Unsupervised Heterogeneous Coupling Learning for Categorical Representation

Chengzhang Zhu, Longbing Cao, *Senior Member, IEEE,* and Jianping Yin

Abstract—Complex categorical data is often hierarchically coupled with heterogeneous relationships between attributes and attribute values and the couplings between objects. Such value-to-object couplings are heterogeneous with complementary and inconsistent interactions and distributions. Limited research exists on unlabeled categorical data representations, ignores the heterogeneous and hierarchical couplings, underestimates data characteristics and complexities, and overuses redundant information, etc. The deep representation learning of unlabeled categorical data is challenging, overseeing such value-to-object couplings, complementarity and inconsistency, and requiring large data, disentanglement, and high computational power. This work introduces a shallow but powerful UNsupervised heTerogeneous couplIng lEarning (UNTIE) approach for representing coupled categorical data by untying the interactions between couplings and revealing heterogeneous distributions embedded in each type of couplings. UNTIE is efficiently optimized w.r.t. a kernel *k*-means objective function for unsupervised representation learning of heterogeneous and hierarchical value-to-object couplings. Theoretical analysis shows that UNTIE can represent categorical data with maximal separability while effectively represent heterogeneous couplings and disclose their roles in categorical data. The UNTIE-learned representations make significant performance improvement against the state-of-the-art categorical representations and deep representation models on 25 categorical data sets with diversified characteristics.

Index Terms—Coupling Learning, Heterogeneity Learning, Non-IID Learning, Representation Learning, Similarity Learning, Categorical Data, Categorical Data Representation, Unsupervised Categorical Representation, Unsupervised Learning

◆

1 INTRODUCTION

THE recent years have seen increasing research on learning the representation of complex categorical data ('categorical representation' for short) [1], [2], [3], [4], [5], which shows critical for downstream tasks, e.g., regression [6], clustering [7], classification [8], and outlier detection [9]. Different from numerical data, the attribute values, attributes and objects of categorical data are often coupled with each other w.r.t. various aspects, e.g., value frequency, co-occurrence and distribution; attribute relations including correlations and dependency, interactions and hierarchy; and other data characteristics [8], [10], [11], [12]. We broadly refer them to *couplings* [10], which are heterogeneous - diverse interactions and distributions, and hierarchical - from values to objects, and drive the complexities and dynamics of categorical data. Learning such heterogeneous and hierarchical couplings shows fundamental for appropriate categorical data representation, however, rarely explored in unsupervised settings.

Besides the critical progress made in learning the similarity and metrics of categorical data w.r.t. value co-occurrences and attribute relations [13], [14], [15], coupling learning [10] explores even more comprehensive and stronger categorical representations by revealing and embedding heterogeneous value-to-object couplings on explicit attributes and latent factors [4], [16], [17], [18]. Typically, categorical data is converted to either a vector [17] or a similarity [19] space to leverage the missing numerical intervals between categorical values, then numerical analytical tools can be used. Such methods demonstrate a significant potential of deeply understanding intrinsic couplings in categorical data. However, rare work is available and it is very challenging to handle diverse data characteristics and complexities, including heterogeneities,

interactions, structures, relations, distributions and nonstationarity, in categorical data representation [10], [12], [20].

A critical question raised in heterogeneous coupling learning is *whether learning more couplings enhances categorical data representation*. This issue has been initially studied by Ienco et al. [14]. They found the redundant information in various couplings may hamper the quality of categorical data representation and proposed the *symmetric uncertainty* (SU) as a criterion to filter redundant couplings w.r.t. the correlations between two attributes to largely reduce the redundant information and enable better representation performance. Other recent work [17], [21] further shows that the redundant couplings can be reduced by methods like principal component analysis and shows better categorical representation performance than vector, similarity and embedding-based methods. However, rare work identifies redundant couplings and decouple them from those important ones.

Another open issue is to capture diverse interactions and relations that are complementary and inconsistent with each other while as many types of couplings are learned as possible. As illustrated by Table 1, if an intra-attribute coupling (i.e., value couplings) is measured in terms of value frequency, the difference between slightly curled and curled watermelons per the attribute *root shape* is 0 due to their same frequency. However, we can easily differentiate them (i.e. difference is not 0) because the curled root is more related to the yellow and green watermelons while the slightly curled root is more associated with the green and black ones when the inter-attribute couplings between *color* and *root shape* are considered.

The heterogeneity and inconsistency between couplings [20], [22] may be caused by (1) different types of couplings corresponding to distinct interactions in data and following different data distributions; and (2) multiple distributions existing in a data set. While our earlier work in [8] analyzes and captures

Chengzhang Zhu and Longbing Cao (corresponding author) are with the Advanced Analytics Institute, University of Technology Sydney, Australia. E-mail: kevin.zhu.china@gmail.com, Longbing.Cao@gmail.com. Jianping Yin is with Dongguan University of Technology, China.

TABLE 1
Toy Example. The watermelon information table. Each watermelon with different sweetness is described w.r.t. three attributes: *Texture*, *Color*, and *Root Shape*.

ID	Texture	Color	Root Shape	Sweetness
A1	clear	white	straight	low
A2	blurry	yellow	straight	low
A3	blurry	yellow	curled	low
A4	clear	green	slightly curled	low
A5	blurry	green	curled	high
A6	clear	black	slightly curled	high

the heterogeneous couplings for supervised learning, no existing methods on similarity, metric and representation learning effectively handle the above challenges in unsupervised categorical representation, which is critical yet challenging for understanding the intrinsic data complexities in unlabeled categorical data. Embedding lookup table and deep representation learning methods [1] such as one-hot embedding, word embedding, autoencoder [23], adversarial learning [24], and deep models such as wide and deep model [25] and auto-instructor MAI [26] significantly outperform shallow methods in capturing latent features and relations. However, their common approaches and advantages are built on simplifying and equally treating input (e.g., by a one-hot encoder), involving a special modeling mechanism or structure, ignoring or disentangling complicated couplings, and a deep abstraction of large data with high computational power. They struggle in representing small yet complex unlabeled categorical data and also ignore the semantics and other diverse explicit characteristics of categorical values, attributes and objects, critical for categorical data representation and learning [4], [8], [10].

