Categorical Data Clustering via Value Order Estimated Distance Metric Learning

YIQUN ZHANG*, School of Computer Science and Technology, Guangdong University of Technology, China and Department of Computer Science, Hong Kong Baptist University, Hong Kong
MINGJIE ZHAO*, Department of Computer Science, Hong Kong Baptist University, Hong Kong
HONG JIA†, College of Electronics and Information Engineering, Shenzhen University, China
MENGKE LI, College of Computer Science and Software Engineering, Shenzhen University, China
YANG LU, School of Informatics, Xiamen University, China
YIU-MING CHEUNG†, Department of Computer Science, Hong Kong Baptist University, Hong Kong

Clustering is a popular machine learning technique for data mining that can process and analyze datasets to automatically reveal sample distribution patterns. Since the ubiquitous categorical data naturally lack a well-defined metric space such as the Euclidean distance space of numerical data, the distribution of categorical data is usually under-represented, and thus valuable information can be easily twisted in clustering. This paper, therefore, introduces a novel order distance metric learning approach to intuitively represent categorical attribute values by learning their optimal order relationship and quantifying their distance in a line similar to that of the numerical attributes. Since subjectively created qualitative categorical values involve ambiguity and fuzziness, the order distance metric is learned in the context of clustering. Accordingly, a new joint learning paradigm is developed to alternatively perform clustering and order distance metric learning with low time complexity and a guarantee of convergence. Due to the clustering-friendly order learning mechanism and the homogeneous ordinal nature of the order distance and Euclidean distance, the proposed method achieves superior clustering accuracy on categorical and mixed datasets. More importantly, the learned order distance metric greatly reduces the difficulty of understanding and managing the non-intuitive categorical data. Experiments with ablation studies, significance tests, case studies, etc., have validated the efficacy of the proposed method. The source code is available at https://github.com/csmjzhao/OCL_Source_Code.

CCS Concepts: • **Computing methodologies** → **Cluster analysis**.

Additional Key Words and Phrases: Cluster analysis, categorical data, subspace distance structure, distance learning, partitional clustering

ACM Reference Format:
Yiqun Zhang, Mingjie Zhao, Hong Jia, Mengke Li, Yang Lu, and Yiu-ming Cheung. 2025. Categorical Data Clustering via Value Order Estimated Distance Metric Learning. *Proc. ACM Manag. Data* 3, 6 (SIGMOD), Article 307 (December 2025), 24 pages. https://doi.org/10.1145/3769772

*Co-first author
†Corresponding author

Authors' Contact Information: Yiqun Zhang, School of Computer Science and Technology, Guangdong University of Technology, Guangzhou, China and Department of Computer Science, Hong Kong Baptist University, Kowloon Tong, Hong Kong, yqzhang@gdut.edu.cn; Mingjie Zhao, Department of Computer Science, Hong Kong Baptist University, Kowloon Tong, Hong Kong, mjzhao@comp.hkbu.edu.hk; Hong Jia, College of Electronics and Information Engineering, Shenzhen University, Shenzhen, China, hongjia1102@szu.edu.cn; Mengke Li, College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China, mengkeli@szu.edu.cn; Yang Lu, School of Informatics, Xiamen University, Xiamen, China, luyang@xmu.edu.cn; Yiu-ming Cheung, Department of Computer Science, Hong Kong Baptist University, Kowloon Tong, Hong Kong, ymc@comp.hkbu.edu.hk.

ACM 2836-6573/2025/12-ART307
https://doi.org/10.1145/3769772

1 Introduction

Categorical attributes with qualitative values (e.g., {lawyer, doctor, computer scientist} of an attribute "occupation") are common in cluster analysis of real datasets. As the categorical values are cognitive conceptual entities that cannot directly attend arithmetic operations [2, 17], most existing clustering techniques [53, 55, 62] proposed for numerical data are not directly applicable to the qualitative-valued categorical data [15, 31]. Although some conventional encoding strategies, such as one-hot encoding, are feasible to convert categorical data into numerical data, information loss is often inevitable. Therefore, the development of categorical data clustering techniques, including clustering algorithms, distance measures, and distance metric learning approaches, continues to attract attention in data management and machine learning fields [19, 45, 54]. Categorical data clustering approaches can be roughly categorized into: 1) k-modes and its variants [5, 6] that partition data samples into clusters based on certain distance measures, and 2) cluster quality assessment-based algorithms [9, 20, 35] that adjust the sample-cluster affiliations based on certain cluster quality measures. Their performance is commonly dominated by the adopted metric space. Thus, the focus of current research has shifted from the design of clustering algorithms to proposing reasonable measures that suit categorical data clustering [4].

The Hamming distance and one-hot encoding are common measure and encoding strategy, respectively, and they both simply represent the distance between two categorical values based on whether the values are identical or not, which is far from informative. Later, distance measures have been proposed to finely indicate the differences at the value level by considering the intra-attribute value probabilities [42], the inter-attribute dependence [3, 24, 33], or both [26, 47]. Advanced encoding strategies that encode distances between data samples [39, 46], and couplings of attribute values [10, 28, 29] have also been proposed. Since they are clustering task independent, learning-based approaches have been further developed to optimize the representation and clustering jointly to circumvent suboptimal solutions. Typical learning-based works include the distance learning-based methods [25, 48] that learn the dissimilarity relationship of categorical values, and representation learning-based methods [56, 58, 60] that learn to project categorical attribute values into a higher-dimensional Euclidean distance space.

Current distance or representation learning approaches typically involve dimensional expansion of categorical attributes, which impedes efficient data management [14, 34]. Furthermore, when processing mixed datasets comprising both categorical and numerical attributes, these approaches inadvertently amplify the weight of categorical attributes. This introduces heterogeneity and thus compromises the reliability of cluster analysis results. To address this, some methods project categorical attribute values into one-dimensional Euclidean distance metric space based on their semantic order [32, 50] (e.g., an attribute "decision" with values {strong_accept, accept, ..., strong_reject}), aiming to align them with numerical attributes. However, this strategy is inapplicable to attributes lacking semantic order. Even when a semantic order exists, it is often subjectively assigned by the data collector. Consequently, these methods demonstrate significant clustering performance improvements only on datasets where the semantic order is highly relevant to the clustering task. This motivates a more general *hypothesis* regarding the concept of "order": *The implicit optimal order of categorical attribute values (even in the absence of semantic order) can substantially enhance clustering performance.*

To validate the hypothesis, a demonstration experiment (illustrated in Figure 1) is conducted on twelve datasets, with statistical characteristics detailed in Table 2 within Section 4.1. The standard k-modes [22] clustering algorithm is employed adopting the following three distance metrics: 1) the original Hamming distance, i.e., treat categorical attributes Without Orders (WO); 2) Semantic Order (SO) distance, i.e., treat ordinal attribute values like {accept, neutral, reject} with normalized

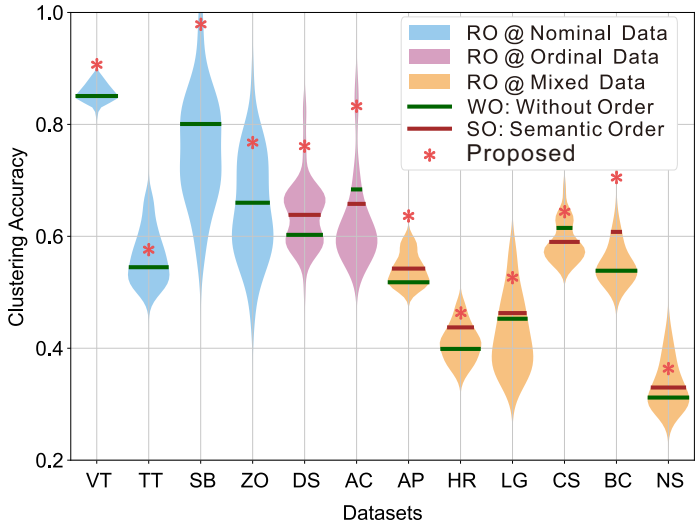

Fig. 1. Clustering accuracy of our method (indicated by red stars) and the variants of k-modes [22]: 1) WO: Without considering semantic Orders indicated by green bars, 2) SO: with Semantic Orders indicated by dark red bars, and 3) RO: with Random Orders generated 1000 times to characterize the clustering accuracy distribution on datasets comprising nominal attributes (blue region), ordinal attributes (purple region), and both (i.e., mixed data indicated by brown region).

order distances $dist(\text{accept}, \text{neutral}) = 0.5$, $dist(\text{neutral}, \text{reject}) = 0.5$, $dist(\text{accept}, \text{reject}) = 1$; and 3) Random Order (RO) distance, i.e., distances derived from randomly assigned orders to the values of each attribute, followed by normalization. Note that SO is inapplicable to the left four datasets, which consist solely of nominal attributes. Due to the random initialization of cluster modes, k-modes+WO and k-modes+SO are each executed 100 times, with average performance reported. k-modes+RO is implemented 1000 times to adequately characterize the distribution of clustering accuracy. Results indicate that although SO is utilized on the right 8 datasets, its performance (red bars) does not consistently surpass that of WO (green bars). This occurs because the semantic order, being subjectively defined by data collectors, may not align with the clustering task. Crucially, the 1000 trials of RO adequately traverse the state space of implicit value orders of the attributes. Thus, the hypothesis is strongly validated by the observation that the upper performance limits of RO are markedly higher than the performance of SO and WO. Therefore, the above evidence suggests that identifying the implicit optimal orders represents a critical factor for achieving accurate categorical data clustering.

This paper, therefore, proposes a new order learning-based clustering paradigm that dynamically estimates orders based on the current data partition to optimize the data partition. To make full use of the learned orders, a distance learning mechanism is also introduced to dynamically update the order distance measure for the sample-cluster affiliation. Since the order learning and the clustering task are connected to form an integrated optimization problem, the sub-optimal clustering solutions can be considerably circumvented. It turns out that the learned orders can significantly improve clustering accuracy and can also intuitively help understand the distribution and clustering effect of categorical data. The convergence of the proposed optimization algorithm and the metric properties of the order distance are rigorously guaranteed. Extensive experiments, including comparative studies, ablation studies, significance tests, case studies, and intuitive visualization, demonstrate

the superiority of the proposed paradigm. The main contributions of this work can be summarized into the following three points:

- A new insight that "obtaining the optimal orders is the decisive factor for accurately clustering categorical data" is proposed. Accordingly, a new clustering paradigm with order distance learning is presented.
- The proposed iterative learning mechanism more thoroughly optimizes the order, distance, and clusters by considering their connections. As a result, it features adaptability and is more robust to various clustering scenarios.
- Both the learning process and learned orders are highly interpretable, which helps understand complex categorical data. Moreover, the algorithm is efficient, strictly convergent, and can be easily adapted for mixed data clustering.

2 Related Work

This section overviews the related works in: 1) **data encoding**, 2) **representation learning**, and 3) **consideration of value order**, from the perspective of categorical data clustering.

The **data encoding** techniques convert categorical data into numerical values based on certain strategies and then cluster numerical encoding. As the conventional one-hot encoding and the Hamming distance cannot distinguish different degrees of dissimilarity [8, 11] of data samples, information loss is usually unavoidable. Some statistic-based distance measures [17, 36] consider the intra-attribute occurrence frequency of values to better distinguish the data samples. Some metrics [3, 24, 33] further define distance considering the relevance between attributes. Later, the measures [26, 27] and encoding strategies [28, 29, 39] that more comprehensively consider both the intra- and inter-attribute couplings reflected by the data statistics have also been presented.

Recently, more advanced **representation learning** methods have been proposed to learn categorical data encoding, facilitating better data representation in data analysis tasks. The work proposed in [60] projects dataset through different kernels for more informative coupling representation, and then learns the kernel parameters to obtain task-oriented representation. In contrast, [9] reveals the complex nested multi-granular cluster effect to obtain multi-granular distribution encoding representation of categorical data. Some approaches [30, 41, 61] have also been designed to simultaneously encode numerical and categorical attributes. However, their performance is highly sensitive to hyper-parameters, making optimal selection labor-intensive across diverse real-world applications. Although a parameter-free approach [25] has been proposed to learn sample-cluster distance based on cluster statistics, it does not form a valid distance metric for categorical data.

Some new advances proposed with the **consideration of value order** further distinguish categorical attributes into nominal and ordinal attributes in clustering tasks. The distance metrics [47, 51] define the distance between attribute values based on the data statistical information with considering the semantic order. Later, a representation learning approach was introduced by [49], which treats the possible values of an ordinal attribute as linearly arranged concepts, and learns the distances between adjacent concepts. Later, the works [48, 50] further unify the distances of nominal and ordinal attributes, and make them learnable with clustering. The most recent work [52, 56] attempts to convert the nominal attributes into ordinal ones through geometric projection, and then learn the latent distance spaces during clustering. The above recent advances achieve considerable clustering performance improvements and empirically prove the importance of value order in categorical data clustering. However, they all assume the availability of the semantic order of attribute values. Moreover, they directly adopt this order for clustering, and overlook that the semantic order may not be suitable under different clustering scenarios as shown in Figure 1.

3 Proposed Method

This section first formulates the problem, then proposes the order distance definition, order learning mechanism, and the joint learning of the order-guided distances and clusters with analysis.

3.1 Problem Formulation

The problem of clustering data with categorical attributes is formulated as follows. Given a dataset $X = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n\}$ with n samples (e.g., a collection of n clients of a bank). Each sample $\mathbf{x}_i$ can be denoted as an s-dimensional row vector $\mathbf{x}_i = [x_{i,1}, x_{i,2}, .., x_{i,s}]^\top$ represented by s attributes (e.g., occupation, priority, income, etc.) including $s^{\{c\}}$ categorical and $s^{\{u\}}$ numerical attributes with $s^{\{c\}} + s^{\{u\}} = s$. For simplicity and without loss of generality, assume that all categorical and numerical attributes come from the corresponding attribute sets $A^{\{c\}}$ and $A^{\{u\}}$, respectively. Each attribute can be denoted as a column vector $\mathbf{a}_r = [x_{1,r}, x_{2,r}, ..., x_{n,r}]$ composed of the description of all n samples, all taking values from a corresponding possible value set $V_r = \{v_{r,1}, v_{r,2}, ..., v_{r,l_r}\}$.

For a categorical attribute $\mathbf{a}_r \in A^{\{c\}}$, the possible value set is qualitative, e.g., {lawyer, doctor, ..., scientist} of an "occupation" attribute, while for a numerical attribute $\mathbf{a}_r \in A^{\{u\}}$, the value set is quantitative, e.g., {\$4228, \$11900, ..., \$2880} of an "income" attribute. We use l_r to indicate the total number of possible values of an attribute, and for a typical categorical attribute, l_r is usually finite and satisfies $l_r < n$. Note that the subscript sequential numbers of possible values only distinguish them, and do not indicate their order. We do not distinguish between nominal and ordinal attributes under categorical attributes in the whole Section 3, because their order will be learned uniformly by the proposed method.

