Received: March 3, 2024. Revised: May 18, 2024.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Enhancing Formal Concept Analysis with the Kernel Concept Set Approach: A Novel Methodology for Efficient Lattice Reduction

Mohammed Alwersh[1]* **László Kovács[1]**

[1]*Department of Information Technology, University of Miskolc, Miskolc, Hungary*
* Corresponding author's Email: alwersh.mohammed.ali.daash@student.uni-miskolc.hu

Abstract: Formal Concept Analysis (FCA) is a key tool in data analysis and knowledge discovery, yet its application is challenged by the complexity of concept lattices in large datasets. This paper presents the Kernel Concept Set Approach (KCS), a novel methodology that overcomes the limitations of traditional lattice reduction techniques by integrating a flexible derivation cost function and focusing on the frequency and structural importance of concepts. Unlike conventional methods, KCS efficiently operates in a general metric space, reducing computational costs and providing a dynamic approach to conceptual clustering. A comparative study with the K-means Dijkstra on Lattice (KDL) method highlights KCS's superiority in simplifying lattice complexity and enhancing clustering quality. KCS not only maintains crucial data structures but also facilitates the approximation of formal concept lattices, establishing it as an efficient alternative for structured data analysis.

Keywords: Formal concept analysis, FCA, Kernel concept set, Concept lattice reduction, Conceptual clustering, Computational efficiency, Algorithm optimization.

1. Introduction

Formal Concept Analysis (FCA), conceived by Wille in 1982 [1], has become a cornerstone in the realm of knowledge extraction and analysis, finding applications across varied domains like data mining [2], neural networks [3], and social network analysis [4]. Central to FCA is the visualization and interpretation of data through formal concepts, each comprising an extension (a set of objects with common attributes) and an intension (a group of attributes shared by objects). This dual structure has proven effective in discerning complex data relationships and patterns. However, the practical application of FCA faces significant challenges, particularly when dealing with large and intricate concept lattices. These lattices, arising from extensive formal contexts, often pose considerable computational burdens and interpretation difficulties, risking the loss of valuable insights amidst a sea of less relevant details.

In the landscape of Formal Concept Analysis (FCA), the reduction of concept lattices has been a focal area of research, with several methodologies being developed to streamline this process. These methodologies, as detailed in [5], encompass a range of techniques specifically designed for concept lattice reduction. Broadly categorized into three main groups: 1) redundant information removal, 2) simplification, and 3) selection, these strategies each adopt a unique approach based on their underlying ideology and methodologies. While in this paper, we mainly focus on a specific group of techniques which are a selection technique [5]. It is defined as follows: A selection technique is one that, from a formal context or concept lattice, selects a subset of formal concepts, objects or attributes that satisfy a set of constraints.

Building on this foundation, various selection-based strategies have been identified, each bringing unique methodologies to simplify and enhance the interpretability of concept lattices. These techniques, foundational in facilitating a deeper understanding of the datasets, leverage a wide range of approaches to refine the concept selection process, integrating supplementary knowledge to guide their

methodologies effectively. Selection approaches in concept lattice reduction, such as attribute weighting [6] and hierarchical structuring [7], have been instrumental in reducing lattice complexity. Attribute weighting assigns varying degrees of importance to attributes, thereby influencing the significance of the concepts within the lattice. Hierarchical structuring, on the other hand, organizes concepts in a tiered system, simplifying the relationships and dependencies among them. Moreover, the field has seen the integration of logical frameworks [8] and the application of filtering techniques [9] to select portions of formal concepts, objects, or attributes from a lattice based on specific constraints. Such methods have been crucial in identifying and retaining only the most relevant concepts within a lattice. Despite the effectiveness of these existing methods, they often fall short in addressing the dynamic aspect of concept derivation within lattices and may yield less reliable results in complex datasets. In an effort to navigate the complexities inherent to Formal Concept Analysis (FCA) and to effectively address its limitations, we are excited to introduce the Kernel Concept Set Approach (KCS), a selection-based methodology designed to significantly enhance the analytical capabilities within this field. Diverging from the conventional methodologies that predominantly focus on attribute relevance or the frequency of concepts, the KCS approach innovatively integrates the frequency of concepts with their derivation cost, thereby offering a holistic and more sophisticated analysis. What sets the KCS methodology apart from existing approaches is its innovative perspective on concept similarity, facilitated by a flexible derivation cost function. This distinctive feature allows for an analysis that is not only confined to the practical application level but also extends to the internal structure level of the concepts. Such a versatile distance measure broadens the scope of application, providing a more comprehensive understanding of concept relationships.

A key strength of our method lies in its ability to identify concepts that assume pivotal roles as cluster centers within the set of formal concepts. This capability effectively positions the KCS approach as a bespoke clustering method tailored specifically for concept sets. Our methodology demonstrates superior performance over traditional clustering techniques in several critical aspects: it eliminates the need for a vector space, operating efficiently within a general metric space; it offers a cost-effective alternative to standard agglutinative clustering methods; it provides a flexible interpretation of distances; and crucially, it identifies not only the cluster members but also the cluster centroids.

The dual assessment of frequency and derivation cost is a cornerstone of the KCS approach, enabling a nuanced understanding of the lattice structure. This in-depth analysis sheds light on deeper insights into the underlying data patterns and relationships. Consequently, the KCS methodology emerges not just as a novel method for the clustering of concepts but also as an innovative approach to clustering categorical data. Through this advanced strategy, the KCS approach promises to enrich the field of Formal Concept Analysis with a more nuanced and practical tool for deciphering complex data landscapes.

The structure of this paper is methodically organized into six sections for clarity and depth of analysis. Section 2 delves into the Foundational Principles of Formal Concept Analysis presenting core concepts. In Section 3, a thorough exploration of the related work is conducted, providing insights into existing methodologies and their context within Formal Concept Analysis. The fourth section introduces our innovative Kernel Concept Set Approach, detailing its unique methodology and applicability in complex lattice analyses. Section 5 outlines the experimental setup and methodology, delineating the framework employed to validate and assess the effectiveness of our approach. Finally, the paper concludes with Section 6, synthesizing key findings and discussing the broader implications of our research.

2. Foundational principles of formal concept analysis: Preliminaries

In this framework, a concept is viewed as a dual entity comprising two key elements: its extension, which refers to the set of objects it encompasses, and its intension, indicating the set of attributes it entails.

Definition 1:

A formal context can be defined as a triple, denoted as $F_C = (G, M, I)$, where G and M represent distinct sets, and I is a binary relation between G and M, $I \subseteq G \times M$. In this context, G comprises elements known as objects, and M consists of elements referred to as attributes. The relationship $(g, m) \in I$ signifies that the object g possesses the attribute m. Given $A \subseteq G$, we define to derivation functions:

$$A' = \{m \in M \mid \forall g \in A, (g, m) \in I\} \quad (1)$$

Similarly, for a subset $B \subseteq M$, we define:

$$B' = \{g \in G \mid \forall m \in B, (g, m) \in I\} \quad (2)$$

Lemma 1. (Properties of Formal Contexts): Let (G, M, I), be a formal context, with $A_1, A_2 \subseteq G$ as sets of objects, and $B_1, B_2 \subseteq M$ as sets of attributes. Then, the following properties are observed:

$$A_1 \subseteq A_2, \Longrightarrow {A_2}' \subseteq {A_1}' \\ , B_1 \subseteq B_2, \Longrightarrow {B_2}' \subseteq {B_1}' \quad (3)$$

$$A \subseteq A'', \quad B \subseteq B'' \quad (4)$$

$$A' = A''' , B' = B''' \quad (5)$$

$$A \subseteq B' \Leftrightarrow B \subseteq A' \Leftrightarrow A \times B \subseteq I \quad (6)$$

Definition 2:

A formal concept within the context of Formal Concept Analysis is defined as a pair (A, B) where $A \subseteq G$ and $B \subseteq M$, satisfying the conditions $A' = B$ and $B' = A$. This definition implies that $A \subseteq G$ and $B \subseteq M$ are maximal with respect to the relation $A \times B \subseteq I$. In this context, A is referred to as the extent of the concept, and B as the intent part.

