and 7.

Table 6. The 2022–2023 season 5 frequent itemsets.

	Antecedent	Consequent	Confidence
1	Schröder, Brown, James, Davis	Reaves	0.622
2	Reaves, Schröder, James, Brown	Davis	0.822
3	Brown, Reaves, Schröder, Davis	James	1.000
4	Schröder, Reaves, Brown	James, Davis	0.730
5	Schröder, Reaves, Davis	James, Brown	0.679

Table 7. The 2023–2024 season 5 frequent itemsets.

	Antecedent	Consequent	Confidence
1	Russell, Reaves, James, Davis	Hachimura	0.726
2	Reaves, Russell, James, Hachimura	Davis	0.876
3	Russell, Reaves, Hachimura, Davis	James	1.000
4	Russell, Reaves, Hachimura	James, Davis	0.769
5	Reaves, Hachimura, Davis	James, Russell	0.815

The results show that, in the 2022–2023 season, when Brown, Reaves, Schröder, and Davis were all on the court simultaneously, James was also on the court. The combination of these five players cooperates in terms of technique and tactics to achieve a 100% probability of scoring an attack. When Brown, Reaves, Schröder, and James were all on the court simultaneously, Davis was also on the court, and the probability of these five people completing the offensive score through technical and tactical coordination was 82.2%. When Brown, Reaves, and Schröder were all on the court, James and Davis were also on the court, and the probability of these five people working together to complete an offensive score was 73%. In the 2023–2024 season, when Russell, Reaves, Hachimura, and Davis were all on the court at the same time, James was also on the court, and the combination of these five players plays with each other's technical and tactical coordination achieved a 100% probability of scoring an offensive goal. When Russell, Reaves, Hachimura, and James were all on the court simultaneously, Davis was also on the court, and the probability of these five people completing the offensive score through technical and tactical coordination was 83.6%. When Russell, Reaves, and Hachimura were all on the court, James and Davis were also on the court, and the probability of these five people working together to complete an offensive score was 76.5%. Based on the above results, according to this conclusion, coaches can select different player combinations to play at specific game stages.

Next, the proposed algorithm was used for data mining. Firstly, a BM was established using 2022–2023 data, with a record of 1 for participating in the attack and 0 for not

participating. The support threshold for the inner product of the Boolean vector (BV) in the proposed algorithm was set to 180, and 1-frequent terms more significant than the minimum support were obtained. The results are shown in Table 8.

Table 8. The 2022–2023 season 1-frequent itemsets.

Number	Frequent Items
1	James
2	Davis
3	Westbrook
4	Schröder
5	Reaves

The minimum confidence level was set to 0.4. The ant colony algorithm had 20 ants, one pheromone importance, two heuristic information importance, 0.1 pheromone volatility, and 100 iterations. The results are shown in Table 9.

Table 9. Experimental results of the proposed algorithm for the 2022–2023 season.

	Antecedent	Consequent	Confidence
1	Brown, Reaves, Schröder, Davis	James	1.000
2	Reaves, Schröder, James, Brown	Davis	0.822
3	Reaves, Schröder, Brown	Davis, James	0.730
4	Reaves, Schröder, Davis	James, Brown	0.679
5	Schröder, James, Brown, Davis	Reaves	0.622

A BM was established using 2023–2024 data, with a record of 1 for participating in an attack and 0 for not participating. The support threshold for the inner product of the BV in the improved algorithm was set to 200, and one-frequent terms more significant than the minimum support were obtained. The results are shown in Tables 10 and 11.

Table 10. The 2023–2024 season 1-frequent itemsets.

Number	Frequent Items
1	Davis
2	James
3	Russell
4	Reaves
5	Prince
6	Hachimura

Table 11. Experimental results of the proposed algorithm for the 2023–2024 season.

	Antecedent	Consequent	Confidence	Pheromone
1	Russell, Reaves, Hachimura, Davis	James	1.000	0.262
2	Reaves, Russell, James, Hachimura	Davis	0.876	0.211
3	Russell, Reaves, Hachimura	James, Davis	0.815	0.192
4	Reaves, Hachimura, Davis	James, Russell	0.769	0.183
5	Russell, Reaves, James, Davis	Hachimura	0.726	0.174

Experiments using the AA based on DL and the CBAFA were conducted, and their results were compared. The results are shown in Tables 12 and 13. Table 12 represents the time spent by each algorithm, and Table 13 represents the similarity between each algorithm and the best lineup determined by the AA, that is, based on the best lineup determined by the AA.

Table 12. Comparison of select the best team time.

Competition Season	Apriori Algorithm	Confidence-Debiased Adversarial Fuzzy Apriori Method	Apriori Algorithm Based on Deep Learning	Proposed Algorithm
2022–2023	1.51 (s)	0.44 (s)	1.42 (s)	0.27 (s)
2023–2024	1.57 (s)	0.45 (s)	1.55 (s)	0.34 (s)

Table 13. Similarity to the experimental results of the Apriori algorithm.

Competition Season	Confidence-Debiased Adversarial Fuzzy Apriori Method	Apriori Algorithm Based on Deep Learning	Proposed Algorithm
2022–2023	33%	100%	100%
2023–2024	25%	100%	100%

The comparison results are shown in Figures 6 and 7.

As shown in the figure, the time required for the AA and the AA based on DL are similar. The CDAFAM takes approximately 25% of the time required for the first two algorithms, while the proposed algorithm takes approximately 20% of the time required for the first two algorithms.