Like numerical attributes with inherent value order, there is also an optimal order indicating an appropriate underlying distance metric of each categorical attribute $\mathbf{a}_r$. The difference is that there exists ambiguity in the relationship among categorical possible values, so their potential optimal order should be learned in a specific context, i.e., clustering here. We denote the optimal order of $\mathbf{a}_r$ as $O_r^* = \{o^*(v_{r,1}), o^*(v_{r,2}), ..., o^*(v_{r,l_r})\}$ where the superscript "$*$" indicates the optimum and the value of $o^*(v_{r,g})$ is an integer reflecting the unique ranking of $v_{r,g}$ among all its l_r possible values. For a numerical $\mathbf{a}_r \in A^{\{u\}}$, since the attribute values themselves already precisely reflect their orders in Euclidean distance space, we have $o(v_{r,g}) = v_{r,g}$ with $g = \{1, 2, ..., l_r\}$, and the Euclidean distance $\text{dist}(v_{r,g}, v_{r,h}) = |o(v_{r,g}) - o(v_{r,h})| = |v_{r,g} - v_{r,h}|$ can be viewed as reflected by the order of values. For a categorical $\mathbf{a}_r \in A^{\{c\}}$, by still taking the "occupation" attribute as an example, if its three possible values $v_{r,1}$ = lawyer, $v_{r,2}$ = doctor, and $v_{r,3}$ = scientist, ranking second, third, and first, respectively, then the corresponding order is $O_r = \{o(\text{lawyer}), o(\text{doctor}), o(\text{scientist})\} = \{2, 3, 1\}$, which reflects value relationship satisfying: $\text{dist}(v_{r,g}, v_{r,h}) \propto |o(v_{r,g}) - o(v_{r,h})|$.

It is noteworthy that fixing $\text{dist}(v_{r,g}, v_{r,h}) \equiv 1$ by the Hamming distance can limit the possibility of obtaining a context-aware distance metric, while most categorical data distance metrics based on statistics do not reveal a value order to ensure compatibility with the naturally ordered values of a numerical attribute under the Euclidean distance metric. This paper, therefore, aims to learn a set of optimal orders $O^* = \{O_1^*, O_2^*, ..., O_s^*\}$ corresponding to the $s^{\{c\}}$ attributes in $A^{\{c\}}$, so that the order-guided distance metric is compatible with numerical attributes and is adaptive to clustering tasks. For crisp clustering that partitions X into k non-overlapping sample sets $C = \{C_1, C_2, ..., C_k\}$, the objective with our order distance learning can be formalized as the problem of minimizing:

$$L(\mathbf{Q}, O) = \sum_{m=1}^{k} \sum_{i=1}^{n} q_{i,m} \cdot \Theta(\mathbf{x}_i, C_m; O) \tag{1}$$

where $\mathbf{Q}$ is an $n \times k$ matrix with its (i, m)-th entry $q_{i,m}$ indicating the affiliation between sample $\mathbf{x}_i$ and cluster C_m. Specifically, $q_{i,m} = 1$ indicates that $\mathbf{x}_i$ belongs to C_m, while $q_{i,m} = 0$ indicates that

Table 1. Frequently used symbols. Lowercase, uppercase, bold lowercase, and bold uppercase are uniformly used to indicate value, set, vector, and matrix, respectively.

Symbol	Explanation
X, $A^{\{c\}}$, and $A^{\{u\}}$	Dataset, categorical attribute set, and numerical attribute set
O and C	order set corresponding to categorical attributes and cluster set
$\mathbf{x}_i$ and $x_{i,r}$	i-th data object and r-th value of $\mathbf{x}_i$
$\mathbf{a}_r$, V_r, and $v_{r,g}$	r-th attribute, its value set, and g-th value of V_r
l_r	Total number of possible values in V_r
O_r	Order values (integers) of $\mathbf{a}_r$'s possible values V_r
$o(v_{r,g})$	Order value (integer) of possible value $v_{r,g}$
$\mathbf{Q}$ and C_m	Object-cluster affiliation matrix and m-th cluster
$q_{i,m}$	(i,m)-th entry of $\mathbf{Q}$ indicating affiliation of $\mathbf{x}_i$ to C_m
$\Theta(\mathbf{x}_i, C_m; O)$	Overall order distance between $\mathbf{x}_i$ and C_m
$\theta(x_{i,r}, \mathbf{p}_{m,r})$	Order distance between $\mathbf{x}_i$ and C_m reflected by $\mathbf{a}_r$
$\mathbf{D}$ and $\mathbf{d}_{i,r}$	Order distance matrix and its (i,r)-th entry
$d_{i,r,g}$	Order distance between $x_{i,r}$ and $v_{r,g}$, also the g-th value of $\mathbf{d}_{i,r}$
$\mathbf{P}$ and $\mathbf{p}_{m,r}$	Conditional probability matrix and its (m,r)-th entry
$p_{m,r,g}$	Occurrence probability of $v_{r,g}$ in C_m, also the g-th value of $\mathbf{p}_{m,r}$
$O_{r,m}$	Order values of $\mathbf{a}_r$'s possible values V_r obtained through C_m
$L_{m,r}$	A fraction of L jointly contributed by C_m and $\mathbf{a}_r$

$\mathbf{x}_i$ belongs to a cluster other than C_m, which can be written as:

$$q_{i,m} = \begin{cases} 1, & \text{if } m = \arg\min_y \Theta(\mathbf{x}_i, C_y; O) \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

with $\sum_{i=1}^{k} q_{i,m} = 1$. The distance $\Theta(\mathbf{x}_i, C_m; O)$ reflects the dissimilarity between $\mathbf{x}_i$ and C_m computed based on order relationship O. Since the Euclidean distance of numerical attributes $A^{\{u\}}$ is known, the problem to be solved by this paper can be specified as how to jointly learn O of categorical attributes $A^{\{c\}}$ and partition $\mathbf{Q}$ to minimize L. Frequently used symbols are sorted out in Table 1.

3.2 Order Distance

Order distance is the key component of the clustering objective in Eq. (1), as it acts to decide the sample-cluster affiliation for data partitioning, indicates whether the current order is conducive to approaching the minimization of the clustering objective, and provides a quantitative basis for order learning. Given the order O representing the attribute value relationship, the sample-cluster distance $\Theta(\mathbf{x}_i, C_m; O)$ can be expressed as:

$$\Theta(\mathbf{x}_i, C_m; O) = \frac{1}{s} \sum_{r=1}^{s} \theta(x_{i,r}, \mathbf{p}_{m,r}) \tag{3}$$

where $\theta(x_{i,r}, \mathbf{p}_{m,r})$ is the order distance between $\mathbf{x}_i$ and C_m reflected by the r-th attribute $\mathbf{a}_r$, indicating how dissimilar is the sample value $x_{i,r}$ to $\mathbf{a}_r$'s values in C_m. To sufficiently exploit the statistics of C_m, probability distribution $\mathbf{p}_{m,r}$ of $\mathbf{a}_r$'s possible values V_r within C_m is considered to derive $\theta(x_{i,r}, \mathbf{p}_{m,r})$:

$$\theta(x_{i,r}, \mathbf{p}_{m,r}) = \mathbf{d}_{i,r}^{\top} \mathbf{p}_{m,r} \tag{4}$$

where $\mathbf{p}_{m,r}$ is an l_r-dimensional vector $[p_{m,r,1}, p_{m,r,2}, ..., p_{m,r,l_r}]$ describing the probability distribution of V_r within cluster C_m, with its g-th value $p_{m,r,g}$ indicating the occurrence probability of $v_{r,g}$

in cluster C_m obtained by:

$$p_{m,r,g} = \frac{card(X_{r,g} \cap C_m)}{card(C_m)}. \tag{5}$$

Here, $X_{r,g} = \{\mathbf{x}_i | x_{i,r} = v_{r,g}\}$ is collection of all the samples with their r-th values equal to $v_{r,g}$ in X, and the cardinality function $card(\cdot)$ counts the number of elements in a set. $\mathbf{d}_{i,r}$ in Eq. (4) is the order differences between $x_{i,r}$ and all the values in V_r, which is also expressed as a vector $[d_{i,r,1}, d_{i,r,2}, ..., d_{i,r,l_r}]$. Its g-th value can be computed by:

$$d_{i,r,g} = \frac{|o(x_{i,r}) - o(v_{r,g})|}{l_r - 1} \tag{6}$$

where $l_r - 1$ normalizes the differences into $[0, 1]$ to ensure comparability across different attributes. Eq. (6) connects the value order to the objective function, and thus the order distance defined in Eq. (4) can quantify the appropriateness of the current order O_r of $\mathbf{a}_r$. That is, O_r corresponding to smaller $\theta(x_{i,r}, \mathbf{p}_{m,r})$ with $\mathbf{x}_i \in C_m$ is preferable as it indicates a more compact C_m and a higher contribution in minimizing L. Subsequently, Section 3.3 introduces the order learning based on order distance.

3.3 Order Learning

Given data partition $\mathbf{Q}$, the optimal order O^* can be expressed as:

$$O^* = \arg\min_{O} L(\mathbf{Q}, O). \tag{7}$$

A simple but laborious solution is to search different order combinations across all the attributes. However, there are a total of $\prod_{r=1}^{d} l_r!$ combinations, and each combination should be evaluated by considering all n samples through the clustering objective function, which makes the search almost infeasible for large-scale data. A relaxed efficient solution is to search the optimal order within clusters by assuming independence of attributes. Accordingly, optimal order of $\mathbf{a}_r$ w.r.t. C_m denoted as $O^*_{r,m} = \{o^*_m(v_{r,1}), o^*_m(v_{r,2}), ..., o^*_m(v_{r,l_r})\}$ can be obtained through within-cluster traversal search:

$$O^*_{r,m} = \arg\min_{O_r} L_{m,r} \tag{8}$$

where $L_{m,r}$ is an inner sum component of the objective L. More specifically, $L_{m,r}$ is jointly contributed by the r-th attribute and m-th cluster, which can be derived from Eqs. (1) and (3) as:

$$L_{m,r} = \sum_{\mathbf{x}_i \in C_m} \theta(x_{i,r}, \mathbf{p}_{m,r}). \tag{9}$$

However, the traversal search for the optimal order is still with an $O(\varsigma!)$ time complexity in the worst case where $\varsigma = \max(l_1, l_2, ..., l_d)$, which can still be laborious for attributes with a large number of possible values ς. Fortunately, such an optimal order searching problem is a simple special case of the well-known Quadratic Assignment Problem (QAP) [37, 43, 59], for which an efficient Optimal Linear Ordering (OLO) [1] algorithm with time complexity $O(\varsigma \log \varsigma)$ has been proposed with optimal solution guarantee. OLO solves a circuit routing cost optimization. It assumes that there are a set of ς pins with $\varsigma(\varsigma - 1)/2$ wire connections, and there are ς holes in a line with adjacent holes at unit distances apart to put the ς pins. If the number of wires connecting the pins is given, OLO can arrange the pins into holes to minimize the overall wire length to save cost. Intuitively, the solution follows a unimodal arrangement, where the pins that are more densely connected with the other pins (i.e., with more wire linkages) are placed in the middle holes. Corresponding to our problem, how to define the number of wires between the pins, i.e., the link density between possible values within the same cluster, is the prerequisite to implement OLO.

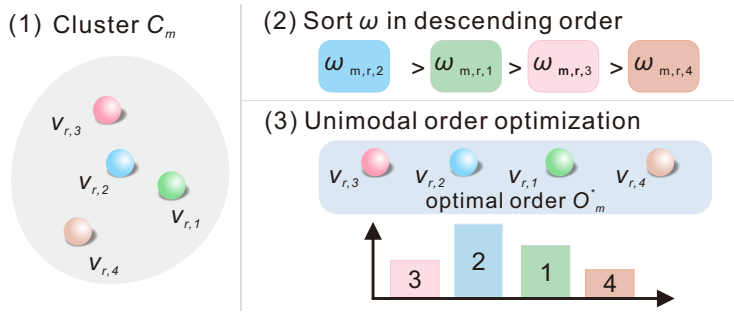

Fig. 2. Illustration of the order learning process. (1) Four values of $\mathbf{a}_r$ appear in C_m; (2) Their link density $\omega_{m,r,u}$ are quantified and ranked; (3) Order optimization by Eq. (13).

Using the occurrence probability $p_{m,r,g}$ of a possible value within a cluster C_m defined in Eq. (5) as the link density is a straightforward possible way. This is equivalent to quantifying the number of wire linkages by the occurrence frequency of each possible value, and the rationality lies in that the values with a larger proportion have a stronger dominance over the objective function. However, taking into account only the within-cluster probabilities of the values makes the OLO process relatively isolated, i.e., less connected with the clustering objective in the alternating optimization process, potentially hindering the convergence of the algorithm. Thus, contribution of a possible value $v_{r,g}$ to a cluster C_m in forming the objective function is also derived as:

$$L_{m,r,g} = \sum_{x_{i,r} \in \{X_{r,g} \cap C_m\}} \theta(x_{i,r}, \mathbf{p}_{m,r}), \tag{10}$$

which is denoted as the form of a component of $L_{m,r}$ satisfying:

$$L_{m,r} = \sum_{g=1}^{l_r} L_{m,r,g}. \tag{11}$$

By simultaneously considering $p_{m,r,g}$ and $L_{m,r,g}$, the link density of $v_{r,g}$ to the other possible values within C_m can be quantified as:

$$\omega_{m,r,g} = \frac{p_{m,r,g}}{L_{m,r,g}}, \tag{12}$$

and the rationality of introducing $L_{m,r,g}$ is analyzed as Remark 1.

Remark 1. ***Rationality of*** $L_{m,r,g}$ ***in quantifying*** $\omega_{m,r,g}$**.** *$L_{m,r,g}$ in Eq. (12) acts as a regularization term to reduce the complexity of the learning process. It is obtained based on the partition of the whole dataset and order distance learned in the previous iteration, and forms a balance with $p_{m,r,g}$ that only considers the value distribution within the current C_m. Intuitively, a smaller $L_{m,r,g}$ indicates that $v_{r,g}$ contributes less to $L_{m,r}$. This results in a larger $\omega_{m,r,g}$, emphasizing its importance in the OLO order optimization process, and further consolidating its role in minimizing the clustering objective function. This effectively circumvents optimizing L in the large order space and improves the stability of the learned order in adjacent iterations.*

With link density defined by Eq. (12), the unimodal order arrangement of OLO to minimize linkage cost can be expressed as:

$$o^*_m(v_{r,g}) = \lceil \frac{l_r}{2} \rceil - (-1)^{\eta+1} \left(\left\lfloor \frac{\eta}{2} \right\rfloor \right) \tag{13}$$

where $\eta = \text{rank}(\omega_{m,r,g})$ is the rank of $\omega_{m,r,g}$ in the descending order of $\{\omega_{m,r,1}, \omega_{m,r,2}, ..., \omega_{m,r,l_r}\}$, and $\lceil\cdot\rceil$ and $\lfloor\cdot\rfloor$ take the ceiling and floor of a value, respectively. Such an arrangement put the values with the highest link density to the center, and arrange the remaining values in descending order of their link density, alternating between the left and right sides of the value with the highest link density. The core idea of this process is to place the value with the highest link density in the middle of the order to effectively reduce its overall order distance with other values, thereby minimizing the linkage cost. For more OLO derivation and its $O(\varsigma \log \varsigma)$ time complexity analysis, please refer to [1].

