mathematics

MDPI

*Article*

# Efficient Associate Rules Mining Based on Topology for Items of Transactional Data

**Bo Li [1,*], Zheng Pei [2], Chao Zhang [3,*] and Fei Hao [4]**

[1] School of Computer and Software Engineering, Xihua University, Chengdu 610039, China
[2] School of Science, Xihua University, Chengdu 610039, China; zhengpei@263.com
[3] Intelligent Policing Key Laboratory of Sichuan Province, Sichuan Police College, Luzhou 646000, China
[4] School of Computer Science, Shaanxi Normal University, Xi'an 710119, China; fhao@snnu.edu.cn
* Correspondence: libo@mail.xhu.edu.cn (B.L.); galoiszhang1@scpolicec.edu.cn (C.Z.)

**Abstract:** A challenge in association rules' mining is effectively reducing the time and space complexity in association rules mining with predefined minimum support and confidence thresholds from huge transaction databases. In this paper, we propose an efficient method based on the topology space of the itemset for mining associate rules from transaction databases. To do so, we deduce a binary relation on itemset, and construct a topology space of itemset based on the binary relation and the quotient lattice of the topology according to transactions of itemsets. Furthermore, we prove that all closed itemsets are included in the quotient lattice of the topology, and generators or minimal generators of every closed itemset can be easily obtained from an element of the quotient lattice. Formally, the topology on itemset represents more general associative relationship among items of transaction databases, the quotient lattice of the topology displays the hierarchical structures on all itemsets, and provide us a method to approximate any template of the itemset. Accordingly, we provide efficient algorithms to generate Min-Max association rules or reduce generalized association rules based on the lower approximation and the upper approximation of a template, respectively. The experiment results demonstrate that the proposed method is an alternative and efficient method to generate or reduce association rules from transaction databases.

**Keywords:** knowledge discovery in database (KDD); frequent itemsets; closed itemsets; association rules; the topology for itemsets

**MSC:** 68W99

check for updates

**Citation:** Li, B.; Pei, Z.; Zhang, C.; Hao, F. Efficient Associate Rules Mining Based on Topology for Items of Transactional Data. *Mathematics* **2023**, *11*, 401. https://doi.org/10.3390/math11020401

Academic Editor: Shih-Wei Lin

Received: 9 December 2022
Revised: 1 January 2023
Accepted: 7 January 2023
Published: 12 January 2023

**Copyright:** © 2023 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

In knowledge discovery in database (KDD), association rules mining (ARM) from transaction databases proposed in [1,2] have received considerable attention and wide applications, such as medical diagnosis [3,4], marketing planning [5,6], etc. ARM can be formally explained as follows: Let $\mathscr{A} = (U, A)$ be a transaction database, where $U$ a non-empty finite set of transactions, $A$ a non-empty finite set of items, each transaction $u_i \in U$ decides a mapping $u_i : A \longrightarrow \{0,1\}$, i.e., for any item $a_j \in A$, the transaction $u_i$ has (or has not) the item $a_j \in A$ if $u_i(a_j) = 1$ (or $u_i(a_j) = 0$), each subset of $A$ is called as an itemset. An association rule describing the co-occurrence relation among items is an implication in the form $A_1 \to A_2$, where itemsets $A_1$ and $A_2$ of $A$ such that $A_1 \cap A_2 = \emptyset$ are called as antecedent and consequent, respectively. Theoretically, ARM from a transaction database is a NP-hard problem because itemsets $A_1$ and $A_2$ are selected from $2^{|A|}$ (powerset of $A$), in real world practices, the existed mining methods usually extract a large number of association rules which are difficult to handle; many kinds of association rules or mining methods have been proposed to generate association rules, although these mining methods are different, their processing is nearly the same, i.e., how to evaluate usefulness of association rules and how to select the antecedent and consequent of association rules.

Quality measures are used to evaluate the usefulness of association rules, and various measures have been provided to discover significant and specific association rules. In a transaction database $\mathscr{A} = (U, A)$, for any $A' \subseteq A$, denote $\tau(A') = \{u_i \in U | \forall a_j \in A', u_i(a_j) = 1\}$, $\neg A'$ is the absence of $A'$, then the following quality measures can be defined

- Support [1,2]: $supp_{\mathscr{A}}(A_1 \cup A_2) = \frac{|\tau(A_1 \cup A_2)|}{|U|}$;
- Confidence [1,2]: $conf_{\mathscr{A}}(A_1 \to A_2) = \frac{supp_{\mathscr{A}}(A_1 \cup A_2)}{supp_{\mathscr{A}}(A_1)}$;
- Netconfidence [7]: $netc_{\mathscr{A}}(A_1 \to A_2) = \frac{supp_{\mathscr{A}}(A_1 \cup A_2) - supp_{\mathscr{A}}(A_1) supp_{\mathscr{A}}(A_2)}{supp_{\mathscr{A}}(A_1)(1 - supp_{\mathscr{A}}(A_1))}$;
- Conviction [8]: $conv_{\mathscr{A}}(A_1 \to A_2) = \frac{supp_{\mathscr{A}}(A_1) supp_{\mathscr{A}}(\neg A_2)}{supp_{\mathscr{A}}(A_1 \cup \neg A_2)}$;
- Added value [9]: $add_{\mathscr{A}}(A_1 \to A_2) = conf_{\mathscr{A}}(A_1 \to A_2) - supp_{\mathscr{A}}(A_2)$;
- Accuracy [9]: $acc_{\mathscr{A}}(A_1 \to A_2) = conf_{\mathscr{A}}(A_1 \to A_2) + conf_{\mathscr{A}}(\neg A_1 \to \neg A_2)$;
- Interestingness [10]: $inte_{\mathscr{A}}(A_1 \to A_2) = \frac{supp_{\mathscr{A}}(A_1 \cup A_2)}{supp_{\mathscr{A}}(A_1)} \times \frac{supp_{\mathscr{A}}(A_1 \cup A_2)}{supp_{\mathscr{A}}(A_2)} \times (1 - supp_{\mathscr{A}}(A_1 \cup A_2))$;
- Comprehensibility [10]: $comp_{\mathscr{A}}(A_1 \to A_2) = \frac{\log(1 + |\tau(A_2)|)}{\log(1 + |\tau(A_1 \cup A_2)|)}$;
- Lift [11]: $lift_{\mathscr{A}}(A_1 \to A_2) = \frac{supp_{\mathscr{A}}(A_1 \cup A_2)}{supp_{\mathscr{A}}(A_1) supp_{\mathscr{A}}(A_2)}$.

By combining with several quality measures to form fitness functions, ARM is transformed into optimization problems, and various optimal algorithms can be used to extract interesting or valid association rules from $\mathscr{A} = (U, A)$. In [12], schema constraints and the opportunistic confidence constraint are enforced to mine generalized association rules in the analyzed data. In [13], confidence, comprehensibility, and interestingness are considered as a multi-objective problem, and the particle swarm optimization algorithm is proposed to extract the best rules. In [10], support, comprehensibility and interestingness are considered as a multi-objective problem, and a Pareto-based genetic algorithm is used to extract some useful and interesting rules from any market-basket type database. In [14], Shafer's theory of evidence is used as two information measures for the quality evaluation of the set of frequent itemset (or frequent pattern). In [15], off-the-shelf constraint programming techniques are employed for modeling and solving a wide variety of constraint-based frequent itemsets. In [16], based on exclusive causal-leverage measure, a data mining algorithm is developed to mine the causal relation between drugs and their associated adverse drug reactions. In [17], the particle swarm optimization algorithm are provided to improve computational efficiency as well as to automatically determine suitable support and confidence threshold values of association rules. In [18], grammar-guided genetic programming models are proposed to deal with the association rule mining problem under a multi-objective perspective. In [19], a new multi-objective evolutionary model, which maximizes the comprehensibility, interestingness and performance of the objectives, is presented to mine a set of quantitative association rules with a good trade-off between interpretability and accuracy. In [20], a principal components analysis is applied to a set of measures that evaluate quantitative association rules' quality; the QARGA algorithm is provided to find out quantitative association rules from a wide variety of datasets. In [21], a new confidence degree of association rules and a discrete bi-level parametric programming are proposed to extract association rules from huge databases. In [22], redundancy analysis, sampling and multivariate statistical analysis are provided to ascertain the discovered rules and discard the non-significant rules. In [23], a systematic assessment of various numerical association rule mining methods and a meta-study of thirty numerical association rule mining algorithms are provided. The authors investigate how far the discretization techniques have been used in the numerical association rule mining methods. In [24], a multi-objective particle swarm optimization is proposed using an adaptive archive grid based on the Pareto optimal strategy for numerical association rule mining. In [25–28], these papers adopt the idea of parallelization and improves the Apriori algorithm based on the MapReduce model.

By analyzing associative relation among items, suitable itemsets can be selected to generate association rules. The most widely used itemsets are frequent itemsets and closed itemsets [15,29–32], an itemset is frequent if its support is not less than the minimum support value, an itemset is closed if and only if no proper superset of the itemset has the same support as the itemset. In [33], frequent itemsets are employed to construct taxonomy over items, and a breadth-first search is adopted to enumerate all frequent itemsets. Closed frequent itemsets can be used to uniquely determine the set of all frequent itemsets and applied to generate a condensed set of association rules [34–36]. Formally, all closed itemsets of $\mathscr{A} = (U, A)$ can be constructed as a closed itemset lattice, and CHARM-L algorithm is proposed to explicitly generate the frequent closed itemset lattice [37]. In [38], the parent-child relation of the closed itemset lattice has been exploited, and the cross-level closed itemset lattice is constructed to mine the most relevant minimal cross-level association rules. In [39], an efficient post-processing method is presented to prune redundant rules by virtue of the property of Galois connection, which inherently constrains rules with respect to objects. At present, itemsets have been widely investigated [40,41], and various itemsets and their generating algorithms for specific association rules mining have been proposed, such as expressive generalized itemsets [42], free itemsets [43], disjunction-free itemsets [44,45], non-derivable itemsets [46,47], disjunctive closed itemsets [48], etc. In fact, we notice that frequent itemsets or closed itemsets are rooted to the co-occurrence relation among items; from the mathematical point of view, topology may be a more suitable tool to express the relation among items, because a topology for the set is used to express a relation among subsets, subsets of the set are granulated as members of the topology, and the topology for the set is generated by its topology base; this means that the base for the topology is the basic relation among elements of the set. Topology-based pattern mining has been widely investigated in the rough set theory and knowledge model, such as in [49]; a nested topology on a crisp set of reference is provided to interpret a fuzzy subset; the approach provides us a unified framework for most of the fuzzy thresholding algorithms. In [50], the lower and upper approximations of the rough set theory are shown as pre-topological invariants and dual to each other, and the corresponding generalized topological closure via the upper approximation is constructed. In [51], the knowledge representation and reasoning method are proposed for identity-based spatial change; the change process is presented by a multistage graph, the binary relation model $BC$ for identity change is defined, and qualitative reasoning for $BC$ and combining $BC$ with topological relations are investigated.

In this paper, we analyze topology and the topology base for items, express more general associative relations among items and extract useful association rules based on the topology and the base; all of these are inspired by our previous works on the formal concept analysis; in [52], we proposed a novel method based on the topology for attributes of a formal context to generate all formal concepts, the topology for attributes has been used to explain the associative relation among attributes of the formal context, and the formal concept lattice can be constructed by the topology for attributes based on an equivalent relation. In [53], we used the topology for attributes of multi-valued information systems to generate Min-Max association rules. In this paper, we construct the topology for the set of items of a transaction database, which can be used to explain more general associative relation among items; moreover, we prove that frequent itemsets, closed itemsets and closed frequent itemsets are included in the topology for the set of items, provide two kinds of lattice on the topology to display the hierarchical structures on itemsets, and propose efficient algorithms based on the base for the topology to generate or reduce many kinds of association rules.

The organization of this paper is as follows: In Section 2, we briefly review some basic notions used in association rules mining. In Section 3, we present the topology for the set of items and lattices on itemsets of a transactional database, in which some important properties of the topology, the quotient lattice and minimal generators of closed frequent itemsets, are provided. In Section 4, we propose several algorithms to generate or reduce association rules, and analyze confidences of generalized association rules. In Section 5, we choose zoo, mushroom, connect-4 and chess as transactional databases to show the proposed method in generating association rules. We conclude the paper in Section 6.

## 2. Preliminaries

In the section, we briefly review some basic concepts used in frequent itemsets, closed itemsets or association rules mining; several concepts are also widely used in rough sets and formal concept analysis. For a uniform expression, we adopt the following set-to-set mapping in a transaction database $\mathscr{A} = (U, A)$, denote $P(U)$ and $P(A)$ as power sets of $U$ and $A$, respectively, then we have

$$\begin{aligned} \tau &: P(A) \longrightarrow P(U),\ \tau(A_1) = \{u_i \in U | \forall a_j \in A_1 \subset A, u_i(a_j) = 1\}, \\ \gamma &: P(U) \longrightarrow P(A),\ \gamma(O) = \{a_j \in A | \forall u_i \in O \subset U, u_i(a_j) = 1\}. \end{aligned}$$

Based on $\tau$ and $\gamma$, we can rewrite the support of an itemset $A_1$ and the confidence of an association rule $A_1 \to A_2$, as follows:

$$\begin{aligned} supp(A_1) &= |\tau(A_1)| (or \ \frac{|\tau(A_1)|}{|U|}), \\ conf(A_1 \to A_2) &= \frac{|\tau(A_1 \cup A_2)|}{|\tau(A_1)|}. \end{aligned}$$

$\tau$ and $\gamma$ also make us define closed itemsets and their generator concisely, i.e., $A_1 \subset A$ is a closed itemset if and only if $\gamma\tau(A_1) = A_1$, an itemset $A_2$ is a generator of the closed itemset $A_1$ if and only if $A_2 \subset A_1$ and $\gamma\tau(A_2) = A_1$, furthermore, $A_2$ is a minimal generator of $A_1$ if and only if $\nexists A_3 \subset A_2$ such that $\gamma\tau(A_3) = A_1$. Formally, $\tau$ and $\gamma$ are also analyzed in rough sets and formal concept analysis [50,54–57], and the following properties of $\tau$ and $\gamma$ are obviously, for any $A_1, A_2, A_3 \subset A$ and $O_1, O_2 \subset U$, (1) If $A_1 \subseteq A_2$, then $\tau(A_1) \supseteq \tau(A_2)$; (2) If $O_1 \subseteq O_2$, then $\gamma(O_1) \supseteq \gamma(O_2)$; (3) $\tau(A_1 \cup A_2) = \tau(A_1) \cap \tau(A_2), \gamma(O_1 \cup O_2) = \gamma(O_1) \cap \gamma(O_2)$; (4) $\tau(\gamma\tau(A_1)) = \tau(A_1), \gamma(\tau\gamma(O_1)) = \gamma(O_2)$. In [52], we use $\tau$ and $\gamma$ to induce a reflexive and transitive relation on the set of attributes from a formal context, the topology for attributes are constructed by using the relation, then the formal concepts are analyzed in the the topology for attributes. Here, we provide the same results in a transaction database.

**Definition 1.** *For any transactional database $\mathscr{A} = (U, A)$, $\tau$ and $\gamma$ decide a point-to-set mapping from $A$ to $P(A)$, i.e., for any $a_i \in A$, $C : A \longrightarrow P(A)$ is*

$$C(a_i) = \{a_j \in A | a_j \in \gamma\tau(\{a_i\})\}.$$

Intuitively, the point-to-set mapping $C$ represents a co-occurrence relation among items, i.e., the following binary relation on $A$ can be decided by the mapping, $C$. For any $a_i, a_j \in A$,

$$R_A(a_i, a_j) = \begin{cases} 1, & \text{if} \quad a_j \in C(a_i), \\ 0, & \text{if} \quad a_j \notin C(a_i). \end{cases} \tag{1}$$

In [52], we have proved that the binary relation $R_A$ on $A$ is reflexive and transitive, and $(A, R_A)$ is an approximation space. For any subset $A_1$ of $A$, denote $\underline{R_A}(A_1) = \{a_i \in A | \forall a_j, (R_A(a_i, a_j) = 1) \to (a_j \in A_1)\}$ and $\overline{R_A}(A_1) = \{a_i \in A | \exists a_j \in A_1 \wedge R_A(a_i, a_j) = 1\}$, then $\overline{R_A}(A_1)$ and $\underline{R_A}(A_1)$ are generalized upper approximation and lower approximation of $A_1$, especially if $R_A$ is an equivalent relation; $\overline{R_A}(A_1)$ and $\underline{R_A}(A_1)$ are Pawlak's upper approximation and lower approximation [55]. More important, we have the following theorem.