We compared the best lineups obtained using the AA with those obtained by the other three algorithms. The optimal lineups obtained via the AA based on DL and the proposed algorithm were consistent with those obtained using the AA, while the CDAFAM obtained three sets of best lineups in the 2022–2023 season and four sets of best lineups in 2023–2024, which is obviously inconsistent with the facts. The results indicate that although the AA based on DL yields the same results as the Apriori algorithm, there is not much improvement in time. At the same time, the CDAFAM generates too many results despite being fast. The research results of the proposed algorithm are consistent with the AA, and because of fewer database scans, the proposed algorithm is significantly better than the other three algorithms.

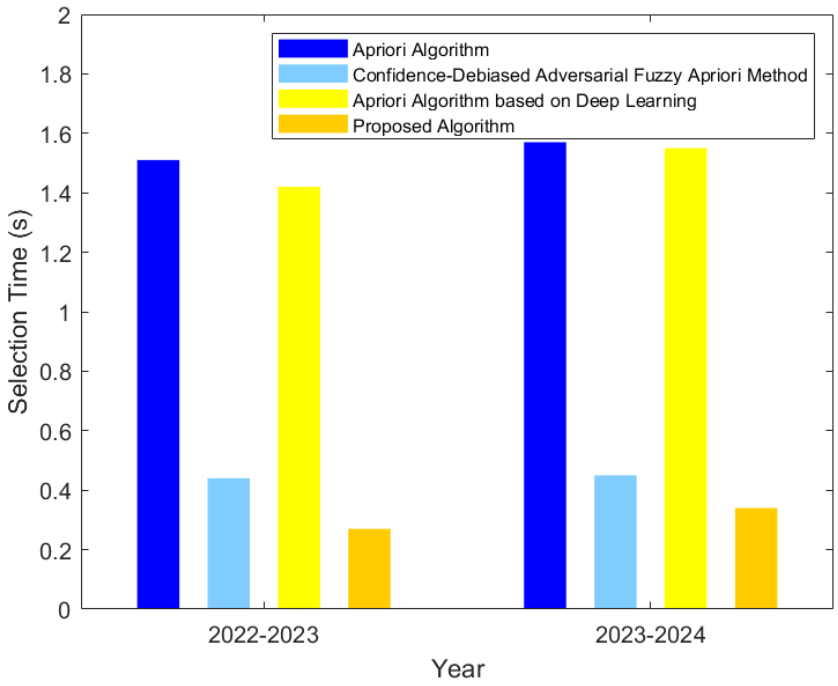


Figure 6. Selection time.

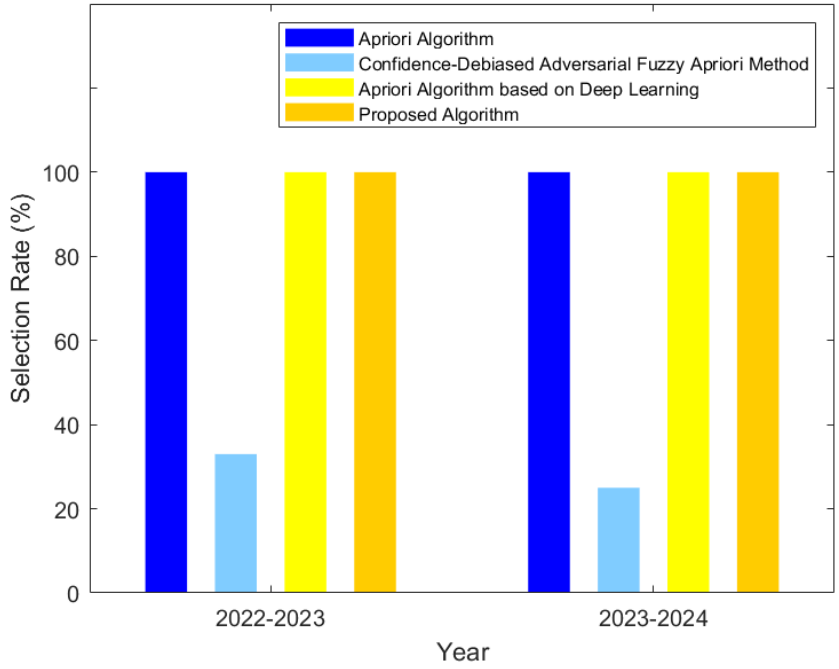


Figure 7. The Apriori algorithm's similarity to the experimental results.

The threshold of pheromones mainly determines the number of association rules mined by the proposed algorithm. The smaller the threshold of pheromones set, the more rules are obtained. As shown in Figure 8, the number of pheromones reflects the strength of the dependency between two frequent terms.

Changing the minimum support can alter the number of algorithm-generated association rules. Lower support means that the rules may only appear by chance, and most are meaningless. In the case of higher support, the improved algorithm and AA have similar reactions to the rules in the dataset, as shown in Figure 9. When reducing the support of the generated rules, the improved algorithm can effectively avoid generating many rules with lower support, and the regulations mined in this way are more practical.