A possible alternative solution is to obtain differentiated orders across different cluster groups. However, each order subtly reflects the relationship between the values of the corresponding attribute. If the order is differentiated w.r.t. different cluster groups, the overly distorted distance metric space derived from the inconsistent orders may make the algorithm prone to falling into local optima or lead to learning divergence. By searching for the optimal ranking in each cluster (as shown in Figure 2) and then performing a linear combination based on the cluster size, the overall consensus ranking of each possible value $v_{r,g}$ can be obtained by:

$$o(v_{r,g}) = \sum_{m=1}^{k} \left(\frac{card(C_m)}{n} \cdot o^*_m(v_{r,g}) \right). \tag{14}$$

Since the value of $o(v_{r,g})$ is not necessarily an integer, the obtained rankings $\{o(v_{r,1}), o(v_{r,2}), ..., o(v_{r,l_r})\}$ of an attribute is sorted in ascending order to obtain the final ranking of V_r, and the corresponding integer order $\tilde{O}_r = \{\tilde{o}(v_{r,1}), \tilde{o}(v_{r,2}), ..., \tilde{o}(v_{r,l_r})\}$ is obtained. Consequently, all orders of $A^{\{c\}}$ are denoted as $\tilde{O} = \{\tilde{O}_1, \tilde{O}_2, ..., \tilde{O}_s\}$. The optimal order learning described by Eqs. (8) - (14) considers both the within-cluster statistics $p_{m,r,g}$ and the optimization of the overall objective function $L_{m,r,g}$ as discussed in Remark 1, ensuring the consistency of the objectives of order learning and clustering. $\tilde{O}$ is utilized to form distance metric by Eqs. (3) and (4) for clustering in the alternating optimization process presented in Section 3.4.

3.4 Joint Order and Cluster Learning

By integrating the order learning with clustering, the objective can be rewritten in a more detailed form based on Eqs. (1) - (4) as:

$$L(\mathbf{Q}, O) = \sum_{m=1}^{k} \sum_{i=1}^{n} q_{i,m} \cdot \left(\frac{1}{s^{\{c\}}} \sum_{\mathbf{a}_r \in \mathbf{A}^{\{c\}}} \mathbf{d}_{i,r}^{\top} \mathbf{p}_{m,r} \right) \tag{15}$$

where the order distance $\mathbf{D} = \{\mathbf{d}_{i,r} | i \in \{1, 2, ..., n\}, \mathbf{a}_r \in \mathbf{A}^{\{c\}}\}$ is dependent to the order O according to Eqs. (6) and (13), and the probability distribution $\mathbf{P} = \{\mathbf{p}_{m,r} | m = \{1, 2..., k\}, r \in \{1, 2, ..., s^{\{c\}}\}\}$ is dependent to the partition $\mathbf{Q}$ according to Eqs. (2) and (5). Typically, $\mathbf{Q}$ and O are iteratively computed to minimize L. Specifically, given $\hat{O}$, $\mathbf{D}$ is updated accordingly, then a new $\mathbf{Q}$ is computed by:

$$q_{i,m} = \begin{cases} 1, & \text{if } m = \arg\min_y \frac{1}{s^{\{c\}}} \sum_{r=1}^{s^{\{c\}}} \mathbf{d}_{i,r}^{\top} \mathbf{p}_{y,r} \\ 0, & \text{otherwise} \end{cases} \tag{16}$$

with $i = \{1, 2, ..., n\}$ and $m = \{1, 2, ..., k\}$. Eq. (16) is a detailed version of Eq. (2) that adopts the order distance defined by Eqs. (3) and (4). Then, $\mathbf{P}$ is updated according to the new $\mathbf{Q}$. When the iterative updating of $\mathbf{Q}$ and $\mathbf{P}$ converges, we obtain O according to Eqs. (8) - (14), and update $\mathbf{D}$ accordingly. To ensure effective initialization of the optimization process, k-modes is implemented once and the clustering results are utilized as the starting point, which has been widely adopted in clustering optimization. In summary, L is minimized by iteratively solving the following two

Algorithm 1 OCL: Order and Cluster Learning Algorithm

Input: Dataset X, number of sought clusters k
Output: Partition $\mathbf{Q}$, order O

1: Initialization: Set outer loop counter and convergence mark by $E \leftarrow 0$ and $Conv\_O \leftarrow False$, respectively; Randomly partition X into k clusters C, obtain $\mathbf{Q}^{\{I\}}$ and $\mathbf{P}^{\{I\}}$ accordingly; Randomly initialize $O^{\{E\}}$, update $\mathbf{D}^{\{E\}}$ accordingly;
2: **while** $Conv\_O = False$ **do**
3: $E \leftarrow E + 1$; Compute $O^{\{E\}}$ by Eqs. (8) - (14); Update $\mathbf{D}^{\{E\}}$ accordingly; Set inner loop counter and convergence mark by $I \leftarrow 0$ and $Conv\_I \leftarrow False$, respectively;
4: **while** $Conv\_I = False$ **do**
5: $I \leftarrow I + 1$; Compute $\mathbf{Q}^{\{I\}}$ by Eq. (16); Update $\mathbf{P}^{\{I\}}$ accordingly; Obtain $L(\mathbf{Q}^{\{I\}}, O^{\{E\}})$;
6: **if** $L(\mathbf{Q}^{\{I\}}, O^{\{E\}}) \geq L(\mathbf{Q}^{\{I-1\}}, O^{\{E\}})$ **then**
7: $Conv\_I \leftarrow True$; $\mathbf{Q}^{\{E\}} = \mathbf{Q}^{\{I\}}$;
8: **end if**
9: **end while**
10: **if** $L(\mathbf{Q}^{\{E\}}, O^{\{E\}}) \geq L(\mathbf{Q}^{\{E-1\}}, O^{\{E-1\}})$ **then**
11: $Conv\_O \leftarrow True$;
12: **end if**
13: **end while**

sub-problems: 1) fix $\hat{\mathbf{Q}}$ and $\hat{\mathbf{P}}$, solve the minimization problem $L(\hat{\mathbf{Q}}, O)$ by searching O and updating $\mathbf{D}$ accordingly, and 2) fix $\hat{O}$ and $\hat{\mathbf{D}}$, solve the minimization problem $L(\mathbf{Q}, \hat{O})$ by iteratively computing $\mathbf{Q}$, and updating $\mathbf{P}$ according to $\mathbf{Q}$, until convergence. The whole algorithm named Order and Cluster Learning (OCL) is summarized as Algorithm 1.

Since the learned order distance of each attribute resides in a one-dimensional metric space compatible with the attribute-level Euclidean distance, it can be seamlessly integrated with numerical attributes for homogeneous computation. This enables straightforward extension of OCL to mixed data clustering. Specifically, the order distance metric of the categorical attributes is first learned using OCL, and then combined with numerical attributes using a conventional clustering algorithm such as k-means [7]. Crucially, this decouples order distance learning of categorical attributes from the deterministic Euclidean distance of numerical attributes. Furthermore, the compatibility of learned ordinal metric and the Euclidean distance space considerably relieves information loss during joint clustering, enhancing performance on mixed datasets.

3.5 Complexity and Convergence Analysis

The time and space complexity and convergence of OCL are analyzed below.

Theorem 1. *The time complexity of OCL is* $O(EInks\varsigma + Enks\varsigma \log \varsigma)$.

Proof. Assume it involves I inner iterations (lines 4 - 9 in Algorithm 1) to compute $\mathbf{Q}$ and $\mathbf{P}$, and E outer iterations (lines 2 - 13 in Algorithm 1) to search O and update $\mathbf{D}$. For worst-case analysis, $\varsigma = \max(l_1, l_2, ..., l_s)$ is adopted to indicate the maximum number of possible values of a dataset.

For each of the I inner iterations, $\mathbf{P}$ is obtained upon the n data samples with complexity $O(nks\varsigma)$, and $\mathbf{D}$ is obtained upon the ς values of s attributes on n samples with complexity $O(nd\varsigma)$. Then the n samples are partitioned to k clusters with complexity $O(nks\varsigma)$. For I iterations in total, the complexity is $O(Inks\varsigma)$.

Table 2. Detailed statistics of the 20 datasets. "$s^{\{n\}}$", "$s^{\{o\}}$", and "$s^{\{u\}}$" indicate the numbers of nominal, ordinal, and numerical attributes, respectively. "l_r" indicates the number of possible values of categorical (ordinal + nominal) attributes. ϑ, ς, and ε indicate the averaged, maximum, and minimum numbers of possible values for the categorical attributes of each dataset, respectively. "n" is the number of samples. k^* is the true number of clusters.

No.	Datasets	Abbrev.	$s^{\{n\}}$	$s^{\{o\}}$	$s^{\{u\}}$	l_r	ϑ	ς	ε	n	k^*
1	Soybeans	SB	35	0	0	7 2 3 3 2 4 4 2 2 3 2 2 4 4 2 2 2 2 2 2 2	2.76	7	2	47	4
2	Nursery School	NS	1	7	0	3 5 4 4 3 2 3 3	3.38	5	2	12960	4
3	Amphibians	AP	6	6	2	8 5 8 7 8 3 5 6 6 6 3 2	5.58	8	2	189	2
4	Inflammations Diagnosis	DS	0	5	1	2 2 2 2 2	2.00	2	2	120	2
5	Caesarian Section	CS	1	3	0	4 3 3 2	3.00	4	2	80	2
6	Hayes-Roth	HR	2	2	0	3 4 4 4	3.75	4	3	132	3
7	Zoo	ZO	15	0	0	2 2 2 2 2 2 2 2 2 2 2 2 2 6 2 2 2	2.25	6	2	101	7
8	Breast Cancer	BC	5	4	0	3 3 2 6 2 6 11 7 3	4.78	11	2	286	2
9	Lym	LG	15	3	0	3 4 8 4 2 2 2 2 2 2 2 3 4 4 8 3 2 2	3.28	8	2	148	4
10	Tic-Tae-Toc	TT	9	0	0	3 3 3 3 3 3 3 3 3	3.00	3	3	958	2
11	Australia Credit	AC	0	8	6	2 2 2 2 3 14 8 3	4.50	14	2	690	2
12	Congressional Voting	VT	16	0	0	3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3	3.00	3	3	435	2
13	Connect-4	CT4	42	0	0	-	3.00	3	3	67557	3
14	Obesity Levels	OB	6	2	0	2 2 2 4 2 2 4 5	2.88	5	2	2111	7
15	Auction Verifications	AV	0	6	0	3 4 2 2 2 6 5	3.88	6	2	2043	2
16	Bank Marking	BM	6	3	0	12 3 4 2 2 2 3 12 4	4.89	12	2	45211	2
17	Chess	CC	36	0	0	-	2.02	3	2	3195	2
18	Coupon	CR	7	4	0	3 4 3 5 5 2 2 5 6 25 9	6.27	25	2	12684	2
19	Heart Failure	HF	6	0	8	2 2 2 2 2 47 208 17 176 40 27 148	56.08	208	2	299	2
20	Multi-attribute-category AC	MA	0	8	6	2 2 2 2 3 14 8 3 350 215 132 23 171 240	83.36	350	2	690	2

For each of the E outer iterations, a new order of ς possible values of each attribute within each cluster is searched by OLO algorithm described by Eqs. (8) - (14), and the complexity for k clusters and s attributes is thus $O(nks\varsigma \log \varsigma)$.

For E outer iterations with considering the I inner iterations, the overall time complexity of OCL is $O(EInks\varsigma + Enks\varsigma \log \varsigma)$. □

Theorem 2. *The space complexity of OCL is* $O(ns + nk + ns\varsigma + ks\varsigma)$.

Proof. OCL requires $O(ns)$ space for the dataset X, $O(nk)$ for the cluster assignment matrix $\mathbf{Q}$, $O(ns\varsigma)$ for the order distance matrix $\mathbf{D}$, and $O(ks\varsigma)$ space each for the conditional probability matrix $\mathbf{P}$ and the link density. □

Remark 2. ***Scalability of OCL.*** *The essential characteristic of categorical data is that the number of possible values ς for an attribute is finite satisfying $\varsigma < n$ for real benchmark datasets as shown in Table 2. Therefore, the time and space complexity of OCL scales linearly with n, k, and s, and are also relatively insensitive to ς, indicating high scalability of the proposed OCL algorithm.*

Theorem 3. *OCL algorithm converges to a local minimum in a finite number of iterations.*

Proof. We prove the convergence of the inner and outer loop, i.e., lines 4 - 9 and lines 2 - 13 of Algorithm 1, respectively.

Lemma 1. *The inner loop converges to a local minimum of $L(\mathbf{Q}, \hat{O})$ in a finite number of iterations with fixed $\hat{O}$. Note that the number of possible partitions $\mathbf{Q}$ is finite for a dataset with a finite number of objects n. We then illustrate that each possible partition $\mathbf{Q}$ occurs at most once by OCL. Assume that two optimizers satisfy $\mathbf{Q}^{\{I_1\}} = \mathbf{Q}^{\{I_2\}}$, where $I_1 \neq I_2$. Then it is clear that $L(\mathbf{Q}^{\{I_1\}}, O^{\{E\}}) = L(\mathbf{Q}^{\{I_2\}}, O^{\{E\}})$. However, the value sequence of the objective function $\{L(\mathbf{Q}^{\{1\}}, O^{\{E\}}), L(\mathbf{Q}^{\{2\}}, O^{\{E\}}), ..., L(\mathbf{Q}^{\{I\}}, O^{\{E\}})\}$ generated by OCL is strictly decreasing. Hence, the result follows.*

Table 3. Characteristics of compared clustering approaches in terms of the utilization/consideration of Hamming Distance (HD), Statistical Information (SI), Distance Metric Learning (DML), Semantic Order (SO), and Order Learning (OL).