Definition 3

Formal concepts can be arranged based on the subconcept-superconcept relation $\leq$, expressed as follows:

$(\mathrm{A}_1, \mathrm{B}_1) \leq (\mathrm{A}_2, \mathrm{B}_2) \Leftrightarrow \mathrm{A}_1 \subseteq \mathrm{A}_2$, (or equivalently $\mathrm{B}_1 \subseteq \mathrm{B}_2$),

where $(\mathrm{A}_1, \mathrm{B}_1)$ is a subconcept (more specific) and $(\mathrm{A}_2, \mathrm{B}_2)$ is a super concept (more general). Within a formal context F_C, the assembly of all formal concepts K, in conjunction with the partial order $\leq$, generally defines the notation $\mathcal{B}(K, \leq)$ signifies the concept lattice derived from a formal context F_C.

Lemma 2. (Concept Lattice Formation): Given a formal context $F_C = (G, M, I)$, the concept lattice $\mathcal{B}(F_C)$ can be expressed as:

$$K(F_C) = \{(B', B'') | \, B \subseteq M)\} \quad (7)$$

This lemma, along with Lemma 1, Eq. (5), indicates that the concept lattice can be constructed from the set of concept intents, offering a method to systematically derive and understand the structure of concepts within a given context. By the initial part of the core theorem on concept lattices [1]. A concept lattice $\mathcal{B}(K, \leq)$ is identified as a complete lattice where the infimum and supremum are present for any arbitrary set; This is represented as:

$$(A_1, B_1) \wedge (A_2, B_2) = (A_1 \cap A_2, (B_1 \cup B_2)'') \quad (8)$$

$$(A_1, B_1) \vee (A_2, B_2) = (\,(A_1 \cup A_2)'', B_1 \cap B_2) \quad (9)$$

For a more comprehensive understanding, readers are directed to the extensive discussions in [10, 11]. Formal Concept Analysis (FCA) is utilized across diverse domains, demonstrating its versatility and effectiveness. It finds applications in areas like knowledge reduction [12], In the context of enhancing classification techniques in natural language processing (NLP) [13], data mining and association rule mining [2, 14], information retrieval [15], neural networks [16, 3], and ontology engineering [17]. Additionally, it plays a significant role in reliability engineering [18] and the analysis of social networks [4, 19, 20]. These fields leverage FCA's modelling capabilities for knowledge extraction and management. For an in-depth exploration of FCA's role in knowledge discovery and information science, readers are directed to a detailed survey available in [21].

3. Related work

Lattice reduction techniques are essential in Formal Concept Analysis (FCA), significantly simplifying and improving the interpretability of concept lattices. Broadly, these methods fall into three categories: redundant information removal, simplification, and selection. Each category employs distinct strategies to streamline the concept lattice, making it easier to analyze and interpret [5].

Concepts Redundant information removal techniques remove redundant information from the concept lattice. In general, they aim to find the minimum set of objects or attributes that keep the structure of the original concept lattice unchanged [22- 24]. Where information removal techniques can be defined as: An object $g \in G$ (set of objects), attribute $m \in M$ (set of attributes) or incidence $i \in I$ ($I \subseteq G \times M$) is considered redundant information if its removal or transformation results in a lattice isomorphic to the original concept lattice. In accordance with this definition, if an object, attribute or incidence can be removed or changed in a way that the resulting concept lattice is isomorphic to the original one, then such elements are redundant.

Simplification methods in Formal Concept Analysis (FCA) are instrumental in distilling the essence of complex concept lattices, aiming to enhance their interpretability and analytical manageability. Techniques such as clustering similar objects or attributes [25], and algebraic reductions like Singular Value Decomposition (SVD) and non-negative matrix factorization, significantly reduce the dimensionality and complexity of concept lattices [26]. The concept lattice reduction algorithm, based on the Discernibility Matrix, utilizes mathematical

structures to streamline concept lattices by identifying and removing redundant attributes through discernibility matrices. This efficiently determines the minimal attribute sets required to preserve the lattice structure. However, the algorithm's notable limitation is its computational demand, particularly with large datasets. The initial calculation of the discernibility matrix and subsequent derivation of the identifiable function are intensive processes. This computational burden can restrict the algorithm's usability in environments where resources are scarce or rapid processing is essential [27]. Neighborhood-based concept lattices present a simplification strategy by using approximation operators to compress the lattice, a method efficient in reducing complexity [28]. The novel approach presented in [29], where the context factorisation is utilized in order to provide a concept set reduction with minimal loss of information. The authors applied the toolset of Boolean factor analysis to decompose the large context matrix into the product of two lower-dimensional matrices. Within the reduction process, the method first generates the set of representative concepts and then constructs the minimal representative concept matrix. Additionally, the work of [30], focuses on information processing in imprecise language environments. The authors propose a linguistic-valued layered concept lattice simplification method using a special three-way clustering. The three-way decision method uses a rough-set oriented approach using three regions, the positive, negative, and boundary regions are viewed as the regions of acceptance, rejection, and non-commitment in a ternary classification. The efficiency of related f-concept analysis in the domain of Pythagorean Fuzzy formal contexts is presented in [31], providing a novel attribute reduction approach, too. The main benefit of this representation format is that it shows imprecision in both objects and attributes at the same time. For the generation of the frequent f-concepts an optimized version of the Apriori-algorithm was utilized. However, its reliance on pseudo similarities may not capture all nuanced relationships, posing a potential drawback in accurately representing the original lattice's structure.

Selection techniques [9], in concept lattice reduction are essential for focusing analysis on the most relevant concepts within extensive lattices, where not all concepts may hold significant value for specific applications. These techniques prioritize concepts based on factors such as the size of a concept's intent or extension [32- 34], or the relationships between specific attributes [35]. By employing such selective filtering, these methods aim to identify and retain concepts that are most pertinent to the analysis at hand, ensuring that the lattice is streamlined for efficiency and relevance. This approach is particularly valuable in contexts where the sheer volume of data can obscure important patterns or relationships, making selection techniques a critical tool in the realm of Formal Concept Analysis (FCA). The tri-granularity model introduced in [36], the concept lattice reduction is based on a tri-granularity model of concept lattices. In this model, the lattice is layered into three granularity levels. The bottom level corresponds to the concept level, the second one covers a set of similar concepts and the top layer represents the whole concept lattice. The work introduces novel methods of local granularity and elementary granularity attribute reduction of three-way concept lattices. These newly proposed two levels of attribute reduction with the existing global granularity attribute reduction together provide a framework for the tri-granularity attribute reduction of three-way concept lattices.

The proposed Kernel Concept Set (KSC) approach is a selection-based strategy. These techniques, involve choosing a portion of formal concepts, objects, or attributes from a lattice or context, based on certain constraints. In a variety of scenarios, deeper understanding of both objects and attributes within a dataset can significantly enhance the process of reducing concept lattices. In the expansive survey of lattice reduction techniques within the realm of Formal Concept Analysis (FCA), various selection-based strategies have been identified, each bringing unique methodologies to simplify and enhance the interpretability of concept lattices. These techniques, foundational in facilitating a deeper understanding of the datasets, leverage a wide range of approaches to refine the concept selection process, integrating supplementary knowledge to guide their methodologies effectively.

The proposed Kernel Concept Set Approach (KCS) differs from the existing approaches in many aspects. It presents a unique approach on concept similarity using a flexible derivation cost function. This distance measure may focus both on the usage level and the internal structure level of the concepts providing a more general application potential.

Another benefit of the proposed approach is that it selects those concepts which have a central role as cluster centers in the set of formal concepts. Thus, this method can be used as a special clustering method on the concepts set.

The method is dominating the standard clustering methods as

- it does not require a vector space; a general metric space is sufficient.

- it has a lower cost compared to the standard agglutinative clustering methods.
- flexible distance interpretation
- it provides the cluster centroids not only the cluster members.

4. Kernel concept set approach

The Kernel Concept Set Approach (KCS) is an innovative response to the challenges posed by the inherently complex nature of concept lattices in Formal Concept Analysis (FCA). This approach is particularly crucial when dealing with large lattices, where traditional methods, such as arbitrary reduction or selection of objects, prove insufficient and potentially overlook key structures within the data. To tackle these complexities, our proposed Kernel Concept Set Approach (KCS) adopts a nuanced methodology, focusing on two essential aspects: the frequency of concepts and their associated derivation cost. The frequency component assesses how prevalent and significant a concept is within the domain, offering a quantifiable metric of its importance. In parallel, the derivation cost aspect evaluates the intricacy and effort required in navigating from one concept to another within the lattice framework.

The essence of the Kernel Concept Set Approach (KCS) is its strategic focus on identifying and prioritizing 'kernel concepts' within the concept lattice. These kernel concepts are distinguished not only by their frequency but also by their critical positioning and role within the lattice's structure, acting as the most informative and structurally significant elements of the domain. Essentially, these kernel concepts form the conceptual backbone of the lattice, guiding the simplification process by highlighting the most essential elements. This methodical focus ensures that the analysis is both efficient and insightful, emphasizing the data's most critical aspects.

