

Research Article

Exploring Structural Characteristics of Lattices in Real World

Yu Chen ¹, **Jinguo You** ^{1,2}, **Benyuan Zou**¹, **Guoyu Gan**¹, **Ting Zhang**³,
and **Lianyin Jia** ^{1,2}

¹Kunming University of Science and Technology, Yunnan 650504, China

²Computer Technology Application Key Lab of Yunnan Province, Yunnan 650504, China

³Jiaozuo University, Henan 454003, China

Correspondence should be addressed to Jinguo You; jgyou@126.com

Received 19 September 2019; Accepted 26 December 2019; Published 21 January 2020

Academic Editor: Rongqing Zhang

Copyright © 2020 Yu Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

There are two important models for data analysis and knowledge system: data cube lattices and concept lattices. They both essentially have lattice structures, which are actually irregular in our real world. However, their structural characteristics and relationship are not yet clear. To the best of our knowledge, no work has paid enough attention to this challenging issue from the perspective of graph data, in spite of the importance of structures in lattice data. In this paper, we first tackle the structural statistics of lattice data from three aspects: the degree distribution, clustering coefficient, and average path length. We demonstrated by various datasets that data cube lattices and concept lattices share similarities underlying their topology, which are, in general, different from random networks and complex networks. Specifically, lattice data follow the Poisson distribution and have smaller clustering coefficient and greater average path length. We further discuss and explain these characteristics intrinsically by building the analytical model and the generating mechanism.

1. Introduction

The data cube (lattice) [1] proposed by Gray et al. is a core data model in data warehousing and online analytical processing (OLAP) [2]. It plays a more critical role in data online analysis and processing, especially in the era of big data. It allows data to be modeled and analyzed intuitively in the context of multiple analysis dimensions and plays a vital role in business intelligence. Based on data cubes, online analysis operations such as rolling up, drilling down, slicing, and rotation can be easily carried out. From a conceptual view, users can analysis the data along a dimension hierarchy to various coarse-grained levels so that a large amount of detailed data can be described in a more concise and summary fashion, which facilitates users to obtain the general view. On the contrary, they can also specialize data along a dimension hierarchy to the fine-grained data. According to the computational dependence relation of rolling up and drilling down, the data cube lattice is generated.

The concept lattice [3] proposed by German mathematician Wille in 1982 is an important model in formal concept

analysis (FCA) and is considered as an essential facility for data analysis. Formally, the concept lattice takes the triple (U, A, I) as its formal background, in which U is the object set, A is the attribute set, and I is the binary relation between the object set and the attribute set. It derives the formal concept (X, B) , where X is called the concept extent and B is called the concept intent; each concept in the concept set forms the concept lattice according to the partial order relation. The node of concept lattices denotes both the intent and the extent of the concept, and the relationship between nodes reflects the generalization and specialization of concepts. According to the dependency of the knowledge body in the intent and the extent, a concept hierarchy model is established. Concept lattices are widely used in machine learning, pattern recognition, expert systems, decision analysis, data mining, information retrieval, computer networks and software engineering, and many other fields [4, 5]. For example, most data mining tasks can generate a large number of concepts. The lattice structure as the organizational form of the concept has many advantages in knowledge discovery. It

facilitates a deep understanding of the dependencies between different concepts selected from a data set.

Essentially, the instances of data cube lattices and concept lattices all belong to lattice structure data. As the core model, data cube lattices and concept lattices are widely used in data analysis. From a higher perspective, their algebraic structures are intrinsically lattices, which imply they share similarities or connections in external characteristics.

Nedjar et al. and Shi [6, 7] demonstrated their relationship in the generation mechanism. Nedjar et al. [6] pointed out that, in data warehousing and data mining, the frequent closed itemset searching, the concise representation of association rules, or data cube lattices can use the formal concept analysis for feature depicting, compress representation, and effective calculation. Furthermore, they put forward Agree Concept Lattice and Quotient Agree Lattice to point out their close relationship between them. Shi [7] made a thorough study of concept lattices and finds that concept lattices are closely related to data cube lattices. They are all based on partial order relations. When base tables are used as formal backgrounds, the covering equivalence class of the data cube lattice and the equivalent feature sets based on formal concept analysis theory have the same partition results, and then the Aggregate Concept, Aggregate Concept Lattice (ACL), and Reductive Aggregate Concept (RAC) were proposed.

The structural analysis of these two crucial and representative models in data analysis under a unified framework will facilitate the discovery of their essence and design more generalized algorithms. Strogatz [8] discussed the reason why network anatomy is so important to characterize is that structure always affects function. Liu et al. and Zhai et al. [9, 10] mentioned the application of degree, distance, and topological statistics in some real networks. Lattice topology characteristics like the degree distribution of nodes affect the communication overhead (virtually link numbers) across various nodes for splitting a lattice in the cloud environment. The clustering or partitioning of lattice structures from the graph data perspective may result in space compression. Until recently, however, very little attention has been devoted to the structural characteristics and the relationship between them.

To address the above challenges, this paper explores and studies the structural characteristics of data cube lattices and concept lattices in terms of the degree distribution, clustering coefficient, average path length through the experimental demonstration, and theoretical analysis. By the analysis of the graph structure, we can find whether the data cube lattice and concept lattice share similar structural characteristics, which are different from other networks like random networks or complex networks and why. Note that the lattices discussed in our paper are generated from the real life—they are not completely regular lattices where all the nodes have the same degrees, and a plot of the degree distribution contains a single sharp spike (delta distribution) [11]. As we discovered in Section 3, the nodes of the real-world lattices have much broader range of degrees, implying randomnesses are added.

Our main contributions are as follows:

- (i) To the best of our knowledge, we first study the graph structural characteristics (such as the degree

distribution, clustering coefficient, and average path length) of data cube lattices and concept lattices, analyze, and demonstrate them based on various real datasets.

- (ii) We present the structure relationship between data cube lattices and concept lattices. Compared with random networks and complex networks, real-world lattices follow the Poisson distribution and have smaller clustering coefficient and greater average path length.
- (iii) We discuss the analytical model and the generation mechanism of data cube lattices and concept lattices.

This paper is structured as follows. Section 2 introduces the model of data cube lattices and concept lattices and relevant definitions. Section 3 presents the structural characteristics of data cube lattices and concept lattices through the experimental study and validates the similarity between data cube lattices and concept lattices on structural characteristics. Section 4 discusses the analytical model and the generation mechanism of data cube lattices and concept lattices. The last section concludes this paper and describes some future works.

2. Preliminary

2.1. Data Cube Lattices. Data cube lattices generalize group-by operators and aggregate each combination of group-by attributes. Among them, the attributes grouped are called dimensions D , and the attributes aggregated are called measurements M ; each grouped attribute combination is called cuboid (also called view). Correspondingly, the cuboid containing i ($0 \leq i \leq n$) dimensions called is i -dimension cuboid. The data cell $c = (x_1, x_2, x_3, \dots, x_n, M)$ is a tuple in the cuboid, where x_i is the i th dimension attribute value.

For a d -dimensional data cube lattice, 2^d group-by views (or cuboids) are generated since each combination of group-by attributes is computed.

Definition 1 (partial order relation). If $\forall A \in D$, $u[A] \neq \text{ALL} \rightarrow u[A] = v[A]$, then u, v have partial order relation, denoted by $v < u$ or $u \geq v$. If $v < u$ and $u \neq v$, then it is denoted as $v < u$ or $u > v$. We can say that u generalizes v , or v specializes u . In other words, u drills down to v or v rolls up to u . If $\forall A \in D$, $u[A] = v[A]$, then $u = v$.