In this paper, we build a shallow but powerful UNsupervised heTerogeneous couplIng lEarning (UNTIE, for short) approach to learning heterogeneous and hierarchical couplings that may be complementary yet inconsistent in small unlabeled categorical data with complex data characteristics. As the first attempt, UNTIE simultaneously represents (1) diverse value-to-object couplings,(2) complex relations between heterogeneous and hierarchical couplings, and (3) heterogeneous distributions of respective couplings by unsupervised multikernel learning. Specifically, complex relations are entangled by the nonlinear mapping of various kernelized coupling functions, and the heterogeneous distributions are sensitively modeled by different kernels respectively. Instead of directly combining heterogeneous couplings, UNTIE first re-models the diverse couplings by multiple kernels to transform the various couplings-based spaces to respective kernelized representation spaces of higher dimensionality. Then, UNTIE learns both the weight of each attribute value in an individual kernel space and the weights of the learned kernel spaces to reflect both the heterogeneous data distributions of respective couplings and the interactions between couplings. Further, to efficiently learn heterogeneous couplings, UNTIE seamlessly wraps the kernel space weights by a positive semi-definite kernel and optimizes this kernel by optimizing an unsupervised kernel k-means objective. Lastly, the optimized kernel is used as the similarity representation of categorical data to generate a vector representation by further decomposing this kernel. We provide theoretical analysis (see Theorem 5.3) to show UNTIE can represent categorical data with maximizing the separability for further learning tasks.

Accordingly, this work delivers the following significant con-

tributions to categorical representation, unsupervised representation, and coupling learning.

- UNTIE is the first unsupervised categorical representation method to learn various value-to-object couplings and their complementarity and inconsistency. UNTIE collectively captures heterogeneous data distributions of diverse couplings and adaptively integrates the couplings by involving their interactions.
- UNTIE maps the heterogeneous intra- and inter-attribute couplings into multiple kernels to capture the coupling heterogeneity (§4.3). In the kernel spaces, learning heterogeneous couplings is formalized as an efficient unsupervised optimization problem by optimizing an UNTIE-enabled kernel k-means objective (§4.4).
- UNTIE works in a completely unsupervised fashion to capture the intrinsic data characteristics in categorical data with both theoretical and experimental verification. Theoretical analysis is provided that shows the UNTIE-represented data has the minimum normalized cut and increases data separability (§5).

We substantially verify the UNTIE effectiveness, representation quality, efficiency, flexibility and stability on 25 real-life categorical data sets with diversified data characteristics (including multi-dimensional, multi-class and multi-valued objects) and four synthetic data sets generated per a variety of data factors. UNTIE is compared with vector, similarity, embedding and deep representation methods. (1) UNTIE can effectively address both complementarity and inconsistency in learning heterogeneous couplings. (2) UNTIE enjoys accuracy gain (up to 51.72% in terms of F-score on these data sets) from the learned heterogeneity and produces substantially better representation performance than the state-of-the-art shallow and deep categorical representation methods. (3) The efficiency of UNTIE is insensitive to the volume of data, which indicates UNTIE is scalable for large data. (4) The learned UNTIE representations can enhance different downstream learning tasks. This work also shows that shallow learning does not lose the ground under handling complex (small or large) data to deep models, particularly for unsupervised settings.

2 RELATED WORK

The quality of categorical data representations affects the performance of representations-based learning tasks. Categorical representations are determined by how well a representer captures the various value-to-object coupling relationships and their heterogeneities within and between categorical values, attributes and objects [2], [10], [19]. For example, embedding methods like one-hot embedding and word embedding only encode the existence of a value or the IDF-based textual vector to a vector space. Matching-based methods treat categorical values equally and overlook their rich differences.

A recent effort made for categorical representation is the coupling learning of complex interactions and relations [10] and demonstrates great potential in (1) intra-attribute couplings-based representations [19], [27] and (2) inter-attribute couplings-based representations [2], [13], [15], [28]. The former reveals the way and degree that values are coupled within an attribute. For example, the method in [19] adopts the conditional probability of the attribute values of an object w.r.t. the attribute cluster centers to represent categorical data, and the method in [27]

introduces set theory for measuring intra-attribute value similarity to represent categorical data. The latter captures the way and degree that attributes are coupled. They typically measure the inter-attribute couplings w.r.t. the conditional probabilities [13], [28] or co-occurrence frequencies [15] between values of different attributes. These two groups of representations outperform other classic methods such as matching-based as they capture richer interactions in categorical data. However, most of such work only considers a single type of couplings and overlooks many other characteristics in categorical data.

The work in [16] shows that representing more couplings in categorical data may significantly upgrade learning performance. However, other recent research also shows that capturing more but duplicated couplings does not guarantee better categorical representation as in [14]. A symmetric uncertainty (SU) criterion enables better representation performance in several categorical data representation methods with multiple couplings [4], [14], [29]. Alternatively, the method in [17] uses the principal component analysis to reduce the redundancy between couplings. These methods make performance gain by reducing redundant couplings but ignore the inconsistency between heterogeneous couplings with diverse interactions and distributions, i.e., heterogeneity [20] of different couplings. This issue was studied in [8] by capturing hierarchical couplings to enhance categorical data representation with label information. None of existing unsupervised categorical representation methods explicitly and effectively model heterogeneous couplings, which brings about significant challenge to representation learning, as explored in this work.