Type	Counterpart	Year	HD	SI	DML	SO	OL
Traditional	KMD [22]	1998	✓				
	LSM [36]	1998		✓		✓	
	CBDM [24]	2012		✓			
Statistical	JDM [26]	2016		✓			
	UDMC [47]	2020		✓		✓	
Learning based (SOTA)	DLC [49]	2020		✓	✓	✓	
	HDC [50]	2022		✓	✓	✓	
	ADC [48]	2023		✓	✓	✓	
	MCDC [9]	2024		✓	✓		
	AMPHM [57]	2025		✓	✓		
	HARR [56]	2025		✓	✓	✓	
Ours	OCL	–		✓	✓		✓

Lemma 2. *The outer loop converges to a local minimum of $L(\mathbf{Q}, O)$ in a finite number of iterations. We note that the status of orders O is in a discrete space and thus the number of possible orders is finite. The number of possible partitions $\mathbf{Q}$ is also finite. Therefore, the number of possible combinations of $\mathbf{Q}$ and O is finite. We then prove that each possible combination of $\mathbf{Q}$ and O appears at most once by OCL. We note that given $\mathbf{Q}^{\{E_1\}} = \mathbf{Q}^{\{E_2\}}$, where $E_1 \neq E_2$, the corresponding $O^{\{E_1+1\}}$ and $O^{\{E_2+1\}}$ can be computed accordingly, satisfying $O^{\{E_1+1\}} = O^{\{E_2+1\}}$. Then we can obtain their corresponding partitions $\mathbf{Q}^{\{E_1+1\}}$ and $\mathbf{Q}^{\{E_2+1\}}$ at the convergence of the next inner loop, satisfying $\mathbf{Q}^{\{E_1+1\}} = \mathbf{Q}^{\{E_2+1\}}$. It is clear that $L(\mathbf{Q}^{\{E_1+1\}}, O^{\{E_1+1\}}) = L(\mathbf{Q}^{\{E_2+1\}}, O^{\{E_2+1\}})$. However, the sequence $\{L(\mathbf{Q}^{\{1\}}, O^{\{1\}}), L(\mathbf{Q}^{\{2\}}, O^{\{2\}}), ..., L(\mathbf{Q}^{\{E\}}, O^{\{E\}})\}$ is monotonically decreasing. Hence, the result follows.*

Since both the inner loop and outer loop of Algorithm 1 converge to a local minimum, convergence of OCL can be confirmed. □

4 Experiments

A series of experiments have been conducted. Experimental settings and the obtained results are introduced as follows.

4.1 Experimental Settings

Six experiments have been conducted: **(1) Clustering performance evaluation with significance tests (Section 4.2):** OCL versus conventional/SOTA methods across diverse datasets, with statistical significance analysis; **(2) Ablation studies (Section 4.3):** Evaluating OCL's core components and learned order impact through algorithm- and data-aspect ablation. **(3) Convergence and efficiency evaluation (Section 4.4):** Objective function trajectory recording for convergence analysis and efficiency assessed under scaling sizes of samples n and attributes s, and number of clusters k. **(4) Visualization of clustering effect (Section 4.5):** t-SNE embeddings generated via OCL's order distance versus baselines for visual comparison. **(5) Clustering performance on mixed datasets (Section 4.6):** OCL-enhanced methods evaluated on numerical-categorical mixed data, demonstrating effectiveness of the orders learned by OCL. **(6) Case study on Hayes-Roth (HR) dataset (Section 4.7):** Validation of meaningfulness and human cognition-aligned semantics of the learned orders through analysis and intuitive demonstration of the clustering results on a

Table 4. Clustering performance evaluated by CA (with range $[0, 1]$). "$\overline{AR}$" row reports the average performance rankings.

Datasets	KMD	LSM	JDM	CBDM	UDMC	DLC	HDC	ADC	MCDC	AMPHM	HARR	OCL
SB	0.8468±0.15	0.8106±0.16	0.7894±0.14	0.7957±0.15	0.7915±0.18	0.8000±0.14	0.7915±0.15	0.7660±0.13	0.5851±0.12	0.7021±0.00	0.9234±0.12	**0.9830±0.05**
NS	0.3208±0.05	0.2975±0.04	0.2980±0.04	–	0.3074±0.05	0.3386±0.07	0.3208±0.05	0.3208±0.05	0.3370±0.01	–	0.3113±0.05	**0.3573±0.05**
AP	0.5392±0.03	0.5656±0.04	0.6159±0.04	0.5926±0.03	0.5698±0.03	0.6032±0.00	0.5608±0.04	0.5529±0.04	0.5566±0.03	0.5026±0.00	0.5603±0.03	**0.6222±0.00**
DS	0.6525±0.08	0.6525±0.08	0.7208±0.12	0.7208±0.12	0.6867±0.12	0.6650±0.12	0.6867±0.12	0.6950±0.12	0.7242±0.08	0.6583±0.00	0.6175±0.04	**0.7458±0.08**
CS	0.5462±0.03	0.5488±0.06	0.5788±0.07	0.5875±0.04	0.5825±0.05	**0.6350±0.03**	0.5837±0.05	0.5525±0.04	0.5863±0.04	0.5000±0.00	0.6075±0.02	0.6313±0.04
HR	0.3758±0.02	0.3886±0.03	0.3917±0.02	0.3924±0.04	0.3773±0.02	0.3697±0.04	0.4000±0.03	0.3758±0.02	0.3939±0.03	0.4167±0.00	0.3333±0.00	**0.4326±0.02**
ZO	0.6624±0.12	0.6733±0.11	0.6743±0.11	0.6752±0.10	0.6693±0.10	0.6802±0.08	0.6733±0.10	0.6743±0.11	0.6228±0.07	0.6832±0.00	0.7059±0.05	**0.7792±0.10**
BC	0.5472±0.04	0.5594±0.07	0.6392±0.09	0.6346±0.09	0.5413±0.06	0.6140±0.09	0.6052±0.10	0.6185±0.10	0.5804±0.08	0.6399±0.00	0.5266±0.06	**0.6650±0.05**
LG	0.4277±0.06	0.4446±0.04	0.4547±0.06	0.4905±0.06	0.4466±0.05	0.5095±0.08	0.4959±0.05	0.5101±0.04	0.4743±0.04	0.5308±0.00	0.4959±0.05	**0.5426±0.02**
TT	0.5569±0.05	0.5593±0.04	0.5563±0.04	0.5408±0.03	0.5386±0.03	0.5637±0.03	0.5465±0.04	0.5471±0.04	**0.6078±0.03**	0.5021±0.00	0.5640±0.04	0.5785±0.03
AC	0.7281±0.12	0.7783±0.08	0.6335±0.12	0.6354±0.12	0.7683±0.08	0.7488±0.14	0.6762±0.14	0.6293±0.14	0.7717±0.10	0.5565±0.00	0.7703±0.08	**0.8206±0.08**
VT	0.8625±0.01	0.8699±0.00	0.8699±0.00	0.8761±0.00	0.8667±0.00	0.8531±0.09	0.8736±0.00	0.8713±0.00	0.8667±0.00	0.6092±0.00	0.8736±0.00	**0.8943±0.00**
CT4	0.4000±0.02	0.3609±0.01	0.3696±0.02	0.3651±0.02	0.3819±0.03	0.3618±0.01	0.3924±0.03	0.3879±0.03	0.3914±0.03	0.3514±0.01	0.3824±0.00	**0.4121±0.02**
OB	0.3715±0.03	0.3697±0.02	0.3594±0.03	0.3673±0.03	0.3720±0.02	0.3724±0.04	0.3797±0.02	0.3617±0.04	0.3765±0.03	0.3717±0.02	0.3512±0.00	**0.3935±0.01**
AV	0.6251±0.08	0.6035±0.10	0.6291±0.12	**0.6695±0.10**	0.6065±0.10	0.6150±0.06	0.6441±0.14	0.6364±0.10	0.6325±0.12	0.6114±0.10	0.6125±0.00	0.6637±0.07
BM	0.5730±0.06	0.5792±0.04	0.5959±0.07	0.6011±0.07	0.5709±0.03	0.5944±0.05	0.6064±0.12	0.5457±0.03	0.5543±0.04	0.5573±0.02	0.5039±0.00	**0.6127±0.00**
CC	0.5562±0.04	0.5572±0.03	0.5632±0.03	0.5433±0.03	0.5587±0.02	0.5342±0.04	0.5248±0.02	0.5438±0.03	0.5395±0.04	0.5668±0.03	0.5012±0.00	**0.5726±0.05**
CR	0.5209±0.02	0.5247±0.02	0.5185±0.02	0.5297±0.02	0.5245±0.01	0.5488±0.03	0.5541±0.00	0.5466±0.02	0.5504±0.02	0.5246±0.02	0.5510±0.00	**0.5590±0.02**
HF	0.5482±0.05	0.5301±0.06	0.5375±0.05	0.5284±0.07	0.5385±0.10	0.5478±0.08	0.5465±0.04	0.5431±0.11	0.5411±0.08	0.5488±0.00	–	**0.5652±0.03**
MA	0.6961±0.09	0.7097±0.03	0.6358±0.12	0.7164±0.07	0.7164±0.03	0.6654±0.10	0.7174±0.06	0.7154± 0.07	0.6649±0.09	0.5594±0.00	-	**0.7333±0.06**
$\overline{AR}$	7.80	7.78	7.12	6.10	7.72	5.95	5.33	7.05	6.33	8.07	7.60	1.15

real dataset. All experiments are coded with MATLAB R2022a and implemented on a workstation with Intel i7-14650 CPU @ 2.20 GHz, 16GB RAM.

20 benchmark datasets in different fields from the UCI machine learning repository [12] have been selected to ensure a comprehensive evaluation. Their detailed statistical information is summarized in Table 2. All datasets are pre-processed by removing samples with missing values. In data ablation studies, we adopted a conservative nominal and ordinal attributes distinguishing protocol: attributes are defaulted to nominal unless exhibiting explicit semantic order (e.g., attribute "Satisfaction" with values {very_satisfied, satisfied, average, ...}). This mitigates subjective bias and crucially prevents distance structure distortion, which is particularly vital for methods that exploit semantic orders, such as DLC, ADC, and HARR. To ensure a fair comparison with existing methods, in all experiments, k is set to the true number of clusters k^* given by the dataset, because automatic determination of the number of clusters remains an open and challenging problem, especially for categorical and mixed data where basic distance metric is still under exploration. 14 attributes of the SB dataset are omitted since each of them has only one possible value, which is meaningless for clustering. For brevity, possible values for the CT4 and CC datasets, which have too many attributes, are not listed in Table 2. Since we focus on categorical data clustering, the numerical attributes of the datasets AP, DS, and AC are omitted, except for the performance evaluation on mixed data in Section 4.6. The MA dataset is derived by treating the integer-valued attributes in the AC dataset as possible values, and it is used together with the HF dataset to evaluate the OCL performance for attributes with numerous possible values.

11 counterparts sorted out in Table 3 have been selected for the comparative study. The counterparts include traditional K-MoDes (KMD) algorithm [22], popular entropy-based metric LSM [36], and context-based distance measurement CBDM [23], statistical-based Jia's Distance Metric (JDM) [26] and Unified Distance Metric based Clustering (UDMC) [47], state-of-the-art HD based Clustering (HDC) [50], Heterogeneous Attribute Reconstruction and Representation based clustering (HARR) [56], Distance Learning based Clustering (DLC) algorithm [49], graph metric based clustering algorithm ADC [48], Multi-granular-guided Categorical Data Clustering (MCDC) algorithm [9], and Adaptive Micro Partition and Hierarchical Merging (AMPHM) clustering algorithm [57]. Among them, LSM and CBDM are traditional distance measures for categorical data, while JDM and UDMC are statistical-based measures. All four are adopted by KMD to perform clustering. DLC, HDC, ADC, MCDC, AMPHM, and HARR are state-of-the-art categorical data

Table 5. Clustering performance evaluated by ARI (with range $[-1, 1]$). "$\overline{AR}$" row reports the average performance rankings.

Datasets	KMD	LSM	JDM	CBDM	UDMC	DLC	HDC	ADC	MCDC	AMPHM	HARR	OCL
SB	0.7638±0.18	0.7555±0.19	0.7232±0.19	0.7407±0.19	0.7307±0.22	0.7148±0.20	0.7350±0.19	0.6968±0.17	0.2736±0.18	0.5949±0.00	0.8650±0.22	**0.9620±0.12**
NS	0.0376±0.03	0.0341±0.02	0.0283±0.02	–	0.0393±0.03	0.0779±0.08	0.0376±0.03	0.0376±0.03	0.0337±0.02	–	0.0309±0.03	**0.1015±0.06**
AP	0.0022±0.01	0.0171±0.02	**0.0530±0.03**	0.0322±0.02	0.0178±0.02	0.0355±0.00	0.0150±0.02	0.0066±0.02	0.0005±0.02	-0.0055±0.00	0.0076±0.02	0.0524±0.00
DS	0.1077±0.13	0.1077±0.13	0.2347±0.20	0.2347±0.20	0.1844±0.20	0.1556±0.19	0.1844±0.20	0.1921±0.19	0.2108±0.14	0.0923±0.00	0.0512±0.05	**0.2590±0.16**
CS	-0.0023±0.01	0.0088±0.04	0.0295±0.04	0.0250±0.03	0.0248±0.04	**0.0647±0.03**	0.0234±0.03	0.0040±0.02	0.0187±0.03	-0.0142±0.00	0.0342±0.02	0.0627±0.03
HR	-0.0081±0.01	-0.0008±0.01	-0.0013±0.01	-0.0010±0.01	-0.0046±0.01	-0.0033±0.01	0.0038±0.02	-0.0064±0.01	-0.0010±0.04	0.0094±0.00	-0.0149±0.00	**0.0368±0.01**
ZO	0.5787±0.17	0.6085±0.17	0.5996±0.17	0.6091±0.15	0.6063±0.16	0.6094±0.12	0.6138±0.15	0.6143±0.15	0.4760±0.16	0.7515±0.00	0.6442±0.07	**0.7536±0.11**
BC	0.0046±0.02	0.0240±0.05	**0.0811±0.07**	0.0777±0.07	0.0128±0.04	0.0523±0.05	0.0605±0.08	0.0692±0.08	0.0084±0.05	0.0393±0.00	-0.0033±0.00	0.0799±0.03
LG	0.0843±0.05	0.0951±0.03	0.1027±0.06	0.1432±0.07	0.1018±0.04	0.1544±0.09	0.1095±0.06	0.1262±0.05	-0.0012±0.02	0.1293±0.00	**0.1715±0.04**	0.1552±0.03
TT	0.0170±0.03	0.0184±0.02	0.0181±0.02	0.0095±0.01	0.0091±0.01	0.0156±0.01	0.0120±0.02	0.0124±0.02	0.0051±0.03	-0.0022±0.00	0.0198±0.02	**0.0226±0.02**
AC	0.2596±0.18	0.3336±0.12	0.1210±0.15	0.1199±0.13	0.3092±0.11	0.3184±0.22	0.1913±0.18	0.1349±0.18	0.3263±0.19	0.0011±0.00	0.3107±0.11	**0.4313±0.16**
VT	0.5247±0.02	0.5463±0.01	0.5463±0.01	0.5648±0.01	0.5368±0.01	0.5200±0.18	0.5572±0.00	0.5503±0.00	0.5367±0.00	0.0013±0.00	0.5572±0.00	**0.6207±0.00**
CT4	-0.0027±0.00	0.0010±0.00	0.0001±0.00	0.0004±0.00	0.0013±0.00	0.0012±0.00	0.0006±0.00	-0.0000±0.00	-0.0005±0.00	0.0002±0.00	0.0005±0.00	**0.0150±0.01**
OB	0.1527±0.03	0.1588±0.02	0.1347±0.04	0.1522±0.04	0.1580±0.02	0.1548±0.04	**0.1727±0.02**	0.1417±0.05	0.1472±0.04	0.1527±0.04	0.1672±0.00	0.1696±0.01
AV	0.0073±0.02	0.0111±0.02	0.0136±0.03	0.0365±0.02	0.0123±0.03	0.0262±0.03	0.0214±0.03	0.0199±0.02	0.0173±0.02	0.0209±0.02	0.0166±0.00	**0.0393±0.02**
BM	0.0146±0.03	0.0107±0.02	0.0183±0.04	0.0054±0.05	0.0074±0.02	0.0178±0.02	0.0186±0.01	-0.0037±0.02	0.0035±0.02	-0.0007±0.02	0.0072±0.00	**0.0233±0.00**
CC	0.0165±0.01	0.0166±0.01	0.0189±0.02	0.0103±0.01	0.0150±0.01	0.0099±0.02	0.0019±0.01	0.0108±0.01	0.0113±0.02	0.0177±0.01	0.0106±0.00	**0.0285±0.02**
CR	0.0021±0.00	0.0033±0.00	0.0024±0.00	0.0039±0.00	0.0030±0.00	0.0120±0.01	0.0114±0.00	0.0101±0.01	0.0106±0.01	0.0096±0.00	0.0122±0.00	**0.0145±0.01**
HF	0.0009±0.01	-0.0025±0.01	-0.0018±0.01	-0.0015±0.01	0.0024±0.02	0.004±0.01	-0.0029±0.01	0.0005±0.01	-0.0021±0.03	0.0063±0.00	–	**0.0072±0.01**
MA	0.1773±0.11	0.2143±0.14	0.1140±0.13	0.2048±0.13	0.2047±0.14	0.1586±0.07	0.1983±0.08	0.2035±0.08	0.1259±0.12	0.0120±0.00	–	**0.2165±0.06**
$\overline{AR}$	8.50	6.40	6.85	6.28	6.78	5.40	5.85	7.40	8.72	8.20	6.38	1.25