Definition 2 (data cube lattice). A data cube is aggregated from a base table of data warehousing on various combinations of dimension attributes. It contains the data cells. Let $c_1 = (x_1, x_2, x_3, \dots, x_n, M)$, $c_2 = (y_1, y_2, y_3, \dots, y_n, M)$ be the data cells. The partial order relation $< : c_2 < c_1$ induces the data cube lattice structure.

Definition 3 (base tuple set). Given a data cell $c \in C$, the base tuple set of c , $\text{BTS}(c) = \{t \mid t \in r \text{ and } t < c\}$, i.e., the set of all base table tuples that roll up to c .

Definition 4 (covering equivalence). Suppose that $c_1, c_2 \in \text{Cube}$, and $\equiv_{\text{cov}} = \{ \langle c_1, c_2 \rangle \mid c_1, c_2 \in \text{Cube}, \text{BTS}(c_1) = \text{BTS}(c_2) \}$ is defined as the covering equivalence relation. The covered equivalence classes is the sets $[c]_{\equiv_{\text{cov}}} = \{x \mid x \in \text{Cube}, c \equiv_{\text{cov}} x\}$.

Definition 5 (upper and lower bounds). Let $\langle A, < \rangle$ be a partially ordered set. Any element a in A is called an upper bound (lower bound) of the subset M of A if for any element m in M , there is $m < a$ ($a < m$).

Example 1. Table 1 is a base table of product sales, which has three dimension attributes of Product, Time, and Store and one measurement attribute, Sales. The data cube lattice generates eight group-by views by aggregation, which form the cube lattice structure based on the relationship (as shown by the connection between nodes) between rolling up (generalization) and drilling down (specialization).

Figure 1 is a data cube lattice generated by aggregation of the Sum operation on sales in Table 1. Note that * denotes the dimension attribute value ALL. The data cells $\{(P1, *, S1, 10)\}$, $\{(P1, T1, S1, 4)\}$, and $\{(P1, T2, S1, 6)\}$ have the semantic relations of rolling up and drilling down each other. The partial order relationship between them forms a data cube lattice.

2.2. Concept Lattices

Definition 6 (concept lattice). Let I be a binary relationship between U and A , where U is the object set and A is the attribute set. Given that $x \in U, y \in A$, when $(x, y) \in I$, we say the object x owns the attribute y , and the triple tuple (U, A, I) was called formal background. On a triple tuple (U, A, I) , we take a subset of U and a subset of A and get a concept set under relation I : $L = \{(A, B) \mid A \in P(U), B \in P(A), (A, B) \text{ is the concept}\}$; among them, A is the extent of L and B is the intent of L . We define a partial ordering relation on L : $(A_1, B_1) < (A_2, B_2) \iff A_1 \subseteq A_2$ (or $B_2 \subseteq B_1$). And then, $(L, <)$ is a complete Galois lattice, called a concept lattice about formal background (U, A, I) .

Definition 7 (equivalent feature set). Let $K = (U, A, I)$ be a formal background, $N \subseteq U, M \subseteq A$. If $M \subseteq N'$ and $M' = N$ are satisfied, then M is an equivalent feature set of N , denoted by $\text{equ}(N) = \{M \mid M \subseteq N', M' \subseteq N\}$.

Definition 8 (Hasse diagram). A Hasse diagram is a kind of mathematical diagram used to represent a finite partially ordered set in the form of a drawing of its transitive reduction. Concretely, for a partially ordered set $(A, <)$, one represents each element of S as a vertex in the plane and draws a line segment or curve that goes upward from v_i to v_j whenever v_j covers v_i (that is, whenever $v_i < v_j$ and there is no v_k such that $v_i < v_k < v_j$). These curves may cross each other but must not touch any vertices other than their endpoints. Such a diagram, with labeled vertices, uniquely determines its partial order.

According to the definition of Concept Lattice, the concept lattice generated by a formal background given in Table 2 is shown in Figure 2.

Figure 2 shows a Hasse diagram representation of concept lattices corresponding to the formal background of Table 2. Each node in the graph represents a concept, each concept is identified by its extent and intent, and the order relationship between concepts is represented by the edges between nodes. Among them, the concept with the largest extent (corresponding to the smallest intent) in the concept lattice is the largest concept in the concept lattice, which is located at the top of the concept lattice; the concept with the largest intent (corresponding to the smallest extent) in the concept lattice is the smallest concept in the concept lattice, which is located at the bottom of the concept lattice.

3. Lattice Structure Characteristics

Since the three spectacular concepts—the degree distribution, clustering coefficient, and average path length—play a more important role in networks than other quantities and measures [11], we verify them on various real datasets for lattices, particularly the main representations—data cube lattices and concept lattices.

3.1. Data Cube Lattices' Structural Characteristics. We used the two classic datasets Foodmart and Weather (<http://cdiac.esd.ornl.gov/cdiac/ndps/ndp026b.html>) for the data cube lattices. Then, we calculated its topology statistics to analyze whether the data cube lattices generated by different data sets have similar structural characteristics.

We randomly extracted 10,000 tuples from Foodmart and generated a data cube lattice Foodmart-1w by adapting the data cube construction algorithm [12]. Then, we randomly extracted 10,000 tuples from Weather data and generated the data cube lattice Weather-1w by the same lattice structure algorithm. They are shown in Table 3.

For the two datasets in Table 3, the structural characteristics of the degree distribution, clustering coefficient, and average path length are calculated.

Figure 3(a) shows the degree distribution of Foodmart-1w and Weather-1w. The horizontal axis represents the degree value of the node, while the vertical axis represents the total number of nodes when the degree is that value. By comparison, it can be found that the two curves in the graph first jump sharply and then decrease exponentially. The average degree of each node of Foodmart-1w is 8.8; the average degree of each node of Weather-1w is 7.9. It can be seen that the average of data cube lattices of different data sets is not very different.

Figure 3(b) shows the clustering coefficient of two data sets in the data cube lattice. The horizontal axis represents the degree value of the node, while the vertical axis represents the average clustering coefficient of nodes when the degree is that value. The average clustering coefficient of Foodmart-1w and Weather-1w is 0.0231 and 0.0042, respectively. All of them have relatively small average clustering coefficients.

TABLE 1: Base table of product sales.

	Product	Time	Store	Sales
1	P1	T1	S1	4
2	P1	T2	S1	6
3	P2	T1	S2	2

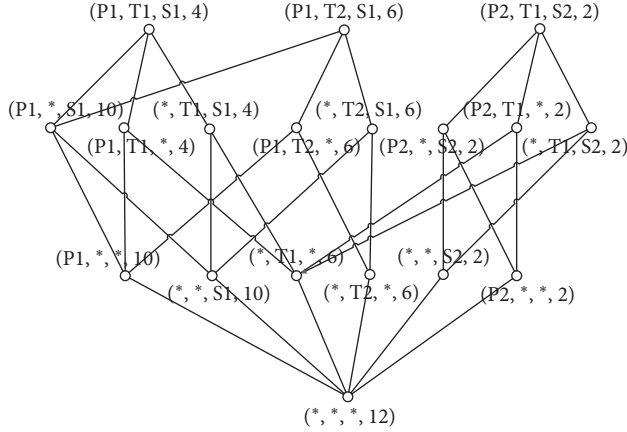


FIGURE 1: Data cube lattice.