Deep representation learning presents increasingly promising power in representing images, text, networks, etc. [1]. However, unsupervised categorical representation learning has not been well explored and presents a challenge to deep learning. Existing methods typically convert categorical input into a vector space through encoding such as one-hot encoding and word embedding and then rely on a deep neural architecture such as autoencoder [23], adversarial learning (e.g., BiGAN and GAN variants, wide and deep network, and variational autoencoder) [24], [25], [30] or auto-instructor [17], [26] to learn the hidden relations and features. Such methods overlook or simplify most of data characteristics of categorical data (e.g., value semantics, frequencies, attribute interactions, distributions, etc.), decouple and disentangle couplings [1], and rely on high computational power and large (partially labeled) data. They are troubled by small, unlabeled, and complicatedly coupled categorical data.

3 PRELIMINARIES

Assume a categorical data set drawn from distributions Φ can be represented as a three-element tuple $C =< O, A, V >$, where $O = \{o_i | i \in N_o\}$ is an object set with n_o objects; $A = \{a_i | i \in N_a\}$ is an attribute set with n_a attributes; and $V = \bigcup_{j=1}^{n_a} V^{(j)}$ is the collection of attribute values with n_v values, in which $V^{(j)} = \{v_i^{(j)} | i \in N_v^{(j)}\}$ is the set of attribute values $v_i^{(j)}$ with $n_v^{(j)}$ values of attribute a_j. N_o, N_a and $N_v^{(j)}$ are the sets of indices for objects, attributes, and values of the j-th attribute, respectively. For the i-th object o_i, the categorical value in the j-th attribute a_j can be represented as $v_i^{(j)}$. The main notations in this paper are defined

in Table 2 [1]. For example, in Table 1, $O = \{A1, A2, A3, A4, A5, A6\}$, $A = \{\text{Texture, Color, Root Shape}\}$, $V = \{\text{clear, blurry, white, yellow, green, black, straight, curled, slightly curled}\}$, $V^{(1)} = \{\text{clear, blurry}\}$, and $v_2^{(3)} = \text{straight}$.

TABLE 2
List of Notations

Symbol	Meaning
Φ	Categorical data distributions
C	Categorical data tuple
O	Object set
A	Attribute set
V	Categorical value set of all attributes
$V^{(j)}$	Categorical value set of the j-th attribute
o_i	The i-th object in O
a_i	The i-th attribute in A
v_i	The i-th categorical value in V
$v_i^{(j)}$	The categorical value of o_i in a_j
$v_i^{(j)}$	The i-th categorical value in $V^{(j)}$
m_i	The vector corresponding to i-th value in a coupling space
c_k	The k-th cluster
ω	The heterogeneity parameter
n_o	The number of objects in O
n_a	The number of attributes in A
n_v	The number of categorical values in V
$n_v^{(j)}$	The number of categorical values in $V^{(j)}$
n_k	The number of kernel matrices transformed from coupling spaces
n_c	The number of clusters
n_{oc}	The size of c-th cluster
n_{mv}	The maximal number of values in attributes
n_{av}	The average number of attribute values
n_ω	The number of elements in ω
n_i	The number of iterations
n_b	The training batch size
$n_m^{(z)}$	The number of couplings for the z-th attribute
N_o	The set of indices for objects in O
N_a	The set of indices for attributes in A
$N_v^{(j)}$	The set of indices for categorical values in $V^{(j)}$
$\mathcal{M}_{Ia}$	Intra-attribute coupling spaces
$\mathcal{M}_{Ie}$	Inter-attribute coupling spaces
$\mathcal{K}_p$	The p-th kernel space transformed from a coupling space
$\mathcal{K}_p'$	The heterogeneous kernel space transformed from $\mathcal{K}_p$
$\mathbf{K}_p$	The kernel matrix that spans $\mathcal{K}_p$
$\mathbf{K}_p'$	The kernel matrix that spans $\mathcal{K}_p'$
$\mathbf{T}_p$	The transformation matrix from $\mathcal{K}_p$ to $\mathcal{K}_p'$
$\mathbf{C}^{(z,k)}$	The k-th coupling matrix of the z-th attribute
α_{pi}	The weight of the i-th value in the p-th kernel space
β_p	The weight of the p-th kernel space
$\mathbf{S}$	UNTIE similarity representation
$\mathbf{X}$	UNTIE vector representation

4 THE UNTIE DESIGN

UNTIE learns unsupervised categorical representation based on the rationale below: (1) a categorical value may belong to multiple distributions; (2) a coupling may make different contributions to different value distributions; and (3) the overall distribution of a categorical value can be described by multiple distributions. We call the above *heterogeneity hypotheses*, which are theoretically supported by Theorem 5.1 in Section 5.1.

---

1. The specifications for symbol styles in this paper are as follows. Element: lowercase with Sans Serif font; value: lowercase; vector: lowercase with bold font; matrix: uppercase with bold font; set: uppercase; function: lowercase with parentheses; space: uppercase with Calligraphic font; value index: subscript; attribute index: superscript with parenthesis.

4.1 The UNTIE Framework

While this paper focuses on a specific instance of UNTIE, as shown in Fig. 1, UNTIE actually presents a framework of unsupervised categorical representation. It represents categorical data in both vector (as a vector representation) and similarity (as a kernel matrix) spaces. To reveal heterogeneous couplings, UNTIE first converts categorical data to several coupling spaces by multiple coupling learning functions. Then, it feeds and transforms each coupling space to multiple kernel spaces. Further, it reduces the redundancy and inconsistency between heterogeneous couplings by learning the heterogeneity between couplings in the kernel spaces. Specifically, UNTIE differentiates the contributions of individual kernel spaces and reveals the kernel-sensitive distribution within each kernel space. To efficiently learn the heterogeneities in an unsupervised way, UNTIE wraps the weight of each kernel space and the weights of the values embedded in a kernel space by a wrapper kernel. It then optimizes this kernel by solving a kernel k-means objective, i.e., regularizing the objects within one cluster to be more similar to each other than to those in others. UNTIE uses the optimized wrapper kernel as the similarity representation of categorical data, and further generates the vector representation by decomposing the optimized wrapper kernel.