Table 6. Clustering performance evaluated by NMI (with range [0,1]). "$\overline{AR}$" row reports the average performance rankings.

Datasets	KMD	LSM	JDM	CBDM	UDMC	DLC	HDC	ADC	MCDC	AMPHM	HARR	OCL
SB	0.8460±0.12	0.8418±0.13	0.8268±0.13	0.8393±0.12	0.8456±0.12	0.8298±0.12	0.8513±0.11	0.8317±0.10	0.5344±0.17	0.7892±0.00	0.9157±0.14	**0.9757±0.08**
NS	0.0458±0.03	0.0441±0.02	0.0412±0.02	–	0.0492±0.03	0.1348±0.10	0.0458±0.03	0.0458±0.03	0.0504±0.03	–	0.0531±0.05	**0.1492±0.10**
AP	0.0147±0.01	0.0321±0.03	0.0746±0.03	0.0469±0.03	0.0276±0.02	0.0649±0.00	0.0384±0.04	0.0131±0.01	0.0194±0.02	0.0004±0.00	0.0185±0.03	**0.0955±0.01**
DS	0.1352±0.14	0.1352±0.14	0.2552±0.21	0.2552±0.21	0.2034±0.20	0.1682±0.21	0.2034±0.20	0.2082±0.20	**0.3083±0.13**	0.0646±0.00	0.0623±0.04	0.2732±0.18
CS	0.0157±0.03	0.0221±0.04	0.0426±0.05	0.0353±0.03	0.0413±0.05	**0.0840±0.03**	0.0346±0.03	0.0155±0.02	0.0310±0.03	0.0034±0.00	0.0344±0.01	0.0808±0.04
HR	0.0088±0.01	0.0144±0.01	0.0133±0.01	0.0110±0.01	0.0091±0.01	0.0100±0.01	0.0182±0.02	0.0080±0.01	0.0407±0.08	0.0304±0.00	0.0000±0.00	**0.0456±0.02**
ZO	0.7172±0.08	0.7438±0.09	0.7395±0.09	0.7439±0.08	0.7541±0.09	0.7799±0.06	0.7620±0.07	0.7731±0.08	0.7057±0.10	0.7190±0.00	0.8324±0.04	**0.8332±0.04**
BC	0.0044±0.01	0.0123±0.02	0.0355±0.03	**0.0356±0.03**	0.0068±0.01	0.0195±0.01	0.0282±0.04	0.0318±0.03	0.0156±0.02	0.0091±0.00	0.0017±0.00	0.0281±0.01
LG	0.1309±0.04	0.1490±0.04	0.1477±0.05	0.1885±0.07	0.1537±0.04	0.1921±0.07	0.1620±0.05	0.1680±0.03	0.0832±0.06	**0.3300±0.00**	0.2047±0.04	0.1919±0.04
TT	0.0105±0.01	0.0148±0.01	0.0146±0.02	0.0084±0.01	0.0074±0.01	0.0078±0.00	0.0092±0.01	0.0094±0.01	0.0075±0.01	0.0008±0.00	**0.0163±0.02**	0.0132±0.02
AC	0.2208±0.13	0.2686±0.10	0.0935±0.11	0.0920±0.10	0.2430±0.08	0.2677±0.19	0.1667±0.11	0.1266±0.12	0.2803±0.12	0.0009±0.00	0.2402±0.08	**0.3643±0.14**
VT	0.4575±0.03	0.4808±0.02	0.4808±0.02	0.4948±0.00	0.4673±0.02	0.4541±0.16	0.4893±0.00	0.4844±0.00	0.4747±0.00	0.0001±0.00	0.4893±0.00	**0.5322±0.00**
CT4	0.0028±0.00	0.0029±0.00	0.0017±0.00	0.0015±0.00	0.0036±0.00	0.0013±0.00	0.0011±0.00	0.0016±0.00	0.0016±0.00	0.0053±0.00	0.0023±0.00	**0.0128±0.01**
OB	0.2256±0.03	0.2330±0.02	0.2081±0.04	0.2262±0.03	0.2342±0.02	0.2391±0.03	**0.2644±0.02**	0.2261±0.04	0.2233±0.03	0.2466±0.03	0.2333±0.00	0.2633±0.01
AV	0.0021±0.00	0.0024±0.00	0.0024±0.00	0.0099±0.01	0.0027±0.00	0.0093±0.01	0.0075±0.01	0.0035±0.00	0.0034±0.00	0.0025±0.01	0.0044±0.00	**0.0110±0.01**
BM	**0.0211±0.00**	0.0143±0.01	0.0135±0.01	0.0198±0.01	0.0179±0.01	0.0121±0.01	0.0155±0.01	0.0192±0.01	0.0204±0.01	0.0103±0.01	0.0177±0.00	0.0194±0.00
CC	0.0150±0.01	0.0149±0.01	0.0149±0.01	0.0084±0.01	0.0119±0.01	0.0076±0.01	0.0023±0.01	0.0148±0.01	0.0141±0.02	0.0156±0.01	0.0144±0.00	**0.0213±0.02**
CR	0.0017±0.00	0.0024±0.00	0.0020±0.00	0.0022±0.00	0.0022±0.00	0.0071±0.01	**0.0115±0.00**	0.0065±0.00	0.0057±0.00	0.0066±0.00	0.0048±0.00	0.0083±0.01
HF	0.0023±0.03	0.0016±0.03	0.0013±0.05	0.0016±0.04	0.0023±0.03	**0.0041±0.03**	0.0021±0.03	0.0026±0.01	0.0018±0.03	0.0019±0.00	–	0.0026±0.01
MA	0.1575±0.08	**0.1705±0.10**	0.0939±0.09	0.1700±0.10	0.1700±0.10	0.1692±0.15	0.1464±0.05	0.1523±0.05	0.1079±0.08	0.0173±0.00	–	0.1582±0.03
$\overline{AR}$	7.84	6.74	7.61	6.55	6.53	5.47	6.03	6.84	7.24	8.13	6.84	2.18

clustering methods. The parameters (if any) of these methods are set to the values recommended in the corresponding paper. Each clustering result was recorded as the average performance with standard deviations of 10 runs of the compared methods.

Three external and one internal validity indices are employed to evaluate clustering performance. The external indices: Clustering Accuracy (CA) [18], Adjusted Rand Index (ARI) [16, 40] and Normalized Mutual Information (NMI) [13] measure alignment with ground truth, where higher values indicate better performance. The internal index, an entropy-based Clustering coMPactness (CMP) is introduced to eliminate the influence of different distance metric magnitudes, which is computed as:

$$\mathrm{CMP} = \frac{1}{s \times k} \sum_{j=1}^{k} \sum_{r=1}^{s} \sum_{g=1}^{V_r} \frac{-p(v_{r,g}|C_j) \log p(v_{r,g}|C_j)}{\log l_r}. \tag{17}$$

Lower CMP indicates higher value concentration of samples within clusters, revealing better cluster compactness. In addition, the Wilcoxon signed rank test [44] is also performed to validate the performance difference between OCL and the competitive counterparts.

Table 7. Clustering performance evaluated by CMP (with range [0, 1]). "$\overline{AR}$" row reports the average performance rankings.

Datasets	KMD	LSM	JDM	CBDM	UDMC	DLC	HDC	ADC	MCDC	AMPHM	HARR	OCL
SB	0.4021±0.05	0.3966±0.04	0.3931±0.04	0.3909±0.04	0.3873±0.03	0.3960±0.03	0.3829±0.03	0.3824±0.04	0.4801±0.07	0.3832±0.00	0.4743±0.02	**0.3659±0.01**
NS	0.8917±0.01	0.8768±0.01	0.8405±0.01	–	0.8799±0.01	0.7855±0.03	0.8917±0.01	0.8917±0.01	0.8036±0.03	–	0.8830±0.06	**0.7796±0.01**
AP	0.6067±0.01	0.6021±0.02	0.5868±0.02	0.5989±0.02	0.6060±0.01	0.5743±0.00	0.5950±0.02	**0.5731±0.09**	0.5882±0.05	0.6211±0.00	0.6885±0.04	0.5737±0.00
DS	0.6535±0.05	0.6535±0.05	0.6186±0.05	0.6136±0.05	**0.6068±0.04**	0.6590±0.06	0.6068±0.04	0.6177±0.05	0.6363±0.13	0.7084±0.00	0.6906±0.09	0.6174±0.06
CS	0.7428±0.02	0.7374±0.05	0.7152±0.06	0.7243±0.04	0.7242±0.04	0.6620±0.03	0.7477±0.01	0.7506±0.01	0.6982±0.09	0.7272±0.00	0.7060±0.05	**0.6618±0.03**
HR	0.7138±0.01	0.7111±0.01	0.7136±0.02	0.7162±0.01	0.7106±0.01	0.6797±0.01	0.7196±0.02	0.7128±0.01	0.7682±0.09	0.7009±0.00	**0.6297±0.00**	0.6994±0.01
ZO	0.2823±0.02	0.2854±0.03	0.2809±0.02	0.2801±0.02	0.2864±0.03	0.2832±0.02	0.2864±0.03	0.2886±0.03	0.3454±0.05	0.2944±0.00	0.2812±0.01	**0.2799±0.02**
BC	0.7163±0.01	0.7180±0.01	0.7139±0.02	0.7174±0.02	0.7172±0.01	0.6960±0.01	0.7201±0.01	0.7137±0.02	0.6858±0.02	0.7554±0.00	0.7547±0.01	**0.6801±0.01**
LG	0.6155±0.01	0.5945±0.01	0.5996±0.02	0.5985±0.01	0.6017±0.02	0.5797±0.01	0.5826±0.02	0.5804±0.03	**0.5470±0.03**	0.5649±0.00	0.6790±0.01	0.5781±0.01
TT	0.9213±0.00	0.9217±0.01	0.9220±0.01	0.9214±0.00	0.9216±0.00	0.8905±0.00	0.9223±0.00	0.9222±0.00	0.8922±0.00	0.9275±0.00	0.9172±0.00	**0.8876±0.00**
AC	0.6560±0.04	0.6448±0.03	0.6605±0.03	0.6644±0.04	0.6602±0.02	0.6462±0.02	0.6707±0.03	0.6800±0.02	0.6468±0.01	0.7878±0.00	0.8044±0.01	**0.6356±0.03**
VT	0.5510±0.00	0.5511±0.00	0.5511±0.00	0.5464±0.00	0.5482±0.00	0.5633±0.06	0.5450±0.00	0.5451±0.00	0.5458±0.00	0.5772±0.00	0.5454±0.00	**0.5427±0.00**
CT4	0.401±0.01	0.4004±0.01	0.3914±0.01	0.3906±0.01	0.3978±0.01	**0.3811±0.01**	0.3858±0.01	0.3872±0.01	0.3988±0.02	0.4013± 0.00	0.3968±0.01	0.3952±0.01
OB	0.3403±0.05	0.3359±0.04	0.3887±0.05	0.3752±0.06	0.3639±0.06	0.3280±0.03	0.3825±0.06	0.3688±0.04	0.3663±0.03	0.3841±0.00	0.3019±0.02	**0.2973±0.01**
AV	0.7039±0.02	0.6861±0.02	**0.6265±0.04**	0.6623±0.05	0.6912±0.02	0.6355±0.02	0.7011±0.03	0.6844±0.05	0.6760±0.02	0.6585±0.00	0.6987 ± 0.07	0.6518±0.02
BM	0.6153±0.01	0.6205±0.03	0.6255±0.02	0.6144±0.02	0.6152±0.02	0.6017±0.03	0.6281±0.01	0.6242±0.01	0.6128±0.03	0.7326±0.00	0.7875±0.01	**0.5762±0.00**
CC	0.5342±0.01	0.5343±0.01	0.5355±0.01	0.5334±0.02	0.5333±0.01	0.5466±0.01	0.5322±0.03	0.5354±0.02	0.5426±0.03	0.5595±0.00	0.5637±0.03	**0.5303±0.01**
CR	0.8339±0.01	0.8278±0.01	0.8238±0.02	0.8288±0.02	0.8257±0.01	0.7880±0.04	0.8047±0.01	0.8008±0.01	0.7826±0.03	0.8045±0.00	0.8645 ± 0.00	**0.7795±0.02**
HF	0.8257±0.01	0.8174±0.01	0.8118±0.01	0.8140±0.01	0.8184±0.01	0.8187±0.01	0.8297±0.01	0.8518±0.01	0.7960±0.07	0.8244±0.00	–	**0.7942±0.01**
MA	0.6857±0.02	0.6865±0.02	0.6925±0.02	0.6850±0.02	0.6850±0.01	0.6957±0.01	**0.6429±0.01**	0.6815±0.01	0.6861±0.01	0.7289±0.00	–	0.6690±0.01
$\overline{AR}$	8.07	7.25	6.53	6.35	6.38	4.70	7.00	6.45	5.70	9.18	8.60	1.80

Table 8. Wilcoxon signed-rank test comparing OCL vs. the best non-OCL method on each dataset based on CA performance. The **Results** column indicates whether the difference is significant at 95% confidence level.