TABLE 2: A formal background.

Object	a	b	c	d
1	1	0	1	1
2	1	1	0	1
3	0	1	1	0
4	0	1	1	0
5	1	0	1	1

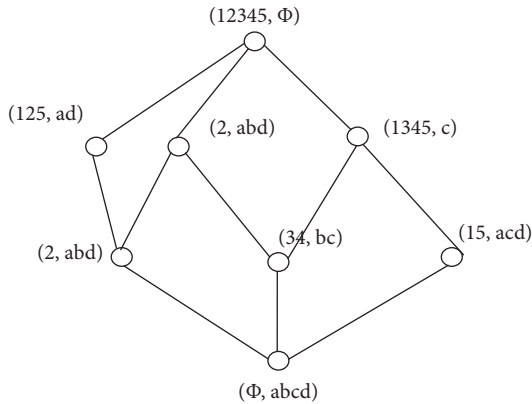


FIGURE 2: Concept lattice.

Figure 3(c) shows the average path length distribution of two data sets in the data cube lattice. The horizontal axis represents the length of the path (hops), and the vertical axis represents the number of pairs of nodes when the path length is that value. The average path length of Foodmart-1w and Weather-1w is 5.25 and 7.21, respectively. Figure 3(c) shows the distribution of the average path length is similar between two structures.

TABLE 3: The first group dataset.

Data cube lattices	Node number	Edge number
Foodmart-1w	17,579	31,676
Weather-1w	13,230	53,424

3.2. *Concept Lattices' Structural Characteristics.* For the concept lattices, we used the Mushroom data in the UCI machine learning library (<http://archive.ics.uci.edu/ml/datasets>), which is the common benchmark dataset in calibrating various concept lattice algorithms. We also analyzed whether the concept lattices generated by different data sets have similar structural characteristics.

We randomly selected 600 tuples from the Mushroom data set of UCI and divided them into two parts, each with 300 tuples. Then, we used In-Close algorithm [13] to generate concept lattices Mushroom-1 and Mushroom-2 and used FcaStone (<http://fcastone.sourceforge.net>) to transform the concept lattices into the graph structures further, forming the second group dataset, as shown in Table 4.

The degree distribution, clustering coefficient, average path length, and other structural characteristics are calculated by using the data in Table 4. The results are shown in Figures 4(a)–4(c).

Figure 4(a) shows the degree distribution of Mushroom-1 and Mushroom-2. By comparison, it is an interesting discovery that the two curves in the graph jump sharply first and then decrease exponentially. The average degree of each node of Mushroom-1 is 8.1, while that of Mushroom-2 is 7.9. It can be seen that the average degree of concept lattices of different data sets is not very different.

Figure 4(b) shows the clustering coefficient of the two data sets in the concept lattice, in which the horizontal axis represents the degree value of the node, and the vertical axis represents the average clustering coefficient of nodes when the degree is the value. The average clustering coefficient of Mushroom-1 and Mushroom-2 is 0.1064 and 0.0842, respectively. They all have relatively small average clustering coefficient.

Figure 4(c) shows the average path length distribution of the two data sets in the concept lattice, in which the horizontal axis represents the path length (hops), and the vertical axis represents the number of node pairs when path length is that value. The average path length of Mushroom-1 and Mushroom-2 is 6.14 and 5.15, respectively. Both of them have smaller average path length.

3.3. *Relationship between Data Cube Lattices and Concept Lattices.* We used the data cube lattice generated by Foodmart data in the first group and the concept lattice generated by Mushroom data in the second group, as shown in Table 5, to analyze whether they share the similar structural characteristics.

For the two kinds of lattice structure data in Table 5, the degree distribution, clustering coefficient, and average path length are calculated. The data cube lattice and concept lattice are abbreviated as CubeLattice and ConceptLattice, respectively.

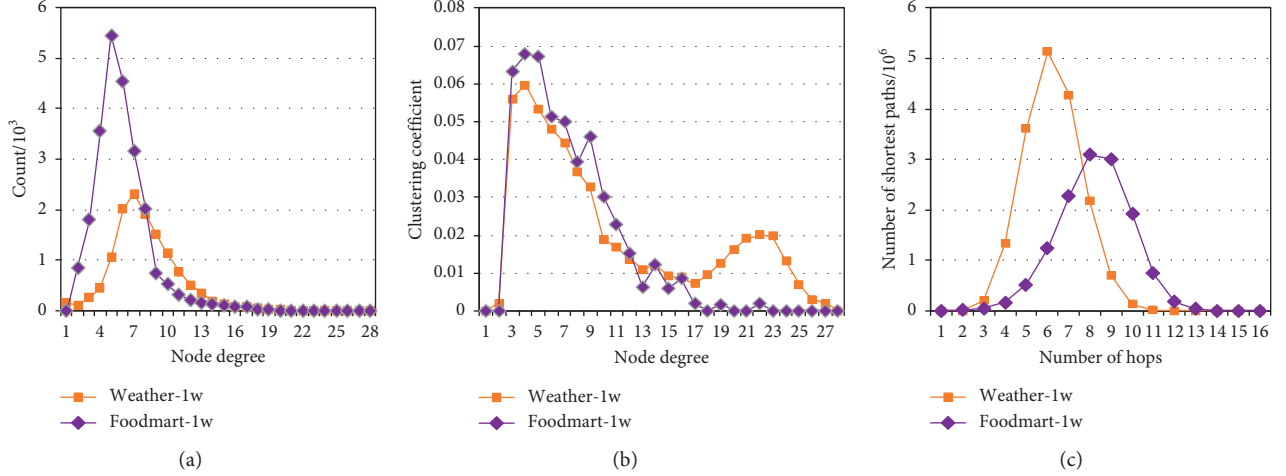


FIGURE 3: The structural characteristics of cube lattices on Foodmart-1w and Weather-1w. (a) Degree distribution. (b) Clustering coefficient. (c) Average path length.

TABLE 4: The second group dataset.

Concept lattices	Node number	Edge number
Mushroom-1	10,486	46,289
Mushroom-2	10,547	60,146

Figure 5(a) shows the degree distribution of two different lattice structure data, CubeLattice and ConceptLattice. The degree distributions of both of them rise sharply, reach a peak value, and then decay exponentially. In the degree distribution of CubeLattice, when the degree of a node is 5, the number of nodes reaches a peak value, which is 5,450; in the degree distribution of ConceptLattice, when the degree of a node is 7, the number of nodes reaches a peak value, which is 2,440. By comparison, we have found that the degree distribution of CubeLattice is similar to that of ConceptLattice.

Figure 5(b) shows the distribution of the clustering coefficient for CubeLattice and ConceptLattice. From Figure 5(b), we can see that both of them rise sharply at first, then decline slowly after a peak. Although there are fluctuations in amplitude during the decline process, the overall trend is downward. The average clustering coefficient of CubeLattice and ConceptLattice is 0.0231 and 0.1064, respectively. So, the average clustering coefficient of lattice structure data is relatively small.

Figure 5(c) shows the distribution of the average path length for the two different lattice structures, CubeLattice and ConceptLattice. By comparison, it has been found that the average path length distribution of them increases slowly at first, reaches a peak value, and then decreases slowly. After calculation, we found that the average path length of CubeLattice is 5.25 and that of ConceptLattice is 6.14. Obviously, the distribution of the average path length about CubeLattice is similar to that of ConceptLattice.