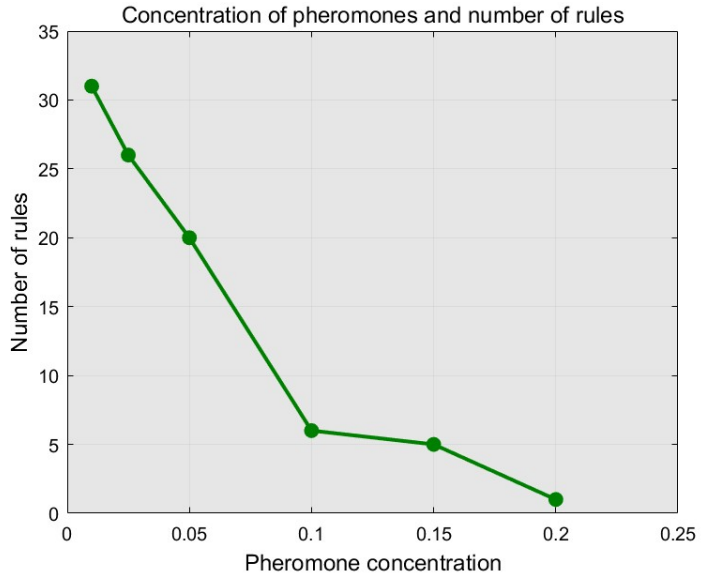


Figure 8. The impact of pheromones on association rules.

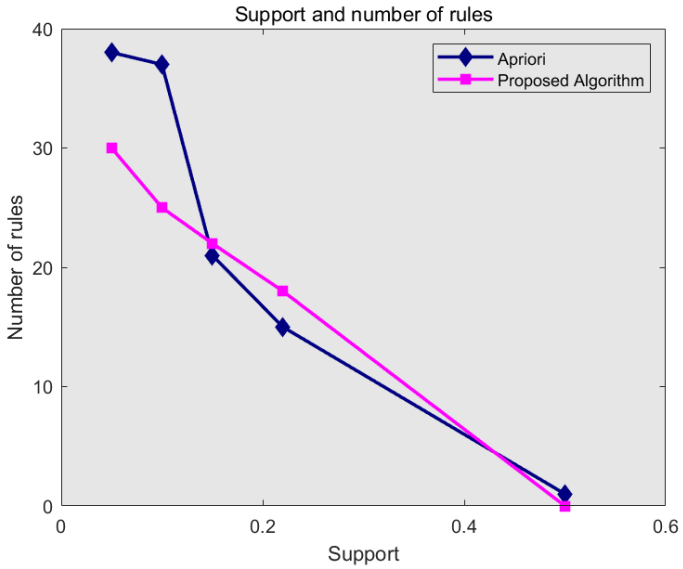


Figure 9. The impact of support on rules.

4.2.2. Disease Symptoms and Patient Data Processing

This dataset from Kaggle can help develop disease diagnosis or monitoring prediction models based on symptoms and patient characteristics. This article first selected five symptom datasets from 100 diseases in the dataset to form a disease feature database. Then, 10,000 patients' symptom transaction sets were chosen randomly, and their diseases were determined through data mining. The results were then compared with the diseases they suffer from. Finally, we compared the accuracy and time of the four algorithms. The feature dataset is shown in Table 14. The test dataset is shown in Table 15.

Table 14. Disease feature dataset.

Disease	Fever	Cough	Fatigue	Dyspnea	Lumbago	Headache	...
Influenza	1	0	1	1	0	1	...
Influenza	1	1	1	1	0	0	...
Influenza	0	1	1	1	0	1	...
Influenza	1	1	0	0	1	0	...
...	...	...	...	...	...	...	...
Asthma	1	1	0	1	0	0	...
Asthma	1	1	0	1	0	0	...
Asthma	0	1	0	0	0	0	...
Asthma	1	1	0	0	0	0	...

Table 15. Symptom dataset.

Number	Fever	Cough	Fatigue	Dyspnea	Lumbago	Headache	...
1	1	0	1	1	0	1	...
2	1	0	1	1	0	0	...
3	0	1	1	0	0	1	...
4	0	0	0	0	1	1	...
...	...	...	...	...	...	...	...
9997	1	1	0	0	0	0	...
9998	1	0	0	1	0	0	...
9999	0	1	1	0	1	0	...
10,000	0	1	1	1	0	1	...

The BV threshold was set to 2, and the minimum support was set to 0.6. The ant colony algorithm had 2000 ants, one pheromone importance, two heuristic pheromone importance, 0.1 pheromone fluctuation, and 100 iterations. We compared the results generated by the four algorithms mentioned earlier. The algorithm time is shown in Table 16, and the algorithm accuracy is shown in Table 17. Respectively, they represent the accuracy of the algorithm in diagnosing diseases based on symptoms and the time it takes.

Table 16. Comparison of diagnosis time.

Apriori Algorithm	Confidence-Debiased Adversarial Fuzzy Apriori Method	Apriori Algorithm Based on Deep Learning	Proposed Algorithm
0.554 (s)	0.132 (s)	1.62 (s)	0.141 (s)

Table 17. Comparison of diagnosis rate.

Apriori Algorithm	Confidence-Debiased Adversarial Fuzzy Apriori Method	Apriori Algorithm Based on Deep Learning	Proposed Algorithm
79.4%	66.6%	88.3%	87.6%

The comparison results are shown in Figures 10 and 11. Figure 10 shows the accuracy of disease diagnosis, and Figure 11 shows the time taken for diagnosis.

The results indicate that the AA based on DL has the highest accuracy. Still, it is less than 1% higher than the proposed algorithm but takes much more time than other algorithms. The CDAFAM has the lowest time and accuracy. Overall, the proposed algorithm balances recognition accuracy and recognition time, with a 10% increase in accuracy and an approximately 74% reduction in time compared with the AA.