**Theorem 1** ([52]). *For any transactional database $\mathscr{A} = (U, A)$.*

1. *$T(R_A) = \{\underline{R_A}(A_1) |\ A_1 \subseteq A\}$ is a topology for $A$;*
2. *For any $a_i \in A$, $\underline{R_A}(C(a_i)) = C(a_i)$;*
3. *$\mathcal{B}_A = \{C(a_i)|a_i \in A\}$ is a base for the topology $T(R_A)$.*

The theorem means that the topology for $A$ has a simple expression, i.e., $T(R_A) = \{\bigcup_{C(a_i)\in\mathcal{B}'_A} C(a_i)|\forall\mathcal{B}'_A \subseteq \mathcal{B}_A\}$, we use the following example to explain the above mentioned concepts and results; the example was initially used to analyze the concept "to-be-a-fruit" [58].

**Example 1.** *Let a fruit data be $\mathscr{A} = (U, A)$, where $U = \{$a chestnut $(u_1)$, an olive $(u_2)$, a pepper $(u_3)$, a strawberry $(u_4)$, an orange $(u_5)$, a tomato $(u_6)\}$ and $A = \{$to-grow-on-trees $(a_1)$, to-be-sweet $(a_2)$, to-be-raw-edible $(a_3)$, to-yield-juice $(a_4)$, to-have-a-skin $(a_5)\}$ be used to understand the concept "to-be-a-fruit" (shown in Table 1). According to Definition 1, we have $C(a_1) = \{a_1\}, C(a_2) = \{a_2, a_3\}$, $C(a_3) = \{a_3\}$, $C(a_4) = \{a_3, a_4, a_5\}$ and $C(a_5) = \{a_5\}$, e.g., for $C(a_2) = \{a_2, a_3\}$, it means that $a_2$ and $a_3$ are co-occurrence in Table 1, i.e., a fruit is "to-be-sweet", it must be "to-be-raw-edible". The binary relation $R_A$ on $A = \{a_1, a_2, a_3, a_4, a_5\}$ is shown in Table 2.*

**Table 1.** A transactional database $\mathscr{A} = (U, A)$.

| $U\backslash A$ | $a_1$ | $a_2$ | $a_3$ | $a_4$ | $a_5$ |
|---|---|---|---|---|---|
| $u_1$ | 1 | 0 | 0 | 0 | 1 |
| $u_2$ | 1 | 0 | 1 | 0 | 0 |
| $u_3$ | 0 | 0 | 1 | 0 | 1 |
| $u_4$ | 0 | 1 | 1 | 0 | 0 |
| $u_5$ | 1 | 1 | 1 | 1 | 1 |
| $u_6$ | 0 | 0 | 1 | 1 | 1 |

**Table 2.** The binary relation $R_A$ on $A$.

| $R_A$ | $a_1$ | $a_2$ | $a_3$ | $a_4$ | $a_5$ |
|---|---|---|---|---|---|
| $a_1$ | 1 | 0 | 0 | 0 | 0 |
| $a_2$ | 0 | 1 | 1 | 0 | 0 |
| $a_3$ | 0 | 0 | 1 | 0 | 0 |
| $a_4$ | 0 | 0 | 1 | 1 | 1 |
| $a_5$ | 0 | 0 | 0 | 0 | 1 |

*Based on Theorem 1 and the mapping $C$, we have $T(R_A) = \{\varnothing, \{a_1\}, \{a_3\}, \{a_5\}, \{a_1, a_3\}, \{a_1, a_5\}, \{a_2, a_3\}, \{a_3, a_5\}, \{a_1, a_2, a_3\}, \{a_1, a_3, a_5\}, \{a_2, a_3, a_5\}, \{a_3, a_4, a_5\}, \{a_1, a_3, a_4, a_5\}, \{a_1, a_2, a_3, a_5\}, \{a_2, a_3, a_4, a_5\}, \{a_1, a_2, a_3, a_4, a_5\}\}$, e.g., the member $\{a_2, a_3, a_5\}$ is obtained by*

$$\{a_2, a_3, a_5\} = C(a_2) \cup C(a_5).$$

*According to Table 2, we can find that $R_A$ on $A$ is not symmetric generally, e.g., $R_A(a_2, a_3) = 1$ but $R_A(a_3, a_2) = 0$, hence, members of the topology $T(R_A)$ for the set of items are not Pawlak's lower approximation in practice. According to Table 1, we can obtain the support of each member in the topology for $A$, e.g., for $\{a_2, a_3\}$, we have $supp(\{a_2, a_3\}) = |\tau(T)| = |\{u_4, u_5\}| = 2$ or $supp(\{a_2, a_3\}) = \frac{|\{u_4,u_5\}|}{|U|} = \frac{1}{3}$. If we fix minimum support $\frac{1}{3}$, then members such as $\{a_2, a_3\}$ and $\{a_1, a_3\}$ are frequent itemsets, which can be used to generate association rules, e.g., we can generate association rules "$a_1 \to a_3$" and "$a_3 \to a_1$"; confidences of them are*

$$\begin{aligned} conf(a_1 \to a_3) &= \frac{|\tau(\{a_1, a_3\})|}{|\tau(a_1)|} = \frac{|\{u_2, u_5\}|}{|\{u_1, u_2, u_5\}|} = \frac{2}{3}, \\ conf(a_3 \to a_1) &= \frac{|\tau(\{a_1, a_3\})|}{|\tau(a_3)|} = \frac{|\{u_2, u_5\}|}{|\{u_2, u_3, u_4, u_5, u_6\}|} = 0.4. \end{aligned}$$

*In $A$ of the fruit data $\mathscr{A} = (U, A)$, "$a_1 \to a_3$" is "to-grow-on-trees implies to-be-raw-edible", "$a_3 \to a_1$" is "to-be-raw-edible implies to-grow-on-trees", by considering their confidences, "$a_1 \to a_3$" is more confident than "$a_3 \to a_1$" due to $\frac{2}{3} > 0.4$.*

From Example 1, we notice that members of the topology $T(R_A)$ for the set of items are itemsets, which is generated by the co-occurrence relation among items. In existed ARM methods, frequent itemsets or closed itemsets are always used to mine association rules, one problem is "can we use members of the topology $T(R_A)$ for the set of items to mine all useful and needful association rules". The problem will be solved in the next section.

## 3. Lattice Structures on the Topology for the Set of Items

In this section, we use set inclusion to construct lattice structures on the topology $T(R_A)$ for the set of items, one is constructed on the topology $T(R_A)$ itself, another is constructed on quotient set of the topology $T(R_A)$, then we analyze minimal elements and minimal generators of closed itemsets in the lattice structures, which are useful information in ARM.

### *3.1. The Lattice on the Topology*

Formally, the topology for the set of items is a proper subset of the power set of $A$—theoretically, the power set of $A$ naturally forms a power set lattice; however, its proper subset may be not a lattice. Here, we use the set inclusion to construct lattice on the topology for the set of items and analyze hierarchical structure of members of the topology.

It is obvious that $T(R_A)$ is an poset by set inclusion, i.e., for any $T_1, T_2 \in T(R_A)$,

$$T_1 \leq T_2 \text{ if and only if } T_1 \subseteq T_2.$$

On the poset $(T(R_A), \leq)$, we define

$$\begin{aligned} T_1 \wedge T_2 &= (\bigcup_{a_i \in A}^{C(a_i) \subseteq T_1} C(a_i)) \cap (\bigcup_{a_j \in A}^{C(a_j) \subseteq T_2} C(a_j)), \\ T_1 \vee T_2 &= (\bigcup_{a_i \in A}^{C(a_i) \subseteq T_1} C(a_i)) \cup (\bigcup_{a_j \in A}^{C(a_j) \subseteq T_2} C(a_j)). \end{aligned}$$

For any $a_k \in T_1 \wedge T_2$, we have $a_k \in \bigcup_{a_i \in A}^{C(a_i) \subseteq T_1} C(a_i)$ and $a_k \in \bigcup_{a_j \in A}^{C(a_j) \subseteq T_2} C(a_j)$; hence, there exists $C(a_i) \subseteq T_1$ and $C(a_j) \subseteq T_2$ such that $a_k \in C(a_i)$ and $a_k \in C(a_j)$, according to Theorem 1, $C(a_k) \subseteq C(a_i)$ and $C(a_k) \subseteq C(a_j)$, we have

$$\begin{aligned} C(a_k) &\subseteq (\bigcup_{a_i \in A}^{C(a_i) \subseteq T_1} C(a_i)) \cap (\bigcup_{a_j \in A}^{C(a_j) \subseteq T_2} C(a_j)) \\ &= \bigcup_{a_i \in A}^{C(a_i) \subseteq T_1} \bigcup_{a_j \in A}^{C(a_j) \subseteq T_2} (C(a_i) \cap C(a_j)) = T_1 \wedge T_2, \end{aligned}$$

this means that we rewrite $T_1 \wedge T_2 = \bigcup_{a_k \in T_1 \cap T_2} C(a_k) \in T(R_A)$. Similarly, we rewrite $T_1 \vee T_2 = \bigcup_{a_k \in T_1 \cup T_2} C(a_k) \in T(R_A) \in T(R_A)$, this means that $(T(R_A), \wedge, \vee)$ is a lattice.

Furthermore, for any subset $\mathcal{T} \subseteq T(R_A)$, we denote

$$\begin{aligned}\wedge\mathcal{T} &= \bigwedge_{T_i\in\mathcal{T}} T_i = \bigcap_{T_i\in\mathcal{T}} \Big( \bigcup_{a_i\in A}^{C(a_i)\subseteq T_i} C(a_i)\Big),\\ \vee\mathcal{T} &= \bigvee_{T_i\in\mathcal{T}} T_i = \bigcup_{T_i\in\mathcal{T}} \Big( \bigcup_{a_i\in A}^{C(a_i)\subseteq T_i} C(a_i)\Big),\end{aligned}$$

then, one can easily prove that $\wedge\mathcal{T} = \bigcup_{a_k\in\bigcap_{T_i\in\mathcal{T}} T_i} C(a_k) \in T(R_A)$ and $\vee\mathcal{T} = \bigcup_{a_k\in\bigcup_{T_i\in\mathcal{T}} T_i} C(a_k) \in T(R_A)$, and $(T(R_A), \wedge, \vee)$ is also a complete lattice, e.g., in Example 1, we have $\{a_1\} \wedge \{a_3\} = \emptyset$, $\{a_1, a_2, a_3\} \wedge \{a_1, a_3, a_5\} = C(a_1) \cup C(a_3) = \{a_1, a_3\}$ due to $\{a_1, a_2, a_3\} \cap \{a_1, a_3, a_5\} = \{a_1, a_3\}$, and $\{a_2, a_3, a_5\} \vee \{a_3, a_4, a_5\} = C(a_2) \cup C(a_3) \cup C(a_4) \cup C(a_5) = \{a_2, a_3, a_4, a_5\}$ due to $\{a_2, a_3, a_5\} \cup \{a_3, a_4, a_5\} = \{a_2, a_3, a_4, a_5\}$. The complete lattice $(T(R_A), \wedge, \vee)$ of Example 1 is shown in Figure 1.

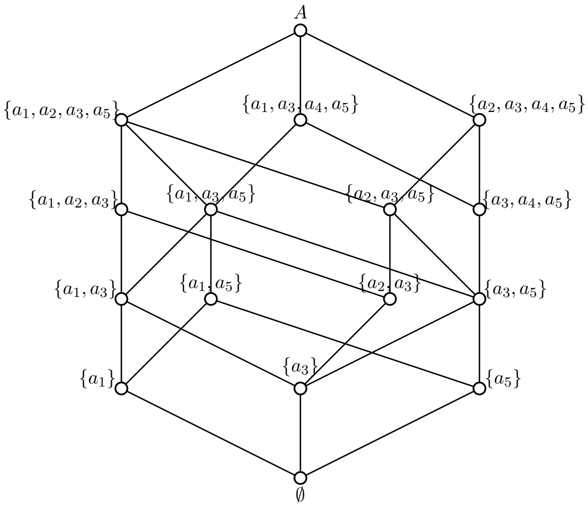

**Figure 1.** The lattice of the topology $T(R_A)$ of Example 1.

From up to down, the complete lattice $(T(R_A), \wedge, \vee)$ provides us hierarchical information among members of the topology, intuitively, hierarchical information means that any member of $T(R_A)$ is contained in its upper hierarchical members, and there are common transactions in the member of $T(R_A)$ and its upper hierarchical members; such information can be used to generate association rules, i.e., suppose that $T_2$ is an upper hierarchical member of $T_1$, then, we have the following association rule

$$\psi = T_1 \to (T_2 - T_1),$$

in which, $supp(\psi) = |\tau(T_2)|$ and $conf(\psi) = \frac{|\tau(T_2)|}{|\tau(T_1)|}$, e.g., in Figure 1, itemset $\{a_3, a_5\}$ is an up hierarchical member of itemset $\{a_5\}$, according to Table 1, we have $a_5 \to a_3$ with $supp(a_5 \to a_3) = |\tau(\{a_3, a_5\})| = 3$ and $conf(a_5 \to a_3) = \frac{|\tau(\{a_3,a_5\})|}{|\tau(\{a_5\})|} = 0.75$. $a_1 \wedge a_3 \to a_5$ with $supp(a_1 \wedge a_3 \to a_5) = |\tau(\{a_1, a_3, a_5\})| = 1$ and $conf(a_1 \wedge a_3 \to a_5) = \frac{|\tau(\{a_1,a_3,a_5\})|}{|\tau(\{a_1,a_3\})|} = 0.5$. In $A$ of the fruit data $\mathscr{A} = (U, A)$, "$a_5 \to a_3$" is "to-have-a-skin implies to-be-raw-edible", "$a_1 \wedge a_3 \to a_5$" is "to-grow-on-trees and to-be-raw-edible implies to-have-a-skin"; obviously, "to-have-a-skin implies to-be-raw-edible" is more confident and useful than "to-grow-on-trees and to-be-raw-edible implies to-have-a-skin" due to $0.75 > 0.5$.

### *3.2. The Lattice on the Quotient Set of the Topology*

To fast mine association rules with high support and confidence from the topology for the set of items, we construct another lattice structure on the topology in the subsection;

then, we analyze minimal elements and minimal generators of closed itemsets in the lattice, which can help us to fast generate association rules with high support and confidence.

For any $T_1, T_2 \in T(R_A)$, we define a binary relation on the topology for the set of items, as follows

$$T_1 \sim_\tau T_2 \text{ if and only if } \tau(T_1) = \tau(T_2).$$

It is obvious that $\sim_\tau$ is an equivalent relation on the topology $T(R_A)$, and an quotient set of the topology $T(R_A)$ can be decided by the equivalent relation, i.e., $T(R_A)/\sim_\tau = \{[T] | T \in T(R_A)\}$; each equivalent class $[T]$ in $T(R_A)/\sim_\tau$ is consisted of members of $T(R_A)$ with the same support $\tau(T)$. According to the property of the topology, we have $T_1 \cup T_2$ is in $T(R_A)$ if $T_1$ and $T_2$ are in $T(R_A)$. On the other hand, according to property of the set-to-set mapping $\tau$, for any $[T] \in T(R_A)/\sim_\tau$ and $T', T'' \in [T]$, $\tau(T' \cup T'') = \tau(T') \cap \tau(T'') = \tau(T)$, we have $T' \cup T'' \in [T]$, i.e., each equivalent class $[T]$ in $T(R_A)/\sim_\tau$ is closed for the $\cup$ operation, hence, we have the following theorem.

**Theorem 2.** *For each equivalent class $[T]$ in $T(R_A)/\sim_\tau$, $[T]$ is a union semi-lattice.*

The maximum element of each equivalent class $[T]$ in $T(R_A)/\sim_\tau$ is important. Formally, we denote the maximum element of $[T]$ as $\cup[T]$, then we confirm that $\cup[T]$ is a closed itemset and each member $T$ of $[T]$ is a generator of $\cup[T]$, in fact, suppose that $T$ is a generator of a closed itemset $A'$, according to the set-to-set mapping $\tau$ and Definition 1; we have

$$\tau(A') = \bigcap_{a_i \in A'} \tau(a_i) = \bigcap_{a_i \in A'} \tau(C(a_i)) = \tau(\bigcup_{a_i \in A'} C(a_i)) = \tau(T),$$

this means that $A'$ is a member of the topology for the set of items and $A' \in [T]$, hence, $A' = \cup[T]$.