3.4. Comparison with Other Main Networks. In order to check whether the lattices' structural statistics are different

with other main networks such as random networks and complex networks [8, 11], we first extracted the 10,000 tuples from Foodmart and generated a data cube lattice. Then, we generated a random graph based on the ER model [14] using SNAP [15]. For the complex network, the data set is a real social network Facebook collected in SNAP data sets.

The comparison results are as follows:

- (1) The degree distribution of lattices is obviously different from that of Facebook but shares certain similarity with the ER random graph. The degree distribution of Facebook generally follows the heavy tail distribution, which indicates the occurrence of nodes with a much higher degree than most other nodes. On the contrary, Lattice and ER follow the Poisson-like degree distribution (Figure 6).
- (2) The average clustering coefficient of lattices is different from that of complex networks. According to the calculation, the average clustering coefficient of Lattice is 0.0231, and the average clustering coefficient of ER is 0.0011. They both have a small clustering coefficient. The clustering coefficient in ER random graphs is evenly and randomly distributed. Different from ER, the nodes with a smaller degree in Lattice have the larger clustering coefficient in lattices. The average clustering coefficient of Facebook is 0.5225, which accords with the large clustering coefficient of the small-world network. That is also different from the average clustering coefficient of Lattice (Figure 7).
- (3) The average path length of Lattice and ER is 7.21 and 4.26, respectively. Both of them have smaller average path length. The longest path between two points in the lattice structure is related to the number of dimensions, while the longest path between two nodes in the ER random graph is related to the number of nodes. Facebook's average path length is 3.7, and the longest path between two points in the network is 6. Obviously, the average path length distribution of

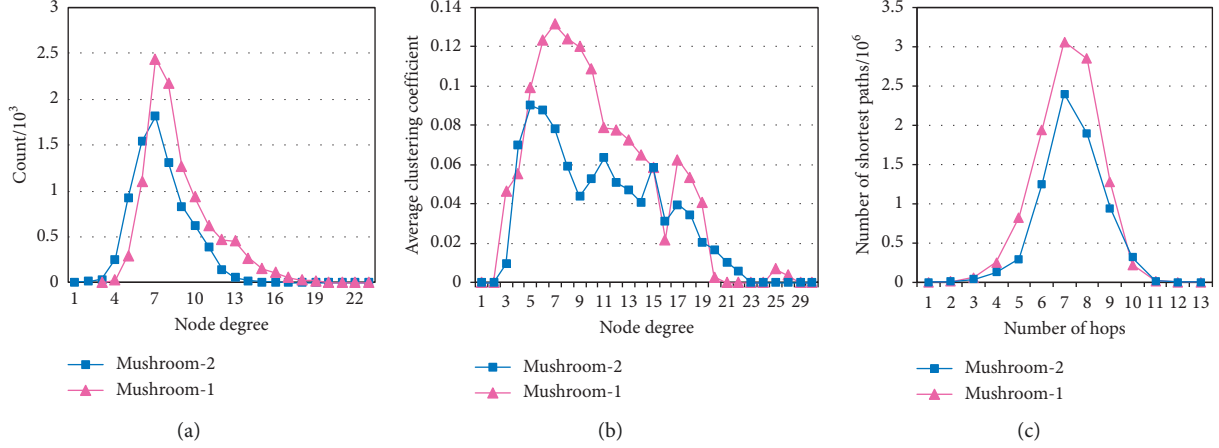


FIGURE 4: The structural characteristics of concept lattices on the mushroom date set. (a) Degree distribution. (b) Clustering coefficient. (c) Average path length.

TABLE 5: The third group dataset.

Lattices	Node number	Edge number
Data cube lattice	17,579	31,676
Concept lattice	10,486	46,289

the lattice structure is different from that of the Facebook social network (Figure 8).

In summary, on the degree distribution, lattices, and random networks all follow the Poisson distribution. Besides, lattices and random networks have the smaller clustering coefficient. But, lattices have the rather larger average path length, while random networks are moderately large in the average path length. For the typical complex networks, they have the larger average clustering coefficient and the smaller average path length, and their degree distribution satisfies the heavy tail distribution.

4. Some Discussions

4.1. Analytical Model. Data cube lattices have different structural characteristics from random networks and complex networks. First, in the process of generating lattice-structured data, nodes do not connect according to a random probability, but there are partial ordering relationships among some different nodes, which generate edges according to the partial ordering relationship; secondly, there is no partial order relationship between data cells in the same layer, that is, the lattice structure data have a clear hierarchical structure; finally, the lattice structure has its unique regular structure, and every two nodes have upper and lower bounds.

We elaborate the analytical model of data cube lattices and concept lattices from three aspects.

4.1.1. Degree Distribution. From the above experiments, we can see that the degree distribution curves of the lattice structure data first jump sharply, then drop exponentially, which tends to be similar to the Poisson distribution curve.

In order to prove this point, we use polynomial distribution and Poisson distribution curves to fit the degree distribution of the lattice structure data.

Firstly, the degree distribution curve of the data cube lattice (it has 10,000 base tuples, 15 dimensions, 13,230 nodes, and 53,424 edges.) is fitted by the polynomial curve. In the process of fitting, when the polynomial is 6th power and more, the R-square reaches the maximum value. As shown in Figure 9, the vertical axis represents the probability of the occurrence of nodes with different degree values, the diamond points represent the discrete points of the degree distribution of the data cube lattice, and the triangle curves represent the fitting polynomial curves of 6th power. At this time, the value of R-square is about 0.89, which fits the original degree distribution curve better.

In order to evaluate the accuracy of curve fitting, the square of the error between the discrete point of the degree distribution curve and the discrete point of the polynomial curve is calculated. The result is about 0.0084.

Then, the Poisson distribution curve is fitted to the degree distribution curve, and the probability expression of Poisson distribution is as follows:

$$P(X) = \frac{\vartheta^x}{x!} e^{-\vartheta}, \quad (1)$$

where ϑ denotes the expected value. The degree values of the nodes in the data cube lattice are multiplied by their corresponding probabilities, and then the sum can be calculated as follows: $\vartheta = \sum d * P(d)$. Then, the Poisson distribution curve can be drawn, as shown in Figure 10. The vertical axis represents the probability of the occurrence of nodes with different degree values, the curve with diamonds represents the curve simulating Poisson distribution, the curve with squares represents the degree distribution curve of the whole lattice structure data, and the curve with triangles is the degree distribution curve of the nodes in the eleventh layer of the lattice structure. It can be seen that both the degree distribution of the whole structure and the degree part of one of the layers tightly fit the Poisson distribution curve. Also, another data cube lattice (it has 10,000 base tuples, 12,088