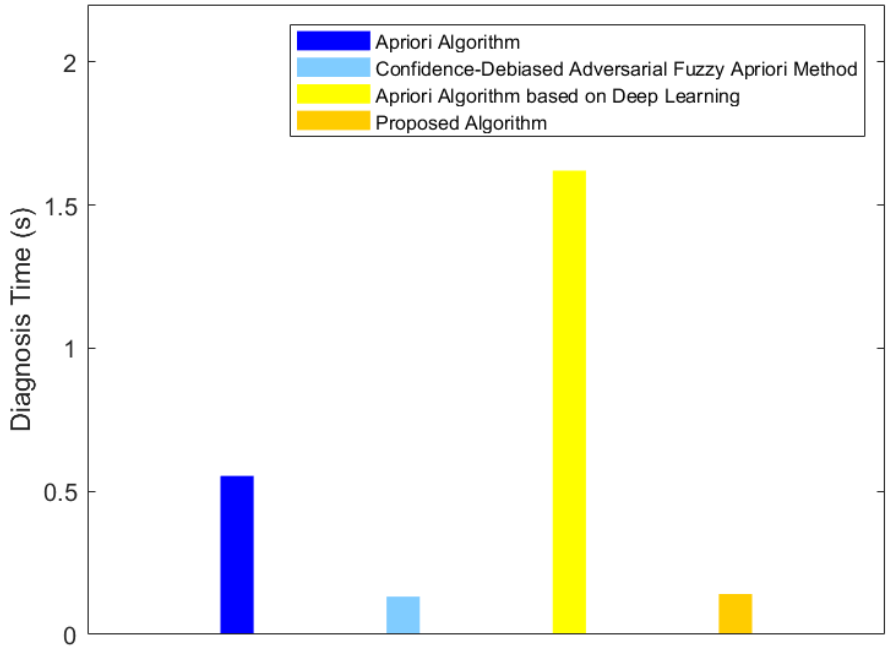


Figure 10. Diagnosis time.

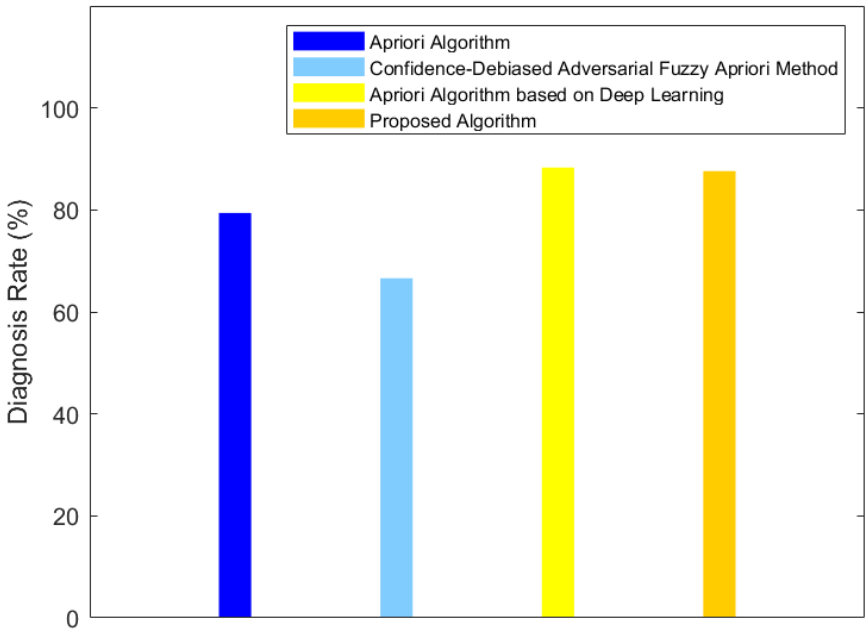


Figure 11. Diagnosis rate.

4.2.3. Radar Simulation Data Processing

The algorithm used simulated radar signal statistical data from a laboratory as the test dataset to mine frequent itemsets.

First, the data were discretized. The algorithm test selected the mean carrier frequency, pulse width, amplitude, heading angle, mean pulse repetition frequency, scan period, angle of arrival, true azimuth, and chord azimuth as targets from the sixteen pieces of hexadecimal statistical data. These targets were then discretized into different intervals and finally mapped to their respective Boolean attributes. The algorithm was then applied to solve the problem. The attributes included in dataset D and their corresponding partial values are shown in Table 18.

Table 18. Radar simulation data.

Number	Mean Carrier	Frequency	Pulse Width Value	Amplitude Value	Mean Pulse Repetition Frequency	Scan Period	...
1	e	b	4	e	5	0	...
2	4	4	3	a	5	c	...
3	7	2	6	3	6	5	...
4	2	e	6	3	6	5	...
...	...	...	...	...	...	...	...
1197	5	f	3	0	3	9	...
1198	6	3	6	3	6	f	...
1199	4	3	6	f	6	e	...
1200	6	d	6	c	6	e	

According to the distribution of each attribute value, each attribute was divided into two parts with a BV of 0 and 1. The part that exceeded the average value was marked as 1, and the other part was marked as 0. Table 17 shows the partial discretization results of the data in Table 19.

Table 19. Mapping BM.