Datasets	Counterpart	p-value	Results	Datasets	Counterpart	p-value	Results
SB	vs. HARR	0.31731	×	AC	vs. LSM	0.00195	✓
NS	vs. DLC	0.01367	✓	VT	vs. CBDM	0.00195	✓
AP	vs. JDM	0.75183	×	CT4	vs. KMD	0.19335	×
DS	vs. CBDM	0.02601	✓	OB	vs. HDC	0.00390	✓
CS	vs. DLC	0.17971	×	AV	vs. CBDM	0.17320	×
HR	vs. AMPHM	0.00195	✓	BM	vs. HDC	0.00977	✓
ZO	vs. HARR	0.00390	✓	CC	vs. AMPHM	0.02734	✓
BC	vs. AMPHM	0.00977	✓	CR	vs. HDC	0.00195	✓
LG	vs. AMPHM	0.00195	✓	HF	vs. AMPHM	0.00195	✓
TT	vs. MCDC	0.05041	×	MA	vs. UDMC	0.03389	✓

4.2 Clustering Performance Evaluation

The clustering performance of various methods is evaluated as shown in Tables 4 - 7. The best and second-best results on each dataset are highlighted in **bold** and underlined, respectively. The results of CBDM and AMPHM are not reported on the NS dataset because all the NS's attributes are independent of each other, making the inter-attribute dependence-based methods like CBDM and AMPHM fail in measuring distances. Moreover, the clustering performance of HARR on the datasets HF and MA is not reported because the projection mechanism of HARR generates excessive endogenous subspaces when handling multiple categorical values, resulting in computationally intractable overhead.

External (label-based) validation: The clustering results in terms of different validity indices are shown in Tables 4 - 6. The observations include the following five aspects: (1) Overall, OCL reaches the best result on most datasets, indicating its superiority in clustering. (2) On the CA index, although OCL does not perform significantly better than the second-best method on the AP and CR datasets, the second-best methods are different on these datasets, but OCL has stable performance on them. (3) On the CA index, although OCL does not perform the best on CS, AV and TT datasets, its performance is close to the best result (the difference is less than 0.005 in CS and AV datasets, less than 0.03 in TT dataset), which still proves OCL's competitiveness. (4) In terms of the ARI index, there are five datasets on which OCL does not perform the best, namely AP, CS, BC, LG, and OB. However, the differences between OCL and the best-performing algorithms on AP, CS,

Table 9. Clustering performance of OCL and its ablation variants. OCL-I, OCL-II, and OCL-III are ablated variants of OCL, obtained by successively removing the consideration of order distance, order learning, and order information from OCL. For more detailed description of the ablation variants, please refer to the main text of Section 4.3.

Datasets	CA				ARI				NMI				CMP			
	OCL-III	OCL-II	OCL-I	OCL	OCL-III	OCL-II	OCL-I	OCL	OCL-III	OCL-II	OCL-I	OCL	OCL-III	OCL-II	OCL-I	OCL
SB	0.8468	0.8681	0.9149	**0.9830**	0.7638	0.8371	0.8718	**0.9620**	0.8460	0.9113	0.9262	**0.9757**	0.4021	0.3854	0.3770	**0.3659**
NS	0.3208	0.3359	0.3364	**0.3573**	0.0376	0.0614	0.0592	**0.1015**	0.0458	0.1122	0.1086	**0.1492**	0.8917	0.7915	0.7915	**0.7796**
AP	0.5392	0.6069	0.6085	**0.6222**	0.0022	0.0384	0.0388	**0.0524**	0.0147	0.0744	0.0919	**0.0955**	0.6067	0.6011	0.5937	**0.5737**
DS	0.6525	0.7125	0.7283	**0.7458**	0.1077	0.2295	0.2407	**0.2590**	0.1352	0.2462	0.2567	**0.2732**	0.6535	0.6401	0.6392	**0.6174**
CS	0.5463	0.6388	**0.6500**	0.6313	-0.0023	0.0670	**0.0780**	0.0627	0.0157	0.0816	**0.0975**	0.0808	0.7428	0.6695	**0.6572**	0.6618
HR	0.3758	0.3712	0.3765	**0.4326**	-0.0081	-0.0003	0.0079	**0.0368**	0.0088	0.0153	0.0236	**0.0456**	0.7138	0.6953	0.7040	**0.6994**
ZO	0.6624	0.7000	0.7525	**0.7792**	0.5787	0.6482	0.7160	**0.7536**	0.7172	0.7965	0.8111	**0.8332**	0.2823	0.2968	0.2906	**0.2799**
BC	0.5472	0.5636	0.5643	**0.6650**	0.0046	0.0247	0.0248	**0.0799**	0.0044	0.0110	0.0112	**0.0281**	0.7163	0.6953	0.6930	**0.6801**
LG	0.4277	0.4453	0.4872	**0.5426**	0.0843	0.0953	0.1054	**0.1552**	0.1309	0.1410	0.1572	**0.1919**	0.6155	0.5983	0.5892	**0.5781**
TT	0.5569	0.5762	0.5784	**0.5785**	0.0170	0.0205	0.0132	**0.0226**	0.0105	0.0106	0.0062	**0.0132**	0.9213	0.8978	0.8886	**0.8876**
AC	0.7281	0.7545	0.7546	**0.8206**	0.2596	0.2879	0.2880	**0.4313**	0.2208	0.2237	0.2238	**0.3643**	0.6560	0.6418	0.6418	**0.6356**
VT	0.8625	0.8828	0.8942	**0.8943**	0.5247	0.5850	**0.6208**	0.6207	0.4575	0.5098	**0.5422**	0.5322	0.5510	0.5451	0.5443	**0.5427**
CT4	0.4000	0.3612	0.4034	**0.4121**	-0.0027	0.0018	0.0024	**0.0150**	0.0028	0.0019	0.0078	**0.0128**	0.4010	0.3991	0.3994	**0.3952**
OB	0.3715	0.3596	0.3910	**0.3935**	0.1527	0.1410	**0.1814**	0.1696	0.2256	0.2433	0.2572	**0.2633**	0.3403	0.2992	0.3036	**0.2973**
AV	0.6251	**0.7144**	**0.7144**	0.6637	0.0073	**0.0483**	**0.0483**	0.0393	0.0021	0.0088	0.0088	**0.0110**	0.7039	**0.6346**	**0.6346**	0.6518
BM	0.5730	0.6029	0.6049	**0.6127**	0.0146	0.0158	0.0165	**0.0233**	**0.0211**	0.0166	0.0181	0.0194	0.6153	0.5859	0.5802	**0.5762**
CC	0.5562	0.5710	0.5711	**0.5726**	0.0165	0.0284	**0.0285**	**0.0285**	0.0150	0.0211	0.0212	**0.0213**	0.5342	0.5298	**0.5295**	0.5303
CR	0.5209	0.5507	0.5512	**0.5590**	0.0021	0.0121	0.0128	**0.0145**	0.0017	0.0072	0.0080	**0.0083**	0.8339	0.7807	0.7852	**0.7795**
HF	0.5482	0.5331	0.5495	**0.5652**	0.0009	0.0002	0.0021	**0.0072**	0.0023	0.0013	0.0008	**0.0026**	0.8257	0.8042	0.7947	**0.7942**
MA	0.6961	**0.7430**	**0.7430**	0.7333	0.1773	**0.2622**	**0.2622**	0.2165	0.1575	**0.2019**	**0.2019**	0.1582	0.6857	0.6771	0.6766	**0.6690**
$\overline{AR}$	3.80	3.00	1.90	1.30	3.85	2.80	1.93	1.43	3.65	2.85	2.20	1.30	3.90	2.62	2.17	1.30

Table 10. Clustering performance of OCL variants with differential treatment of nominal and ordinal attributes: LNRO learns orders only for nominal attributes, while RNRO preserves the semantic order of ordinal attributes and does not perform order learning for all attributes. For more detailed description of the two OCL variants, please refer to the main text of Section 4.3.

Datasets	CA			ARI			NMI			CMP		
	RNRO	LNRO	OCL	RNRO	LNRO	OCL	RNRO	LNRO	OCL	RNRO	LNRO	OCL
NS	0.3308	0.3281	**0.3573**	0.0734	0.0705	**0.1015**	0.1098	0.1068	**0.1492**	0.7954	0.7954	**0.7796**
AP	0.5984	0.6074	**0.6222**	0.0315	0.0385	**0.0524**	0.0619	0.0791	**0.0955**	0.5737	0.5726	**0.5712**
CS	0.6225	**0.6350**	0.6313	0.0545	**0.0647**	0.0627	0.0726	0.0840	**0.0808**	**0.6618**	0.6776	0.6776
HR	0.3902	0.3985	**0.4326**	0.0066	0.0131	**0.0368**	0.0189	0.0230	**0.0456**	0.6994	0.6965	**0.6946**
BC	0.5276	0.5476	**0.6650**	0.0062	0.0156	**0.0799**	0.0047	0.0083	**0.0281**	0.6801	0.6771	**0.6573**
LG	0.4493	0.4818	**0.5426**	0.1076	0.1122	**0.1552**	0.1610	0.1500	**0.1919**	0.5797	**0.5733**	0.5781
CR	0.5533	0.5583	**0.5590**	0.0129	0.0141	**0.0145**	0.0076	0.0082	**0.0083**	0.7835	0.7795	**0.7777**
OB	0.3786	0.3586	**0.3935**	0.1648	0.1487	**0.1696**	0.2363	0.2239	**0.2633**	0.3165	0.3248	**0.2973**
BM	0.5882	0.5970	**0.6127**	0.0223	**0.0239**	0.0233	0.0178	0.0179	**0.0194**	0.5817	0.5805	**0.5762**
$\overline{AR}$	2.78	2.22	**1.11**	2.83	2.11	**1.22**	2.67	2.33	**1.11**	2.56	2.00	**1.22**

and BC datasets are very small. (5) As for the NMI index, although OCL does not have the best NMI performance on some datasets, it is not surpassed by much by the winners and is still very competitive.

Internal validation: As can be seen in Table 7, the proposed OCL still performs the best in general in exploring compact and meaningful clusters, as it infers the ordinal structure of attribute values and finely adjusts the distance metric according to each cluster to better serve the clustering objective.

4.3 Ablation Studies

To demonstrate the effectiveness of the core components of OCL and to verify our hypothesis on the orders of categorical data attributes, several ablated variants of OCL are compared from both algorithm and data perspectives.

From the algorithm aspect, to evaluate the effectiveness of the proposed Order Distance (OD), we compare OCL with its variant OCL-I, which uses normalized equidistant order distance without considering the probability distribution of possible values. To evaluate the effectiveness of the Order Learning (OL) mechanism, we further remove the process of order learning from OCL-I and only update the order once in the first iteration, which forms OCL-II. To completely discard Order Information (OI), we further make OCL-II use only the traditional Hamming distance, thus forming OCL-III. Table 9 compares the three OCL variants. It can be observed that the performance of OCL is superior to all variants. More specifically, OCL outperforms OCL-I on 17, 14, 17, and 17 out of 20 datasets in terms of CA, ARI, NMI, and CMP, respectively, indicating that OD effectively leverages the distribution of attribute values across clusters for more reasonable order distance measurement. Furthermore, OCL-I outperforms OCL-II on 17, 16, 15, and 13 out of 20 datasets in terms of CA, ARI, NMI, and CMP, respectively, verifying the effectiveness of the OL mechanism in searching for optimal orders during clustering. In addition, compared to OCL-III, which completely disregards OI, OCL-II achieves better performance on 16, 18, 17, and 19 out of 20 datasets in terms of CA, ARI, NMI, and CMP, respectively, demonstrating that even a roughly estimated initial order can effectively improve clustering performance.

Significant studies: Table 8 demonstrates the Wilcoxon signed rank test [44] results based on the CA results reported in Table 4. For each dataset, OCL is compared with the best-performing counterpart using the one-sided Wilcoxon test under a 95% confidence level. It can be observed that OCL significantly outperforms the second-best method in most cases, providing strong statistical evidence for its overall superiority. Moreover, on some datasets where OCL does not perform the best (i.e., CS, TT, and AV), its performance is not significantly worse than the best-performing counterparts, demonstrating the competitiveness of OCL.

From the data aspect, to validate our hypothesis that correct order information is crucial to categorical data clustering, ablation experiments are conducted on datasets containing both nominal and ordinal attributes. The OCL that learns order for both nominal and ordinal attributes is compared with its two variants: (1) LNRO (Learn Nominal Reserve Ordinal), which retains the semantic order of the original ordinal attribute and only learns orders for the nominal attributes, and (2) RNRO (Reserve Nominal Reserve Ordinal), which preserves the original semantic order for ordinal attributes without learning, and similarly imposes no order learning for nominal attributes. Table 10 compares the clustering performance of the three variants in CA, ARI, and NMI indices. It can be observed that, in terms of CA and ARI, on seven out of the nine datasets, LNRO outperforms RNRO. This verifies our hypothesis that the nominal attributes also contain potential order information that is effective for clustering. AS for the NMI and CMP indices, LNRO outperforms RNRO on six out of nine datasets, the performance gap between LNRO and RNRO is tiny, which can still verify the effectiveness of learning orders for nominal attributes in general. Moreover, we can also find that the OCL outperforms LNRO on eight out of the nine datasets in CA and ARi indices, and seven out of the nine datasets in NMI and CMP indices. This indicates that the orders learned by the proposed mechanism for ordinal attributes are more conducive to clustering than the original semantic orders.

4.4 Convergence and Efficiency Evaluation

The convergence of OCL is evaluated by executing it on the SB, NS, and AP datasets, and present the convergence curves in Figure 3. The horizontal and vertical axes represent the number of learning iterations and the value of the objective function L, respectively. The blue triangles mark the iterations of order updates, while the red square indicates the iteration that OCL converges. It can be observed that, after each order update, the value of L further decreases. This indicates that the proposed order learning mechanism is consistent with the optimization of the clustering

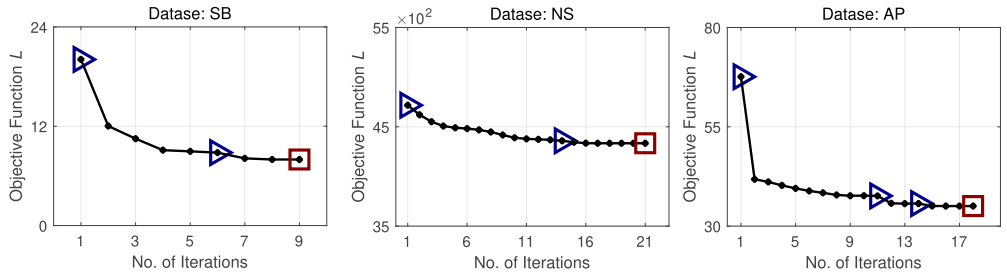

Fig. 3. Convergence curves of OCL on three datasets.