Incorporating the proposed method's distinct advantages, KCS introduces a novel perspective on concept similarity through a flexible derivation cost function. This approach allows for an analysis that considers both the practical application level and the internal structure of concepts, offering broad applicability. Furthermore, KCS identifies kernel concepts that assume central roles as cluster centers, establishing this method as a specialized clustering technique for concept sets. This unique clustering approach provides several advantages over traditional methods: it operates within a general metric space without the need for a vector space, offers a cost-effective alternative to standard agglutinative clustering methods, allows for a flexible interpretation of distances, and importantly, it identifies cluster centroids in addition to cluster members.

By honing in on these kernel concepts, KCS not only streamlines the lattice for more manageable analysis but also guarantees that the most informative and essential aspects of the data are emphasized. Thus, KCS presents a highly efficient, insightful, and practical solution for navigating the intricacies of large concept lattices, significantly enhancing the process of knowledge discovery and ensuring a deeper understanding of the underlying data.

Definition 4:

The Extended Concept Lattice in Formal Concept Analysis (FCA) introduces advanced components that enrich the standard concept lattice framework explained in Definition 3. This extension primarily involves two key elements: the Frequency Value function and the Derivation Cost Function:

Definition 5:

Function $f: K \rightarrow \mathbb{R}^+$, assigns a positive real number to each concept in the lattice, representing its frequency within the domain. f (c) denotes the frequency value of c.

Definition 6:

The Derivation Cost Function (d) is defined as $d: K \times K \rightarrow \mathbb{R}^+$, this function calculates the cost of deriving one concept from another within the lattice. Properties:

- Self-Cost: $d\ (c, c) = 0$, for any concept c within the lattice, indicating no cost for self-derivation.
- Asymmetric Cost: For two different concepts c_1 and c_2, $d(c_1, c_2) \neq d\ (c_2, c_1)$, reflecting the directional nature of derivation within the lattice.
- Integration of Dijkstra-Based Distance Measure: To refine the calculation of asymmetric costs between concepts, we have employed the Dijkstra-Based Distance Measure from [34]. This approach computes the shortest path in the lattice considering the direction and cost of the path. Specifically, we have set the cost for upward transitions (parent-to-child) in the lattice as 2 and for downward transitions (child-to-parent) as 1. This integration adds a layer of sophistication to our function d, allowing it to more accurately represent the complexities involved in navigating the concept lattice.

Definition 7:

Distance $d\ (K_1, c) = min\ \{d\ (c_1, c) \mid c_1 \in K_1\}$, calculates the minimum derivation cost from any concept in the subset K_1 to a specific concept c within the lattice. This calculation is vital for accurately determining the extended cost, reflecting

the cost of reaching concept c from the nearest concept within K_1.

Definition 8:

Frequency-Weighted Derivation Cost combines the frequency of concept c with its derivation cost from K_1, offering a holistic measure that encompasses both significance and relational complexity.

$$d^f(K_1, c) = f(c) \cdot d(K_1, c) \quad (10)$$

Definition 9:

The Kernel Concept Set is defined as a triplet $\mathfrak{B}(\boldsymbol{d}, \boldsymbol{f}, \boldsymbol{d}^f)$ extended with the following properties:

- Capacity Constraint: The size of the kernel concept set K_C is fixed and equal to S_c.
- Optimization Constraint: The set is optimized to minimize the aggregated derivation cost, quantifying the conceptual "distance" within the lattice from the kernel set.
- Composition: $K_C \subseteq K$, where K_C is derived as:
- $K_C = argmin_{K_1 \subset K}\{\sum_{c \in K} d^f(K_1, c) | |K_1| \leq S_c\}$ (11)
- Role: KCS focuses on these kernel concepts that provide the most comprehensive structuring of the domain, essentially forming the backbone of the concept lattice. By emphasizing these kernel concepts, the approach streamlines the lattice to its core elements, ensuring that the analysis is both manageable and retains the most critical and informative aspects of the data. This results in a more efficient, insightful, and practical approach for handling the complexities of large concept lattices, enhancing data understanding and knowledge discovery.

4.1 Optimized greedy algorithm for determining kernel concept set

In addressing the computational challenges posed by large concept lattices in Formal Concept Analysis (FCA), we propose an optimized Greedy Algorithm as shown in Algorithm 1, to effectively identify the Kernel Concept Set (KCS). The optimization of the greedy algorithm is designed to systematically construct an optimal Kernel Concept Set that minimizes the total derivation cost across a concept lattice. This cost is a composite measure reflecting the cumulative effort needed to derive all other concepts from a set of core concepts. This refined approach aims to streamline the lattice by focusing on the most significant concepts, thereby reducing its size and complexity. The algorithm's optimization process involves advanced techniques such as employing the ancestors and descendants' relationships of concepts and constructing efficient sub-lattices.

Algorithm 1: Optimized Greedy Algorithm

Input:

- Concept Lattice *B (K, ≤)*
- Frequency Value Function $f: K \rightarrow R^+$
- Maximum Core Set Size S_c
- Transition Cost: $upward \leftarrow 2, downward \leftarrow 1$

Output:

- Kernel Concept Set K_C

Algorithm Steps:

1. Initialization:

- Construct the Concept Lattice $B(K, \leq)$.
- Initialize Kernel Set K_C as an empty set.
- Assign Frequency Values $f(c)$ to each concept c in the lattice.

2. Ancestors and Descendants Preprocessing:

- For each concept c in the lattice, identify its ancestors and descendants.
- Prepare a memoization dictionary to store the minimal derivation costs.

3. Derivation Cost Calculation:

- For each concept c in the lattice:
  - Use Dijkstra's algorithm to calculate the minimal derivation cost $d(K_1, c)$ to every other concept.
  - Store the costs in a structured way for quick retrieval and use memoization to avoid redundant calculations.

4. Core Set Identification with Sub-Lattice Optimization:

- Define S_c as the maximum size for the Kernel set.
- Initialize $best\_cost$ as ∞ and *best_candidate* as None.
- Iteratively expand K_C:
  - For each candidate concep§t not in K_C, construct or retrieve a relevant sub-lattice.
  - Calculate the potential reduction in aggregated derivation cost if the candidate were added to K_C.
  - Update $best\_cost$ and *best_candidate* accordingly.
  - Add the *best_candidate* to K_C and update the cost.
  - Continue until $|K_C| = S_c$ or no further reduction in cost is possible.

5. Result Analysis:

Return the final K_C as the kernel concept set that minimizes the aggregated derivation cost while adhering to the size constraint $|K_C|=S_c$.

The complexity of the optimized greedy algorithm for identifying a kernel concept set within a concept lattice involves several key factors, primarily influenced by the structure and characteristics of the lattice itself. The preprocessing step, which includes determining ancestors and descendants for each concept, generally scales with the square of the number of concepts $O(V^2)$, assuming a densely connected lattice. However, the introduction of sub-lattice optimization significantly reduces the computational burden in subsequent steps. The core computational task involves calculating minimal derivation costs across the lattice using Dijkstra's algorithm. Traditionally, this would imply a cubic complexity $O(V^3)$ when considering all possible paths within the lattice. However, the optimization strategy restricts each calculation to smaller sub-lattices, dramatically reducing the average size of the problem space. If the average size of these sub-lattices is denoted as 's', the complexity for the derivation cost calculations adjusts to $O(V * s)$, a marked improvement, especially if 's' is substantially smaller than 'V', the total number of concepts.

Further efficiency is gained in the iterative kernel set identification process. Here, the algorithm assessively expands the kernel set, each time recalculating the aggregated derivation cost but confined to relevant sub-lattices. This iterative process, while potentially linear in nature $O(s)$, is tempered by the maximum size constraint of the kernel set (S_c) and the use of memoization, which avoids redundant calculations.

Consequently, the overall time complexity of the algorithm is predominantly governed by the derivation cost calculation and the core set identification steps, combining to $O(V * s) + O(S_c * s)$. This represents a substantial optimization over the naive approach, especially in lattices with a large number of concepts but relatively smaller and well-defined sub-lattices.