**Corollary 1.** *For any transactional database $\mathscr{A} = (U, A)$ and each equivalent class $[T]$ in $T(R_A)/\sim_\tau$, (1) all closed itemsets of $A$ are in $T(R_A)$; (2) $\cup[T]$ is a closed itemset; (3) $\forall T' \in [T]$ and $T' \neq \cup[T]$, $T'$ is a generator of $\cup[T]$.*

Because $T(R_A)$ is a proper subset of powerset of $A$ and the existed methods search closed itemsets in powerset of $A$, the corollary means that we can reduce the searching range of closed itemsets in the topology for the set of items. On the other hand, frequent itemsets own the downward closed property, i.e., any subset of a frequent itemset is still frequent; hence, we can use closed frequent itemsetsto obtain its all frequent itemsets, any subset of a closed frequent itemset is a frequent itemset. The following corollaries help us to obtain minimal generators of closed itemsets, which is useful information in ARM.

In each union semi-lattice $[T]$, we denote minimal members of $[T]$ as $min[T] = \{T' \in [T] | \nexists T'' \in [T] \wedge T'' \subseteq T'\}$.

**Corollary 2.** *For any $T' \in min[T]$, if there exists no $P \subset T'$ such that $\tau(P) = \tau(T')$, then $T'$ is a minimal generator of $T'' \in [T]$ such that $T'' \notin min[T]$ and $T' \subset T''$.*

In Example 1, we can easily check $[\{a_1, a_2, a_3\}] = \{\{a_1, a_2, a_3\}, \{a_1, a_3, a_5\}, \{a_2, a_3, a_5\}, \{a_1, a_3, a_4, a_5\}, \{a_1, a_2, a_3, a_5\}, \{a_2, a_3, a_4, a_5\}, \{a_1, a_2, a_3, a_4, a_5\}\}$, in which, $min[\{a_1, a_2, a_3\}] = \{\{a_1, a_2, a_3\}, \{a_1, a_3, a_5\}, \{a_2, a_3, a_5\}\}$, due to for any $P \subset \{a_1, a_3, a_5\}$, $\tau(P) \neq \tau(\{a_1, a_3, a_5\}) = \{u_5\}$, $\{a_1, a_3, a_5\}$ is a minimal generator of $\{a_1, a_3, a_4, a_5\}$, $\{a_1, a_2, a_3, a_5\}$ and $\{a_1, a_2, a_3, a_4, a_5\}$.

For $T' \in min[T]$, if there exists $P \subset T'$ such that $\tau(P) = \tau(T')$, we denote $\mathcal{P}_\tau(T') = \{P \subset T' | T' \in min[T], \tau(P) = \tau(T')\}$, $\mathcal{P}_\tau(T')$ is an poset by set inclusion, i.e., $\forall P_1, P_2 \in \mathcal{P}_\tau(T')$, $P_1 \leq P_2$ if and only if $P_1 \subseteq P_2$, minimal elements of $\mathcal{P}_\tau(T')$ is denoted by

$$min\mathcal{P}_\tau(T') = \{P' \in \mathcal{P}_\tau(T') | \nexists P'' \in \mathcal{P}_\tau(T') \wedge P'' \subseteq P'\}.$$

**Corollary 3.** *For $P' \in min\mathcal{P}_\tau(T')$, $P'$ is a minimal generator of $T'' \in [T]$, where $T'$ is a generator of $T''$.*

In Example 1, $\{a_1, a_2, a_3\} \in min[\{a_1, a_2, a_3\}]$, due to $\tau(\{a_1, a_2\}) = \tau(\{a_1, a_2, a_3\}) = \{u_5\}$, $\{a_1, a_2\} \in min\mathcal{P}_\tau(\{a_1, a_2, a_3\})$ and $\{a_1, a_2\}$ is a minimal generator of $\{a_1, a_2, a_3\}$, $\{a_1, a_2, a_3, a_5\}$ and $\{a_1, a_2, a_3, a_4, a_5\}$.

From the algebraic point of view, we can construct a lattice structure on the quotient set of the topology $T(R_A)$, i.e., for any $[T_1], [T_2] \in T(R_A)/\sim_\tau$, we define $[T_1] \vee [T_2] = [(\cup[T_1]) \cap (\cup[T_2])]$ and $[T_1] \wedge [T_2] = [T_1 \cup T_2]$, one can easily check that operators $\vee$ and $\wedge$ on $T(R_A)/\sim_\tau$ are well defined, and we have the following theorem.

**Theorem 3** ([52]). *$(T(R_A)/\sim_\tau, \wedge, \vee)$ is a complete lattice, in which the maximum and minimum elements are $[\emptyset]$ and $[A]$, respectively.*

In Example 1, $T(R_A)/\sim_\tau = \{[\emptyset], [\{a_1\}], [\{a_3\}], [\{a_5\}], [\{a_1, a_3\}], [\{a_1, a_5\}], [\{a_2, a_3\}], [\{a_3, a_5\}], [\{a_3, a_4, a_5\}], [\{a_1, a_2, a_3\}](= [A])\}$, the lattice $(T(R_A)/\sim_\tau, \wedge, \vee)$ of Example 1 is shown in Figure 2.

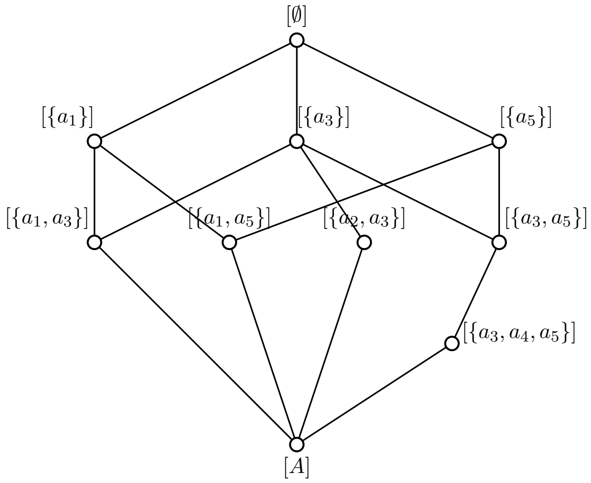

**Figure 2.** The complete lattice $(T(R_A)/\sim_\tau, \wedge, \vee)$ of Example 1.

## 4. Association Rules Mining from the Quotient Set of the Topology

In summary, we can confirm the follows facts about itemsets of a transactional database $\mathscr{A} = (U, A)$ based on the above mentioned results:

1. The topology $T(R_A)$ for the set of items is a complete lattice and displays a hierarchical structure on some itemsets, it can be generated by the base $\mathcal{B}_A = \{C(a_i) | a_i \in A\}$; generally, each $C(a_i)$ is an itemset and can more fast generate closed itemsets than single items in the existed methods;
2. All closed itemsets are included in the topology $T(R_A)$, moreover, a closed itemset is the maximum element of an equivalent class $[T] \in T(R_A)/\sim_\tau$;
3. Each itemsets in $[T]$ has the same support; moreover, generators and minimal generators of a closed itemset can be obtained from $[T]$;
4. The complete lattice $(T(R_A)/\sim_\tau, \wedge, \vee)$ displays the hierarchical structures on closed itemsets.

Inspired by rough set theory, each equivalent class in $T(R_A)/\sim_\tau$ can be understood as a granular knowledge on the set of itemsets of $\mathscr{A} = (U, A)$; on the one hand, each granular knowledge can be used to mine association rules because it includes a closed itemset. On the other hand, all granular knowledge can be also used to approximate any itemset $A_1 = \{a_{i_1}, a_{i_2}, \cdots, a_{i_k}\} \subseteq A$, i.e.,

1. The lower approximation of $A_1$:

$$L(A_1) = \bigcup_{a_i \in A}^{C(a_i) \subseteq A_1} C(a_i);$$

2. The upper approximation of $A_1$:

$$G(A_1) = \bigcup_{a_i \in A}^{C(a_i) \cap A_1 \neq \emptyset} C(a_i).$$

It is obvious that for any itemset $A_1 \subseteq A$, $L(A_1)$ and $G(A_1)$ are in the topology $T(R_A)$ for the set of items, if $L(A_1) = G(A_1)$, then the itemset $A_1$ is in $T(R_A)$; furthermore, supports of $L(A_1)$, $A_1$ and $G(A_1)$ are such that

$$supp(L(A_1)) \geq supp(A_1) \geq supp(G(A_1)),$$

such as in Example 1, for itemset $\{a_1, a_2, a_5\} \subset A$, $L(\{a_1, a_2, a_5\}) = \{a_1, a_5\}$, $G(\{a_1, a_2, a_5\}) = \{a_1, a_2, a_3, a_5\}$, and $supp(\{a_1, a_5\}) = |\{u_1, u_5\}| = 2 \geq supp(\{\ a_1, a_2, a_5\}) = |\{u_5\}| = 1 \geq supp(\{a_1, a_2, a_3, a_5\}) = |\{u_5\}| = 1$. Formally, the lower approximation and the upper approximation of an itemset provide us an alternative method to mine generalized association rules or reduce association rules. All of these will be discussed in the rest of the section.

### *4.1. Min-Max Association Rules Mining*

In the subsection, we provide an useful method to mine Min-Max association rules from closed frequent itemsets; here, an association rule is called as a Min-Max association rule if and only if there does not exist an association rule with the same quality measures as the association rule, but with a more specific antecedent part and a more general consequent part. Because closed itemsets are in equivalent classes of $T(R_A)/\sim_\tau$, we propose Algorithm 1 to generate all Min-Max association rules with confidence $c = 1$ from equivalent classes of $T(R_A)/\sim_\tau$.

**Algorithm 1** Min-Max association rules mining from closed itemsets

```
Input: A transactional database $\mathscr{A} = (U, A)$.
Output: Min-Max association rules with confidence $c = 1$.
while The stop condition is not satisfied do
    (1) Generate $\mathcal{B}_A = \{C(a_i) | a_i \in A\}$ according to $C(a_i) = \{a_j \in A | a_j \in \gamma\tau(\{a_i\})\}$
    $(\forall a_i \in A)$.
    (2) Generate the topology $T(R_A) = \{\bigcup_{C(a_i) \in \mathcal{B}'_A} C(a_i) | \forall \mathcal{B}'_A \subseteq \mathcal{B}_A\}$ according to $\mathcal{B}_A$.
    (3) Generate $T(R_A)/\sim_\tau = \{[T] | T \in T(R_A), \forall T_1, T_2 \in [T], \tau(T_1) = \tau(T_2)\}$ according
    to $T(R_A)$ and the mapping $\tau$.
    (4) Generate $min[T] = \{T' \in [T] | \nexists T'' \in [T] \wedge T'' \subsetneq T'\}$ according to set inclusion for
    any $[T] \in T(R_A)/\sim_\tau$.
    (5) Generate $P \in min\mathcal{P}_\tau(T')$ according to set inclusion for any $T' \in min[T]$.
    (6) Select $P \in min\mathcal{P}_\tau(T')$ $(T' \in min[T])$, generate Min-Max association rule

                $\phi \equiv P \longrightarrow Q,$

    , in which $P \in min\mathcal{P}_\tau(T')$, $T' \in min[T]$, $Q = \cup[T] - P$ and $conf(\phi) = 1$.
    (7) Return to (6).
end while
Output All $\phi \equiv P \longrightarrow Q$ with $conf(\phi) = 1$.
```

The pseudocode provided in Algorithm 1, is responsible for mining Min-Max association rules from closed itemsets. Step (1) generates the base $\mathcal{B}$, step (2) generates the topology $T(R_A)$, step (3) generates the equivalent class $T(R_A)/\sim_\tau$, step (4) generates the minimum of $[T]$.

**Example 2.** *A transactional database is shown in Table 3. According to Definition 1, we obtain* $C(a_1) = \{a_1, a_5\}$, $C(a_2) = \{a_2, a_5\}$, $C(a_3) = \{a_3\}$, $C(a_4) = \{a_3, a_4\}$ *and* $C(a_5) = \{a_5\}$, *i.e., the base for the topology for the set of items is* $\mathcal{B}_A = \{\{a_1, a_5\}, \{a_2, a_5\}, \{a_3\}, \{a_3, a_4\}, \{a_5\}\}$, *the co-occurrence relation* $R_A$ *among items decided by Equation (1) is shown in Table 4. The topology* $T(R_A)$ *for the set of items generated by* $\mathcal{B}_A$ *and the quotient set* $T(R_A)/\sim_\tau$ *of the topology are*

$$\begin{aligned}
T(R_A) &= \{\emptyset, \{a_1, a_5\}, \{a_2, a_5\}, \{a_3\}, \{a_3, a_4\}, \{a_5\}, \{a_3, a_5\}, \{a_1, a_2, a_5\},\\
&\quad \{a_1, a_3, a_5\}, \{a_2, a_3, a_5\}, \{a_3, a_4, a_5\}, \{a_1, a_2, a_3, a_5\}, \{a_1, a_3, a_4,\\
&\quad a_5\}, \{a_2, a_3, a_4, a_5\}, A(= \{a_1, a_2, a_3, a_4, a_5\})\},\\
T(R_A)/\sim_\tau &= \{[\emptyset], [\{a_1, a_5\}], [\{a_2, a_5\}], [\{a_3\}], [\{a_3, a_4\}], [\{a_5\}], [\{a_3, a_5\}],\\
&\quad [\{a_1, a_2, a_5\}], [\{a_2, a_3, a_5\}], [\{a_3, a_4, a_5\}], [A]\}.
\end{aligned}$$

*The complete lattice* $(T(R_A)/\sim_\tau, \wedge, \vee)$ *is shown in Figure 3, in which, each equivalent class is with its support, e.g.,* $([\{a_5\}], 7)$ *means* $supp(\{a_5\}) = 7$*. Closed itemset and minimal generators of each equivalent class are shown in Table 5; accordingly, Min-Max association rules with confidence* $c = 1$ *generated from each equivalent class are shown in Table 6.*

**Table 3.** A transactional database $\mathscr{A} = (U, A)$.

| $U \backslash A$ | $a_1$ | $a_2$ | $a_3$ | $a_4$ | $a_5$ |
|---|---|---|---|---|---|
| $u_1$ | 1 | 0 | 1 | 0 | 1 |
| $u_2$ | 0 | 0 | 1 | 1 | 0 |
| $u_3$ | 0 | 1 | 0 | 0 | 1 |
| $u_4$ | 1 | 0 | 1 | 0 | 1 |
| $u_5$ | 0 | 0 | 1 | 1 | 0 |
| $u_6$ | 1 | 1 | 1 | 0 | 1 |
| $u_7$ | 1 | 0 | 1 | 1 | 1 |
| $u_8$ | 1 | 1 | 0 | 0 | 1 |
| $u_9$ | 1 | 0 | 0 | 0 | 1 |
| $u_{10}$ | 0 | 0 | 1 | 1 | 0 |

**Table 4.** A binary relation on $A$.

| $R_A$ | $a_1$ | $a_2$ | $a_3$ | $a_4$ | $a_5$ |
|---|---|---|---|---|---|
| $a_1$ | 1 | 0 | 0 | 0 | 1 |
| $a_2$ | 0 | 1 | 0 | 0 | 1 |
| $a_3$ | 0 | 0 | 1 | 0 | 0 |
| $a_4$ | 0 | 0 | 1 | 1 | 0 |
| $a_5$ | 0 | 0 | 0 | 0 | 1 |

**Table 5.** Closed itemset and minimal generators of each equivalent class.

| **Equivalent Class** | $\cup[T]$ | $min[T]$ | $min\mathcal{P}_\tau(T')$ |
|---|---|---|---|
| $[\{a_1, a_5\}]$ | $\{a_1, a_5\}$ | $\{a_1, a_5\}$ | $\{a_1\}$ |
| $[\{a_2, a_5\}]$ | $\{a_2, a_5\}$ | $\{a_2, a_5\}$ | $\{a_2\}$ |
| $[\{a_3, a_4\}]$ | $\{a_3, a_4\}$ | $\{a_3, a_4\}$ | $\{a_4\}$ |
| $[\{a_1, a_2, a_5\}]$ | $\{a_1, a_2, a_5\}$ | $\{a_1, a_2, a_5\}$ | $\{a_1, a_2\}$ |
| $[\{a_3, a_5\}]$ | $\{a_1, a_3, a_5\}$ | $\{a_3, a_5\}$ | $\{a_3, a_5\}$ |
| $[\{a_2, a_3, a_5\}]$ | $\{a_1, a_2, a_3, a_5\}$ | $\{a_2, a_3, a_5\}$ | $\{a_2, a_3\}$ |
| $[\{a_3, a_4, a_5\}]$ | $\{a_1, a_3, a_4, a_5\}$ | $\{a_3, a_4, a_5\}$ | $\{a_4, a_5\}$ |