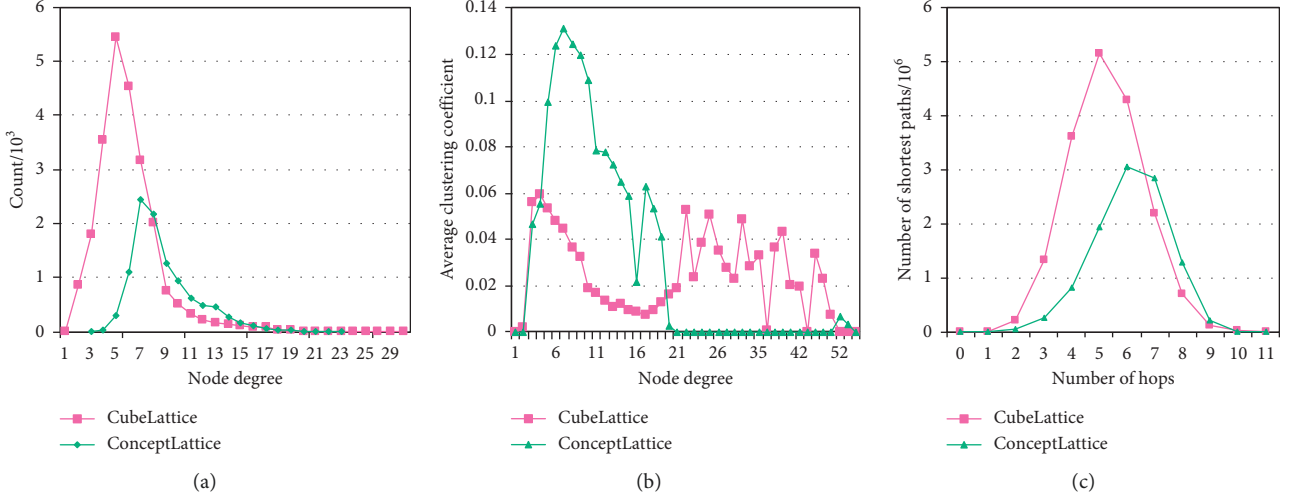


FIGURE 5: The structural characteristics between cube lattices and concept lattices. (a) Degree distribution. (b) Clustering coefficient. (c) Average path length.

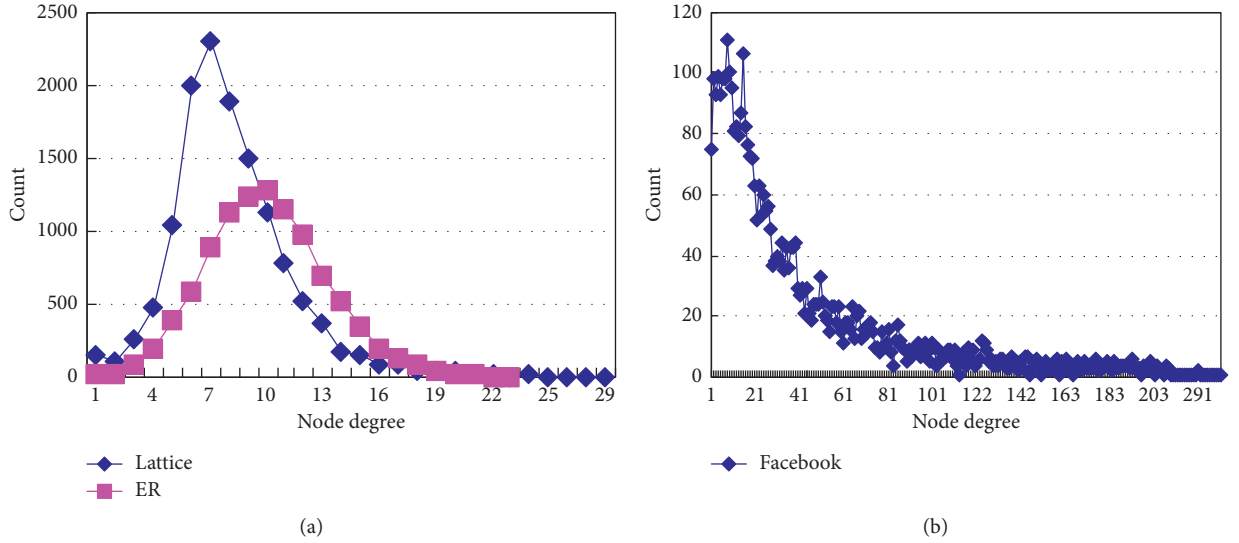


FIGURE 6: Degree distribution of three kinds of network structures. (a) Lattice and ER. (b) Facebook.

nodes, and 54,658 edges.) is also fitted with the Poisson distribution curve. As shown in Figure 11, the degree distribution of different lattice structure data is tightly fitted with the Poisson distribution curve.

The square of the errors of the discrete points of the degree distribution curve and the Poisson distribution is calculated. The result is about 0.0038, and the error is much smaller than the fitting curve of the polynomial. Combined with the additivity of Poisson distribution, it is further confirmed that the degree distribution of the lattice structure is similar to Poisson distribution.

4.1.2. Clustering Coefficient. If a node i is chosen arbitrarily in the lattice, and the degree of i is d_i , then the number of possible connected edges of the d_i adjacent nodes of node i is $d_i(d_i - 1)/2$. If the actual number of edges between the d_i

adjacent nodes is r , the clustering coefficient of node i is as follows:

$$C_i = \frac{r}{d_i(d_i - 1)/2}. \quad (2)$$

The average clustering coefficient of all nodes in the whole lattice structure is as follows:

$$C = \frac{1}{N} \sum_{i=1}^N C_i, \quad (3)$$

where N denotes the total number of nodes in the lattice structure.

It is assumed that the node i is connected with the node j and the node k in the same layer, but there is no edge between j and k , so the value of r is rather smaller.

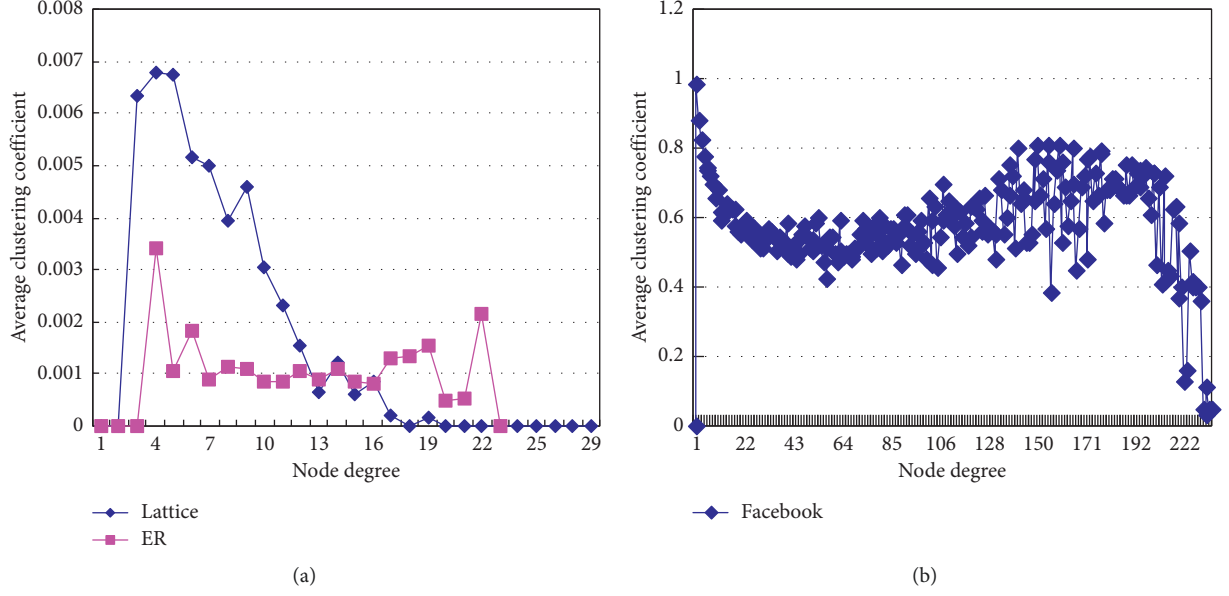


FIGURE 7: Clustering coefficient of three kinds of network structures. (a) Lattice and ER. (b) Facebook.