The process of sub-lattice construction is a fundamental optimization step in algorithms designed for navigating and analyzing concept lattices, particularly in the context of minimizing derivation costs. This process can be detailed as follows:

1. Defining the Sub-Lattice:

- Conceptualization: A sub-lattice is essentially a smaller, more concentrated segment of the original lattice. It includes only those concepts and their interconnections (edges) that are pertinent to the specific computational task at hand. This selective focus allows for a more manageable and relevant section of the lattice to be processed, rather than the entire structure.
- Purpose: The main aim of creating a sub-lattice is to isolate the essential part of the lattice that contains all the necessary information for the current calculation. This targeted approach eliminates the need to process extraneous parts of the lattice, thus optimizing computational efficiency.

2. Utilization in Derivation Cost Calculation:

- Selective Inclusion: During the process of calculating derivation costs, the algorithm constructs a sub-lattice. This sub-lattice selectively incorporates the concepts and edges (connections) that are directly relevant to the calculation. It typically includes the specific concept under examination, the concepts that are already part of the kernel set, and their respective hierarchical relations (ancestors and descendants).
- Reduction of Complexity: By constructing this sub-lattice, the algorithm significantly reduces the number of paths and concepts that need to be considered in the calculation. This reduction is critical in decreasing the overall computational complexity, particularly in lattices with a dense of connections.

3. Strategic Implementation in Algorithmic Processes:

- Dynamic Construction: As the algorithm progresses, especially in iterative processes like the greedy algorithm for kernel set identification, the sub-lattice is dynamically reconstructed or updated to reflect changes in the kernel set or the target concepts. This dynamic nature ensures that the algorithm always works with the most current and relevant subset of the lattice.
- Impact on Efficiency and Accuracy: The use of sub-lattices enhances both the efficiency and accuracy of the algorithm. By focusing on a smaller, more relevant set of data, the algorithm can more quickly and accurately perform calculations related to derivation costs, leading to better optimization of the kernel set.
- Scalability: The sub-lattice construction as described in Algorithm 2, is scalable and can be effectively applied to lattices of various sizes and complexities. This scalability is essential for ensuring that the algorithm remains efficient and

effective even as the size and complexity of the lattice increase.

Algorithm 2: Steps for Building a Sub-Lattice

1. Initialize Relevant Concepts:
- Start with an empty set to hold all relevant concepts.
- Add the two concepts, A and B, to the relevant concepts set.
2. Add Ancestors and Descendants:
- Include all ancestors of A into the relevant concepts set.
- Include all descendants of A into the relevant concepts set.
- Repeat the process for node B, adding both its ancestors and descendants to the relevant concepts set.
3. Create Sub-Lattice:
- Initialize an empty dictionary to represent the sub-lattice.
- For each concept in the relevant concepts set, do the following:
    - Initialize an empty list to store the neighbors of the concept.
    - Retrieve the list of neighbors from the full lattice dictionary.
    - Include a neighbor in the concept's neighbor list only if the neighbor is also in the relevant concepts set.
- Assign the neighbor list to the concept in the sub-lattice dictionary.
4. Return Sub-Lattice:
- The sub-lattice containing only the relevant concepts and edges is now constructed.
- Return the sub-lattice dictionary.

By harnessing these optimization strategies, the algorithm judiciously narrows the computational workload while maintaining a thorough and representative search through the lattice. The result is a kernel set that optimizes cost-effectiveness, a testament to the algorithm's ability to balance depth and breadth in analyzing complex concept lattices.

5. Experimental setup and methodology

The development and testing of our algorithm were carried out in the Python environment, chosen for its broad acceptance and the extensive range of development tools it offers. Our experiments were conducted on a Mac system equipped with an Apple M1 chip and 8GB of RAM, running on Mac OS 14.3.1. This setup provided a stable and robust platform for evaluating the algorithm's performance across different scenarios.

5.1 Clustering performance

In our study, we conduct a meticulous experimental analysis that includes a comparative study of the clustering performance between the Kernel Concept Set Approach (KCS) and the K-means Dijkstra on Lattice (KDL) method [34], within the framework of Formal Concept Analysis (FCA). To evaluate the efficacy of clustering without relying on ground truth labels, which are often unavailable in real-world scenarios, we utilize the Silhouette Coefficient and the Davies-Bouldin Index (DBI) as our metrics of choice.

The Silhouette Coefficient is a measure that evaluates how well a data point has been assigned to its cluster relative to other clusters. This coefficient ranges between -1 and 1, where a high positive value suggests that the data point is well matched to its own cluster and poorly matched to neighboring clusters. The Silhouette Coefficient is calculated as follows:

$$\text{Silhouette Score} = (b - a) \,/\, max\,(a, b) \qquad (11)$$

Where $'a'$ is represents the mean intra-cluster distance, and $'b'$ is the mean nearest-cluster distance. A higher score indicates a data point is appropriately clustered, while a negative score may suggest incorrect cluster assignment. Conversely, the Davies-Bouldin Index (DBI) assesses both the compactness and separation of clusters. Optimal clustering is indicated by lower DBI values, calculated through a series of steps:

1. Compute the average distance (SC_i) between each point in a cluster (S_i) and all other points in the same cluster, representing the intra-cluster distance.
   $SC_i = (1 / n_i)\, \Sigma\, ||x - Z_i||\; for\; x \in S_i$
   where n_i is the number of points in cluster S_i, x is a point in cluster S_i, Z_i is the centroid of cluster S_i, $||x - Z_i||$ is the distance between point x and centroid Z_i.
2. Determine the distance (d_{ij}) between cluster S_i and S_j, using an appropriate distance measure between their centroids.
3. Calculate the ratio R_{ij} between the sum of intra-cluster distances of S_i and S_j, and the inter-cluster distance between S_i and S_j.
   $R_{ij} = (SC_i + SC_j) / d_{ij}$
4. For each cluster S_i, identify the maximum ratio R_i which is the maximum R_{ij} for all $j \neq i$.
   $R_i = max(R_{ij})$ for all $j \neq i$

5. The Davies-Bouldin Index (DBI) is then the average of all R_i values.

$$\text{DBI} = (1 / S) \sum R_i \quad (12)$$

here: S represents the total number of clusters.

A lower DBI signifies superior clustering by indicating clusters that are more compact (lower SC_i) and better separated (higher d_{ij}). Employing these metrics allows for an in-depth evaluation of KCS and KDL methods, highlighting their performance across different datasets in terms of cluster quality and structure without the bias of predefined labels.

The analysis is based on four real-world datasets described in Table 1, ensuring consistency and relevance for the comparative study. These lattices (datasets), each with unique attributes such as object count, attribute number, and lattice density, provide a comprehensive and challenging testbed for algorithm evaluation. The clustering performance of the Kernel Concept Set (KCS) approach, compared to the K-means Dijkstra on Lattice (KDL) method, is highlighted through the evaluation of Silhouette Coefficient scores and Davies-Bouldin Index (DBI) across these datasets as shown in Tables 2 and 3. The Silhouette Coefficient, indicative of how well each data point fits within its cluster relative to other clusters, shows that KCS outperforms KDL in all cases, with scores of 0.406, 0.351, 0.393, and 0.680 for Balance-Scale, Breast Cancer, Tae, and Car Evaluation datasets, respectively. These higher Silhouette scores suggest that KCS not only places data points more appropriately within clusters but also enhances the overall compactness and separation between clusters.

Similarly, the DBI, which assesses the clustering solution based on the compactness and separation of clusters, where lower values indicate better clustering quality, reinforces the superiority of the KCS approach. For Balance-Scale, Breast Cancer, Tae, and Car Evaluation datasets, KCS achieves lower DBI scores of 1.72, 1.35, 1.70, and 1.41, respectively, compared to KDL. This indicates that clusters formed by KCS are more tightly knit and better delineated from each other, signifying an optimal clustering solution.

The enhanced clustering performance of KCS can be attributed to its strategic selection of kernel concepts as cluster centers. This methodology not only leverages the inherent structure and significance within the data but also ensures that clusters are formed around the most pivotal elements of the dataset. By focusing on kernel concepts characterized by their frequency and derivation cost, KCS emphasizes the most informative aspects of the data, resulting in clusters that are not only coherent and compact but also meaningfully distinct.