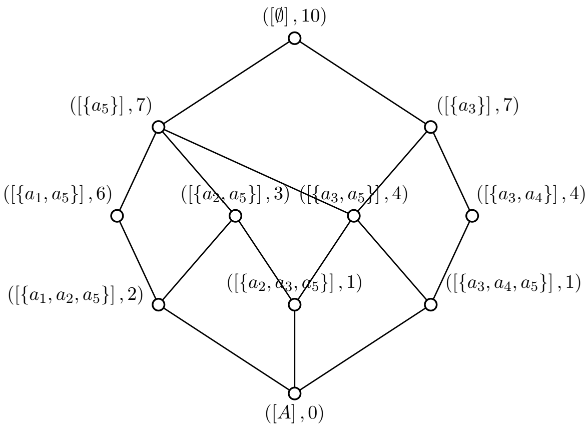

**Figure 3.** The complete lattice $(T(R_A)/\sim_\tau, \wedge, \vee)$ of Example 2.

**Table 6.** Min-Max association rules with support $s$ and confidence $c = 1$.

| Min-Max Association Rule | (Support, Confidence) |
|---|---|
| $a_1 \longrightarrow a_5$ | $(6, 1)$ |
| $a_2 \longrightarrow a_5$ | $(3, 1)$ |
| $a_4 \longrightarrow a_3$ | $(4, 1)$ |
| $a_1 \wedge a_2 \longrightarrow a_5$ | $(2, 1)$ |
| $a_3 \wedge a_5 \longrightarrow a_1$ | $(4, 1)$ |
| $a_2 \wedge a_3 \longrightarrow a_5$ | $(1, 1)$ |
| $a_2 \wedge a_3 \wedge a_5 \longrightarrow a_1$ | $(1, 1)$ |
| $a_2 \wedge a_3 \longrightarrow a_1 \wedge a_5$ | $(1, 1)$ |
| $a_4 \wedge a_5 \longrightarrow a_3$ | $(1, 1)$ |
| $a_3 \wedge a_4 \wedge a_5 \longrightarrow a_1$ | $(1, 1)$ |
| $a_4 \wedge a_5 \longrightarrow a_1 \wedge a_3$ | $(1, 1)$ |

In Table 6, $a_2 \wedge a_3 \longrightarrow a_5$ is generated by a minimal generator $\{a_2, a_3\}$ of $\{a_2, a_3, a_5\}$, $a_2 \wedge a_3 \wedge a_5 \longrightarrow a_1$ is generated by generator $\{a_2, a_3, a_5\}$ of $\{a_1, a_2, a_3, a_5\}$, which are association rules with confidence $c = 1$; however, they are not Min-Max association rules with confidence $c = 1$, only $a_2 \wedge a_3 \longrightarrow a_1 \wedge a_5$ is Min-Max association rules with confidence $c = 1$. Generally, there are many association rules with confidence $c = 1$ generated from each equivalent class; however, Min-Max association rules $P \longrightarrow Q$ with confidence $c = 1$ must be generated by $P \in min\mathcal{P}_\tau(T')$, $T' \in min[T]$ and $Q = \cup[T] - P$, i.e., the advantage of our method is that searching minimal generator is limited in $T' \in min[T]$ but not in all subsets of $\cup[T]$.

According to the hierarchical structures on closed itemsets displayed in the complete lattice $(T(R_A)/\sim_\tau, \wedge, \vee)$, we provide Algorithm 2 to generate Min-Max association rules with high support and confidence from a fixed itemset.

The pseudocode provided in Algorithm 2 is responsible for mining Min-Max association rules from a fixed itemsets. Step (1) generates $L(A_1)$, step (2) generates the set of equivalent classes, and step (4) generates the Min-Max association rules.

In Example 2, several fixed itemsets and their lower approximations are shown in Table 7, the corresponding Min-Max association rules generated by the itemsets are shown in Table 8.

**Algorithm 2** Min-Max association rules mining from a fixed itemset

```
Input: An itemset $A_1 = \{a_{i_1}, a_{i_2}, \cdots, a_{i_k}\} \subseteq A$ with $supp(A_1) \geq s$.
Output: Min-Max association rules with confidence $c \in (0, 1)$.
while The stop condition is not satisfied do
    (1) Generate $L(A_1)$ in $T(R_A)$.
    (2) Generate the set of equivalent classes such that $A_1^{\sim_\tau} = \{[T_i] | [T_i] \geq_{\sim_\tau} [L(A_1)]\}$.
    (3) Select $[T_i]$ from $A_1^{\sim_\tau}$ such that $\cup[T_i] \subseteq A_1$,
    (4) Generate association rule
            $\phi \equiv P \longrightarrow Q,$
where, $P \in min\mathcal{P}_\tau(T')$, $T' \in min[T_i]$, $Q = A_1 - P$ and $conf(\phi) = \frac{supp(A_1)}{supp(T_i)}$.
    (5) Return to (3).
end while
Output All $\phi \equiv P \longrightarrow Q$ with $supp(A_1) \geq s$ and $conf(\phi) = \frac{supp(A_1)}{supp(P)}$.
```

**Table 7.** Itemsets and their lower approximations of Example 2.

| Itemsets | Lower Approximations | The Set of Equivalent Classes |
|---|---|---|
| $\{a_1, a_3\}$ | $\{a_3\}$ | $\{[\{a_3\}]\}$ |
| $\{a_3, a_5\}$ | $\{a_3, a_5\}$ | $\{[\{a_3\}], [\{a_5\}]\}$ |
| $\{a_1, a_3, a_4\}$ | $\{a_3, a_4\}$ | $\{[\{a_3\}], [\{a_3, a_4\}]\}$ |

**Table 8.** Min-Max association rules generated by the itemsets.

| Itemsets | Min-Max Association Rules | (Support, Confidence) |
|---|---|---|
| $\{a_1, a_3\}$ | $a_3 \longrightarrow a_1$ | $(4, \frac{4}{7})$ |
| $\{a_3, a_5\}$ | $a_3 \longrightarrow a_5$ | $(4, \frac{4}{7})$ |
| $\{a_3, a_5\}$ | $a_5 \longrightarrow a_3$ | $(4, \frac{4}{7})$ |
| $\{a_1, a_3, a_4\}$ | $a_3 \longrightarrow a_1 \wedge a_4$ | $(1, \frac{1}{7})$ |
| $\{a_1, a_3, a_4\}$ | $a_4 \longrightarrow a_1 \wedge a_3$ | $(1, \frac{1}{4})$ |

Finally, advantages of Algorithms 1 and 2 can be summarized as follows:

1. Min-Max association rules are always mined from closed itemsets, in this paper, we prove that closed itemsets are maximum elements of equivalent classes, i.e., equivalent classes can be used to mine Min-Max association rules with confidence 1;
2. The shortest length antecedents of Min-Max association rules are searched from minimal members of equivalent classes, i.e., $P \in min\mathcal{P}_\tau(T')$ and $T' \in min[T]$; in this paper, searching minimal generators are in smaller scope than in all subsets of closed itemsets;
3. Lower approximations and their minimal generators help us to fast mine Min-Max association rules from a fixed itemset.

### *4.2. Generalized Association Rules Based on the Lower Approximation*

Generalized association rules are important extension of association rules. By using taxonomy over items of transactional databases, generalized items are aggregated according to different granularity levels. In practice, generalized itemsets provide a high level view of the patterns hidden in the analyzed data and a high level abstraction of the mined knowledge in different application domains [12,30,59,60]. In this paper, each equivalent class $[T]$ in $T(R_A)/\sim_\tau$ is understood as a granular knowledge on the set of itemsets; it can also be considered as a kind of generalized itemsets. Accordingly, we provide an alternative method to mine generalized association rules based on the lower approximation of any itemset, and analyze the changing confidence of generalized association rules. Formally, for any association rule $\psi \equiv A_1 \longrightarrow A_2$, we propose the following three kinds of generalized association rules of $\psi$ based on the lower approximation:

1. Generalized antecedent association rule (GAR): $\psi_{GA} \equiv L(A_1) \longrightarrow A_2$;
2. Generalized conclusion association rule (GCR): $\psi_{GC} \equiv A_1 \longrightarrow L(A_2)$;
3. Generalized antecedent and conclusion association rule (GACR): $\psi_{GAC} \equiv L(A_1) \longrightarrow L(A_2)$.

It is obvious that if $A_1 \in T(R_A)$, then $L(A_1) = A_1$ and $\psi = \psi_{GA}$; if $A_2 \in T(R_A)$, then $L(A_2) = A_2$ and $\psi = \psi_{GC}$; if $A_1, A_2 \in T(R_A)$, then $L(A_1) = A_1$, $L(A_2) = A_2$ and $\psi = \psi_{GAC}$.

**Corollary 4.** *For any association rule $\psi \equiv A_1 \longrightarrow A_2$, $conf_{\mathscr{A}}(\psi) = c$, $l = |\tau(L(A_1)) - \tau(A_1)|$ and $k = |(\tau(L(A_1)) - \tau(A_1)) \cap \tau(A_2)|$.*

1. *$conf_{\mathscr{A}}(\psi) \leq conf_{\mathscr{A}}(\psi_{GC})$;*
2. *$conf_{\mathscr{A}}(\psi_{GA}) \leq conf_{\mathscr{A}}(\psi_{GAC})$;*
3. *If $k \geq cl$, then $conf_{\mathscr{A}}(\psi) \leq conf_{\mathscr{A}}(\psi_{GA})$;*
4. *If $k < cl$, then $conf_{\mathscr{A}}(\psi) > conf_{\mathscr{A}}(\psi_{GA})$.*

**Proof.** Based on $\tau(A_1) \subseteq \tau(L(A_1))$ and $\tau(A_2) \subseteq \tau(L(A_2))$, (1) and (2) can be easily proved. For (3) and (4), due to $\tau(L(A_1)) = \tau(A_1) \cup (\tau(L(A_1)) - \tau(A_1))$ and $\tau(A_1) \cap (\tau(L(A_1)) - \tau(A_1)) = \emptyset$, we have $conf_{\mathscr{A}}(\psi_{GA}) = \frac{|\tau(L(A_1)) \cap \tau(A_2)|}{|\tau(L(A_1))|} = \frac{|(\tau(A_1) \cup (\tau(L(A_1)) - \tau(A_1))) \cap \tau(A_2)|}{|\tau(L(A_1))|} = \frac{|(\tau(A_1) \cap \tau(A_2)) \cup ((\tau(L(A_1)) - \tau(A_1)) \cap \tau(A_2))|}{|\tau(A_1) \cup (\tau(L(A_1)) - \tau(A_1))|} = \frac{k + |\tau(A_1) \cap \tau(A_2)|}{l + |\tau(A_1)|}$, by $\frac{|\tau(A_1) \cap \tau(A_2)|}{|\tau(A_1)|} = c$, if $k \geq cl$, then $\frac{k + |\tau(A_1) \cap \tau(A_2)|}{l + |\tau(A_1)|} \geq \frac{cl + c|\tau(A_1)|}{l + |\tau(A_1)|} = c$. If $k < cl$, then $\frac{k + |\tau(A_1) \cap \tau(A_2)|}{l + |\tau(A_1)|} < \frac{cl + c|\tau(A_1)|}{l + |\tau(A_1)|} = c$. □

**Corollary 5.** *Let $conf_{\mathscr{A}}(\psi_{GC}) = c$, $l = |\tau(L(A_1)) - \tau(A_1)|$ and $k = |(\tau(L(A_1)) - \tau(A_1)) \cap \tau(L(A_2))|$. 1) If $k \geq cl$, then $conf_{\mathscr{A}}(\psi_{GC}) \leq conf_{\mathscr{A}}(\psi_{GAC})$; 2) If $k < cl$, then $conf_{\mathscr{A}}(\psi_{GC}) > conf_{\mathscr{A}}(\psi_{GAC})$.*

Based on $P \in min\mathcal{P}_\tau(T')$, $T' \in min[L(A_1)]$ and $\cup[L(A_2)]$, $\psi_{GA}$, $\psi_{GC}$ and $\psi_{GAC}$ of association rule $\psi \equiv A_1 \longrightarrow A_2$ have the following redundant rules with the same confidence and the same support:

1. Redundant association rule of GAR: $\psi_{GAN} \equiv P \longrightarrow A_2$;
2. Redundant association rule of GCR: $\psi_{GCN} \equiv A_1 \longrightarrow \cup[L(A_2)]$;
3. Redundant association rule of GACR: $\psi_{GACN} \equiv P \longrightarrow \cup[L(A_2)]$.

Accordingly, we provide Algorithm 3 to mine generalized association rules of any association rule $\psi$ and its corresponding redundant association rules.

**Algorithm 3** Mining generalized association rules and redundant association rules based on the lower approximation

```
Input: Association rule $\psi \equiv A_1 \longrightarrow A_2$.
Output: Generalized association rules of $\psi$ and corresponding redundant association rules.
while The stop condition is not satisfied do
    (1) Generate $L(A_1)$ and $L(A_2)$ in $T(R_A)$.
    (2) Generate $\cup[L(A_1)]$ and $\cup[L(A_2)]$ according to set inclusion of $[L(A_1)]$ and $[L(A_2)]$, respectively.
    (3) Generate $T' \in min[L(A_1)]$ and $P \in min\mathcal{P}_\tau(T')$ according to set inclusion of $[L(A_1)]$ and the power set of $T'$, respectively.
    (4) Obtain generalized association rules $\psi_{GA} \equiv L(A_1) \longrightarrow A_2$, $\psi_{GC} \equiv A_1 \longrightarrow L(A_2)$ and $\psi_{GAC} \equiv L(A_1) \longrightarrow L(A_2)$, respectively.
    (5) Generate redundant association rules $\psi_{GAN} \equiv P \longrightarrow A_2$, $\psi_{GCN} \equiv A_1 \longrightarrow \cup[L(A_2)]$ and $\psi_{GACN} \equiv P \longrightarrow \cup[L(A_2)]$, respectively.
    (6) Return to (5).
end while
Output Generalized association rules and corresponding redundant association rules based on the lower approximation with $(supp(\psi_*), conf(\psi_*))$.
```

The pseudocode provided in Algorithm 3 is responsible for mining generalized association rules and redundant association rules based on the lower approximation. Step (1) generates $L(A_1)$ and $L(A_2)$, step (2) generates $\cup[L(A_1)]$ and $\cup[L(A_2)]$ , step (4) generates association rules.

### *4.3. Generalized Association Rules Based on the Upper Approximation*

Similarly, for any association rule $\psi \equiv A_1 \longrightarrow A_2$, we propose the following three kinds of generalized association rules of $\psi$ based on the upper approximation:

1. Generalized antecedent association rule (gar): $\psi_{ga} \equiv G(A_1) \longrightarrow A_2$;
2. Generalized conclusion association rule (gcr): $\psi_{gc} \equiv A_1 \longrightarrow G(A_2)$;
3. Generalized antecedent and conclusion association rule (gacr): $\psi_{gac} \equiv G(A_1) \longrightarrow G(A_2)$.

It is obvious that if $A_1 \in T(R_A)$, then $\psi = \psi_{ga}$; if $A_2 \in T(R_A)$, then $\psi = \psi_{gc}$; if $A_1, A_2 \in T(R_A)$, and then $\psi = \psi_{gac}$.