To effectively address the coupling inconsistency problem, UNTIE learns heterogeneous couplings by multiple kernels, which transforms a coupling from its original space to several kernel spaces. Since a kernel is sensitive to a distribution [31], these kernel spaces reflect different value distributions. In a kernel space, a coupling is preserved if it matches the kernel-sensitive distribution while other couplings are filtered. For this, UNTIE learns the weights of values in each kernel space to effectively reveal the multiple distributions corresponding to a coupling. The multiple distributions of a categorical value jointly contribute to the overall distribution of the value. Therefore, the learned weights of these kernel spaces filter the redundant information but integrate the complementary information.

To learn the heterogeneous couplings in an unsupervised manner, without loss of generality, UNTIE takes the assumption that the objects within one cluster are more similar to each other than to those in other clusters. This assumption is commonly taken in most of clustering and classification methods and shows its validity with real data distributions. Accordingly, UNTIE learns heterogeneous couplings for categorical data representation in an iterative way. In each iteration, UNTIE first analyzes the clusters based on its generated representation, and then tunes the representation based on the obtained clusters. To efficiently cluster data, UNTIE wraps the weight of each kernel space and the weight of the values embedded in kernel spaces by a wrapper kernel and optimizes this kernel by solving a kernel k-means objective. In addition to efficiency, the kernel k-means objective also brings other benefits for categorical data representation, such as good separability, which will be discussed in Section 5.3.

4.2 Coupling Learning

As discussed in Introduction, we broadly refer *couplings* to any interactions and relations within and between values, attributes, and objects [4], [8], [10]. In categorical data, possible types of couplings include *intra-attribute couplings* (i.e., value couplings, such as per value frequency, co-occurrence and matching), inter-attribute couplings (e.g., attribute correlation and dependency and unknown linkage), and object couplings built on the value and

attribute couplings. UNTIE learns the value-to-attribute-to-object hierarchical couplings after learning and fusing various intra- and inter-attribute couplings in unlabeled categorical data.

Learning Intra-attribute Couplings. *Intra-attribute couplings* represent the interactions between the values of an attribute and the value distributions in an attribute [2], [32]. We measure intra-attribute couplings in terms of intra-attribute distributions by a value frequency function and calculate the Euclidean distance in a numerical space. Although the value frequency function has only one input value, it measures the value distribution against all values. For a categorical value $\mathsf{v}_i^{(j)}$ in the j-th attribute, the value frequency function $m_{Ia}^{(j)}(\mathsf{v}_i^{(j)})$ maps an intra-attribute coupling between this value and the other categorical values in this attribute to a one-dimensional intra-attribute coupling vector $m_{Ia}^{(j)}(\mathsf{v}_i^{(j)})$.

$$m_{Ia}^{(j)}(\mathsf{v}_i^{(j)}) = \big[\frac{|g^{(j)}(\mathsf{v}_i^{(j)})|}{n_o}\big], \tag{1}$$

where $g^{(j)}(\cdot) : V^{(j)} \to O$ maps the value $\mathsf{v}_i^{(j)}$ to a set of objects that have value $\mathsf{v}_i^{(j)}$ in the j-th attribute, n_o is the number of objects, and $|\cdot|$ refers to the count of a set. For example, in Table 1, a relationship between value *yellow* and object attribute *color* is $g^{(j)}(yellow) = \{A2, A3\}$. The intra-attribute coupling vector of value *yellow* is $m_{Ia}^{(2)}(yellow) = [\frac{|\{A2, A3\}|}{6}] = [\frac{1}{3}]$.

An intra-attribute coupling space $\mathcal{M}_{Ia}^{(j)}$ is spanned by the intra-attribute coupling vectors obtained in an attribute by Eq. (1) and is defined below:

$$\mathcal{M}_{Ia}^{(j)} = \{m_{Ia}^{(j)}(\mathsf{v}_i^{(j)})|\mathsf{v}_i^{(j)} \in V_j\}. \tag{2}$$

For categorical data with n_a attributes, the intra-attribute coupling spaces are $\mathcal{M}_{Ia} = \{\mathcal{M}_{Ia}^{(1)}, \cdots, \mathcal{M}_{Ia}^{(n_a)}\}$. The intra-attribute coupling spaces only present a one-dimensional embedding of the categorical data space w.r.t. each attribute. The following inter-attribute couplings consider the interactions between attributes.

Learning Inter-attribute Couplings. *Inter-attribute couplings* refer to the interactions between attributes and the contextual (and/or semantic) information of attribute values w.r.t. other attributes [10], [14], [33]. This attribute-based interactive and contextual information complements the value distributions and interactions captured by intra-attribute couplings. For example, in Table 1, white and black watermelons have the same frequency but can be distinguished by involving their *root shapes* which are significantly different.