Furthermore, the flexibility of KCS, which operates within a general metric space without requiring a vector space and at a lower computational cost compared to traditional methods, contributes to its effectiveness in handling complex datasets. The method's capability to provide cluster centroids alongside cluster members facilitates a deeper understanding of the data's intrinsic patterns and relationships

Table 1. Lattice characteristics

Formal Contexts	#Object	#Attributes	Density	# Formal concepts	#Edges
Balance-Scale	625	20	0.18	1070	3822
Breast Cancer	286	43	0.20	2132	7818
Tae	151	101	0.05	276	619
Car Evaluation	1728	21	0.20	3596	14917

Table 2. Silhouette scores comparing KDL and KCS methods across datasets

Datasets	KDL	KCS	#Clusters
Balance-Scale	0.275	**0.406**	3
Breast Cancer	0.125	**0.351**	2
Tae	0.163	**0.393**	3
Car Evaluation	0.382	**0.680**	4

Table 3. DBI index scores comparing KDL and KCS methods across datasets

Datasets	KDL	KCS	# Clusters
Balance-Scale	2.67	**1.72**	3
Breast Cancer	2.88	**1.35**	2
Tae	2.12	**1.70**	3
Car Evaluation	3.34	**1.41**	4

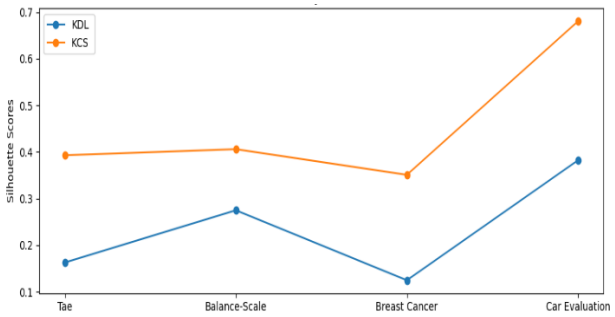

Figure. 1 Silhouette scores by dataset and method

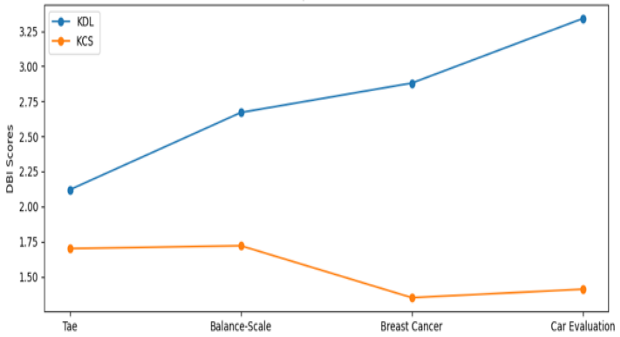

Figure. 2 DBI scores by dataset and method

The data presented in Tables 2 and 3, alongside their graphical representations in Figs. 1 and 2, unequivocally showcase the Kernel Concept Set (KCS) approach's superior clustering performance over the K-means Dijkstra on Lattice (KDL) method across various datasets. This enhanced performance of KCS is attributable to its innovative approach in leveraging concept lattice's intrinsic complexity for clustering, providing a more nuanced and effective analysis of categorical data. Unlike the KDL method, which capitalizes on the lattice structure and Dijkstra's algorithm for clustering, KCS introduces a novel method focused on kernel concept identification, emphasizing the frequency and derivation cost of concepts. This strategy not only simplifies the analysis by reducing the lattice to its most informative elements but also ensures a higher quality of clustering by selecting kernel concepts as cluster centers. The improvement in clustering quality is clearly reflected through higher Silhouette Coefficients and lower Davies-Bouldin Index scores for KCS, indicating more cohesive and well-separated clusters compared to those generated by the KDL method. These results validate the premise that a more targeted and insightful approach, such as KCS, that directly engages with the core elements of categorical data and their hierarchical relationships, significantly enhances clustering outcomes.

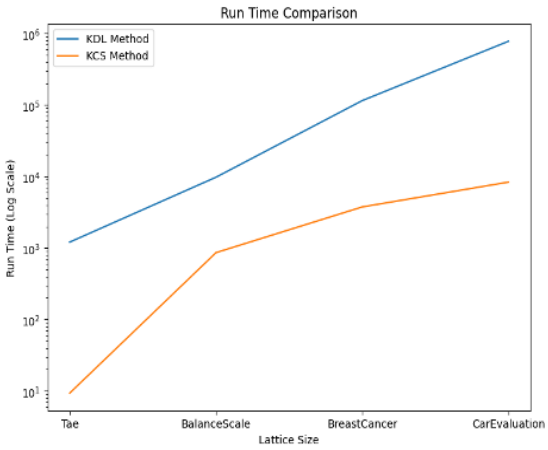

Figure. 3 Comparative performance analysis of KCS and KDL methods across diverse lattice sizes

5.2 Influence of lattice size on runtime

In our focused experimental analysis, we evaluated the runtime efficiency of the Kernel Concept Set Approach (KCS) compared to the K-means Dijkstra on Lattice (KDL) method [34], within the domain of Formal Concept Analysis (FCA). Our objective centered on understanding how these methodologies respond to the challenge posed by varying sizes of lattice structures, leveraging datasets delineated in Table 1 for a cohesive and direct comparison.

The core of our investigation was to uncover the adaptability and efficiency of the KCS and KDL methods when dealing with lattices of increasing complexity. By examining runtime as the primary metric of evaluation, we aimed to provide clear insights into the scalability and operational performance of these approaches across different lattice sizes. The detailed runtime comparison is visually presented in Fig. 3, which illustrates the stark differences in performance metrics across the evaluated datasets. The results from our study, as highlighted in Fig. 3, showcase a distinctive contrast in performance between the KCS and KDL methods, particularly as lattice sizes escalate. While the KDL method showed reasonable efficiency in managing smaller lattice sizes, indicating its potential in less complex scenarios, it struggled significantly as the lattice size increased. This was evident in the pronounced rise in runtime, which underscored scalability and efficiency challenges inherent in the KDL approach for larger and more complex lattice structures.

Conversely, the KCS method demonstrated remarkable efficiency across the entire range of lattice sizes. It consistently outperformed the KDL method in terms of runtime, even in scenarios

involving larger and more intricate lattices. For example, in processing the "Tae" dataset with 276 concepts, the KDL method required 1210.14 seconds, whereas the KCS method drastically reduced this to just 9.35 seconds. This pattern of enhanced efficiency with the KCS method persisted across all evaluated datasets, with the "Car Evaluation" dataset showing a notable decrease in runtime from 781799.93 seconds (KDL) to 8361.93 seconds (KCS) for 3596 concepts.

These findings underscore the KCS method's superior adaptability and scalability, presenting it as a more efficient solution for FCA applications across varied lattice complexities. The efficiency of the KCS method in minimizing runtime, irrespective of the lattice size, speaks volumes about its potential to handle complex data-intensive environments effectively. The analysis exclusively focusing on runtime reveals the KCS method as a highly scalable and efficient approach for managing the intricacies of large concept lattices in FCA. By significantly reducing the runtime needed to process extensive lattices, the KCS method enhances the practicality and applicability of FCA in analyzing complex datasets, setting a new benchmark for future advancements in the field, as clearly demonstrated in Fig. 3.

5.3 Experiment with the teaching assistant evaluation dataset

In this example, we turn our attention to the Teaching Assistant Evaluation dataset from the UCI KDD Archive. This dataset provides an in-depth analysis of 151 teaching assistant performances at the University of Wisconsin-Madison's Statistics Department, captured across a range of semesters, including both regular and summer sessions. Available at UCI KDD (https://archive.ics.uci.edu/dataset/100/teaching+assistant+evaluation), this dataset has become an invaluable asset in educational research, particularly in the assessment of teaching effectiveness. The dataset is characterized by 6 categorical attributes. These attributes cover a variety of aspects: the TA's native language (English or non-English speaker), the course instructor (across 25 categories), the specific course (26 different types), the type of semester (summer or regular), the class size. This comprehensive attribute set aims to capture the multifaceted nature of teaching performance evaluation.

For the purpose of applying Formal Concept Analysis (FCA), these categorical attributes are converted into Boolean values, creating a formal context that includes 151 instances (TA assignments) and their respective 101 attributes with density of 0.05. This conversion allows for a more granular analysis of the dataset, facilitating the identification of patterns and relationships crucial for understanding the dynamics of teaching performance. In demonstrating the principles of FCA in Section 2, we initially focus on a subset of these instances, comprising the first 10 TA assignments and a selection of 8 attributes. The concept lattice displayed in Fig. 4, constructed from the formal context given in Table 4, is visually represented through a line diagram. This lattice comprises various formal concepts, each emerging from the relationships and interactions within the formal context, adhering to the subconcept-superconcept framework as per reference [10]. In this diagram, every node signifies a formal concept, with these concepts being divisible into two primary categories: object concepts (denoted as $\gamma(g) = (\{g\}'', \{g\}')$) and attribute concepts (denoted as $\mu(m) = (\{m\}', \{m\}'')$), which are related to specific objects or attributes respectively.