**Corollary 6.** *For $\psi \equiv A_1 \longrightarrow A_2$, $conf_{\mathscr{A}}(\psi) = c$, $l = |\tau(A_1) - \tau(G(A_1))|$ and $k = |(\tau(A_1) - \tau(G(A_1))) \cap \tau(A_2)|$.*

1. *$conf_{\mathscr{A}}(\psi) \geq conf_{\mathscr{A}}(\psi_{gc})$;*
2. *$conf_{\mathscr{A}}(\psi_{ga}) \geq conf_{\mathscr{A}}(\psi_{gac})$*
3. *If $k > cl$, then $conf_{\mathscr{A}}(\psi) > conf_{\mathscr{A}}(\psi_{ga})$;*
4. *If $k \leq cl$, then $conf_{\mathscr{A}}(\psi) \leq conf_{\mathscr{A}}(\psi_{ga})$.*

**Proof.** Based on $\tau(A_1) \supseteq \tau(G(A_1))$ and $\tau(A_2) \supseteq \tau(G(A_2))$, (1) and (2) can be easily proved. For (3) and (4), due to $\tau(A_1) = \tau(G(A_1)) \cup (\tau(A_1) - \tau(G(A_1)))$ and $\tau(G(A_1)) \cap (\tau(A_1) - \tau(G(A_1))) = \emptyset$, we have

$$\begin{aligned} conf_{\mathscr{A}}(\psi) &= \frac{|\tau(A_1) \cap \tau(A_2)|}{|\tau(A_1)|} = \frac{|(\tau(G(A_1)) \cup (\tau(A_1) - \tau(G(A_1)))) \cap \tau(A_2)|}{|\tau(G(A_1)) \cup (\tau(A_1) - \tau(G(A_1)))|} \\ &= \frac{|(\tau(G(A_1)) \cap \tau(A_2)) \cup ((\tau(A_1) - \tau(G(A_1))) \cap \tau(A_2))|}{|\tau(G(A_1)) \cup (\tau(A_1) - \tau(G(A_1)))|} \\ &= \frac{|\tau(G(A_1)) \cap \tau(A_2)| + k}{|\tau(G(A_1))| + l} = c, \end{aligned}$$

i.e., $|\tau(G(A_1)) \cap \tau(A_2)| - c|\tau(G(A_1))| = cl - k$. If $k > cl$, then $|\tau(G(A_1)) \cap \tau(A_2)| - c|\tau(G(A_1))| < 0$, i.e., $conf_{\mathscr{A}}(\psi_{ga}) = \frac{|\tau(G(A_1)) \cap \tau(A_2)|}{|\tau(G(A_1))|} < c$. If $k \leq cl$, then $|\tau(G(A_1)) \cap \tau(A_2)| - c|\tau(G(A_1))| \geq 0$, i.e., $conf_{\mathscr{A}}(\psi_{ga}) = \frac{|\tau(G(A_1)) \cap \tau(A_2)|}{|\tau(G(A_1))|} \geq c$. □

**Corollary 7.** *Let $conf_{\mathscr{A}}(\psi_{gc}) = c$, $l = |\tau(A_1) - \tau(G(A_1))|$ and $k = |(\tau(A_1) - \tau(G(A_1))) \cap \tau(G(A_2))|$. (1) If $k > cl$, then $conf_{\mathscr{A}}(\psi_{gc}) > conf_{\mathscr{A}}(\psi_{gac})$; (2) If $k \leq cl$, then $conf_{\mathscr{A}}(\psi_{gc}) \leq conf_{\mathscr{A}}(\psi_{gac})$.*

Based on $P \in min\mathcal{P}_\tau(T')$, $T' \in min[G(A_1)]$ and $\cup[G(A_2)]$, $\psi_{ga}$, $\psi_{gc}$ and $\psi_{gac}$ of association rule $\psi \equiv A_1 \longrightarrow A_2$ have the following redundant rules with the same confidence and support:

1. Redundant association rule of gar: $\psi_{gan} \equiv P \longrightarrow A_2$;
2. Redundant association rule of gcr: $\psi_{gcn} \equiv A_1 \longrightarrow \cup[G(A_2)]$;
3. Redundant association rule of gacr: $\psi_{gacn} \equiv P \longrightarrow \cup[G(A_2)]$.

Accordingly, we provide Algorithm 4 to mine generalized association rules of $\psi$ and corresponding redundant association rules.

The pseudocode provided in Algorithm 4, is responsible for mining generalized association rules and redundant association rules based on the upper approximation. Step (1) generates $G(A_1)$ and $G(A_2)$ , step (2) generates $\cup[G(A_1)]$ and $\cup[G(A_2)]$, step (4) generates association rules, and step (5) generates redundant association rules.

**Algorithm 4** Mining generalized association rules and redundant association rules based on the upper approximation

```
Input: Association rule $\psi \equiv A_1 \longrightarrow A_2$.
Output: Generalized association rules of $\psi$ and corresponding redundant association rules.
while The stop condition is not satisfied do
    (1) Generate $G(A_1)$ and $G(A_2)$ in $T(R_A)$.
    (2) Generate $\cup[G(A_1)]$ and $\cup[G(A_2)]$ according to set inclusion of $[G(A_1)]$ and $[G(A_2)]$, respectively.
    (3) Generate $T' \in min[G(A_1)]$ and $P \in min\mathcal{P}_\tau(T')$ according to set inclusion of $[G(A_1)]$ and the power set of $T'$, respectively.
    (4) Obtain generalized association rules $\psi_{gar} \equiv G(A_1) \longrightarrow A_2$, $\psi_{gcr} \equiv A_1 \longrightarrow G(A_2)$ and $\psi_{gacr} \equiv G(A_1) \longrightarrow G(A_2)$, respectively.
    (5) Generate redundant association rules $\psi_{gan} \equiv P \longrightarrow A_2$, $\psi_{gcn} \equiv A_1 \longrightarrow \cup[G(A_2)]$ and $\psi_{gacn} \equiv P \longrightarrow \cup[G(A_2)]$, respectively.
    (6) Return to (5).
end while
Output Generalized association rules and corresponding redundant association rules based on the upper approximation with $(supp(\psi_*), conf(\psi_*))$.
```

## 5. Example Analysis

Experiments were made to compare the execution time, memory usage and numbers of association rules of the Apriori [1,61] algorithms and our method. They were implemented on a Thinkpad X1 laptop with Intel i5 Core Duo (2 × 2.4 GHz), 4 GB of RAM and running Windows 10. The algorithms were coded in Matlab 2015b. Four databases from the UCI databases [62] were used for the experiments, of which the features are shown in Table 9.

### *5.1. The Execution Time*

Experiments were made to compare the execution time of the algorithms Apriori and ours.The minConf was set to 50%. The results of the four databases for various minSup values are shown in Figures 4–7, which shows that the execution of our algorithm was faster than Apriori in all cases. For example, given $minSup = 75\%$, for the Chess database, the mining time of Apriori and ours were 3035.53($s$) and 507.33($s$), respectively. The time ratio is $\frac{507.33}{3035.53} \times 100\% = 16.71\%$. Besides, as the minSup is decreased, the time ratio is reduced also. For example, consider the Connect database with $minSup$ set at 98%, 96% and 94%, the speed up of the time ratio were 91.93%, 7.26% and 1.56%, respectively. The results demonstrate that the execution time of mining rules from ours was more efficient than Apriori.

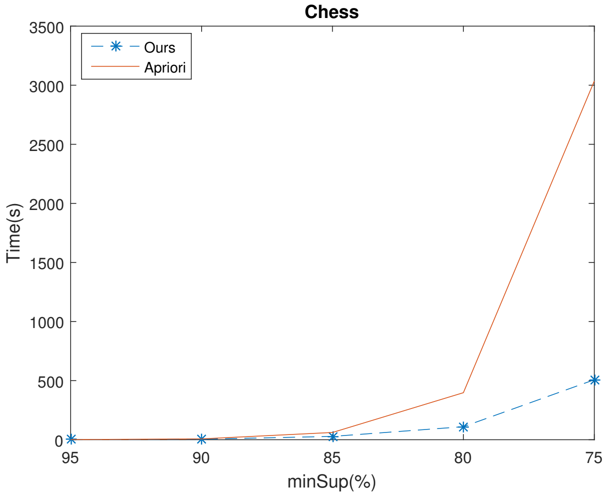

**Figure 4.** Execution time of the two algorithms in Chess dataset for various minSup values.

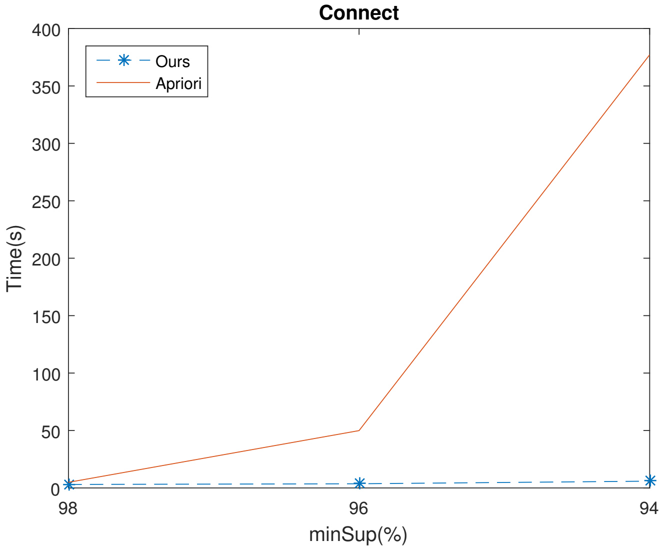

**Figure 5.** Execution time of the two algorithms in Connect dataset for various minSup values.

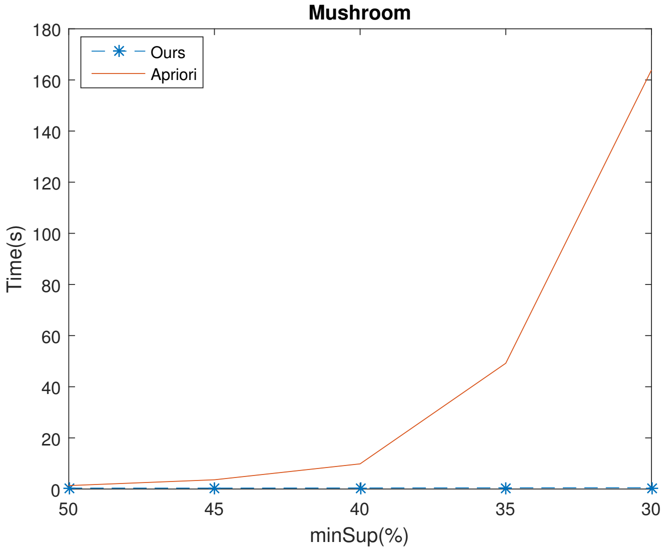

**Figure 6.** Execution time of the two algorithms in Mushroom dataset for various minSup values.

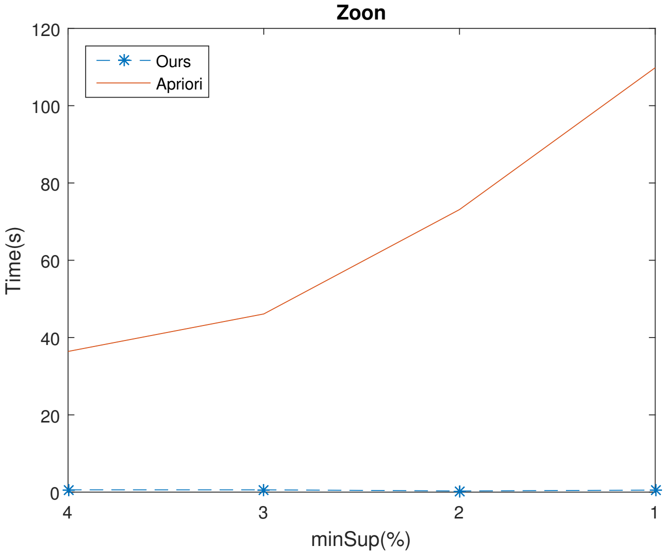

**Figure 7.** Execution time of the two algorithms in Zoon dataset for various minSup values.

### *5.2. The Memory Usage*

Experiments were made with the same databases and same parameters as the execution time experiments. The results show that the Apriori algorithm consumed more memory than ours in almost all cases, which are shown if Figures 8–11. For example, given $minSup = 75\%$, for Chess database, the memory usage of Apriori and ours were 672 (Mb) and 136.54 (Mb), respectively. The time ratio is $\frac{136.54}{672} \times 100\% = 20.32\%$. Moreover, when we decrease the $minSup$, the time ratio will reduce as well. For example, consider the Connect database with $minSup$ set at 98%, 96% and 94%, the speed up of the time ratio were 87.55%, 75.84% and 9.18%, respectively.

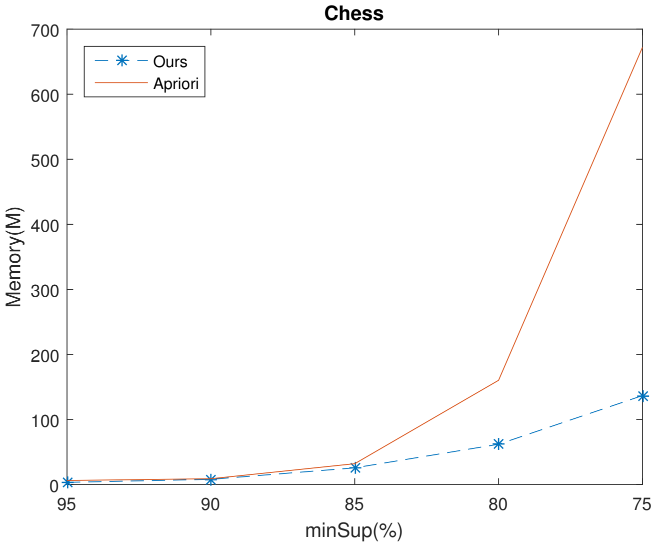

**Figure 8.** Memory usage of the two algorithms in Chess dataset for various minSup values.

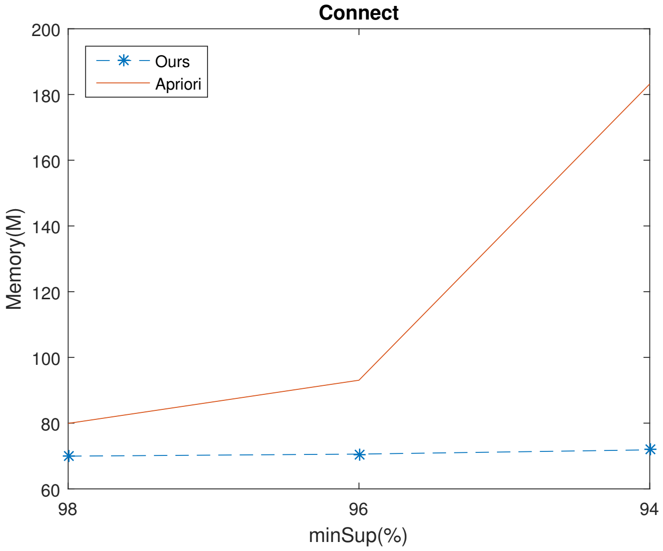

**Figure 9.** Memory usage of the two algorithms in Connect dataset for various minSup values.

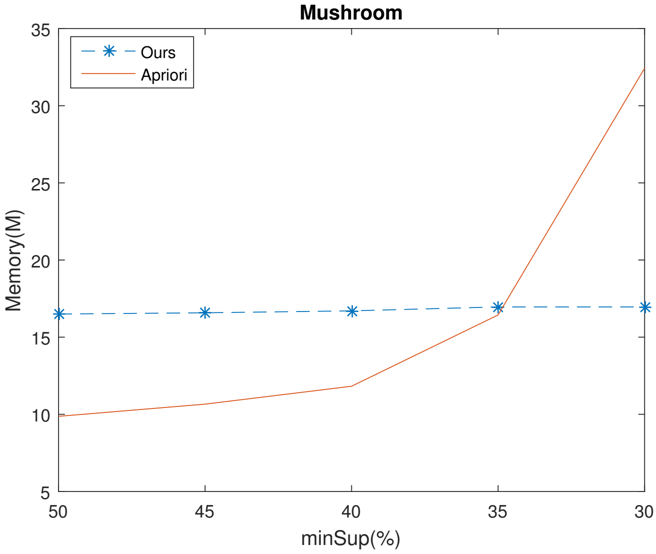

**Figure 10.** Memory usage of the two algorithms in Mushroom dataset for various minSup values.

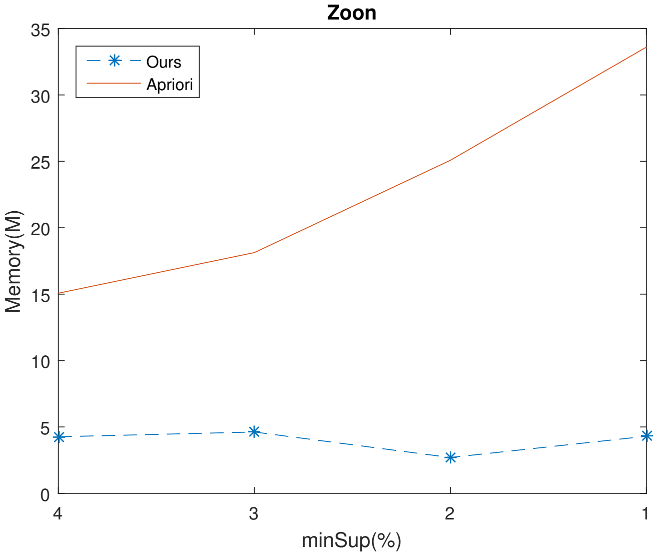

**Figure 11.** Memory usage of the two algorithms in Zoon dataset for various minSup values.

### 5.3. Numbers of Rules

We compare the numbers of association rules of Apriori with ours with the same databases and same parameters as the execution time experiments in Table 10. The results demonstrate that the numbers of ours is always smaller than those of Apriori. For example, given $minSup = 75\%$, for Chess database, the number of rules of Apriori and ours were 2,336,556 and 253,836, respectively. The time ratio is $\frac{253836}{2336556} \times 100\% = 10.86\%$. For our algorithm generating Min-Max association rules, the numbers of rules of ours is smaller than Apriori.