Number	Mean Carrier	Frequency	Pulse Width Value	Amplitude Value	Mean Pulse Repetition Frequency	Scan Period	...
1	1	1	0	1	0	0	...
2	0	0	0	1	0	1	...
3	1	0	1	0	1	0	...
4	0	1	1	0	1	0	...
...	...	...	...	...	...	...	...
1197	0	1	0	0	0	1	...
1198	1	0	1	0	1	1	...
1199	0	0	1	1	1	1	...
1200	1	1	1	1	1	1	

First, 100 sets of 1200 sets of detection data were randomly generated based on the performance of each of the 12 selected detection platforms. The BV threshold was set to 1, and the minimum support was set to 0.6. The ant colony algorithm had 60 ants, one pheromone importance, two heuristic pheromone importance, 0.1 pheromone fluctuation, and 100 iterations. Then, data mining methods were used to establish recognition association rules for each detection platform based on signal feature types, selecting five association rules with the highest confidence for each section, for a total of 60 rules. Finally, the proposed algorithm was compared with the three algorithms proposed in the previous section, and the results are shown in Tables 20 and 21. The most standard answer is the radar raw data type, and the accuracy is the radar type correctly identified by the algorithm.

Table 20. Comparison of target recognition time.

Target Platform Type	Number of Detection Platforms	Apriori Algorithm	Confidence-Debiased Adversarial Fuzzy Apriori Method	Apriori Algorithm Based on Deep Learning	Proposed Algorithm
p1	2	8.74 (ms)	1.22 (ms)	5.42 (ms)	1.17 (ms)
p2	3	10.94 (ms)	1.73 (ms)	8.34 (ms)	1.63 (ms)
p3	2	9.71 (ms)	1.99 (ms)	9.66 (ms)	2.11 (ms)
p4	2	8.98 (ms)	2.39 (ms)	10.19 (ms)	2.6 (ms)
p5	3	12.65 (ms)	3.42 (ms)	14.35 (ms)	3.11 (ms)

Table 21. Comparison of target recognition rate.

Target Platform Type	Number of Detection Platforms	Apriori Algorithm	Confidence-Debiased Adversarial Fuzzy Apriori Method	Apriori Algorithm Based on Deep Learning	Proposed Algorithm
p1	2	82.46%	69.47%	95.38%	95.22%
p2	3	81.93%	68.88%	94.33%	93.65%
p3	2	81.20%	68.66%	94.88%	94.21%
p4	2	79.07%	69.11%	91.87%	92.07%
p5	3	79.73%	66.35%	91.69%	92.73%

The comparison results are shown in Figures 12 and 13. Figure 12 shows the recognition rate, and Figure 13 shows the recognition time.

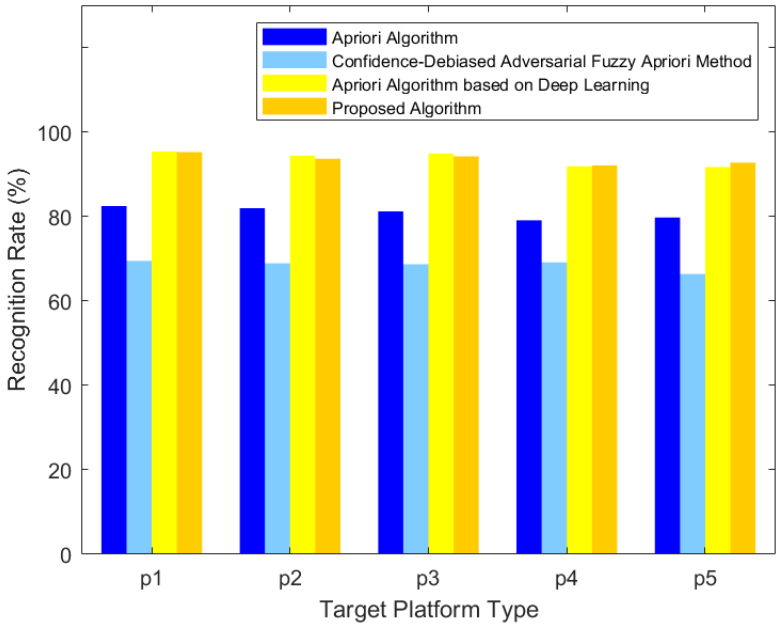


Figure 12. Recognition rate.

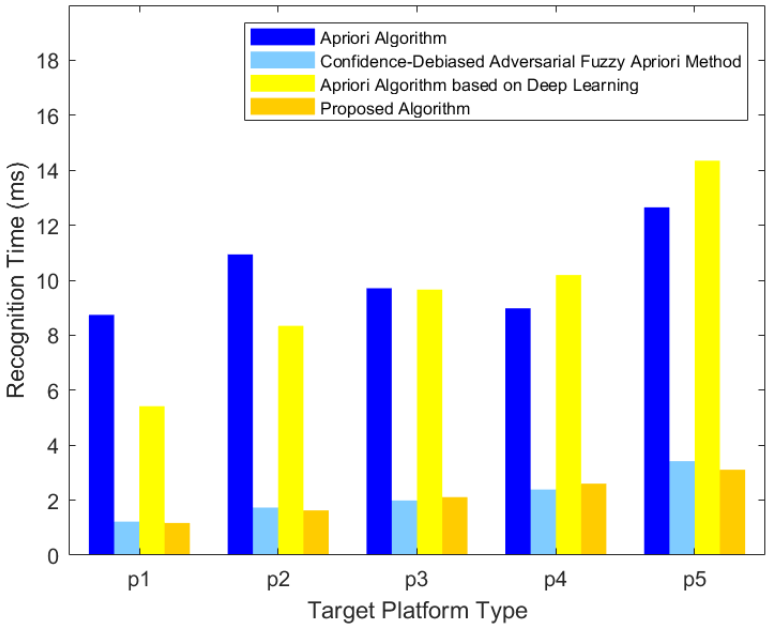


Figure 13. Recognition time.