Here, the inter-attribute couplings are represented by the information conditional probability, which reveals the distributions of an attribute value in the spaces spanned by the values of the other attributes. Given a value $\mathsf{v}^{(j)}$ of attribute a_j and a value $\mathsf{v}^{(k)}$ of attribute a_k, the information conditional probability function is defined as follows:

$$p(\mathsf{v}^{(j)}|\mathsf{v}^{(k)}) = \frac{|g^{(j)}(\mathsf{v}^{(j)}) \cap g^{(k)}(\mathsf{v}^{(k)})|}{|g^{(k)}(\mathsf{v}^{(k)})|}, \tag{3}$$

where $\cap$ returns the intersection of two sets. Based on the information conditional probability function, the inter-attribute coupling learning function $m_{Ie}^{(j)}(\mathsf{v}_i^{(j)})$ embeds interactions between value $\mathsf{v}_i^{(j)}$ and other attributes as a $|V_*|$-dimensional inter-attribute coupling vector $m_{Ie}^{(j)}(\mathsf{v}_i^{(j)})$,

$$m_{Ie}^{(j)}(\mathsf{v}_i^{(j)}) = \big[\; p(\mathsf{v}_i^{(j)}|\mathsf{v}_{*1}), \quad \cdots, \quad p(\mathsf{v}_i^{(j)}|\mathsf{v}_{*|V_*|}) \; \big]^\top, \tag{4}$$
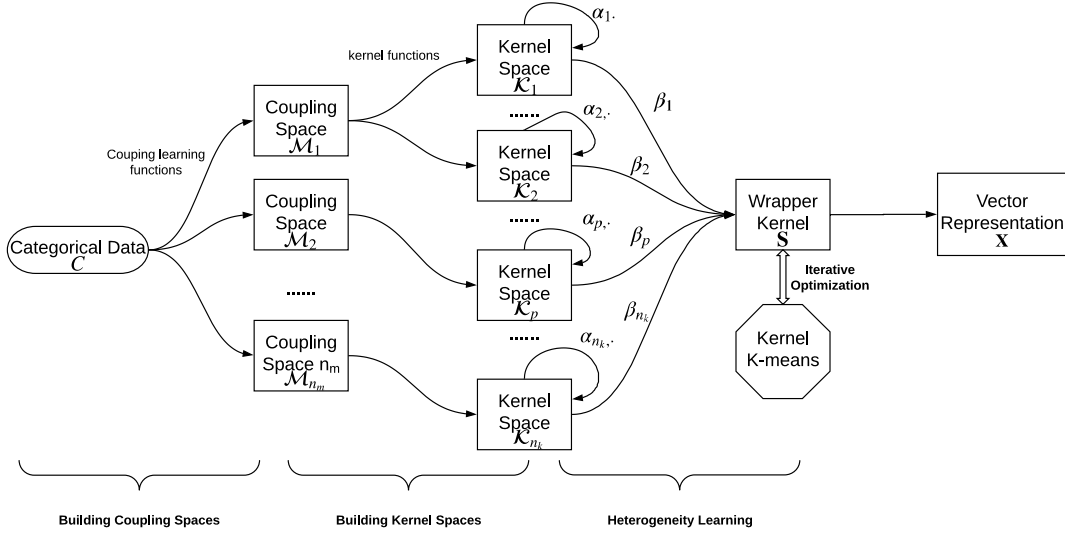
Fig. 1. The UNTIE Framework. It first transforms the coupling spaces to multiple kernel spaces and then learns the heterogeneities within and between couplings in these kernel spaces by solving a kernel k-means objective.

where $V_* = \{V^{(k)}|k \in N_a, k \neq j\}$ is the set of values in all attributes except a_j, and $\mathsf{v}_{*i} \in V_*$ is a categorical value in set V_*. For example, the information condition probability between the yellow watermelons and those with curled root shape is $p(yellow|curled) = \frac{|\{A2,A3\} \cap \{A3,A5\}|}{|\{A3,A5\}|} = \frac{|\{A3\}|}{|\{A3,A5\}|} = \frac{1}{2}$. The inter-attribute coupling vector of value *yellow* is calculated as

$$m_{Ie}^{(2)}(yellow)$$
$$= [p(yellow|clear), \cdots, p(yellow|slightly\ curled)]$$
$$= [0, \frac{2}{3}, \frac{1}{2}, \frac{1}{2}, 0]$$

An inter-attribute coupling space $\mathcal{M}_{Ie}^{(j)}$ is spanned by the inter-attribute coupling vectors obtained in an attribute by Eq. (4):

$$\mathcal{M}_{Ie}^{(j)} = \{m_{Ie}^{(j)}(\mathsf{v}_i^{(j)})|\mathsf{v}_i^{(j)} \in V^{(j)}\}. \tag{5}$$

For categorical data with n_a attributes, the inter-attribute coupling spaces $\mathcal{M}_{Ie} = \{\mathcal{M}_{Ie}^{(1)}, \cdots, \mathcal{M}_{Ie}^{(n_a)}\}$.

An inter-attribute coupling learning function projects categorical values into a higher dimensional space if $|V| > 2|V^{(j)}| - 1$, because the dimensionality of inter-attribute coupling space equals $|V| - |V^{(j)}|$, while the degree of freedom (equivalent to the dimensionality of transforming a categorical value to a dummy variable) of the j-th attribute is $|V^{(j)}| - 1$. In this way, the value couplings incurred by other attributes are captured, which complements with the intra-attribute couplings to form a complete representation of categorical attribute space.

4.3 Heterogeneity Learning in Kernel Spaces

With the coupling spaces built above from the intra- and inter-attribute perspectives, UNTIE further constructs an entire coupling space $\mathcal{M}$ which is a collection of heterogeneous intra-attribute coupling spaces $\mathcal{M}_{Ia}$ and inter-attribute coupling spaces $\mathcal{M}_{Ie}$,

$$\mathcal{M} = \mathcal{M}_{Ia} \cup \mathcal{M}_{Ie}. \tag{6}$$

To effectively integrate the heterogeneous couplings in the learned coupling space set, UNTIE transforms the learned heterogeneous coupling spaces to uniform spaces, in which heterogeneous couplings are comparable. Specifically, UNTIE uses multiple kernels to transform each coupling space into its corresponding kernel spaces, where each kernel space corresponds to the transformed coupling space w.r.t. a particular kernel mapping function. It generates a set of n_k ($n_k = |\mathcal{M}| \times |F|$, where F is the set of kernel functions for the transformation) kernel spaces $\{\mathcal{K}_1, \mathcal{K}_2, \cdots, \mathcal{K}_{n_k}\}$, and the p-th ($p \leq n_k$) space is spanned by a kernel matrix $\mathbf{K}_p$, which is constructed from a coupling space $\mathcal{M}_j$ by a kernel function $k_p(\cdot, \cdot)$ for an attribute. Denoting m_i as a vector in $\mathcal{M}_j$ corresponding to the i-th categorical value, $\mathbf{K}_p$ is represented as follows,