Table 4. Formal context about subset of TAs dataset

	Class_Size_17	Eng_Nat_spk_1	Eng_Nat_Spk_2	Summer_or_Regular_1	Summer_or_Regular_2	Course_3	Course_Instructor_13	Course_Instructor_23
TA 1	X	X		X		X		X
TA 2	X	X			X	X		X
TA 3	X		X		X	X	X	
TA 4	X		X	X		X		X
TA 5	X	X			X	X	X	
TA 6	X		X		X	X		X
TA 7	X		X		X	X		X
TA 8	X	X		X		X	X	
TA 9	X	X		X		X	X	
TA 10	X	X		X		X	X	

In our notation, each object g is labeled directly at the node representing the smallest concept containing g in its extent, while every attribute m is labeled at the node corresponding to the largest concept including m in its intent. This labeling strategy is pivotal for interpreting the relationships within the context: an object g possesses an attribute m if there's a direct ascending path in the diagram from g's node to m's node. The extent of any given concept is formed by all objects situated lower in the hierarchical structure, and similarly, the intent of a concept is constituted by all attributes that are

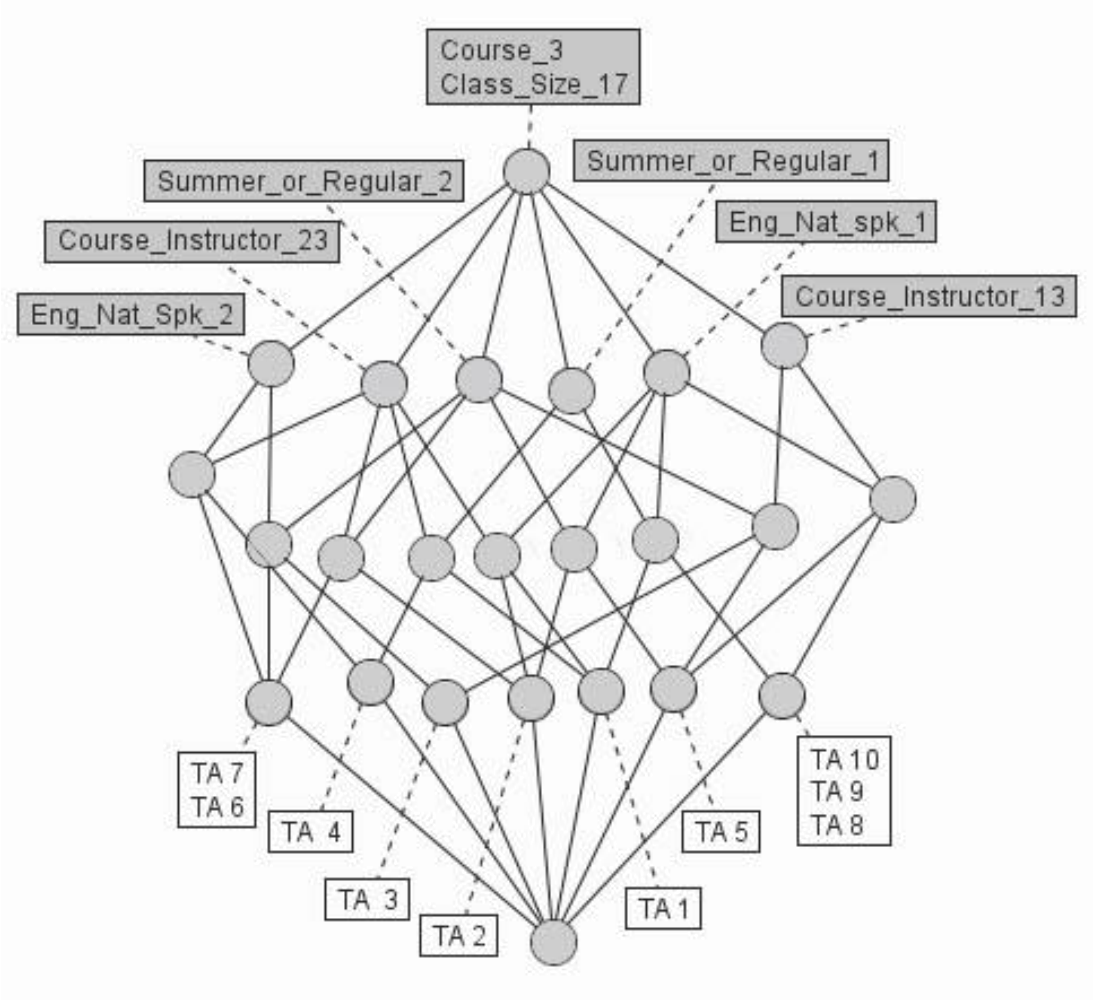

Figure. 4 Concept lattice derived from the formal context of tae dataset

Table 5. Kernel concept set analysis of TA assignments ($\boldsymbol{S_c}$ set to 5%)

Concept ID	Number of TAs Sharing Attributes	Highlighted Attributes
1	2	Course_3, Summer, Course_Instructor_15, Class_Size_17, Eng_Nat_spk_2
2	2	Class_Size_19, Course_3, Summer, Course_Instructor_23, Eng_Nat_spk_1
3	3	regular, Eng_Nat_spk_2, Course_1, Class_Size_51
4	3	Summer_or_regular_2, Eng_Nat_spk_2, Course_3, Course_Instructor_8
5	3	Summer_or_regular_2, Eng_Nat_spk_2, Course_5, Course_Instructor_9
6	3	Summer_or_regular_2, Course_3, Course_Instructor_22, Eng_Nat_spk_1
7	4	Course_7, Eng_Nat_spk_2, Summer_or_regular_2, Course_Instructor_25
8	4	Summer_or_regular_2, Eng_Nat_spk_2, Course_3, Course_Instructor_23
9	6	Eng_Nat_spk_2, Class_Size_20, Summer_or_regular_1, Course_3
10	7	Summer_or_regular_2, Eng_Nat_spk_2, Course_15
11	8	Summer_or_regular_2, Eng_Nat_spk_2, Course_Instructor_7, Course_11
12	14	Summer_or_regular_2, Eng_Nat_spk_2, Course_2
13	108	Summer_or_regular_2, Eng_Nat_spk_2
14	128	Summer_or_regular_2

positioned higher up in the hierarchy. For instance, the concept labeled [Course_Instructor_13] in the lattice depicted in Fig. 4 has {TA 3, TA 5, TA 8, TA 9, TA 10} as extent, and {Course_3, Class_Size_17, Course_Instructor_13} as intent of the concept. It's important to note that in this context, not every concept is exclusively an object or attribute concept; some may represent a combination of both or neither, as indicated in references [11, 9]. This nuanced categorization allows for a detailed and comprehensive understanding of the relationships and structures within the formal context as depicted in the concept lattice.

In the comprehensive analysis of the Teaching Assistant Evaluation dataset, the application of the Kernel Concept Set (KCS) methodology showcases its effectiveness in simplifying the dataset's complex concept lattice, which comprises 276 concepts. The methodology's focus is on a kernel concept set, carefully selected by applying a size parameter S_c, initially set to capture 5% of the total concepts. This process results in the identification of 14 pivotal concepts, as detailed in Table 5, collectively bearing an aggregate derivation cost of 30,808. It's pivotal to mention that the frequencies of these concepts were randomly assigned to illustrate the KCS approach, underscoring the methodology's adaptability to various analytical scenarios. This initial kernel concept set offers an insightful glimpse into the dataset's core structure and relationships, revealing distinct patterns and preferences in teaching assistant (TA) assignments across the Statistics Department. Notably, Concepts 13 and 14, focusing exclusively on semester type and TA language proficiency, emerge as critical, encapsulating 138 unique TAs out of the total 151. This selection process demonstrates the KCS method's capacity to distill essential assignment characteristics into a manageable and informative framework, highlighting a departmental trend towards favoring regular semester courses and non-English-speaking TAs. Moreover, the analysis illuminates the structured approach to TA assignments within the department, with attributes such as "Course_3", "Summer_or_regular_1", and "Summer_or_regular_2" frequently appearing across the kernel concepts. The specific detailing of class sizes and instructor identifiers within these concepts suggests a deliberate consideration of class dynamics and teaching effectiveness in the TA allocation process.