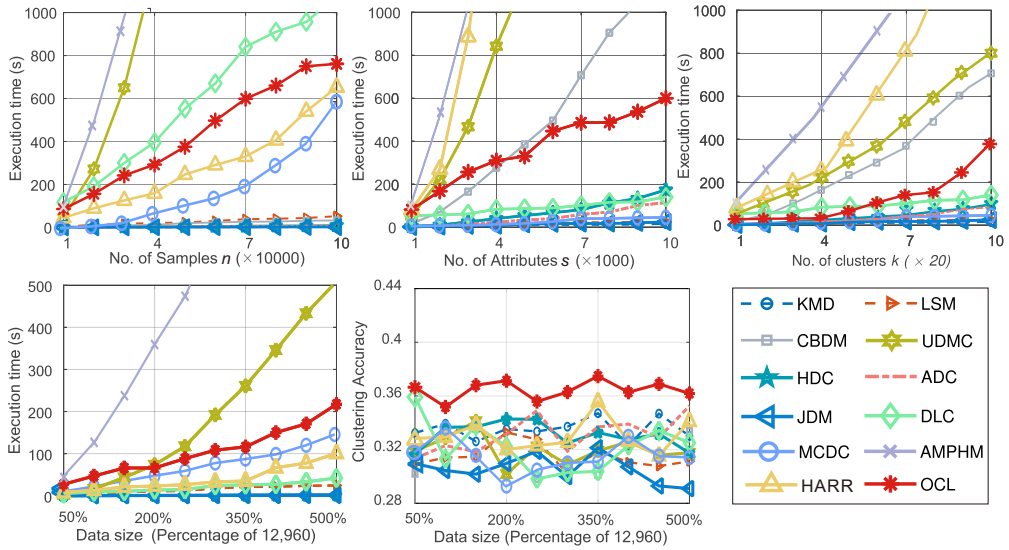

Fig. 4. Efficiency evaluation of different methods on synthetic datasets with varying n, s, k (upper row). Execution time and accuracy on the real NS dataset with varying n is also reported (lower row).

objective function. Moreover, for all the tested datasets, OCL completes the learning process within 40 iterations with at most three updates of the orders. This reflects that OCL converges efficiently with guarantee, which conforms to the analysis in Section 3.5.

To evaluate the efficiency of OCL, large synthetic datasets are generated with varying numbers of samples (n), attributes (s), and clusters (k) with the following configurations: (1) fix $s = 20$ and $k = 5$, and vary n from 10k to 100k in steps of 10k, (2) fix $n = 2k$ and $k = 5$, and vary s from 1k to 10k in steps of 1k, and (3) fix $n = 2k$ and $s = 20$, and vary k from 20 to 200 in steps of 20. All sample values are randomly generated from five possible values per attribute. Overall, Figure 4 shows that OCL achieves higher efficiency than state-of-the-art methods while maintaining comparable scalability to traditional baseline methods. More specifically, on the synthetic datasets, OCL maintains a nearly linear growth rate in execution time, which is consistent with the time complexity analysis in Theorem 1, and its efficiency outperforms: (1) UDMC, AMPHM, and DLC on large-scale datasets; (2) HARR, UDMC, AMPHM, and CBDM on high-dimensional datasets; (3) the same competitors on datasets with numerous clusters. In addition, the evaluation is also

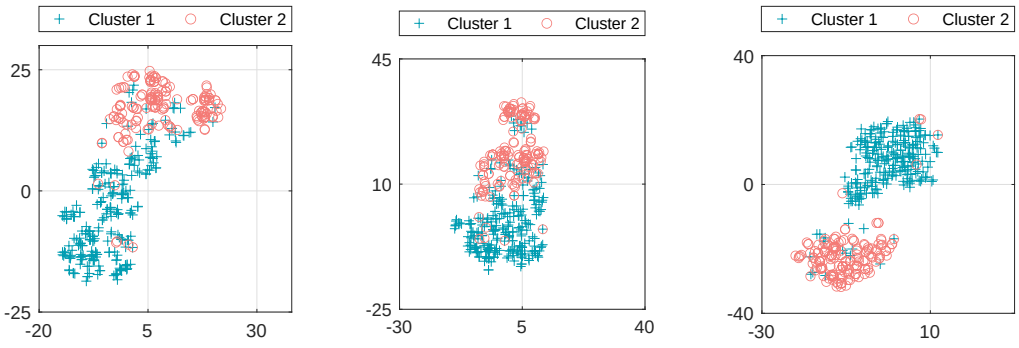

Fig. 5. t-SNE visual comparison of CBDM (left), ADC (middle), and OCL (right) on the VT dataset.

Table 11. Clustering performance of mixed data clustering approaches and their OCL-enhanced versions (adopting the distance metric learned by OCL) on AP, AC, and DS datasets. The green arrow indicates an improvement achieved by the OCL-enhanced version; the gray "-" indicates no performance change.

Datasets	Indices	KPT	KMS+OCL	HARR	HARR+OCL	ADC	ADC+OCL
AP	CA	0.5323	0.5550↑	0.5688	0.6042↑	0.5487	0.5677↑
	ARI	0.0015	0.0104↑	0.0164	0.0373↑	0.0049	0.0144↑
	NMI	0.0051	0.0106↑	0.0371	0.0593↑	0.0070	0.0126↑
AC	CA	0.7425	0.7975↑	0.8213	0.8232↑	0.6445	0.6480↑
	ARI	0.2794	0.3537↑	0.4121	0.4169↑	0.0797	0.0839↑
	NMI	0.2154	0.2738↑	0.3184	0.3225↑	0.0607	0.0644↑
DS	CA	0.6842	0.6842-	0.8117	0.8117-	0.8667	0.8667-
	ARI	0.1684	0.1684-	0.4364	0.4364-	0.5336	0.5336-
	NMI	0.1888	0.1888-	0.4277	0.4277-	0.4244	0.4244-

conducted on the real benchmark NS dataset with varying sampling rates (from 50% to 500% in steps of 50%) to show that OCL achieves scalability without compromising clustering accuracy. In summary, OCL is effective and efficient compared to state-of-the-art methods, without incurring much additional computational cost in comparison with the traditional simple approaches.

4.5 Visualization of Clustering Effect

To demonstrate the cluster discrimination ability of the OCL method, We use OCL, CBDM, and ADC to learn a distance metric for the VT dataset, then compute a new distance matrix for the data samples, which are projected into 2-D space for visualization via t-SNE [38], as shown in Figure 5. We use different markers (distinguished in colors and shapes) to indicate ground-truth clusters, with point positions representing the samples projected into the 2-D space by t-SNE. This effectively visualizes how well each method captures the distinguishable cluster distributions for intuitive comparison. It can be seen from Figure 5 that OCL has significantly better cluster discrimination ability, as it performs clustering task-oriented distance learning to better suit the exploration of the k^* clusters.

4.6 Clustering Performance on Mixed Datasets

To demonstrate the potential of OCL in boosting the clustering of mixed data comprising both numerical and categorical attributes, we compare the original k-ProTotypes(KPT) [21], HARR [56],

and ADC [48] methods proposed for mixed data clustering, with their corresponding variants enhanced by the orders learned by OCL. More specifically, for KPT, we first encode the categorical attribute values using the orders learned by the proposed OCL method and then treat the dataset as a pure numerical dataset for clustering using the k-means (KMS) algorithm [7]. Actually, KPT enhanced by OCL is equivalent to KMS with the categorical attributes encoded into numerical ones by OCL, and thus we call the enhanced version KMS+OCL. For HARR and ADC, since they are originally competent in processing ordinal attributes, the orders learned by OCL are assigned to the categorical attributes, and we let HARR and ADC treat all the categorical attributes as ordinal attributes. Accordingly, their enhanced versions are named HARR+OCL and ADC+OCL, respectively. Table 11 demonstrates the performance of the above-mentioned approaches and their OCL-enhanced versions.

It can be observed that OCL considerably enhances the clustering performance of the three approaches in terms of AP and AC datasets. This significantly indicates that the learned orders can effectively improve the clustering performance on mixed data. The improvement on the AP dataset is more obvious because the AP contains both nominal and ordinal attributes as shown in Table 2. This situation allows OCL to leverage its superiority in learning orders for nominal attributes and potential corrections on the semantic orders of ordinal attributes. A noteworthy special case can be seen on the DS dataset. Since DS has only two possible values on each of its attributes, OCL cannot further optimize the orders. This is why the clustering performance on the DS dataset remains unchanged after using the orders learned by OCL. This observation suggests that when the original semantic order of an ordinal attribute is already optimal, OCL exerts no measurable effect on clustering performance. In short, the proposed OCL is promising for more complex mixed-data clustering scenarios.

4.7 Case Study on Hayes-Roth (HR) Dataset

It is worth noting that the proposed order learning method not only improves the clustering performance but also mines the hidden order information when facing datasets with unclear semantic information. This case study investigates whether the value order learned by the proposed method is interpretable and consistent with human subjective understanding. The HR dataset with one nominal and three ordinal attributes records personal information, such as the nominal "Hobby" and the ordinal "Education level", "Age", and "Marital status". The true labels divide the samples into three groups of club members for psychological evaluation. The original dataset information is visualized in Figure 6 (a).

It can be seen from Figure 6 (b) that OCL can correctly learn the semantic order of the values from attributes "Education level" and "Age", as the learned orders are consistent with the original semantic orders. Moreover, for the nominal attribute "Hobby", although there is no original obvious semantic order, the learned order is very meaningful in data analysis and knowledge acquisition. First, the learned order distance between "chess" and "stamp collection" hobbies is closer, while the distance between "stamp collection" and "sport" is larger. This is consistent with cognitive intuition that "chess" is a quieter sport, so it ranks between the quieter hobby "stamp collection" and its sibling "sport". This indicates that OCL can learn latent semantic orders for nominal attributes, which is an important characteristic for categorical data understanding.

As shown in the clustering results obtained by OCL in Figure 6 (c), it can be seen that there are more cases of people who enjoy playing chess and collecting stamps appearing in the same cluster, which is consistent with human intuition and the learned order of "Hobby" attribute. This further confirms the prospects of our OCL method in discovering the implicit order information of unlabeled data with no obvious semantic order.

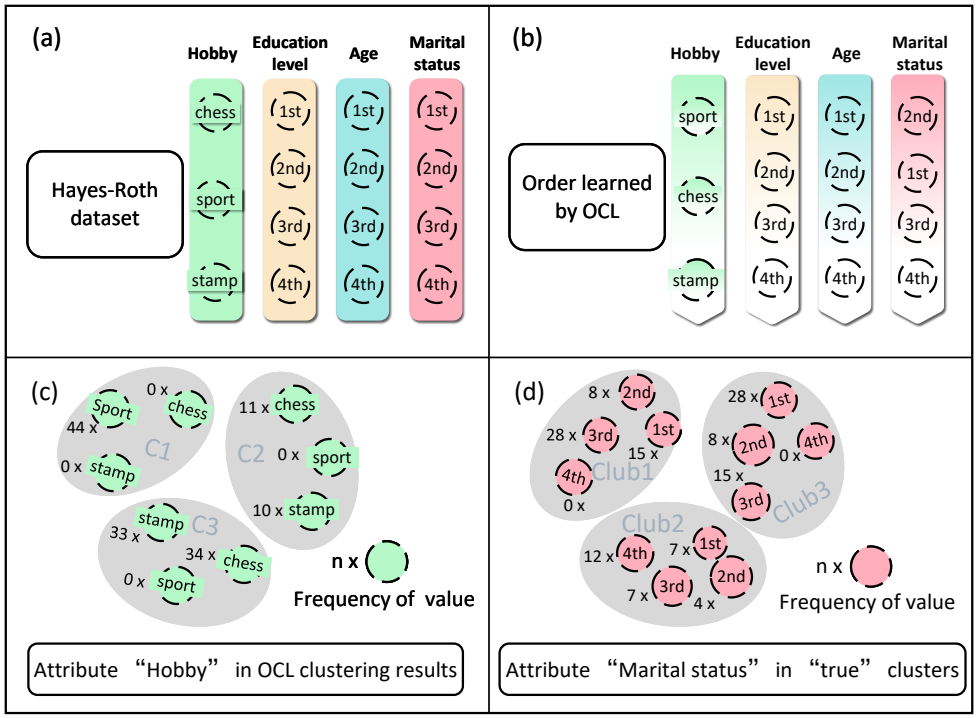

Fig. 6. A case study on HR dataset. "Hobby" is a nominal attribute with possible values {chess, sport, stamp}, and the remaining attributes are ordinal attributes with the semantic order of their possible values indicated by {1st, 2nd, 3rd, 4th}. Sub-figure (a) demonstrates the original data. Sub-figure (b) showcases the orders learned by the OCL method. Sub-figure (c) illustrates the possible values of the "Hobby" attribute in the clusters obtained by OCL. Sub-figure (d) presents the possible values of the "Marital status" attribute in the true clusters indicated by the data labels.

For the "Marital status" attribute, the order learned by OCL is inconsistent with the original semantic order as demonstrated in Figure 6 (b). The order learned by OCL indicates that the possible values semantically ranked first and third in "Marital status" should be more similar to better serve clustering. From the true clusters given by the dataset shown in Figure 6 (d), it can be found that more of the first and third values of the "Marital status" attribute simultaneously appear in the same clusters, which confirms the reasonableness of the learned order. This characteristic allows the users to better understand the cluster distribution of data, or even explore deeper meanings and relationships of values.