**Table 9.** Dataset characteristics.

| Dataset | Transactions | Original Attributes | Attributes after Conversion |
|---|---|---|---|
| Zoo | 101 | 17 | 15 |
| Mushroom | 8124 | 23 | 126 |
| Connect-4 | 67,557 | 43 | 129 |
| Chess | 3196 | 36 | 108 |

**Table 10.** Number of rules of Apriori and Ours.

| Dataset | $minSup(\%)$ | Number of Rules | |
|---|---|---|---|
| | | Apriori | Ours |
| Chess | 95 | 472 | 700 |
| | 90 | 10,742 | 9482 |
| | 85 | 95,482 | 43,116 |
| | 80 | 552,564 | 111,768 |
| | 75 | 2,336,556 | 253,836 |
| Mushroom | 50 | 1146 | 172 |
| | 45 | 2704 | 291 |
| | 40 | 5006 | 483 |
| | 35 | 14,107 | 903 |
| | 30 | 45,145 | 903 |
| Connect | 98 | 1544 | 380 |
| | 96 | 27,340 | 1480 |
| | 94 | 201,928 | 3848 |
| Zoon | 4 | 29,288 | 8136 |
| | 3 | 35,204 | 8826 |
| | 2 | 48,578 | 5110 |
| | 1 | 64,868 | 8174 |

In this section, some examples are provided to show our method in association rules mining; transactional databases come from [62] and their characteristics are shown in Table 9. The Zoo database is initially used for classification of animals; in this paper, association rules among attributes of animals are considered, in which Min-Max association rules with confidence $c = 1$ and confidence $c \in (0, 1)$ will be shown, respectively. For mushroom database, connect-4 and chess, the topology, basis for the topology and equivalent classes with thresholds of support are obtained, and Min-Max association rules and generalized association rules are generated. Because the Reliable basis used to retrieve and reduce association rules in [63] is similar to our method, our results are compared with the method based on the Reliable basis limited in the number of Min-Max association rules and redundant rules.

**Example 3.** *In the Zoo database, there are 101 objects (animals) and 17 attributes (15 boolean, 2 numeric). Here, we discover association rules among 15 boolean attributes of animals, i.e., hair (ha), feathers ($fe$), eggs (eg), milk (mi), airborne (ai), aquatic (aq), predator (pr), toothed (to), backbone (ba), breathes (br), venomous (ve), fins ($fi$), tail (ta), domestic (do) and catsize (ca).*

*According to Definition 1, 15 bases are generated (shown in Table A1 of Appendix A). Members of topology and equivalent classes with $supp_r(*)$ are shown in Table A2 of Appendix A. Equivalent classes used to generate association rules with confidence $c = 1$ are shown in Table A3 of Appendix A. According to Table A3, 7 Min-Max association rules with $supp_r(*) \geq 0.4$ and confidence $c = 1$ are generated (shown in Table A4 of Appendix A), in which $mi \longrightarrow ba \wedge br$ (rule 4) is considered redundant of $mi \wedge ba \longrightarrow br$ (rule 5) and $mi \wedge br \longrightarrow ba$ (rule 6) due to $\{mi\} \subset \{mi, ba\}$, $\{mi\} \subset \{mi, br\}$, $\{br\} \subset \{ba, br\}$ and $\{ba\} \subset \{ba, br\}$; i.e., we use less conditions to achieve more results by $mi \longrightarrow ba \wedge br$ with the same support and confidence. Furthermore, some Min-Max association rules generated in the quotient lattice of the topology with $supp_r(*) \geq 0.4$ and confidence $c \in (0, 1)$ are shown in Table A5 of Appendix A, in which the Min-Max association rule $br \longrightarrow ba$ and $ta \longrightarrow ba$ can be considered as a GAR of $br \wedge ta \longrightarrow ba$, respectively.*

**Example 4.** *The mushroom database consists of a database with 8124 objects (mushrooms) and 22 nominally valued attributes. Here, we convert 22 nominally valued attributes as 126 boolean attributes and generate 111 bases. Change of bases with different $supp_r(*)$ is shown in Figure A1 of Appendix A. Members of topology, equivalent classes and its generating time by using different bases are shown in Figure A2 of Appendix A.*

*Table A6 of Appendix A shows Min-Max association rules with* $supp_r(*) \geq 0.8$ *and* $c = 1$. *Compared with results in [63], rules 1–8 and 11 are same with [63], and rules 9, 10, 12 and 13 are new rules, in which rule 8 can be considered as a redundant rule of rules 9 and 10, rule 11 being a redundant rule of rules 12 and 13.*

*Table A7 of Appendix A show Min-Max association rules with* $supp_r(*) \geq 0.8$ *and* $c \in [0.8, 1)$, *in which rules 1–25 are also generated in [63]; rules 26-41 are new rules.*

*Table A8 of Appendix A shows comparative results, in which* $TRX$ *means the number of all approximate rules in [63], Min-Max* $(N, Rt)$ *and Reliable* $(N, Rt)$ *means number and reduction ratio of Min-Max basis rules and reliable basis rules generated by Min-Max approximate basis, Min-Max exact basis, reliable approximate basis and reliable exact basis in [63].* $TR$ *means the number of all rules generated in the quotient lattice* $(T(R_A)/\sim_\tau, \wedge, \vee)$, *N-Min-Max* $(N, Rt)$ *means the number and reduction ratio of Min-Max association rules generated in* $(T(R_A)/\sim_\tau, \wedge, \vee)$, *the reduction ratio is calculated as* $Rt = \frac{TRX-N}{TRX}$ *or* $\frac{TR-N}{TR}$. *In Table A8, it can be noticed that the association rules used to obtain a reduction in the quotient lattice* $(T(R_A)/\sim_\tau, \wedge, \vee)$ *are less than in [63], such as for* $supp_r(*) \geq 0.4$, *2528 association rules are used to obtain reduction. The lower* $supp_r(*)$ *and confidence c is corresponding to more generated association rules and higher reduction ratio in the quotient lattice* $(T(R_A)/\sim_\tau, \wedge, \vee)$, *such as for* $supp_r(*) \geq 0.4$ *and confidence* $c \geq 0.5$, *1825 rules are generated and the reduction ratio is 0.77.*

**Example 5.** *In the connect-4 database, the number of reduction association rules with confidence* $c \geq 0.5$ *and compared with in [63] are shown in Table A9 of Appendix A. For template* $A = \{d_5 - b, d_6 - b, e_5 - b, f_5 - b, f_6 - b\}$ *and rule* $\phi \equiv a_6 - b \wedge e_4 - b \wedge f_5 - b \longrightarrow e_6 - b \wedge f_6 - b$,

$$\begin{aligned}
L(\{d_5 - b, d_6 - b, e_5 - b, f_5 - b, f_6 - b\}) &= \{d_5 - b, d_6 - b, f_5 - b, f_6 - b\}, \\
min[\{d_5 - b, d_6 - b, f_5 - b, f_6 - b\}] &= \{d_5 - b, f_5 - b\}, \\
L(\{a_6 - b, e_4 - b, f_5 - b\}) &= \{a_6 - b\}, \\
G(\{a_6 - b, e_4 - b, f_5 - b\}) &= \{a_6 - b, e_4 - b, e_5 - b, f_5 - b\}, \\
L(\{e_6 - b, f_6 - b\}) &= \{e_6 - b, f_6 - b\}, \\
G(\{e_6 - b, f_6 - b\}) &= \{e_6 - b, f_6 - b\}.
\end{aligned}$$

*Accordingly, Table A10 shows the association rule of the template and generalized association rules in the connect-4 database.*

*In the chess database, the number of reduction association rules with confidence* $c \geq 0.5$ *and compared with in [63] are shown in Table A11 of Appendix A. For template* $A = \{mulch\_f, skach\_f, wkna8\_f\}$ *and rule* $\phi \equiv qxmsq\_f \wedge spcop\_f \wedge wkna8\_f \longrightarrow bkon8\_f \wedge hdchk\_f$,

$$\begin{aligned}
L(\{mulch\_f, skach\_f, wkna8\_f\}) &= \{mulch\_f, skach\_f\}, \\
min[\{mulch\_f, skach\_f\}] &= \{mulch\_f, skach\_f\}, \\
L(\{qxmsq\_f, spcop\_f, wkna8\_f\}) &= \{qxmsq\_f, spcop\_f\}, \\
G(\{qxmsq\_f, spcop\_f, wkna8\_f\}) &= \{qxmsq\_f, spcop\_f, wkna8\_f, stlmt\_f\}, \\
L(\{bkon8\_f, hdchk\_f\}) &= \{bkon8\_f, hdchk\_f, reskd\_f\}, \\
G(\{bkon8\_f, hdchk\_f\}) &= \{bkon8\_f, hdchk\_f, reskd\_f, thrsk\_f, spcop\_f\}.
\end{aligned}$$

*Accordingly, Table A12 shows the association rule of the template and generalized association rules in the chess database.*

## 6. Conclusions

Association rules are often generated by frequent itemsets or closed itemsets from transactional databases. In order to obtain these itemsets, many methods have been proposed. In this article, for representing more general associative relations, which are among the items of transaction databases, the topology on itemset of a transactional database can been constructed. The topology on itemset includes frequent itemsets and closed itemsets. The topology on itemset includes frequent itemsets and closed itemsets, which has been proved.

Most important of all, the basis of the topology can be used to generate the topology on the itemset, which deduced from the transactional database. Using the basis of the topology can efficiently avoid scanning databases many times in extracting association rules. The quotient lattice of the topology displays the hierarchical structures on all itemsets, because every closed itemset and its generators or minimal generators are limited in an element of the quotient lattice, valid Min-Max association rules can be easily generated in the element. Because the quotient lattice of the topology provides granular concepts to approximate any template of itemset, reductant association rules can be easily generated by granular concepts. The experiment demonstrates that association rules mining using topology for itemset is an efficient method.

**Author Contributions:** Conceptualization, Z.P.; methodology, Z.P. and B.L.; validation, C.Z. and F.H.; Writing—original draft, B.L.; Writing—review and editing, Z.P., formal analysis, Z.P. and B.L.; All authors have read and agreed to the published version of the manuscript.

**Funding:** This work has been partially supported by Talent introduction project of Xihua University (Z202104) and the Opening Project of Intelligent Policing Key Laboratory of Sichuan Province (Grant no. ZNJW2022KFMS004, ZNJW2022KFQN002, ZNJW2023KFQN007).

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Appendix A

**Table A1.** Basis $\mathcal{B}_A$ of Zoo database.

| **Attributes** | $ha$ | $fe$ | $eg$ | $mi$ | $ai$ |
|---|---|---|---|---|---|
| Basis | $\{ha, br\}$ | $\{fe, eg, ba, br, ta\}$ | $\{eg\}$ | $\{mi, ba, br\}$ | $\{ai, br\}$ |
| Attributes | $aq$ | $pr$ | $to$ | $ba$ | $br$ |
| Basis | $\{aq\}$ | $\{pr\}$ | $\{to, ba\}$ | $\{ba\}$ | $\{br\}$ |
| Attributes | $ve$ | $fi$ | $ta$ | $do$ | $ca$ |
| Basis | $\{ve\}$ | $\{aq, to, ba, fi\}$ | $\{ta\}$ | $\{do\}$ | $\{ca\}$ |

**Table A2.** Members of topology and equivalent classes with $supp_r(*) \geq c$.

| $c$ | **0** | **0.2** | **0.3** | **0.5** |
|---|---|---|---|---|
| Members of topology | 4159 | 98 | 54 | 11 |
| Equivalent classes | 237 | 75 | 44 | 11 |

**Table A3.** Equivalent classes used to generate association rules with $supp_r(*) \geq 0.3$ and confidence $c = 1$.

| Equivalent Classes | $\|[*]\|$ | $min[*]$ | $\mathcal{P}_\tau(T')$ |
|---|---|---|---|
| $[\{ba, br, ca\}]$ | 2 | $\{\{br, ca\}\}$ | $\{\{br, ca\}\}$ |
| $[\{ba, br, ta, ca\}]$ | 2 | $\{\{br, ta, ca\}\}$ | $\{\{br, ta, ca\}\}$ |
| $[\{ba, br, to\}]$ | 1 | $\{\{ba, br, to\}\}$ | $\{\{br, to\}\}$ |
| $[\{ha, br\}]$ | 1 | $\{\{ha, br\}\}$ | $\{\{ha\}\}$ |
| $[\{to, ba, br, ta\}]$ | 1 | $\{\{to, ba, br, ta\}\}$ | $\{\{br, to, ta\}\}$ |
| $[\{mi, to, ba, br\}]$ | 1 | $\{\{mi, to, ba, br\}\}$ | $\{\{mi, to\}, \{mi, to, ba\}, \{mi, to, br\}\}$ |
| $[\{ha, mi, ba, br\}]$ | 2 | $\{\{ha, ba, br\}\}$ | $\{\{ha, ba\}, \{ha, ba, br\}\}$ |
| $[\{mi, to, ba, br, ta\}]$ | 1 | $\{\{mi, to, ba, br, ta\}\}$ | $\{\{mi, to, ta\}, \{mi, to, ba, ta\}, \{mi, to, br, ta\}\}$ |
| $[\{eg, ba, ta\}]$ | 2 | $\{\{eg, ta\}\}$ | $\{\{eg, ta\}\}$ |
| $[\{ha, mi, ba, br, ta\}]$ | 3 | $\{\{ha, br, ta\}\}$ | $\{\{ha, ta\}, \{ha, br, ta\}\}$ |
| $[\{ha, mi, to, ba, br, ta\}]$ | 2 | $\{\{ha, to, ba, br, ta\}\}$ | $\{\{ha, to, ta\}, \{ha, to, ba, br, ta\}, \{ha, to, br, ta\}, \{ha, to, ba, ta\}\}$ |
| $[\{pr, to, ba, ta\}]$ | 1 | $\{\{pr, to, ba, ta\}\}$ | $\{\{pr, to, ta\}\}$ |
| $[\{to, ba\}]$ | 1 | $\{\{to, ba\}\}$ | $\{\{to\}\}$ |
| $[\{pr, to, ba\}]$ | 1 | $\{\{pr, to, ba\}\}$ | $\{\{pr, to\}\}$ |
| $[\{mi, ba, br\}]$ | 1 | $\{\{mi, ba, br\}\}$ | $\{\{mi\}, \{mi, ba\}, \{mi, br\}\}$ |
| $[\{mi, ba, br, ta\}]$ | 1 | $\{\{mi, ba, br, ta\}\}$ | $\{\{mi, ta\}, \{mi, ba, ta\}, \{mi, br, ta\}\}$ |
| $[\{to, ba, ca\}]$ | 1 | $\{\{to, ba, ca\}\}$ | $\{\{to, ca\}\}$ |
| $[\{mi, ba, br, ca\}]$ | 1 | $\{\{mi, ba, br, ca\}\}$ | $\{\{mi, ca\}, \{mi, ba, ca\}, \{mi, br, ca\}\}$ |
| $[\{mi, to, ba, br, ca\}]$ | 2 | $\{\{to, ba, br, ca\}\}$ | $\{\{to, ba, ca\}, \{to, ba, br, ca\}\}$ |
| $[\{to, ba, ta\}]$ | 1 | $\{\{to, ba, ta\}\}$ | $\{\{to, ta\}\}$ |
| $[\{ha, mi, to, ba, br\}]$ | 2 | $\{\{ha, to, ba, br\}\}$ | $\{\{ha, to\}, \{ha, to, ba\}, \{ha, to, br\}, \{ha, to, ba, br\}\}$ |
| $[\{ba, ta, ca\}]$ | 2 | $\{\{ta, ca\}\}$ | $\{\{ta, ca\}\}$ |