According to the results, the accuracy of the AA based on DL and the proposed algorithm is comparable, with an increase of about 12% compared with the AA and an increase of about 25% compared with the CDAFAM. In terms of time, the CDAFAM is comparable to the proposed algorithm, accounting for approximately 20% of the AA. The AA based on DL, because of the content of the data, sometimes takes more time than the AA and sometimes takes less time than the AA. The proposed algorithm takes a short time and has high recognition accuracy.

Next, we performed statistical analysis on the radar simulation data experimental results using the Friedman test and post-Holm test to analyze the differences between the methods used.

We calculated the average accuracy of the four algorithms, as shown in Table 22.

Table 22. Average accuracy.

Apriori Algorithm	Confidence-Debiased Adversarial Fuzzy Apriori Method	Apriori Algorithm Based on Deep Learning	Proposed Algorithm
80.88%	68.49%	93.63%	93.58%

First, a Friedman test was performed to determine if there is a statistically significant difference among the adopted methods according to the average testing accuracies in Table 22. The statistical value of Friedman's test was 13.56, corresponding to a *p*-value of 0.00357. Because the *p*-value is less than the significance level of 0.05, it means that the recognition performance of the algorithm used has a statistically significant difference.

Second, a post-Holm test was used to compare the proposed algorithm with all comparison methods. Table 23 lists the *p*-values and the statistical magnitudes obtained using the post-Holm test. Clearly, there is a statistically significant difference between the proposed algorithm and all the comparative algorithms. As a result, the Friedman test with the post-Holm test statistically indicates the effectiveness of the proposed algorithm in improving radar signal recognition capability.

Table 23. *p*-value table based on the post-Holm test.

Algorithm	Difference in Average Rank	*p*-Value	Null Hypothesis
Apriori Algorithm vs. Proposed Algorithm	1.4	0.00119	Rejected
Confidence-Debiased Adversarial Fuzzy Apriori Method vs. Proposed Algorithm	2.4	0.00178	Rejected
Apriori Algorithm based on Deep Learning vs. Proposed Algorithm	−0.2	0.00357	Rejected

The main reason for the difference in the performance of several algorithms based on the above results is roughly that the CDAFAM reduces uncertainty through confidence value adjustment and fuzzy logic, which can quickly process data. However, because of its debiasing and fuzzy processing strategies, some valuable patterns may be blurred, especially in low-noise data, resulting in a decrease in recognition accuracy. However, it has an advantage in processing speed because fuzzy processing typically reduces overly complex calculations. The AA based on DL combines the traditional AA with DL. The AA is good at discovering association rules; however, its search space is large, and its computational complexity is high. DL further enhances its pattern recognition ability, especially in complex data situations where it can extract more accurate rules. However, DL requires a large amount of computation, especially for training models, so its running time is relatively long. This method prioritizes accuracy and sacrifices runtime. The proposed algorithm achieves rapid matching and pattern recognition through the inner product operation of the BV, while the ant colony algorithm performs well in path optimization. The BV inner product itself is computationally simple, and the operation of handling 0s and 1s is very efficient. The ant colony algorithm can quickly find the optimal solution through multiple iterations. The advantage of combining the two is that it can quickly process large amounts of binary data and effectively identify patterns in the data, thereby achieving high-precision recognition. This method achieves dual optimization of accuracy and speed by reducing redundant calculations and introducing intelligent optimization strategies.

5. Conclusions

This study presents an enhanced version of the traditional Apriori algorithm (AA) by integrating a Boolean vector (BV) inner product and an ant colony optimization approach. The enhanced algorithm was tested on diverse datasets from the medical, sports,

and military domains. The results indicate that the proposed algorithm outperforms the traditional AA in terms of time efficiency and accuracy, demonstrating significant potential for applications in object recognition. The reliance of the traditional AA on pairwise comparisons across all datasets leads to considerable delays in response times. In contrast, the proposed algorithm streamlines the process by limiting comparisons to the generated rules, thereby enhancing both speed and accuracy in target identification. Additionally, it effectively addresses the necessity for multiple database scans inherent in the traditional AA. The incorporation of ant colony optimization further facilitates a dynamic exploration of the itemset space, allowing for the adaptive adjustment of pheromone levels to optimize the search trajectory. This innovation enables the discovery of longer frequent itemsets without the exhaustive generation of all possible candidates. When applied to the three diverse data types, the algorithm demonstrated distinct advantages, including a broad range of applicability and accelerated mining speeds, positioning it favorably for potential applications in various unexplored fields.

However, certain limitations remain. The necessity to construct a comprehensive graph of the support for all frequent subsets requires the imposition of a support threshold on selected rules. Without this constraint, the rapid expansion of the problem space due to a high number of attributes in frequent sets can lead to substantial space consumption. Future research should focus on refining the determination of experimental parameters and developing strategies for compressing the problem space to further improve algorithmic efficiency in scenarios with numerous attribute columns. The proposed algorithm can also be combined with other algorithms and applied to new fields, for instance, integration with text mining [46] techniques could enhance sentiment analysis, public opinion management, and keyword extraction, among other areas.