$$\mathbf{K}_p = \begin{bmatrix} k_p(\mathsf{m}_1, \mathsf{m}_1) & k_p(\mathsf{m}_1, \mathsf{m}_2) & \cdots & k_p(\mathsf{m}_1, \mathsf{m}_{n_v^*}) \\ k_p(\mathsf{m}_2, \mathsf{m}_1) & k_p(\mathsf{m}_2, \mathsf{m}_2) & \cdots & k_p(\mathsf{m}_2, \mathsf{m}_{n_v^*}) \\ \vdots & \vdots & \ddots & \vdots \\ k_p(\mathsf{m}_{n_v^*}, \mathsf{m}_1) & k_p(\mathsf{m}_{n_v^*}, \mathsf{m}_2) & \cdots & k_p(\mathsf{m}_{n_v^*}, \mathsf{m}_{n_v^*}) \end{bmatrix}, \tag{7}$$

where n_v^* is the number of categorical values represented by $\mathcal{M}_j$. For example, in Table 1, if $k_p(\cdot, \cdot)$ is a linear kernel and $\mathcal{M}_j$ is $\mathcal{M}_{Ie}^{(2)}$, let m_2 correspond to value $yellow$, $k_p(\mathsf{m}_2, \mathsf{m}_2) = [0, \frac{2}{3}, \frac{1}{2}, \frac{1}{2}, 0]^\top \cdot [0, \frac{2}{3}, \frac{1}{2}, \frac{1}{2}, 0] = \frac{17}{18}$.

To reveal the heterogeneity within a coupling, UNTIE learns the weights of values in each kernel space. Specifically, it learns a set of transformation matrices $\{\mathbf{T}_1, \mathbf{T}_2, \cdots, \mathbf{T}_{n_k}\}$ to reconstruct the kernel spaces $\{\mathcal{K}_1', \cdots, \mathcal{K}_{n_k}'\}$, in which the p-th kernel matrix $\mathbf{K}_p'$ only contains the p-th kernel sensitive distribution that suits for the corresponding coupling. We call the reconstructed kernel spaces *heterogeneous kernel spaces*. $\mathbf{K}_p'$ is defined as:

$$\mathbf{K}_p' = \mathbf{T}_p \cdot \mathbf{K}_p. \tag{8}$$

UNTIE regulates $\mathbf{T}_p$ as a diagonal matrix:

$$\mathbf{T}_p = \begin{bmatrix} \alpha_{p1} & 0 & \cdots & 0 \\ 0 & \alpha_{p2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \alpha_{pn_v} \end{bmatrix}. \tag{9}$$

As a result, α_{pi} is the weight of the i-th value in the p-th kernel space, i.e., $[k_p(\mathsf{m}_1, \mathsf{m}_1), k_p(\mathsf{m}_1, \mathsf{m}_2), \cdots, k_p(\mathsf{m}_1, \mathsf{m}_{n_v^*})]$. The larger α_{pi} implies stronger coupling of the i-th value revealed by the coupling space corresponding to the p-th kernel space.

To further capture the heterogeneity between couplings, UN-TIE learns the contribution of each heterogeneous kernel space to a final representation. It first defines a similarity measure between objects in the heterogeneous kernel space and then learns the weight of each kernel space based on this similarity measure to reflect their contribution. Given a categorical data set, considering the p-th kernel matrix, let i and j represent the indices of values in the p-th kernel space corresponding to the i-th and j-th objects respectively, and using $\mathbf{K}'_{p,\mathsf{i}\cdot}$ (the i-th row in $\mathbf{K}'_p$) to denote o_i in the p-th heterogeneous kernel space, the similarity $S_{p,ij}$ measured by the linear kernel of the i-th and j-th objects in this space is

$$S_{p,ij} = \mathbf{K}'^{\top}_{p,\mathsf{i}\cdot} \mathbf{K}'_{p,\mathsf{j}\cdot}. \tag{10}$$

By considering Eq. (8), Eq. (10) equals

$$S_{p,ij} = \mathbf{K}^{\top}_{p,\mathsf{i}\cdot} \mathbf{T}_p^{\top} \mathbf{T}_p \mathbf{K}_{p,\mathsf{j}\cdot}. \tag{11}$$

UNTIE defines the final similarity representation S_{ij} between the i-th and j-th objects as a linear combination of base similarity measures from heterogeneous spaces to filter redundant information and integrate complementary information between couplings:

$$S_{ij} = \sum_{p=1}^{n_k} \beta_p S_{p,ij}, \tag{12}$$

where $\beta_p \geq 0$ is the weight for the p-th base similarity. Denoting a diagonal matrix $\boldsymbol{\omega}_p = \beta_p \mathbf{T}^{\top}\mathbf{T}$, Eq. (12) is rewritten as:

$$S_{ij} = \sum_{p=1}^{n_k} \mathbf{K}^{\top}_{p,\mathsf{i}\cdot} \boldsymbol{\omega}_p \mathbf{K}_{p,\mathsf{j}\cdot}. \tag{13}$$

Accordingly, UNTIE simultaneously learns α and β by learning $\boldsymbol{\omega}$, a heterogeneity parameter. The optimized $\boldsymbol{\omega}$ guides to integrate the heterogeneous couplings into the similarity representation S_{ij}.

4.4 Kernel K-means-based Representation Learning

In an unsupervised way, UNTIE learns the heterogeneous couplings by wrapping α which reveals the heterogeneity within couplings and β which reveals the heterogeneity between couplings into a wrapper kernel. It then further optimizes this kernel by solving a kernel k-means objective [34].