5 Concluding Remarks

This paper introduces and validates a new finding that knowing the correct order relation of attribute values is critical to cluster analysis of categorical data. Accordingly, the OCL paradigm is proposed, allowing joint learning of data partitions and order relation-guided distances. Its learning mechanism can infer a more appropriate order among attribute values based on the current clusters, thus yielding more compact clusters based on the updated order. The data partition and order learning are iteratively executed to approach the optimal clustering solution. As a result, more accurate clustering results and insightful orders can be obtained. OCL converges quickly

with a guarantee, and extensive experiments on 20 real benchmark datasets fully demonstrate its superiority in comparison with the state-of-the-art clustering methods. A case study has also been provided to show the rationality of the learned order in understanding categorical data clusters. The proposed OCL is also not exempt from limitations. For instance, the natural connection between the treatment of categorical and numerical attributes remains to be explored. Moreover, it would be promising to extend OCL into more complex scenarios, e.g., processing non-stationary mixed dataset with unknown number of imbalanced clusters.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (NSFC) under grants: 62476063, 62376233, 61806131 and 62306181, the NSFC/Research Grants Council (RGC) Joint Research Scheme under the grant N_HKBU214/21, the Natural Science Foundation of Guangdong Province under grant: 2025A1515011293, the Natural Science Foundation of Fujian Province under grant: 2024J09001, the National Key Laboratory of Radar Signal Processing under grant: JKW202403, the General Research Fund of RGC under grants: 12201321, 12202622, and 12201323, the RGC Senior Research Fellow Scheme under grant: SRFS2324-2S02, the Shenzhen Science and Technology Program under grant: RCBS20231211090659101, the Guangdong Provincial Key Laboratory under grant: 2023B1212060076, and the Xiaomi Young Talents Program.

References

[1] D Adolphson and Tien Chung Hu. 1973. Optimal linear ordering. *SIAM J. Appl. Math.* 25, 3 (1973), 403–423.
[2] Alan Agresti. 2003. *Categorical data analysis*. John Wiley & Sons.
[3] Amir Ahmad and Lipika Dey. 2007. A method to compute distance between two categorical values of same attribute in unsupervised learning for categorical data set. *Pattern Recognition Letters* 28, 1 (2007), 110–118.
[4] Madhavi Alamuri, Bapi Raju Surampudi, and Atul Negi. 2014. A survey of distance/similarity measures for categorical data. In *Proceedings of the 2014 International Joint Conference on Neural Networks*. 1907–1914.
[5] Liang Bai and Jiye Liang. 2022. A categorical data clustering framework on graph representation. *Pattern Recognition* 128 (2022), 108694.
[6] Liang Bai, Jiye Liang, Chuangyin Dang, and Fuyuan Cao. 2011. A novel attribute weighting algorithm for clustering high-dimensional categorical data. *Pattern Recognition* 44, 12 (2011), 2843–2861.
[7] Geoffrey H Ball and David J Hall. 1967. A clustering technique for summarizing multivariate data. *Behavioral Science* 12, 2 (1967), 153–155.
[8] Shyam Boriah, Varun Chandola, and Vipin Kumar. 2008. Similarity measures for categorical data: A comparative evaluation. In *Proceedings of the 2008 SIAM International Conference on Data Mining*. 243–254.
[9] Shenghong Cai, Yiqun Zhang, Xiaopeng Luo, Yiu-ming Cheung, Hong Jia, and Peng Liu. 2024. Robust Categorical Data Clustering Guided by Multi-Granular Competitive Learning. In *Proceedings of the 44th International Conference on Distributed Computing Systems*. 288–299.
[10] Junyang Chen, Yuzhu Ji, Rong Zou, Yiqun Zhang, and Yiu-ming Cheung. 2024. QGRL: Quaternion Graph Representation Learning for Heterogeneous Feature Data Clustering. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 297–306.
[11] Tiago RL dos Santos and Luis E Zárate. 2015. Categorical data clustering: What similarity measure to recommend? *Expert Systems with Applications* 42, 3 (2015), 1247–1260.
[12] Dheeru Dua and Efi Karra Taniskidou. 2017. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml
[13] Pablo A Estévez, Michel Tesmer, Claudio A Perez, and Jacek M Zurada. 2009. Normalized mutual information feature selection. *IEEE Transactions on Neural Networks* 20, 2 (2009), 189–201.
[14] Zijin Feng, Miao Qiao, Chengzhi Piao, and Hong Cheng. 2025. On Graph Representation for Attributed Hypergraph Clustering. *Proceedings of the ACM on Management of Data* 3, 1 (2025), 1–26.
[15] Francisco Fernandez-Navarro, Pilar Campoy-Munoz, César Hervás-Martínez, and Xin Yao. 2013. Addressing the EU sovereign ratings using an ordinal regression approach. *IEEE Transactions on Cybernetics* 43, 6 (2013), 2228–2240.
[16] Alexander J Gates and Yong-Yeol Ahn. 2017. The impact of random models on clustering similarity. *The Journal of Machine Learning Research* 18, 1 (2017), 3049–3076.
[17] David Gibson, Jon Kleinberg, and Prabhakar Raghavan. 2000. Clustering categorical data: An approach based on dynamical systems. *The VLDB Journal* 8, 3 (2000), 222–236.

[18] Xiaofei He, Deng Cai, and Partha Niyogi. 2005. Laplacian score for feature selection. In *Proceedings of the 18th International Conference on Neural Information Processing Systems*. 507–514.
[19] Xiao He, Jing Feng, Bettina Konte, Son T. Mai, and Claudia Plant. 2014. Relevant overlapping subspace clusters on categorical data. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 213–222.
[20] Lianyu Hu, Mudi Jiang, Yan Liu, and Zengyou He. 2022. Significance-Based Categorical Data Clustering. *ArXiv* abs/2211.03956 (2022).
[21] Zhexue Huang. 1997. Clustering large data sets with mixed numeric and categorical values. In *Proceedings of the First Pacific-Asia Conference on Knowledge Discovery and Data Mining*. 21–34.
[22] Zhexue Huang. 1998. Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Mining and Knowledge Discovery* 2, 3 (1998), 283–304.
[23] Dino Ienco, Ruggero G Pensa, and Rosa Meo. 2009. Context-Based Distance Learning for Categorical Data Clustering. In *Proceedings of the Eighth International Symposium on Intelligent Data Analysis*. 83–94.
[24] Dino Ienco, Ruggero G Pensa, and Rosa Meo. 2012. From context to distance: Learning dissimilarity for categorical data clustering. *ACM Transactions on Knowledge Discovery from Data* 6, 1 (2012), 1–25.
[25] Hong Jia and Yiu-ming Cheung. 2018. Subspace clustering of categorical and numerical data with an unknown number of clusters. *IEEE Transactions on Neural Networks and Learning Systems* 29, 8 (2018), 3308–3325.
[26] Hong Jia, Yiu-ming Cheung, and Jiming Liu. 2016. A new distance metric for unsupervised learning of categorical data. *IEEE Transactions on Neural Networks and Learning Systems* 27, 5 (2016), 1065–1079.
[27] Songlei Jian, Longbing Cao, Kai Lu, and Hang Gao. 2018. Unsupervised coupled metric similarity for non-IID categorical data. *IEEE Transactions on Knowledge and Data Engineering* 30, 9 (2018), 1810–1823.
[28] Songlei Jian, Longbing Cao, Guansong Pang, Kai Lu, and Hang Gao. 2017. Embedding-based representation of categorical data by hierarchical value coupling learning. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*. 1937–1943.
[29] Songlei Jian, Guansong Pang, Longbing Cao, Kai Lu, and Hang Gao. 2018. CURE: Flexible categorical data representation by hierarchical coupling learning. *IEEE Transactions on Knowledge and Data Engineering* 31, 5 (2018), 853–866.
[30] Xuechun Jing, Fuyuan Cao, Kui Yu, and Jiye Liang. 2025. CM-CaFE: A Clustering Method with Causality-based Feature Embedding. *ACM Transactions on Knowledge Discovery from Data* 19, 3 (2025), 1–23.
[31] Valen E Johnson and James H Albert. 2006. *Ordinal data modeling*. Springer Science & Business Media.
[32] Amit Kumar Kar, Mohammad Maksood Akhter, Amaresh Chandra Mishra, and Sraban Kumar Mohanty. 2024. EDMD: An Entropy based Dissimilarity measure to cluster Mixed-categorical Data. *Pattern Recognition* (2024), 110674.
[33] Si Quang Le and Tu Bao Ho. 2005. An association-based dissimilarity measure for categorical data. *Pattern Recognition Letters* 26, 16 (2005), 2549–2557.
[34] Hongyu Li, Lefei Zhang, Kehua Su, and Wei Yu. 2024. MICCF: A Mutual Information Constrained Clustering Framework for Learning Clustering-Oriented Feature Representations. *ACM Transactions on Knowledge Discovery from Data* 18, 8 (2024), 1–22.
[35] Tao Li, Sheng Ma, and Mitsunori Ogihara. 2004. Entropy-based criterion in categorical clustering. In *Proceedings of the 21st International Conference on Machine Learning*. 536–543.
[36] Dekang Lin. 1998. An information-theoretic definition of similarity. In *Proceedings of the 15th International Conference on Machine Learning*. 296–304.
[37] Eliane Maria Loiola, Nair Maria Maia De Abreu, Paulo Oswaldo Boaventura-Netto, Peter Hahn, and Tania Querido. 2007. A survey for the quadratic assignment problem. *European journal of operational research* 176, 2 (2007), 657–690.
[38] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research* 9, 11 (2008), 2579–2605.
[39] Yuhua Qian, Feijiang Li, Jiye Liang, Bing Liu, and Chuangyin Dang. 2015. Space structure and clustering of categorical data. *IEEE Transactions on Neural Networks and Learning Systems* 27, 10 (2015), 2047–2059.
[40] William M Rand. 1971. Objective criteria for the evaluation of clustering methods. *J. Amer. Statist. Assoc.* 66, 336 (1971), 846–850.
[41] Hafiz Tayyab Rauf, Andre Freitas, and Norman William Paton. 2025. Tabledc: Deep clustering for tabular data. *Proceedings of the ACM on Management of Data* 3, 3 (2025), 1–28.
[42] R. Sandler and M. Lindenbaum. 2011. Nonnegative Matrix Factorization with Earth Mover's Distance Metric for Image Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33, 8 (2011), 1590–1602.
[43] Runzhong Wang, Junchi Yan, and Xiaokang Yang. 2021. Neural graph matching network: Learning lawler's quadratic assignment problem with extension to hypergraph and multiple-graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 9 (2021), 5261–5279.
[44] Frank Wilcoxon. 1945. Individual comparisons by ranking methods. *Biometrics bulletin* 1, 6 (1945), 80–83.

[45] Chuanbin Zhang, Long Chen, Yin-Ping Zhao, Yingxu Wang, and C. L. Philip Chen. 2023. Graph Enhanced Fuzzy Clustering for Categorical Data Using a Bayesian Dissimilarity Measure. *IEEE Transactions on Fuzzy Systems* 31 (2023), 810–824.
[46] Chunyong Zhang, Long Chen, Yin-Ping Zhao, Yingxu Wang, and C. L. Philip Chen. 2023. Graph Enhanced Fuzzy Clustering for Categorical Data Using a Bayesian Dissimilarity Measure. *IEEE Transactions on Fuzzy Systems* 31 (2023), 810–824.
[47] Yiqun Zhang and Yiu-ming Cheung. 2022. A new distance metric exploiting heterogeneous inter-attribute relationship for ordinal-and-nominal-attribute data clustering. *IEEE Transactions on Cybernetics* 52, 2 (2022), 758–771.
[48] Yiqun Zhang and Yiu-ming Cheung. 2023. Graph-Based Dissimilarity Measurement for Cluster Analysis of Any-Type-Attributed Data. *IEEE Transactions on Neural Networks and Learning Systems* 34, 9 (2023), 6530–6544.
[49] Yiqun Zhang, Yiu-ming Cheung, and et al. 2020. An ordinal data clustering algorithm with automated distance learning. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*. 6869–6876.
[50] Yiqun Zhang, Yiu-ming Cheung, and et al. 2022. Learnable Weighting of Intra-Attribute Distances for Categorical Data Clustering with Nominal and Ordinal Attributes. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 7 (2022), 3560–3576.
[51] Yiqun Zhang, Yiu-ming Cheung, and Kaychen Tan. 2020. A unified entropy-based distance metric for ordinal-and-nominal-attribute data clustering. *IEEE Transactions on Neural Networks and Learning Systems* 31, 1 (2020), 39–52.
[52] Yiqun Zhang, Yiu-ming Cheung, and An Zeng. 2022. Het2Hom: Representation of Heterogeneous Attributes into Homogeneous Concept Spaces for Categorical-and-Numerical-Attribute Data Clustering. In *Proceedings of the 31st International Joint Conference on Artificial Intelligence*. 3758–3765.
[53] Yiqun Zhang, Sen Feng, Pengkai Wang, Zexi Tan, Xiaopeng Luo, Yuzhu Ji, Rong Zou, and Yiu-ming Cheung. 2025. Learning self-growth maps for fast and accurate imbalanced streaming data clustering. *IEEE Transactions on Neural Networks and Learning Systems* 36, 9 (2025), 16049–16061.
[54] Yiqun Zhang, Zhanpei Huang, Mingjie Zhao, Chuyao Zhang, Yang Lu, Yuzhu Ji, Fangqing Gu, and An Zeng. 2025. Learning Unbiased Cluster Descriptors for Interpretable Imbalanced Concept Drift Detection. *IEEE Transactions on Emerging Topics in Computational Intelligence* (2025), 1–14.
[55] Yunfan Zhang, Yiqun Zhang, Yang Lu, Mengke Li, Xi Chen, and Yiu-ming Cheung. 2025. Asynchronous federated clustering with unknown number of clusters. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 39. 22695–22703.
[56] Yiqun Zhang, Mingjie Zhao, Yizhou Chen, Yang Lu, and Yiu-ming Cheung. 2025. Learning unified distance metric for heterogeneous attribute data clustering. *Expert Systems with Applications* 273 (2025), 126738.
[57] Yunfan Zhang, Rong Zou, Yiqun Zhang, Yue Zhang, Yiu-ming Cheung, and Kangshun Li. 2025. Adaptive micro partition and hierarchical merging for accurate mixed data clustering. *Complex & Intelligent Systems* 11 (2025), 124–128.
[58] Mingjie Zhao, Sen Feng, Yiqun Zhang, Mengke Li, Yang Lu, and Yiu-ming Cheung. 2024. Learning Order Forest for Qualitative-Attribute Data Clustering. In *Proceedings of 27th European Conference on Artificial Intelligence*. 1943 – 1950.
[59] Yangming Zhou, Jin-Kao Hao, and Béatrice Duval. 2020. Frequent pattern-based search: A case study on the quadratic assignment problem. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 52, 3 (2020), 1503–1515.
[60] Chengzhang Zhu, Longbing Cao, and Jianping Yin. 2022. Unsupervised heterogeneous coupling learning for categorical representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 1 (2022), 553–549.
[61] Chengzhang Zhu, Qi Zhang, Longbing Cao, and Arman Abrahamyan. 2020. Mix2Vec: Unsupervised Mixed Data Representation. In *Proceedings of the 7th International Conference on Data Science and Advanced Analytics*. 118–127.
[62] Rong Zou, Yunfan Zhang, Mingjie Zhao, Zexi Tan, Yiqun Zhang, and Yiu-ming Cheung. 2025. SDENK: Unbiased Subspace Density–Clustering. *Neurocomputing* (2025), 131225.

Received April 2025; revised July 2025; accepted August 2025