**Table A4.** All Min-Max association rules with $supp_r(*) \geq 0.4$ and $c = 1$.

| Numbers | Rule $\phi(supp_r(\phi), c)$ |
|---|---|
| 1 | $br \wedge to \longrightarrow ba\ (0.4653, 1)$ |
| 2 | $ha \longrightarrow br\ (0.4257, 1)$ |
| 3 | $to \longrightarrow ba\ (0.604, 1)$ |
| 4 | $mi \longrightarrow ba \wedge br\ (0.4059, 1)$ |
| 5 | $mi \wedge ba \longrightarrow br\ (0.4059, 1)$ |
| 6 | $mi \wedge br \longrightarrow ba\ (0.4059, 1)$ |
| 7 | $to \wedge ta \longrightarrow ba\ (0.5149, 1)$ |

**Table A5.** Some Min-Max association rules with $supp_r(*) \geq 0.4$ and $0.8 \leq c < 1$.

| Numbers | Rule $\phi(supp_r(\phi), c)$ |
|---|---|
| 1 | $pr \wedge ba \longrightarrow ta\ (0.4059, 0.8723)$ |
| 2 | $pr \wedge ta \longrightarrow ba\ (0.4059, 0.9762)$ |
| 3 | $ta \longrightarrow br\ (0.604, 0.8133)$ |
| 4 | $pr \longrightarrow ba\ (0.4653, 0.8393)$ |
| 5 | $ca \longrightarrow ba\ (0.4257, 0.9773)$ |
| 6 | $to \longrightarrow ba \wedge ta\ (0.5149, 0.8525)$ |
| 7 | $to \wedge ba \longrightarrow ta\ (0.5149, 0.8525)$ |
| 8 | $ba \longrightarrow br\ (0.6832, 0.8313)$ |
| 9 | $br \longrightarrow ba\ (0.6832, 0.8625)$ |
| 10 | $ba \longrightarrow ta\ (0.7327, 0.8916)$ |
| 11 | $ta \longrightarrow ba\ (0.7327, 0.9867)$ |
| 12 | $ta \longrightarrow ba \wedge br\ (0.5941, 0.8)$ |
| 13 | $ba \wedge br \longrightarrow ta\ (0.5941, 0.8696)$ |
| 14 | $ba \wedge ta \longrightarrow br\ (0.5941, 0.8108)$ |
| 15 | $br \wedge ta \longrightarrow ba\ (0.5941, 0.9836)$ |

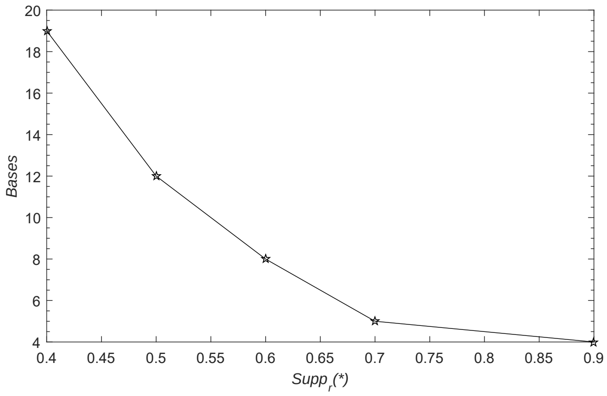

**Figure A1.** Change of bases in the mushroom database corresponding to $supp_r(*)$.

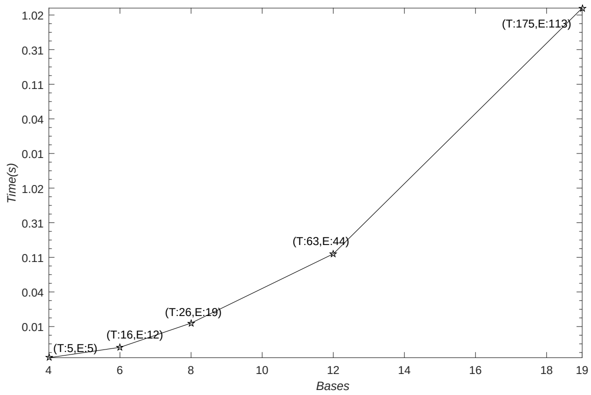

**Figure A2.** Members of topology, equivalent classes and its generating time in the mushroom database corresponding to bases.

**Table A6.** All Min-Max association rules in the mushroom database with $supp_r(*) \geq 0.8$ and $c = 1$.

| $N$ | **Rule** $\boldsymbol{\phi(supp_r(\phi), c)}$ |
|---|---|
| 1 | $gill\_attachment\_f \longrightarrow veil\_type\_p$ (0.9742, 1) |
| 2 | $gill\_spacing\_c \longrightarrow veil\_type\_p$ (0.8385, 1) |
| 3 | $veil\_color\_w \longrightarrow veil\_type\_p$ (0.9754, 1) |
| 4 | $ring\_number\_o \longrightarrow veil\_type\_p$ (0.9217, 1) |
| 5 | $gill\_attachment\_f \wedge veil\_color\_w \longrightarrow veil\_type\_p$ (0.9732, 1) |
| 6 | $gill\_attachment\_f \wedge ring\_number\_o \longrightarrow veil\_type\_p$ (0.8981, 1) |
| 7 | $gill\_spacing\_c \wedge veil\_color\_w \longrightarrow veil\_type\_p$ (0.8149, 1) |
| 8 | $gill\_attachment\_f \wedge gill\_spacing\_c \longrightarrow veil\_color\_w \wedge veil\_type\_p$ (0.8127, 1) |
| 9 | $gill\_attachment\_f \wedge gill\_spacing\_c \wedge veil\_type\_p \longrightarrow veil\_color\_w$ (0.8127, 1) |
| 10 | $gill\_attachment\_f \wedge gill\_spacing\_c \wedge veil\_color\_w \longrightarrow veil\_type\_p$ (0.8127, 1) |
| 11 | $veil\_color\_w \wedge ring\_number\_o \longrightarrow gill\_attachment\_f \wedge veil\_type\_p$ (0.8971, 1) |
| 12 | $gill\_attachment\_f \wedge veil\_color\_w \wedge ring\_number\_o \longrightarrow veil\_type\_p$ (0.8971, 1) |
| 13 | $veil\_type\_p \wedge veil\_color\_w \wedge ring\_number\_o \longrightarrow gill\_attachment\_f$ (0.8971, 1) |

**Table A7.** Min-Max association rules with $supp_r(*) \geq 0.8$ and $0.8 \leq c < 1$.

| Numbers | Rule $\boldsymbol{\phi(supp_r(\phi), c)}$ |
|---|---|
| 1 | $veil\_type\_p \longrightarrow gill\_attachment\_f$ (0.9742, 0.9742) |
| 2 | $veil\_type\_p \longrightarrow gill\_spacing\_c$ (0.8385, 0.8385) |
| 3 | $veil\_type\_p \longrightarrow veil\_color\_w$ (0.9754, 0.9754) |
| 4 | $veil\_type\_p \longrightarrow ring\_number\_o$ (0.9217, 0.9217) |
| 5 | $veil\_type\_p \longrightarrow gill\_attachment\_f \wedge veil\_color\_w$ (0.9732, 0.9732) |
| 6 | $veil\_type\_p \longrightarrow gill\_attachment\_f \wedge ring\_number\_o$ (0.8981, 0.8981) |
| 7 | $veil\_type\_p \longrightarrow gill\_spacing\_c \wedge veil\_color\_w$ (0.8149, 0.8149) |
| 8 | $veil\_type\_p \longrightarrow gill\_attachment\_f \wedge gill\_spacing\_c \wedge veil\_color\_w$ (0.8127, 0.8127) |
| 9 | $veil\_type\_p \longrightarrow gill\_attachment\_f \wedge veil\_color\_w \wedge ring\_number\_o$ (0.8971, 0.8971) |
| 10 | $gill\_attachment\_f \longrightarrow veil\_type\_p \wedge veil\_color\_w$ (0.9732, 0.999) |
| 11 | $gill\_attachment\_f \longrightarrow veil\_type\_p \wedge ring\_number\_o$ (0.8981, 0.9219) |
| 12 | $gill\_attachment\_f \longrightarrow gill\_spacing\_c \wedge veil\_color\_w \wedge veil\_type\_p$ (0.8127, 0.8342) |
| 13 | $gill\_attachment\_f \longrightarrow veil\_color\_w \wedge ring\_number\_o \wedge veil\_type\_p$ (0.8971, 0.9209) |
| 14 | $gill\_spacing\_c \longrightarrow gill\_attachment\_f \wedge veil\_color\_w \wedge veil\_type\_p$ (0.8127, 0.9692) |
| 15 | $gill\_spacing\_c \longrightarrow veil\_color\_w \wedge veil\_type\_p$ (0.8149, 0.9718) |
| 16 | $veil\_color\_w \longrightarrow gill\_spacing\_c \wedge veil\_type\_p$ (0.8149, 0.8354) |
| 17 | $veil\_color\_w \longrightarrow gill\_attachment\_f \wedge gill\_spacing\_c \wedge veil\_type\_p$ (0.8127, 0.8332) |
| 18 | $veil\_color\_w \longrightarrow gill\_attachment\_f \wedge veil\_type\_p$ (0.9732, 0.9977) |
| 19 | $veil\_color\_w \longrightarrow gill\_attachment\_f \wedge ring\_number\_o \wedge veil\_type\_p$ (0.8971, 0.9197) |
| 20 | $ring\_number\_o \longrightarrow gill\_attachment\_f \wedge veil\_type\_p$ (0.8981, 0.9744) |
| 21 | $ring\_number\_o \longrightarrow gill\_attachment\_f \wedge veil\_color\_w \wedge veil\_type\_p$ (0.8971, 0.9733) |
| 22 | $gill\_attachment\_f \wedge veil\_color\_w \longrightarrow ring\_number\_o \wedge veil\_type\_p$ (0.8971, 0.9218) |
| 23 | $gill\_attachment\_f \wedge veil\_color\_w \longrightarrow gill\_spacing\_c \wedge veil\_type\_p$ (0.8127, 0.8351) |
| 24 | $gill\_attachment\_f \wedge ring\_number\_o \longrightarrow veil\_color\_w \wedge veil\_type\_p$ (0.8971, 0.9989) |
| 25 | $gill\_spacing\_c \wedge veil\_color\_w \longrightarrow gill\_attachment\_f \wedge veil\_type\_p$ (0.8127, 0.9973) |
| 26 | $gill\_attachment\_f \wedge veil\_type\_p \longrightarrow veil\_color\_w \wedge ring\_number\_o$ (0.8971, 0.9209) |
| 27 | $gill\_attachment\_f \wedge veil\_type\_p \longrightarrow ring\_number\_o$ (0.8981, 0.9219) |
| 28 | $veil\_type\_p \wedge veil\_color\_w \longrightarrow gill\_attachment\_f \wedge ring\_number\_o$ (0.8971, 0.9197) |
| 29 | $veil\_type\_p \wedge ring\_number\_o \longrightarrow gill\_attachment\_f \wedge veil\_color\_w$ (0.8971, 0.9733) |
| 30 | $veil\_type\_p \wedge ring\_number\_o \longrightarrow gill\_attachment\_f$ (0.8981, 0.9744) |
| 31 | $gill\_attachment\_f \wedge veil\_type\_p \wedge veil\_color\_w \longrightarrow ring\_number\_o$ (0.8971, 0.9218) |
| 32 | $gill\_attachment\_f \wedge veil\_type\_p \wedge ring\_number\_o \longrightarrow veil\_color\_w$ (0.8971, 0.9989) |
| 33 | $gill\_attachment\_f \wedge veil\_type\_p \longrightarrow gill\_spacing\_c \wedge veil\_color\_w$ (0.8127, 0.8342) |
| 34 | $gill\_spacing\_c \wedge veil\_type\_p \longrightarrow gill\_attachment\_f \wedge veil\_color\_w$ (0.8127, 0.9692) |
| 35 | $veil\_type\_p \wedge veil\_color\_w \longrightarrow gill\_attachment\_f \wedge gill\_spacing\_c$ (0.8127, 0.8332) |
| 36 | $gill\_attachment\_f \wedge veil\_type\_p \wedge veil\_color\_w \longrightarrow gill\_spacing\_c$ (0.8127, 0.8351) |
| 37 | $gill\_spacing\_c \wedge veil\_type\_p \wedge veil\_color\_w \longrightarrow gill\_attachment\_f$ (0.8127, 0.9973) |
| 38 | $gill\_attachment\_f \wedge veil\_type\_p \longrightarrow veil\_color\_w$ (0.9732, 0.999) |
| 39 | $veil\_type\_p \wedge veil\_color\_w \longrightarrow gill\_attachment\_f$ (0.9732, 0.9977) |
| 40 | $gill\_spacing\_c \wedge veil\_type\_p \longrightarrow veil\_color\_w$ (0.8149, 0.9718) |
| 41 | $veil\_type\_p \wedge veil\_color\_w \longrightarrow gill\_spacing\_c$ (0.8149, 0.8354) |

**Table A8.** The number of approximate rules and reduction in the mushroom database with confidence $c \geq 0.5$.

| $supp_r(*)$ | $TRX$ | Min-Max<br>$(N, Rt)$ | Reliable<br>$(N, Rt)$ | $TR$ | N-Min-Max<br>$(N, Rt)$ |
|---|---|---|---|---|---|
| 0.4 | 2528 | (465, 0.82) | (361, 0.86) | 1825 | (420, 0.77) |
| 0.5 | 835 | (175, 0.79) | (135, 0.84) | 514 | (190, 0.63) |
| 0.6 | 228 | (59, 0.74) | (52, 0.77) | 136 | (65, 0.52) |
| 0.7 | 161 | (39, 0.74) | (34, 0.79) | 90 | (41, 0.54) |
| Average | | 0.77 | 0.82 | | 0.62 |

**Table A9.** Number of approximate rules and reduction in the connect-4 database with confidence $c \geq 0.5$.

| $supp_r(*)$ | $TRX$ | Min-Max<br>$(N, Rt)$ | Reliable<br>$(N, Rt)$ | $TR$ | N-Min-Max<br>$(N, Rt)$ |
|---|---|---|---|---|---|
| 0.94 | 199,560 | (49,407, 0.75) | (10,220, 0.95) | 88,116 | (7914, 0.91) |
| 0.95 | 77,206 | (24,794, 0.68) | (5245, 0.93) | 39,768 | (4731, 0.88) |
| 0.96 | 26,856 | (11,452, 0.57) | (2538, 0.91) | 16,356 | (2535, 0.85) |
| 0.97 | 7895 | (4439, 0.44) | (1214, 0.85) | 5690 | (1294, 0.77) |
| Average | | 0.61 | 0.91 | | 0.85 |

**Table A10.** Templates, association rules and generalized association rules in the connect-4 database.

| **Template 1** | $\boldsymbol{A = \{d_5 - b, d_6 - b, e_5 - b, f_5 - b, f_6 - b\}\ (supp_r(A) = 0.9573)}$ |
|---|---|
| $\psi_1$ | $d_5 - b \wedge d_6 - b \wedge f_5 - b \wedge f_6 - b \longrightarrow e_5 - b\ (0.9573, 0.99)$ |
| Reduction of $\psi_1$ | $d_5 - b \wedge f_5 - b \longrightarrow e_5 - b\ (0.9573, 0.99)$ |
| Rule 1 | $\phi \equiv a_6 - b \wedge e_4 - b \wedge f_5 - b \longrightarrow e_6 - b \wedge f_6 - b\ (0.9502, 0.9694)$ |
| | $\phi_{GA} \equiv a_6 - b \longrightarrow e_6 - b \wedge f_6 - b\ (0.9502, 0.9576)$ |
| | $\phi_{ga} \equiv a_6 - b \wedge e_4 - b \wedge e_5 - b \wedge f_5 - b \longrightarrow e_6 - b \wedge f_6 - b\ (0.9502, 1)$ |