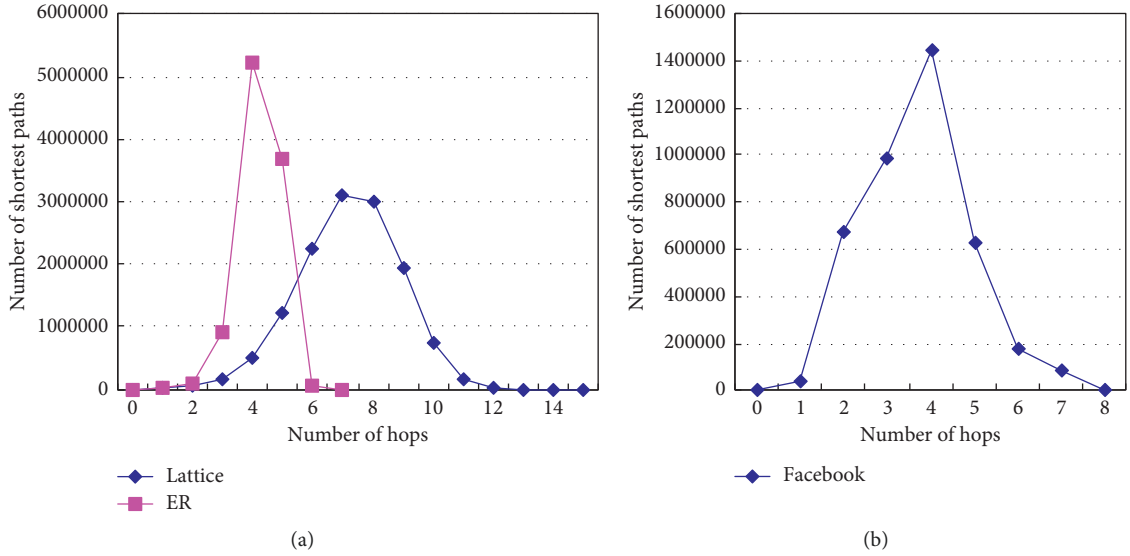


FIGURE 8: Average path length of three kinds of network structures. (a) Lattice and ER. (b) Facebook.

Fewer nodes are connected across layers, so there are fewer edges between the nodes connected with the same node.

In concept lattices, there are few nodes connected across layers and few edges between the nodes connected with the same node. Thus, the clustering coefficient of concept lattices in the graph structure is also very small.

Therefore, combining equations 2 and 3, we can see that data cube lattice and concept lattice have smaller clustering coefficients.

4.1.3. Average Path Length. If the dimension number of nodes in the lattice structure is h , the number of layers is also h since it rolls up from the bottom cuboid to the top cuboid.

Taking any two nodes i and j , the minimum number of edges to pass from i to j is called the shortest distance from i to j and is denoted as $l_{i,j}$. The average distance of all pairs of nodes in the data cube lattice is as follows:

$$l = \frac{2}{M(M-1)} \sum_{i < j} l_{i,j}, \quad (4)$$

where M is the sum-up node of the data cube lattice, and there must be $l_{i,j} \leq h$. Therefore, the data cube lattice has a relatively small average path length.

For concept lattices, the following definitions are given to facilitate the analysis of their graph structure model:

The Hasse diagram of an n -layer concept lattice consists of a triple $G = (N, E, m)$, where N is the set of nodes, E is

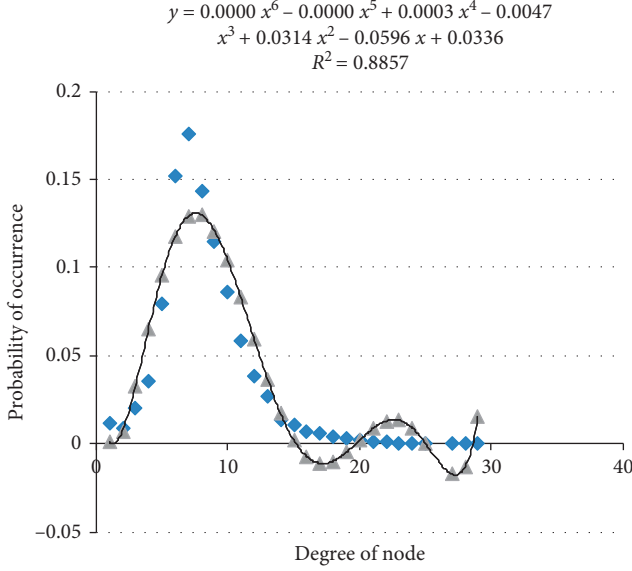


FIGURE 9: Polynomial curve fitting.

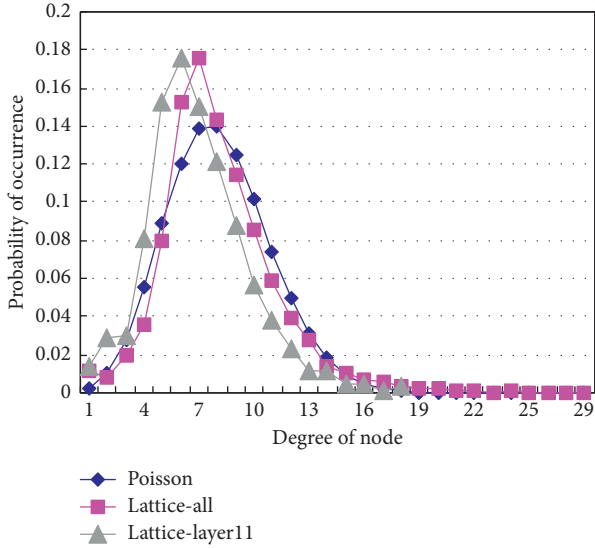


FIGURE 10: Poisson distribution curve fitting I.

set of edges, and m is the number of layers of the concept lattice graph. Therefore, in this hierarchical structure, there are the following properties:

- (1) N consists of m nonempty subsets, $N = N_1 \cup N_2 \cup \dots \cup N_m$ and $N_i \cap N_j = \emptyset$, where $i \neq j$, N_i is the i th layer in the graph
- (2) If the edge $(x, y) \in E$, then $x \in N_i, y \in N_j$ and $i \leq j$, then $j - i$ is the span of edge (x, y) , that is, the distance between the node x and the node j

Let $l_{x,y}$ be the distance from the node x to the node y ; we have $l_{x,y} \leq m$. Combining equation 4, it can be deduced that the graph structure of the concept lattice also has a relatively small average path length.

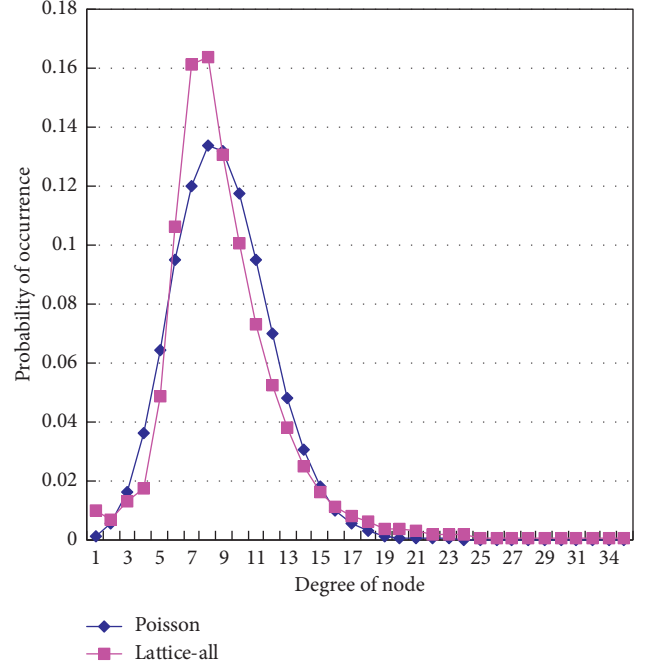


FIGURE 11: Poisson distribution curve fitting II.