To obtain a more granular view of the Teaching Assistant Evaluation dataset, an adjustment to the size parameter S_c is necessary. By setting S_c to 8% of the total concepts within the lattice, as depicted in Table 6, we not only preserve the initial set outlined in Table 5, which corresponds to S_c at 5%, but also incorporate 8 new concepts, now totaling 22, with an aggregate derivation cost of 26,768. These additional concepts, highlighted for emphasis, afford deeper insights into the dataset's structural nuances. This enhanced kernel concept set reveals intricate attributes and their relationships with teaching assistants (TAs), shedding light on specific class sizes, course types, semester types, and instructor preferences that the preliminary analysis might have overlooked. The inclusion of these new concepts unravels more complex patterns in TA allocations, such as the discernible preference for non-English-speaking assistants during regular semesters across a diversity of courses and settings, alongside the strategic placement of English-speaking TAs in summer semesters. The infusion of Concepts 1, 2, 3, 4, 13, 15, 16, and 20 into our extended analysis elucidates departmental strategies aimed at diversifying TA assignments, with a particular emphasis on linguistic abilities and instructional requisites. This comprehensive exploration, enabled by the recalibration of the S_c parameter to 8%, significantly amplifies our understanding of the intricate criteria steering TA assignments, thereby affirming the Kernel Concept Set Approach's role in facilitating a more robust knowledge discovery process in educational data analysis.

The significant decrease in derivation cost from 30,808 to 26,768 not only underscores the KCS method's adeptness at refining the lattice's structure but also highlights its strategic acumen in isolating kernel concepts that encapsulate the most salient patterns and relationships within the dataset. This methodological precision significantly minimizes the analytical effort needed to explore and interpret the extensive concept set, thus facilitating a more streamlined, clear, and insightful analysis. The efficiency and depth provided by the KCS approach enhance the interpretability of complex datasets, enriching the analytical process with more nuanced insights into teaching assistant assignments and their underlying dynamics.

This efficiency is further elucidated as we delve into the systematic expansion of the kernel set size from 5% to 20%, a process vividly depicted in Fig. 5. The trend captured therein illustrates a key characteristic of the Kernel Concept Set (KCS) approach: as the kernel set size increases, the derivation cost consistently decreases. Beginning with S_c at 5%, the derivation cost stands at its peak of 30,808, which then progressively diminishes as the kernel set expands—reaching 24,274 for 10%, 19,782 for 15%, and eventually 16,132 for an S_c of 20%. This descending trend underscores a principle

Table 6. Kernel concept set analysis of TA assignments (S_c set to 8%)

Concept ID	Number of TAs Sharing Attributes	Highlighted Attributes
1	**1**	**Class_Size_11, Course_19, Summer_or_regular_2, Eng_Nat_spk_2, Course_Instructor_16**
2	**1**	**Course_Instructor_1, Summer_or_regular_2, Eng_Nat_spk_2, Course_8, Class_Size_18**
3	**1**	**Class_Size_39, Summer_or_regular_2, Course_2, Eng_Nat_spk_2, Course_Instructor_9**
4	**2**	**Course_3, Class_Size_13, Summer_or_regular_1, Eng_Nat_spk_1, Course_Instructor_13**
5	2	Course_3, Summer, Course_Instructor_15, Class_Size_17, Eng_Nat_spk_2
6	2	Class_Size_19, Course_3, Summer, Course_Instructor_23, Eng_Nat_spk_1
7	3	regular, Eng_Nat_spk_2, Course_1, Class_Size_51
8	3	Summer_or_regular_2, Eng_Nat_spk_2, Course_3, Course_Instructor_8
9	3	Summer_or_regular_2, Eng_Nat_spk_2, Course_5, Course_Instructor_9
10	3	Summer_or_regular_2, Course_3, Course_Instructor_22, Eng_Nat_spk_1
11	4	Course_7, Eng_Nat_spk_2, Summer_or_regular_2, Course_Instructor_25
12	4	Summer_or_regular_2, Eng_Nat_spk_2, Course_3, Course_Instructor_23
13	**5**	**Summer_or_regular_2, Eng_Nat_spk_2, Course_3, Course_Instructor_10**
14	6	Eng_Nat_spk_2, Class_Size_20, Summer_or_regular_1, Course_3
15	**7**	**Course_Instructor_18, Eng_Nat_spk_2, Summer_or_regular_2**
16	**7**	**Course_Instructor_13, Eng_Nat_spk_2, Summer_or_regular_2**
17	7	Summer_or_regular_2, Eng_Nat_spk_2, Course_15
18	8	Summer_or_regular_2, Eng_Nat_spk_2, Course_Instructor_7, Course_11
19	14	Summer_or_regular_2, Eng_Nat_spk_2, Course_2
20	**20**	**Summer_or_regular_2, Eng_Nat_spk_1**
21	108	Summer_or_regular_2, Eng_Nat_spk_2
22	128	Summer_or_regular_2

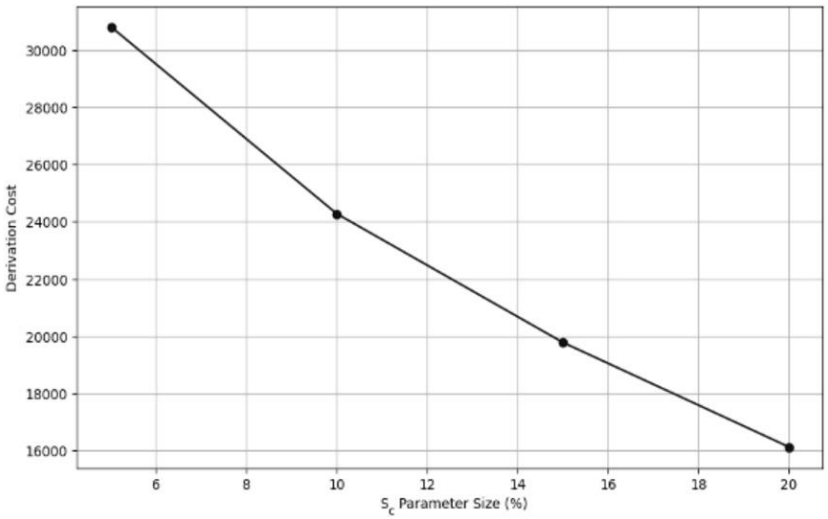

Figure. 5 Trend of decreasing derivation cost with incremental expansion of kernel set size (S_c)

of efficiency inherent in the KCS method; larger kernel sets can seamlessly integrate additional concepts without disproportionately amplifying the complexity of navigating through the concept lattice. Such efficiency implies that the newly included concepts are intricately woven into the existing lattice structure, thereby streamlining the entire analytical framework. This harmonious integration of broader data aspects into the kernel set, without overburdening the analytical endeavor, showcases the method's robust scalability and adaptability. It positions the KCS as an effective tool for in-depth exploration of complex datasets, enabling richer pattern extraction and more informed decision-making.

6. Conclusion

This work introduces the Kernel Concept Set Approach (KCS), a significant advancement in the field of Formal Concept Analysis (FCA) that addresses the inherent challenges of analyzing large and complex concept lattices. Through a novel integration of concept frequency and derivation cost, KCS transcends traditional lattice reduction techniques, offering a dynamic and efficient strategy for simplifying and understanding data structures. Our comparative study with the K-means Dijkstra on Lattice (KDL) method underscores KCS's superior ability to reduce computational complexity while retaining essential data integrity. This enhances the practicality of FCA across various domains, facilitating deeper insights into data analysis and knowledge discovery.

KCS stands out for its ability to operate within a general metric space, significantly lowering computational costs compared to standard methods. Moreover, it introduces a flexible approach to conceptual clustering, centering on kernel concepts that serve as pivotal cluster centroids. This methodology not only streamlines the clustering process but also ensures that the most informative and structurally significant elements of the data are highlighted. The findings from our tests demonstrate that KCS is not merely a tool for lattice reduction but also an effective method for the approximation of formal concept lattices. It presents a comprehensive solution that broadens the application potential of FCA, making it a valuable asset for researchers and practitioners seeking to navigate the complexities of large-scale datasets.