K-means is a popular clustering algorithm that minimizes the distance between an object and its assigned cluster center, which was also used for information integration. Given a set of objects $O = \{\mathsf{o}_i \in \mathcal{R}^{n_a} | i = 1, \cdots, n_o\}$, the k-means objective is formalized as:

$$\underset{\mathbf{Z} \in \{0,1\}^{n_o \times n_c}}{\text{minimize}} \quad \sum_{i=1,c=1}^{n_o,n_c} z_{ic} \|\mathsf{o}_i - \boldsymbol{\mu}_c\|_2^2$$
$$\text{subject to} \quad \sum_{c=1}^{n_c} z_{ic} = 1, \tag{14}$$

where z_{ic} indicates whether o_i belongs to the c-th cluster, $\boldsymbol{\mu}_c = \frac{1}{n_{oc}} \sum_{i=1}^{n_o} z_{ic}\mathbf{x}_i$ is the centroid of the c-th cluster, and $n_{oc} = \sum_{i=1}^{n_o} z_{ic}$ refers to the size of the c-th cluster.

To address the issue that k-means cannot cluster data with a nonlinear boundary, the kernel k-means first uses a mapping function to map data to a higher dimensional space and then adopts k-means to cluster the mapped data. With a kernel function $k(\cdot)$, the kernel k-means is formalized as:

$$\underset{\mathbf{Z} \in \{0,1\}^{n_o \times n_c}}{\text{minimize}} \quad \sum_{i=1,c=1}^{n_o,n_c} z_{ic} \|k(\mathsf{o}_i) - \boldsymbol{\mu}_c\|_2^2$$
$$\text{subject to} \quad \sum_{c=1}^{n_c} z_{ic} = 1, \tag{15}$$

where $\boldsymbol{\mu}_c = \frac{1}{n_{oc}} \sum_{i=1}^{n_o} z_{ic} k(\mathsf{o}_i)$.

Eq. (15) is rewritten in the following form:

$$\underset{\mathbf{Z} \in \{0,1\}^{n_o \times n_c}}{\text{minimize}} \quad \text{Tr}(\mathbf{K}) - \text{Tr}(\mathbf{L}^{\frac{1}{2}} \mathbf{Z}^{\top} \mathbf{K} \mathbf{Z} \mathbf{L}^{\frac{1}{2}})$$
$$\text{subject to} \quad \mathbf{Z}\mathbf{1}_{n_c} = \mathbf{1}_{n_o}, \tag{16}$$

where $\text{Tr}(\cdot)$ calculates the trace of a matrix, $\mathbf{K}$ is a matrix with $k_{ij} = k(\mathsf{o}_i^{\top})k(\mathsf{o}_j)$, $\mathbf{L} = \text{diag}([n_{o1}^{-1}, n_{o2}^{-1}, \cdots, n_{on_c}^{-1}])$ and $\mathbf{1}_{\ell} \in \{1\}^{\ell}$ is a column vector with all elements being 1.

Directly solving Eq. (16) is difficult since the values of $\mathbf{Z}$ are limited to either 0 or 1. Typically, Eq. (16) is relaxed by letting $\mathbf{Z}$ take real values. Denoting $\mathbf{H} = \mathbf{Z}\mathbf{L}^{\frac{1}{2}}$, the above problem is restated as

$$\underset{\mathbf{H}}{\text{minimize}} \quad \text{Tr}(\mathbf{K}(\mathbf{I}_{n_o} - \mathbf{H}\mathbf{H}^{\top}))$$
$$\text{subject to} \quad \mathbf{H} \in \mathcal{R}^{n_o \times n_c},$$
$$\mathbf{H}^{\top}\mathbf{H} = \mathbf{I}_{n_c}, \tag{17}$$

where $\mathbf{I}_{n_c}$ is an identity matrix with size $n_c \times n_c$. The optimal $\mathbf{H}$ for Eq. (17) can be obtained by taking the n_c eigenvectors having large eigenvalues of $\mathbf{K}$ [35].

UNTIE integrates heterogeneous coupling learning into the kernel k-means seamlessly by wrapping α and β to a wrapper kernel $s(\cdot, \cdot) : O \times O \to \mathcal{R}$, which is defined below,

$$s(\mathsf{o}_i, \mathsf{o}_j) = S_{ij}. \tag{18}$$

Accordingly, UNTIE constructs a kernel matrix $\mathbf{S}$ w.r.t. kernel $s(\cdot, \cdot)$ and categorical object set O:

$$\mathbf{S} = \begin{bmatrix} s(\mathsf{o}_1, \mathsf{o}_1) & s(\mathsf{o}_1, \mathsf{o}_2) & \cdots & s(\mathsf{o}_1, \mathsf{o}_{n_o}) \\ s(\mathsf{o}_2, \mathsf{o}_1) & s(\mathsf{o}_2, \mathsf{o}_2) & \cdots & s(\mathsf{o}_2, \mathsf{o}_{n_o}) \\ \vdots & \vdots & \ddots & \vdots \\ s(\mathsf{o}_{n_o}, \mathsf{o}_1) & s(\mathsf{o}_{n_o}, \mathsf{o}_2) & \cdots & s(\mathsf{o}_{n_o}, \mathsf{o}_{n_o}) \end{bmatrix}. \tag{19}$$