**Table A11.** The number of approximate rules and reduction in the chess database with confidence $c \geq 0.5$.

| $supp_r(*)$ | $TRX$ | **Min-Max** $\boldsymbol{(N, Rt)}$ | **Reliable** $\boldsymbol{(N, Rt)}$ | $TR$ | **N-Min-Max** $\boldsymbol{(N, Rt)}$ |
|---|---|---|---|---|---|
| 0.90 | 10,614 | (8371, 0.21) | (2483, 0.77) | 9230 | (2034, 0.78) |
| 0.91 | 5785 | (5050, 0.13) | (1571, 0.73) | 5354 | (1357, 0.75) |
| 0.93 | 2338 | (1948, 0.17) | (688, 0.71) | 2110 | (648, 0.69) |
| 0.95 | 468 | (459, 0.02) | (196, 0.58) | 466 | (195, 0.58) |
| Average | | 0.13 | 0.70 | | 0.70 |

**Table A12.** Templates, association rules and generalized association rules in the chess database.

| **Template 1** | $\boldsymbol{A = \{mulch\_f, skach\_f, wkna8\_f\}\ (supp_r(A) = 0.9002)}$ |
|---|---|
| $\psi_1$ | $mulch\_f \wedge skach\_f \longrightarrow wkna8\_f\ (0.9002, 0.9492)$ |
| Rule 1 | $\phi \equiv qxmsq\_f \wedge spcop\_f \wedge wkna8\_f \longrightarrow bkon8\_f \wedge hdchk\_f \wedge reskd\_f \wedge thrsk\_f\ (0.8326, 0.9103)$ |
| | $\phi_{GA} \equiv qxmsq\_f \wedge spcop\_f \longrightarrow bkon8\_f \wedge hdchk\_f \wedge reskd\_f \wedge thrsk\_f\ (0.8827, 0.9107)$ |
| | $\phi_{GC} \equiv qxmsq\_f \wedge spcop\_f \wedge wkna8\_f \longrightarrow bkon8\_f \wedge hdchk\_f \wedge reskd\_f\ (0.8698, 0.9136)$ |
| | $\phi_{GAC} \equiv qxmsq\_f \wedge spcop\_f \longrightarrow bkon8\_f \wedge hdchk\_f \wedge reskd\_f\ (0.9215, 0.9507)$ |
| | $\phi_{ga} \equiv qxmsq\_f \wedge spcop\_f \wedge wkna8\_f \wedge stlmt\_f \longrightarrow bkon8\_f \wedge hdchk\_f \wedge reskd\_f \wedge thrsk\_f\ (0.8326, 0.9103)$ |

## References

1. Agrawal, R.; Imieliński, T.; Swami, A. Mining Association Rules between Sets of Items in Large Databases. In Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, Washington, DC, USA, 25–28 May 1993; Association for Computing Machinery: New York, NY, USA, 1993; pp. 207–216.
2. Agrawal, R.; Srikant, R. Fast Algorithms for Mining Association Rules in Large Databases. In Proceedings of the 20th International Conference on Very Large Data Bases, Santiago de Chile, Chile, 12–15 September 1994; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 1994; pp. 487–499.
3. Thamer, M.B.; El-Sappagh, S.; El-Shishtawy, T. A Semantic Approach for Extracting Medical Association Rules. *Int. J. Intell. Eng. Syst.* **2020**, *13*, 280–292. [CrossRef]
4. Razzak, M.I.; Imran, M.; Xu, G. Big data analytics for preventive medicine. *Neural Comput. Appl.* **2020**, *32*, 4417–4451. [CrossRef] [PubMed]
5. Zhang, H.N.; Dwivedi, A.D. Precise Marketing Data Mining Method of E-Commerce Platform Based on Association Rules. *Mob. Netw. Appl.* **2022**. [CrossRef]
6. Reddy, R.V.; Venkateswara Rao, K.; Kameswara Rao, M.; Deepak Kumar, B.P. A Review on Stock Market Analysis Using Association Rule Mining. In *Cybernetics, Cognition and Machine Learning Applications*; Gunjan, V.K., Suganthan, P.N., Haase, J., Kumar, A., Eds.; Springer Nature Singapore: Singapore, 2023; pp. 171–183.
7. Ahn, K.I.; Kim, J.Y. Efficient Mining of Frequent Itemsets and a Measure of Interest for Association Rule Mining. *J. Inf. Knowl. Manag.* **2004**, *3*, 245–257. [CrossRef]
8. Brin, S.; Motwani, R.; Ullman, J.D.; Tsur, S. Dynamic itemset counting and implication rules for market basket data. In Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data, Tucson, AZ, USA, 13–15 May 1997; ACM: New York, NY, USA, 1997; pp. 255–264. [CrossRef]
9. Geng, L.; Hamilton, H.J. Interestingness measures for data mining: A survey. *ACM Comput. Surv.* **2006**, *38*, 9. [CrossRef]
10. Ghosh, A.; Nath, B. Multi-objective rule mining using genetic algorithms. *Inf. Sci.* **2004**, *163*, 123–133. [CrossRef]
11. Silverstein, C.; Brin, S.; Motwani, R. Beyond market baskets: Generalizing association rules to dependence rules. *Data Min. Knowl. Discov.* **1998**, *2*, 39–68. [CrossRef]
12. Baralis, E.; Cagliero, L.; Cerquitelli, T.; Garza, P. Generalized association rule mining with constraints. *Inf. Sci.* **2012**, *194*, 68–84. [CrossRef]
13. Beiranvand, V.; Mobasher-Kashani, M.; Bakar, A.A. Multi-objective PSO algorithm for mining numerical association rules without a priori discretization. *Expert Syst. Appl.* **2014**, *41*, 4259–4273. [CrossRef]

14. Guil, F.; Marín, R. A Theory of Evidence-based method for assessing frequent patterns. *Expert Syst. Appl.* **2013**, *40*, 3121–3127. [CrossRef]
15. Guns, T.; Nijssen, S.; De Raedt, L. Itemset Mining: A Constraint Programming Perspective. *Artif. Intell.* **2011**, *175*, 1951–1983. [CrossRef]
16. Ji, Y.; Ying, H.; Tran, J.; Dews, P.; Mansour, A.; Massanari, R.M. A Method for Mining Infrequent Causal Associations and Its Application in Finding Adverse Drug Reaction Signal Pairs. *IEEE Trans. Knowl. Data Eng.* **2013**, *25*, 721–733. [CrossRef]
17. Kuo, R.J.; Chao, C.M.; Chiu, Y.T. Application of particle swarm optimization to association rule mining. *Appl. Soft Comput.* **2011**, *11*, 326–336. [CrossRef]
18. Luna, J.M.; Romero, J.R.; Ventura, S. Grammar-based multi-objective algorithms for mining association rules. *Data Knowl. Eng.* **2013**, *86*, 19–37. [CrossRef]
19. Rodríguez, D.M.; Rosete, A.; Alcalá-Fdez, J.; Herrera, F. QAR-CIP-NSGA-II: A new multi-objective evolutionary algorithm to mine quantitative association rules. *Inf. Sci.* **2014**, *258*, 1–28.
20. Martínez-Ballesteros, M.; Martínez-Álvarez, F.; Lora, A.T.; Riquelme, J.C. Selecting the best measures to discover quantitative association rules. *Neurocomputing* **2014**, *126*, 3–14. [CrossRef]
21. Pei, Z. Extracting association rules based on intuitionistic fuzzy special sets. In Proceedings of the FUZZ-IEEE, Hong Kong, China, 1–6 June 2008; pp. 873–878.
22. Shaharanee, I.N.M.; Hadzic, F.; Dillon, T.S. Interestingness measures for association rules based on statistical validity. *Knowl. Based Syst.* **2011**, *24*, 386–392. [CrossRef]
23. Kaushik, M.; Sharma, R.; Peious, S.A.; Shahin, M.; Yahia, S.B.; Draheim, D. A Systematic Assessment of Numerical Association Rule Mining Methods. *SN Comput. Sci.* **2021**, *2*, 348. [CrossRef]
24. Kuo, R.J.; Gosumolo, M.; Zulvia, F.E. Multi-objective particle swarm optimization algorithm using adaptive archive grid for numerical association rule mining. *Neural Comput. Appl.* **2019**, *31*, 3559–3572. [CrossRef]
25. Wang, H.B.; Gao, Y.J. Research on parallelization of Apriori algorithm in association rule mining. *Procedia Comput. Sci.* **2021**, *183*, 641–647. [CrossRef]
26. Bazai, S.U.; Jang-Jaccard, J. In-Memory Data Anonymization Using Scalable and High Performance RDD Design. *Electronics* **2020**, *9*, 1732. [CrossRef]
27. Bazai, S.U.; Jang-Jaccard, J.; Alavizadeh, H. A Novel Hybrid Approach for Multi-Dimensional Data Anonymization for Apache Spark. *ACM Trans. Priv. Secur.* **2021**, *25*, 1–25. [CrossRef]
28. Bazai, S.U.; Jang-Jaccard, J.; Alavizadeh, H. Scalable, High-Performance, and Generalized Subtree Data Anonymization Approach for Apache Spark. *Electronics* **2021**, *10*, 589. [CrossRef]
29. Calders, T.; Dexters, N.; Gillis, J.J.M.; Goethals, B. Mining frequent itemsets in a stream. *Inf. Syst.* **2014**, *39*, 233–255. [CrossRef]
30. Han, J.; Cheng, H.; Xin, D.; Yan, X. Frequent pattern mining: Current status and future directions. *Data Min. Knowl. Discov.* **2007**, *15*, 55–86. [CrossRef]
31. Pei, J.; Han, J.; Mao, R. CLOSET: An Efficient Algorithm for Mining Frequent Closed Itemsets. In Proceedings of the ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, Dallas, TX, USA, 14 May 2000; pp. 21–30.
32. Wang, J.; Han, J.; Pei, J. CLOSET+: Searching for the best strategies for mining frequent closed itemsets. In Proceedings of the KDD, Washington, DC, USA, 24–27 August 2003; Getoor, L., Senator, T.E., Domingos, P.M., Faloutsos, C., Eds.; ACM: New York, NY, USA, 2003; pp. 236–245.
33. O'Sullivan, D.; Smyth, B.; Wilson, D.C.; McDonald, K.; Smeaton, A. Improving the Quality of the Personalized Electronic Program Guide. *User Model. User-Adapt. Interact.* **2004**, *14*, 5–36. [CrossRef]
34. Kryszkiewicz, M.; Rybinski, H.; Gajek, M. Dataless Transitions Between Concise Representations of Frequent Patterns. *J. Intell. Inf. Syst.* **2004**, *22*, 41–70. [CrossRef]
35. Pasquier, N.; Bastide, Y.; Taouil, R.; Lakhal, L. Efficient mining of association rules using closed itemset lattices. *Inf. Syst.* **1999**, *24*, 25–46. [CrossRef]
36. Zaki, M.J. Scalable algorithms for association mining. *IEEE Trans. Knowl. Data Eng.* **2000**, *12*, 372–390. [CrossRef]
37. Zaki, M.J.; Hsaio, C.J. Efficient Algorithms for Mining Closed Itemsets and Their Lattice Structure. *IEEE Trans. Knowl. Data Eng.* **2005**, *17*, 462–478. [CrossRef]
38. Hashem, T.; Ahmed, C.F.; Samiullah, M.; Akther, S.; Jeong, B.S.; Jeon, S. An efficient approach for mining cross-level closed itemsets and minimal association rules using closed itemset lattices. *Expert Syst. Appl.* **2014**, *41*, 2914–2938. [CrossRef]
39. Liu, H.; Liu, L.; Zhang, H. A fast pruning redundant rule method using Galois connection. *Appl. Soft Comput.* **2011**, *11*, 130–137. [CrossRef]
40. Cagliero, L.; Cerquitelli, T.; Garza, P.; Grimaudo, L. Misleading Generalized Itemset discovery. *Expert Syst. Appl.* **2014**, *41*, 1400–1410. [CrossRef]
41. Cagliero, L.; Garza, P. Itemset generalization with cardinality-based constraints. *Inf. Sci.* **2013**, *244*, 161–174. [CrossRef]
42. Baralis, E.; Cagliero, L.; Cerquitelli, T.; D'Elia, V.; Garza, P. Expressive generalized itemsets. *Inf. Sci.* **2014**, *278*, 327–343. [CrossRef]
43. Boulicaut, J.F.; Bykowski, A.; Rigotti, C. Free-sets: A condensed representation of boolean data for the approximation of frequency queries. *Data Min. Knowl. Discov.* **2003**, *7*, 5–22. [CrossRef]
44. Bykowski, A.; Rigotti, C. DBC: A condensed representation of frequent patterns for efficient mining. *Inf. Syst.* **2003**, *28*, 949–977. [CrossRef]

45. Chiang, D.A.; Wang, Y.F.; Wang, Y.H.; Chen, Z.Y.; Hsu, M.H. Mining disjunctive consequent association rules. *Appl. Soft Comput.* **2011**, *11*, 2129–2133. [CrossRef]
46. Calders, T.; Goethals, B. Non-derivable itemset mining. *Data Min. Knowl. Discov.* **2007**, *14*, 171–206. [CrossRef]
47. Li, H.; Chen, H. Mining non-derivable frequent itemsets over data stream. *Data Knowl. Eng.* **2009**, *68*, 481–498. [CrossRef]
48. Hamrouni, T.; Yahia, S.B.; Nguifo, E.M. Sweeping the disjunctive search space towards mining new exact concise representations of frequent itemsets. *Data Knowl. Eng.* **2009**, *68*, 1091–1111. [CrossRef]
49. Barrenechea, E.; Sola, H.B.; Campión, M.J.; Induráin, E.; Knoblauch, V. Topological interpretations of fuzzy subsets. A unified approach for fuzzy thresholding algorithms. *Knowl. Based Syst.* **2013**, *54*, 163–171. [CrossRef]
50. Syau, Y.R.; Lin, E.B. Neighborhood systems and covering approximation spaces. *Knowl. Based Syst.* **2014**, *66*, 61–67. [CrossRef]
51. Wang, S.; Liu, D. Knowledge representation and reasoning for qualitative spatial change. *Knowl. Based Syst.* **2012**, *30*, 161–171. [CrossRef]
52. Pei, Z.; Ruan, D.; Meng, D.; Liu, Z. Formal concept analysis based on the topology for attributes of a formal context. *Inf. Sci.* **2013**, *236*, 66–82. [CrossRef]
53. Zhang, Y.; Pei, Z.; Shi, P. Association rule mining based on topology for attributes of multi-valued information systems. *Int. J. Innov. Comput. Inf. Control. Ijicic* **2013**, *9*, 1679–1690.
54. Ganter, B.; Wille, R. *Formal Concept Analysis: Mathematical Foundations*; Springer: Berlin/Heidelberg, Germany, 1999.
55. Pawlak, Z.; Skowron, A. Rough sets and Boolean reasoning. *Inf. Sci.* **2007**, *177*, 41–73. [CrossRef]
56. Qin, K.; Yang, J.; Pei, Z. Generalized rough sets based on reflexive and transitive relations. *Inf. Sci.* **2008**, *178*, 4138–4141. [CrossRef]
57. Zhang, H.P.; Ouyang, Y.; Wang, Z. Note on "Generalized rough sets based on reflexive and transitive relations". *Inf. Sci.* **2009**, *179*, 471–473. [CrossRef]
58. Freund, M. On the notion of concept I. *Artif. Intell.* **2008**, *172*, 570–590. [CrossRef]
59. Srikant, R.; Agrawal, R. Mining Generalized Association Rules. In Proceedings of the 21st International Conference on Very Large Databases, Zurich, Switzerland, 11–15 September 1995; pp. 407–419.
60. Wu, C.M.; Huang, Y.F. Generalized association rule mining using an efficient data structure. *Expert Syst. Appl.* **2011**, *38*, 7277–7290. [CrossRef]
61. Apriori Algorithm. Available online: http://www.mathworks.com/matlabcentral/fileexchange/42541-association-rules/ (accessed on 1 September 2022).
62. UCI Machine Learning Repository. Available online: http://archive.ics.uci.edu/ml/ (accessed on 1 September 2022).
63. Xu, Y.; Li, Y.; Shaw, G. Reliable representations for association rules. *Data Knowl. Eng.* **2011**, *70*, 555–575. [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.