4.2. Generation Mechanism. According to the definition of data cube lattices in Definition 2 and the definition of concept lattices in Definition 6, data cube lattices are derived from base tables in the data warehouse, and concept lattices are established based on formal background. When the base table is stored in the relational model, it describes the relationship between attributes and tuples, and formal background describes the relationship between attributes and objects. Attributes Product, Time, and Store in Figure 1 can be mapped to attributes a, b , and c in Figure 2, and the tuple “1” in Figure 1 can be mapped to the object “1” in Figure 2, and so on. By this, attributes in the base table can correspond to attributes and objects in the formal background one by one. Therefore, the structure of the base table in the data cube lattice and the formal background in the concept lattice are the same.

According to [16], when the base table in the data warehouse is taken as the formal background, there is one-to-one correspondence between the covering equivalence class in the data cube lattice and the equivalent feature set in the concept lattice. They both have the same covering tuple set, and the upper bound of each covering equivalence class corresponds to the concept contained in the equivalent feature set. Combined with the definition of equivalent feature sets in Definition 7, the following theorem holds in [16].

Theorem 1. When the base table is taken as the formal background, i.e., $K = (U, A, I)$, where U corresponds to the tuple set of the base table, A corresponds to the dimension attribute set of the base table (without measurement attributes). Let $N \subseteq U, c \in \text{Cube}(N', N'')$ be the concept of the corresponding N , where Cube is a data cube lattice derived from the base table, and c is a data cell. If $\text{BTS}(c) = N$ is

satisfied, then $[c]_{\equiv_{\text{cov}}} = \text{equ}(N)$ and the upper bound u of $[c]_{\equiv_{\text{cov}}}$ is the corresponding conceptual intent N' of N .

Based on Theorem 1, we have the following corollary:

Corollary 1. Let $L = \{u \mid u \text{ is the upper bound of } [c]_{\equiv_{\text{cov}}}\}$ and $L' = \{(A, B) \mid A \in P(U), B \in P(A), (A, B) \text{ is the concept}\}$, then the corresponding data cube lattice $(L, <)$ is equivalent to the concept lattice $(L', <)$, denoted by $(L, <) \equiv (L', <)$.

For example, let $L = \{1(P1, T1, S1), 2(P1, T2, S1)\}$, then the data cell c may take any value of $(P1, *, *)$, $(*, *, S1)$, and $(P1, *, S1)$ in Figure 1. The corresponding concept of L is $((1, 2), (P1, S1))$, and $(P1, *, S1)$ is not only the upper bound of $[c]_{\equiv_{\text{cov}}}$ but also the intent of the concept $((1, 2), (P1, S1))$. Corollary 1 proves that if the base table of the data warehouse is taken as the formal background, the concept lattice derived from the base table is the same as the structure of the data cube lattice which only preserves the upper bounds in equivalence classes (as shown in Figure 12).

5. Related Work

In order to improve the performance of online analysis and decision-making of data warehousing, Gray et al. proposed the data cube operator CUBE [1], which generalizes group-by, cross-tab and subtotal operators, and preaggregates and materializes the attributes (i.e., dimensions) of group-bys.

Laks et al. [12] proposed quotient cubes to compress data cubes efficiently. It uses cover partition to partition data cells with the same upper bound into classes to preserve the semantics of the data cube. The upper and lower bounds of each class are preserved to achieve data cube compression. The closed data cube is proposed in [17], which also implements compression of data cubes through equivalent classes. The difference is that the closed data cube only keeps its upper bound for each class, so it is more efficient to compress the data cube.

In 1982, German mathematician Wile first proposed the theory of formal concept analysis based on concepts and conceptual levels. The core data model is the concept lattice, also known as Galois lattice, which is used to discover, sort, and display concepts [3]. At present, the construction methods of concept lattices are divided into the batch processing algorithm [18] and the progressive algorithm [13]. Zhang et al. [19] studied how to quickly and effectively adjust the original concept lattice to get the concept lattice of the new formal background after some attribute reduction of the formal concept, rather than the reconstruction algorithm in the traditional way. Sarmah et al. [20] proposed an incremental algorithm to reduce multiple attributes. Compared with the incremental algorithm to reduce single attributes, the algorithm only needs to be executed once. With the further improvement and development of the concept lattice theory and methods, the fields of fuzzy theory, spatial clustering, granular computing, and other fields have been intersected and integrated with formal concept analysis and concept lattices, resulting in new applications [16, 21].

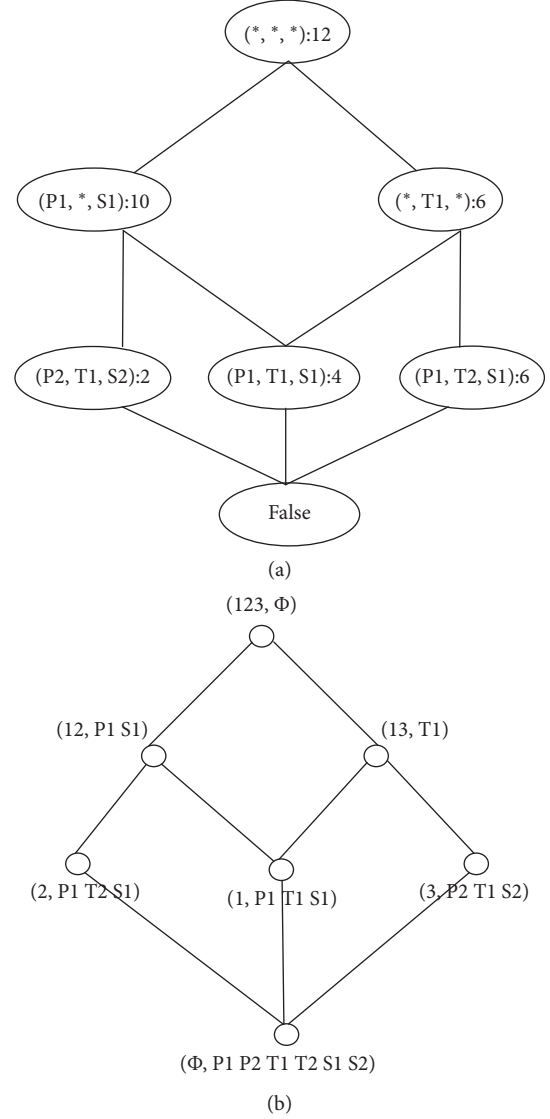


FIGURE 12: Data cube lattice and the corresponding concept lattice. (a) Data cube lattice. (b) Concept lattice.

Until now, none of the above work studies the structural characteristics of data cube lattices and concept lattices in view of the importance of structures.

6. Conclusion and Future Work

This paper analyzes and demonstrates the structural characteristics of data cube lattices and concept lattices based on the synthetic and real data sets. We found the similarities between them in the degree distribution, clustering coefficient, and average path length. We further discuss their similarity of the analytical model and the generation mechanism in the intrinsic perspective.