Author Contributions: Q.Z. was responsible for primary writing, data preparation, experimental reasoning, and revisions. Q.W., B.K., W.Z., J.S. and S.G. provided technical and writing method guidance as instructors. Q.W. and B.K. were responsible for revising this paper; they are the corresponding authors. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within this article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Lu, M.; Zhou, Y.H.; Li, X.N.; Sun, S.D.; Guo, J.D.; Wu, B.W.; Wu, M.S. Research on regularity of traditional Chinese medicine in treatment of Alzheimer's disease based on data mining. *China J. Chin. Mater. Medica* **2021**, *46*, 1558–1563.
2. Chen, H.Q.; Yang, M.H.; Tang, X. Association rule mining of aircraft event causes based on the Apriori algorithm. *Sci. Rep.* **2024**, *14*, 13440. [CrossRef] [PubMed]
3. Chen, M.; Yin, Z.X. Classification of Cardiotocography Based on the Apriori Algorithm and Multi-Model Ensemble Classifier. *Front. Cell Dev. Biol.* **2022**, *10*, 888859. [CrossRef]
4. Zeng, J.; Bao, J. Live Multiattribute Data Mining and Penalty Decision-Making in Basketball Games Based on the Apriori Algorithm. *Appl. Bionics Biomech.* **2022**, *2022*, 6968789. [CrossRef]
5. Chen, R.; Hu, F.; Wang, F.; Bai, L.B. Hypergraph-Clustering Method Based on an Improved Apriori Algorithm. *Appl. Sci.* **2023**, *13*, 10577. [CrossRef]
6. Li, F.L.; Meng, C.; Wang, C.; Fan, S.Y. Equipment Quality Information Mining Method Based on Improved Apriori Algorithm. *J. Sens.* **2023**, *2023*, 080005. [CrossRef]
7. Li, L.W.; Wu, Y.P.; Huang, Y.P.; Li, B.; Miao, F.S.; Deng, Z.Q. Optimized Apriori Algorithm for Deformation Response Analysis of Landslide Hazards. *Comput. Geosci.* **2023**, *170*, 105261.
8. Dasgupta, S.; Saha, B. Big data analysis on medical field for drug recommendation using apriori algorithm and deep learning. *Multimed. Tools Appl.* **2024**, *83*, 83029–83051. [CrossRef]
9. Niu, W.N.; Zhou, J.; Zhao, Y.B.; Zhang, X.S.; Peng, Y.J.; Huang, C. Uncovering APT Malware Traffic Using Deep Learning Combined with Time Sequence and Association Analysis. *Comput. Secur.* **2022**, *120*, 102809. [CrossRef]
10. Troncoso-García, A.R.; Martínez-Ballesteros, M.; Martínez-Álvarez, F.; Troncoso, A. A New Approach Based on Association Rules to Add Explainability to Time Series Forecasting Models. *Inf. Fusion* **2023**, *94*, 169–180. [CrossRef]