Author Contributions

Author Contributions: Conceptualization, László Kovács; methodology, Mohammed Alwersh; validation, Mohammed Alwersh; supervision, László Kovács; investigation, Mohammed Alwersh; formal analysis, Mohammed Alwersh. and László Kovács; data curation, Mohammed Alwersh; writing—original draft preparation, Mohammed Alwersh; writing—review and editing, Mohammed Alwersh and László Kovács. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement

The data we used for evaluation can be found at the following links:
Balance-Scale dataset:
https://archive.ics.uci.edu/dataset/12/balance+scale (accessed on 14 December 2023),
Breast Cancer dataset:
https://archive.ics.uci.edu/dataset/14/breast+cancer (accessed on 14 December 2023),
Tae Dataset:
https://archive.ics.uci.edu/dataset/100/teaching+assistant+evaluation (accessed on 14 December 2023),
Car Evaluation dataset:
https://archive.ics.uci.edu/dataset/19/car+evaluation (accessed on 14 December 2023)

Conflicts of Interest

The authors declare no conflict of interest.

References

[1] R. Wille, “Restructuring lattices theory: an approach on hierarchies of concepts”, *Dordrecht*, Holland: Springer, 1982.

[2] K. Sumangali, and C. A. Kumar, “Critical analysis on open source LMSs using FCA”, *International Journal of Distance Education Technologies (IJDET)*, Vol. 11, No. 4, pp. 97-111, 2013.

[3] M. Priya, and A. K. Ch, “A novel method for merging academic social network ontologies using formal concept analysis and hybrid semantic similarity measure”, *Library Hi Tech*, 2019.

[4] F. Hao, Y. Yang, G. Min, and V. Loia, “Incremental construction of three-way concept lattice for knowledge discovery in social networks”, *Inf Sci (N Y)*, Vol. 578, pp. 257-280, 2021.

[5] S. M. Dias, and N. J. Vieira, “Concept lattices reduction: Definition, analysis and classification”, *Expert Syst Appl*, Vol. 42, No. 20, pp. 7084-7097, 2015.

[6] R. Belohlavek, and J. Macko, “Selecting important concepts using weights”, In: *Proc. of*

International Conf on Formal Concept Analysis, Springer, pp. 65-80, 2011.

[7] R. Bělohlávek, V. Sklenář, and J. Zacpal, "Formal concept analysis with hierarchically ordered attributes", *Int J Gen Syst*, Vol. 33, No. 4, pp. 383-394, 2004.

[8] R. Belohlavek, and V. Vychodil, "Formal concept analysis with background knowledge: attribute priorities", *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, Vol. 39, No. 4, pp. 399-409, 2009.

[9] M. Alwersh, and L. Kovács, "Survey on attribute and concept reduction methods in formal concept analysis", *Indonesian Journal of Electrical Engineering and Computer Science*, Vol. 30, No. 1, pp. 366-387, 2023.

[10] B. Ganter and R. Wille, *Formal Concept Analysis: Mathematical Foundations*, Springer-Verlag, Berlin, Heidelberg, 1999.

[11] K. Sumangali, and C. A. Kumar, "A comprehensive overview on the foundations of formal concept analysis", *Knowledge Management & E-Learning: An International Journal*, Vol. 9, No. 4, pp. 512-538, 2017.

[12] K. Sumangali, and C. Aswani, Kumar, "Knowledge reduction in formal contexts through CUR matrix decomposition", *Cybern Syst*, Vol. 50, No. 5, pp. 465-496, 2019.

[13] L. Kovács, "Concept lattice-based classification in NLP", In: *Proc. of the 14th International Conference on Interdisciplinarity in Engineering*, Târgu Mureș, Romania, Vol. 63, No. 1, p. 48, 2020.

[14] Y. Liu and X. Li, "Application of formal concept analysis in association rule mining", In: *Proc. of 2017 4th International Conference on Information Science and Control Engineering (ICISCE)*, pp. 203-207, 2017.

[15] S. Hao, C. Shi, Z. Niu, and L. Cao, "Concept coupling learning for improving concept lattice-based document retrieval", *Eng Appl Artif Intell*, Vol. 69, pp. 65-75, 2018.

[16] L. E. Zárate, and S. M. Dias, "Qualitative behavior rules for the cold rolling process extracted from trained ANN via the FCANN method", *Eng Appl Artif Intell*, Vol. 22, No. 4-5, pp. 718-731, 2009.

[17] L. González, and A. Hogan, "Modelling dynamics in semantic web knowledge graphs with formal concept analysis", In: *Proc. of the 2018 World Wide Web Conference*, pp. 1175-1184, 2018.

[18] C. M. Rocco, E. Hernandez-Perdomo, and J. Mun, "Introduction to formal concept analysis and its applications in reliability engineering", *Reliab Eng Syst Saf*, Vol. 202, p. 107002, 2020.

[19] F. Hao, Y. Yang, B. Pang, N. Y. Yen, and D.-S. Park, "A fast algorithm on generating concept lattice for symmetry formal context constructed from social networks", *J Ambient Intell Humaniz Comput*, pp. 1-8, 2019.

[20] Y. Yang, F. Hao, B. Pang, G. Min, and Y. Wu, "Dynamic maximal cliques detection and evolution management in social internet of things: A formal concept analysis approach", *IEEE Trans Netw Sci Eng*, Vol. 9, No. 3, pp. 1020-1032, 2021.

[21] J. Poelmans, D. I. Ignatov, S. O. Kuznetsov, and G. Dedene, "Formal concept analysis in knowledge processing: A survey on applications", *Expert Syst Appl*, Vol. 40, No. 16, pp. 6538-6560, 2013.

[22] J. Medina, "Relating attribute reduction in formal, object-oriented and property-oriented concept lattices", *Computers & Mathematics with Applications*, Vol. 64, No. 6, pp. 1992-2002, 2012.

[23] J. Li, C. Mei, and Y. Lv, "A heuristic knowledge-reduction method for decision formal contexts", *Computers & Mathematics with Applications*, Vol. 61, No. 4, pp. 1096-1106, 2011.

[24] S. Peng, and A. Yamamoto, "Concept Lattice Reduction Using Integer Programming", *Knowledge-based systems research group/Japan Society for Artificial Intelligence [edited]*, Vol. 123, pp. 38-43, 2021.

[25] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review", *ACM Computing Surveys (CSUR)*, Vol. 31, No. 3, pp. 264-323, 1999.

[26] K. S. K. Cheung, and D. Vogel, "Complexity reduction in lattice-based information retrieval", *Inf Retr Boston*, Vol. 8, pp. 285-299, 2005.

[27] C. Wang, Y. Bo, and C. Xu, "Attribute reduction algorithm on concept lattice and application in smart city energy consumption analysis", *Wirel Commun Mob Comput*, Vol. 2022, 2022.

[28] H. Yang, K. Qin, Q. Hu, and L. Yang, "Neighborhood based concept lattice", *Applied Intelligence*, Vol. 53, No. 5, pp. 6025-6040, 2023.

[29] S. Zhao, J. Qi, J. Li, and L. Wei, "Concept reduction in formal concept analysis based on representative concept matrix", *International Journal of Machine Learning and Cybernetics*, Vol. 14, No. 4, pp. 1147-1160, 2023.

[30] K. Pang, P. Liu, S. Li, L. Zou, M. Lu, and L. Martínez, "Concept lattice simplification with fuzzy linguistic information based on three-way

clustering", *International Journal of Approximate Reasoning*, Vol. 154, pp. 149-175, 2023.
[31] M. Akram, H. S. Nawaz, and M. Deveci, "Attribute reduction and information granulation in Pythagorean fuzzy formal contexts", *Expert Syst Appl*, Vol. 222, p. 119794, 2023.
[32] J.-F. Boulicaut and J. Besson, "Actionability and formal concepts: A data mining perspective," In: *Proc. of International Conf on Formal Concept Analysis*, Springer, pp. 14-31, 2008.
[33] L. PISKOVÁ, T. HORVÁTH, and S. KRAJČI, "RANKING FORMAL CONCEPTS BY UTILIZING MATRIX FACTORIZATION", *Studia Universitatis Babes-Bolyai, Informatica*, Vol. 59, 2014.
[34] M. Alwersh, and L. Kovács, "K-Means Extensions for Clustering Categorical Data on Concept Lattice", *International Journal of Advanced Computer Science and Applications*, Vol. 14, No. 9, 2023.
[35] R. Belohlávek, V. Sklenar, and J. Zacpal, "Concept Lattices Constrained by Attribute Dependencies", In: *Proc. of DATESO*, Citeseer, pp. 63-73, 2004.
[36] Z. Wang, C. Shi, L. Wei, and Y. Yao, "Tri-granularity attribute reduction of three-way concept lattices", *Knowl Based Syst*, Vol. 276, p. 110762, 2023.