Our results demonstrate initial promise for exploring the structural characteristics of data cube lattices and concept lattices in data analysis; however, there are many directions of future research. Next, we will investigate if data cube lattices and concept lattices can be unified under the same

framework since they are externally and intrinsically similar. Thus, some efficient algorithms such as construction, reduction, or query in data cube lattices or concept lattices could be applied to each other or be generalized. Besides, we intend on utilizing the lattice structural characteristics such as the degree distribution to facilitate the lattice structure data partitioning in the distributed cloud environment as the structure is the key factor in the load balancing and the communication cost minimizing.

Data Availability

The data used to support the findings of this study are available.

Conflicts of Interest

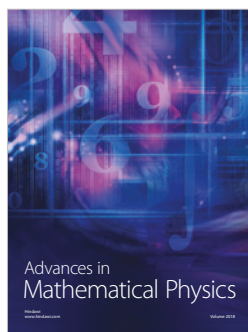
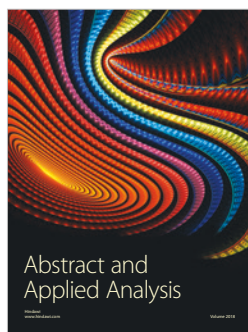
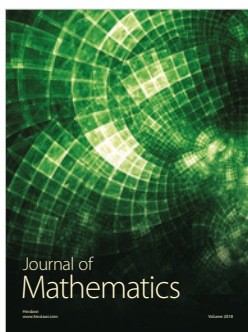
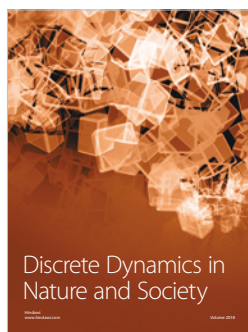
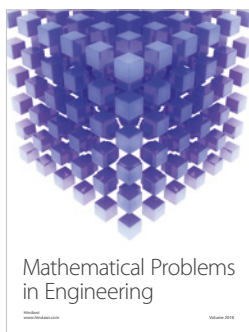
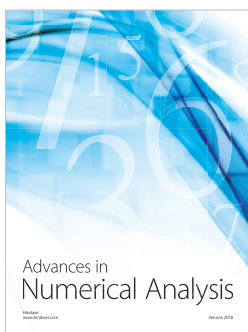
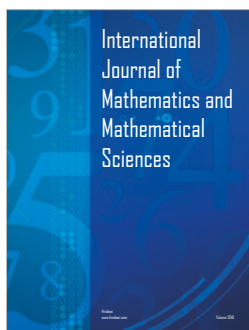
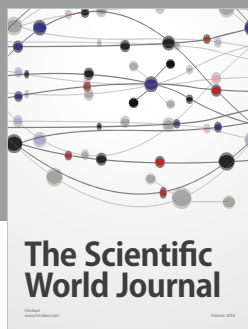
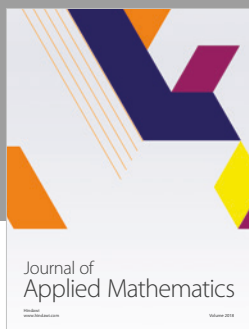
There are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The authors are grateful to Yang Wang for his early work in part related to this paper when he is a graduate in our lab. This work was partially supported by the Natural Science Foundation of China (61462050 and 61562054) and the Natural Science Foundation of Yunnan Province (KKS201603016).

References

- [1] J. Gray, S. Chaudhuri, A. Bosworth et al., "Data cube: a relational aggregation operator generalizing group-by, cross-tab, and sub-totals," *Data Mining and Knowledge Discovery*, vol. 1, no. 1, pp. 29–53, 1997.
- [2] J. Xin, G. Wang, G. Li, Y. Gao, and Z. Zhang, "State of the art data model and its research progress," *Journal of Software*, vol. 30, no. 1, pp. 142–163, 2019.
- [3] R. Wile, "Restructuring lattice theory: an approach based on hierarchies of concepts," in *Ordered Sets*, I. Rival, Ed., Springer, Dordrecht, Netherlands, pp. 445–470, 1982.
- [4] J. Han, M. Kamber, and J. Pei, *Data Mining Concepts and Techniques*, China Machine Press, Beijing, China, 2012.
- [5] X. Zhang, Y. Yao, and Y. Liang, *Rough Set and Concept Lattice*, Xi'an JiaoTong University Press, Xi'an, China, 2006.
- [6] S. Nedjar, F. Pesci, L. Lakhal, and R. Cicchetti, "The agree concept lattice for multidimensional database analysis," in *Proceedings of the International Conference on Formal Concept Analysis*, Nicosia, Cyprus, May 2011.
- [7] Z. Shi, *Research on High Performance Data Cube and Its Semantics*, Ph.D thesis, Beijing Jiaotong University, Beijing, China, 2010.
- [8] S. H. Strogatz, "Exploring complex networks," *Nature*, vol. 410, pp. 268–276, 2001.
- [9] J.-B. Liu, S. Wang, C. Wang, and S. Hayat, "Further results on computation of topological indices of certain networks," *IET Control Theory & Applications*, vol. 11, no. 13, pp. 2065–2071, 2017.
- [10] Y. Zhai, J.-B. Liu, and S. Wang, "Structure properties of koch networks based on networks dynamical systems," *Complexity*, vol. 2017, Article ID 6210878, 7 pages, 2017.
- [11] X. F. Wang and G. Chen, "Complex networks: small-world, scale-free and beyond," *IEEE Circuits and Systems Magazine*, vol. 3, no. 1, pp. 6–20, 2003.
- [12] V. S. Laks, J. Pei, and J. Han, "Quotient cube: how to summarize the semantics of a data cube," in *Proceedings of the 28th International Conference on Very Large Data Bases*, pp. 778–789, VLDB Endowment, Hong Kong, August, 2002.
- [13] S. Anderws, "In-close, a fast algorithm for computing formal concepts," in *Proceedings of the International Conference on Conceptual Structures (ICCS)*, Moscow, Russia, July 2009.
- [14] Béla Bollobás, *Random Graphs*, Cambridge University Press, Cambridge UK, 2001.
- [15] J. Leskovec and R. Sosič, "Snap," *ACM Transactions on Intelligent Systems and Technology*, vol. 8, no. 1, pp. 1–20, 2016.
- [16] X. Kang and D. Miao, "A knowledge acquisition method based on concept lattice in set-valued information systems," *CAAI Transactions on Intelligent Systems*, vol. 11, no. 3, pp. 287–293, 2016.
- [17] S. Li and S. Wang, "Research on closed data cube technology," *Journal of Software*, vol. 15, no. 8, pp. 1165–1171, 2004.
- [18] J. P. Bordat, "Calcul pratique du treillis de galois d'une correspondance," *Mathématiques et Sciences Humaines*, vol. 24, no. 96, pp. 31–47, 1986.
- [19] L. Zhang, H. Zhang, and L. Yin, "Theory and algorithms of attribute decrement for concept lattice," *Journal of Computer Research and Development*, vol. 50, no. 2, pp. 248–259, 2013.
- [20] Y. Ma and W. Ma, "Construction of multi-attributes decrement for concept lattice," *Journal of Software*, vol. 26, no. 12, pp. 3162–3173, 2015.
- [21] A. K. Sarmah, S. M. Hazarika, and S. K. Sinha, "Formal concept analysis: current trends and directions," *Artificial Intelligence Review*, vol. 44, no. 1, pp. 47–86, 2015.



Submit your manuscripts at
www.hindawi.com