11. Qi, Y.X.; Wu, J.; Xu, H.S.; Guizani, M. Prediction Blockchain Data Mining with Graph Learning: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2024**, *46*, 729–748. [CrossRef] [PubMed]
12. Zhang, Z.; He, Q.; Gao, J.; Ni, M. A Deep Learning Approach for Detecting Traffic Accidents from Social Media Data. *Transp. Res. Part C* **2018**, *86*, 580–596. [CrossRef]
13. Mu, Y.H.; Wu, Y. Multimodal Movie Recommendation System Using Deep Learning. *Mathematics* **2023**, *11*, 895. [CrossRef]
14. Xie, R.S.; Chung, F.L.; Wang, S.T. A Cognitively Confidence-Debiased Adversarial Fuzzy Apriori Method. *IEEE Trans. Fuzzy Syst.* **2023**, *32*, 1303–1317. [CrossRef]
15. Wu, Q.H.; You, X.M.; Liu, S. Multi-ant colony optimization algorithm based on game strategy and hierarchical temporal memory model. *Clust. Comput.* **2023**, *27*, 3113–3133. [CrossRef]
16. Ntakolia, C.; Lyridis, D.V. A comparative study on Ant Colony Optimization algorithm approaches for solving multi-objective path planning problems in case of unmanned surface vehicles. *Ocean. Eng.* **2022**, *255*, 111418. [CrossRef]
17. Rafsanjani, M.K.; Varzaneh, Z.A. Improvement on Image Edge Detection Using a Novel Variant of the Ant Colony System. *J. Circuits Syst. Comput.* **2019**, *28*, 1950080.
18. Yu, Y.H.; Xie, X.L.; Tang, Y.G.; Liu, Y.R. Feature Selection for Cross-Scene Hyperspectral Image Classification via Improved Ant Colony Optimization Algorithm. *IEEE Access* **2022**, *10*, 102992–103012. [CrossRef]
19. Ren, T.; Luo, T.Y.; Jia, B.B.; Yang, B.H.; Wang, L.; Xie, L.N. Improved ant colony optimization for the vehicle routing problem with split pickup and split delivery. *Swarm Evol. Comput.* **2023**, *77*, 101228. [CrossRef]
20. Gendreau, M.; Louarn, F.X. GENI ants for the traveling salesman problem. *Ann. Oper. Res.* **2004**, *131*, 187–201.
21. Jiang, X.H.; Fang, X. A novel high-utility association rule mining method and its applications. *Multimed. Tools Appl.* **2023**, *83*, 41033–41049. [CrossRef]
22. Petr, M.; Jan, R. A novel algorithm weighting different importance of classes in enhanced association rules. *Knowl. -Based Syst.* **2024**, *294*, 111741.
23. Sridhar, R.; Prasad, M.; Li, Y.F.; Kang, N.; Kang, N.; Zou, L.; Lu, M.Y. Association rule mining with fuzzy linguistic information based on attribute partial ordered structure. *Soft Comput.* **2023**, *27*, 17447–17472.
24. Patel, S.; Zhang, Y.; Balakrishnan, R. Spatio-Temporal association rule based deep annotation-free clustering (STAR-DAC) for unsupervised person re-identification. *Pattern Recognit.* **2022**, *122*, 108287.
25. Mokkadem, A.; Pelletier, M.; Raimbault, L. Association rules and decision rules. *Stat. Anal. Data Min.* **2023**, *16*, 411–435. [CrossRef]
26. Parisa, A.; Amir, T. An intelligent decision support system for groundwater supply management and electromechanical infrastructure controls. *Heliyon* **2024**, *10*, e25036.
27. Ali, S.C.; Mohammad, G. Enhancing flood risk mitigation by advanced data-driven approach. *Heliyon* **2024**, *10*, e37758.
28. Dong, Y.M.; Li, Z.Y.; Xie, C.Z. Enhancing Forest Fire Risk Assessment: An Ontology-Based Approach with Improved Continuous Apriori Algorithm. *Forests* **2024**, *15*, 967. [CrossRef]
29. Tan, Y.A.; Mona, F. ASCF: Optimization of the Apriori Algorithm Using Spark-Based Cuckoo Filter Structure. *Int. J. Intell. Syst.* **2024**, *2024*, 8781318.
30. Liu, S.S. Application of entertainment E-learning mode based on Apriori algorithm in intelligent English reading assistance mode. *Entertain. Comput.* **2024**, *51*, 100744. [CrossRef]
31. Pradeep, K.S.; Esam, O.; Rafeeq, A. Optimized recommendations by user profiling using apriori algorithm. *Appl. Soft Comput.* **2021**, *106*, 107272.
32. Shi, M.M.; Wang, Q.Y.; Feng, G. Application of medical intelligence based on Apriori algorithm in the management of rehabilitation nursing personnel. *Soft Comput.* 2023. [CrossRef]
33. Takuya, S.; Koji, N.; Zhang, X. Analytical Expression of Capon Spectrum for Two Uncorrelated Signals Using the Inner Product of Mode Vectors. *IEICE Trans. Commun.* **2020**, *103*, 442–457.
34. Li, M.; Liu, Y. Improving Random Projections with Extra Vectors to Approximate Inner Products. *IEEE Access* **2020**, *8*, 78590–78607. [CrossRef]
35. Zhang, M.W.; Zhen, A.L.; Zhang, P.H. A secure and privacy-preserving word vector training scheme based on functional encryption with inner-product predicates. *Comput. Stand. Interfaces* **2023**, *86*, 103734. [CrossRef]
36. Anil, K.; Samrat, L.; Pramod, K.M. Low-Complexity Distributed Arithmetic-Based Architecture for Inner-Product of Variable Vectors. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **2023**, *31*, 1368–1376.
37. Wu, H.C. Inner product of fuzzy vectors. *Soft Comput.* **2022**, *26*, 283–307. [CrossRef]
38. Zhang, Y.N. Application of data mining based on improved ant colony algorithm in college students' employment and entrepreneurship education. *Soft Comput.* 2023. [CrossRef]
39. Antonio, M.T.; Jose, M.L.; Philippe, M.; Sebastian, V. Data heterogeneity's impact on the performance of frequent itemset mining algorithms. *Inf. Sci.* **2024**, *678*, 120981.
40. Tamvakis, P.N.; Sakkopoulos, E.; Verykios, V.S. On the Inverse Frequent Itemset Mining Problem for Condensed Representations of Itemsets. *Int. J. Artif. Intell. Tools* **2023**, *32*, 2350006. [CrossRef]
41. Lázaro, B.M.; Alfredo, M.B. A novel multi-core algorithm for frequent itemsets mining in data streams. *Pattern Recognit. Lett.* **2019**, *125*, 241–248.

42. Praveen Kumar, B.; Padmavathy, T. An optimized fuzzy based FP-growth algorithm for mining temporal data. *J. Intell. Fuzzy Syst.* **2023**, *46*, 41–51. [CrossRef]
43. 2022–23 Los Angeles Lakers Roster and Stats. Available online: https://www.basketball-reference.com/teams/LAL/2023.html (accessed on 17 October 2024).
44. 2023–24 Los Angeles Lakers Roster and Stats. Available online: https://www.basketball-reference.com/teams/LAL/2024.html (accessed on 17 October 2024).
45. Disease and Symptoms dataset. Available online: https://www.kaggle.com/datasets/choongqianzheng/disease-and-symptoms-dataset (accessed on 17 October 2024).
46. Hossein, H.; Beneki, C.; Unger, S.; Mazinani, M.T.; Yeganegi, M.R. Text Mining in Big Data Analytics. *Big Data Cogn. Comput.* **2020**, *4*, 1. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.