Since $s(\cdot, \cdot)$ is proved as a valid positive semi-definite kernel (see details in Section 5.2), $\mathbf{S}$ can replace $\mathbf{K}$ in Eq. (17). In this way, the objective function of kernel k-means-based representation learning can be formalized as:

$$\underset{\mathbf{H}, \boldsymbol{\omega}}{\text{minimize}} \quad \text{Tr}(\mathbf{S}(\mathbf{I}_{n_o} - \mathbf{H}\mathbf{H}^{\top}))$$
$$\text{subject to} \quad \mathbf{H} \in \mathcal{R}^{n_o \times n_c},$$
$$\mathbf{H}^{\top}\mathbf{H} = \mathbf{I}_{n_c}, \tag{20}$$

where $\boldsymbol{\omega}$ is a heterogeneity parameter to learn, and we obtain the similarity representation of categorical data as $\mathbf{S}$. The corresponding vector representation can be obtained by

$$\mathbf{x}_i = [\sqrt{\boldsymbol{\omega}_{1,11}}\mathbf{K}_{1,\mathsf{i}1}, \sqrt{\boldsymbol{\omega}_{1,22}}\mathbf{K}_{1,\mathsf{i}2}, \cdots, \sqrt{\boldsymbol{\omega}_{n_k,n_v^* n_v^*}}\mathbf{K}_{n_k,\mathsf{i}n_v^*}], \tag{21}$$

where $\boldsymbol{\omega}_{i,jj}$ refers to the value of the (j, j)-th entry in $\boldsymbol{\omega}_i$ and n_v^* refers to the number of values in the attribute corresponding to the n_k-th kernel. The learned representation $\mathbf{x}_i$ is a numerical approximation of categorical data, which can be fed into vector-based learning methods.

4.5 The UNTIE Algorithm

The UNTIE objective function in Eq. (20) can be solved by alternatively updating $\mathbf{H}$ and $\boldsymbol{\omega}$: (1) Optimizing $\mathbf{H}$ given $\boldsymbol{\omega}$: by fixing the parameter $\boldsymbol{\omega}$, $\mathbf{H}$ can be obtained by solving a kernel k-means clustering optimization problem shown in Eq. (17) by eigenvalue decomposition; (2) Optimizing $\boldsymbol{\omega}$ given $\mathbf{H}$: with H fixed, the objective function of learning $\boldsymbol{\omega}$ is

$$\underset{\boldsymbol{\omega}}{\text{minimize}} \quad \text{Tr}(\mathbf{S}(\mathbf{I}_{n_o} - \mathbf{H}\mathbf{H}^\top)), \tag{22}$$

which can be optimized by linear programming. For large-scale data, Eq. (22) can be solved by the stochastic gradient descent (SGD) method, e.g., AdaGrad [36] and Adam [37]. We analyze the computational cost of UNTIE w.r.t different optimization methods in Section 5.5. Algorithm 1 explains the UNTIE working process.

Algorithm 1 The UNTIE Algorithm for Unsupervised Categorical Representation

---

Input: Categorical data set C, a set of kernel functions $K = \{k_1(\cdot, \cdot), \cdots, k_{n_k^*}(\cdot, \cdot)\}$, the number of clusters n_c, and convergence rate δ.
Output: Similarity representation $\mathbf{S}$, and vector representation $\mathbf{X}$.
1: Mapping categorical data to coupling spaces according to Eqs. (2) and (5).
2: Mapping coupling spaces to multiple kernel spaces $\{\mathbf{K}_1, \cdots, \mathbf{K}_{n_k}\}$ by using K according to Eq. (7).
3: Initializing the wrapper kernel matrix $\mathbf{S}$ by setting α and β as 1, and setting $l' = +\infty$ and $\Delta = +\infty$.
4: **for** $\Delta > \delta$ **do**
5:     Calculating the n_c eigenvectors that have the largest eigenvalues of $\mathbf{S}$. Constructing $\mathbf{H}$ by these eigenvectors.
6:     Optimizing $\boldsymbol{\omega}$ by solving Eq. (22).
7:     Calculating loss per $l = \text{Tr}(\mathbf{S}(\mathbf{I}_{n_o} - \mathbf{H}\mathbf{H}^\top))$.
8:     Calculating loss change per $\Delta = |l - l'|$
9:     Setting $l' = l$.
10:     $n_i = n_i + 1$.
11: **end for**
12: Calculating the vector representation $\mathbf{X}$ per Eq. (21).
13: **return** $\mathbf{S}$, $\mathbf{X}$

---

5 THEORETICAL ANALYSIS

5.1 The Fitness of Heterogeneity Hypotheses

To discuss the fitness of heterogeneity hypotheses, we first introduce the following theorem.

Theorem 5.1. *The distribution Φ of a categorical data set can be described as a probability tensor $\boldsymbol{\Phi}$, where each entry corresponds to the joint probability of a set of categorical values from n_a different attributes. $\boldsymbol{\Phi}$ can be decomposed as $\boldsymbol{\Phi} = \sum_{h=1}^{k} \pi_h \Theta_h$, where $\Theta_h = \boldsymbol{\theta}_h^{(1)} \otimes \boldsymbol{\theta}_h^{(2)} \otimes \cdots \otimes \boldsymbol{\theta}_h^{(n_a)}$, π_h is the weight of Θ_h for composing $\boldsymbol{\Phi}$, $\otimes$ refers to the outer product, $\boldsymbol{\theta}_h^{(j)}$ is a probability vector of categorical values in the j-th attribute with a size of $n_v^{(j)} \times 1$ for $h = 1, \cdots, k$ and $j = 1, \cdots, n_a$.*

Theorem 5.1 can be proved by Corollary 1 in [38]. The categorical data distribution Φ is a joint distribution of categorical values in each attribute. It is defined as $\Phi = \{\phi_{v^{(1)}v^{(2)}\dots v^{(n_a)}} | v^{(j)} \in V^{(j)}\}$, where $\phi_{v^{(1)}v^{(2)}\dots v^{(n_a)}}$ are the probabilities of values $v^{(1)}, v^{(2)}, \cdots, v^{(n_a)}$ co-occur. Φ is a probability tensor, where each entry is an element in Φ. Theorem 5.1 indicates the following categorical data characteristics:

- *A value may belong to multiple distributions*. For different h values in Theorem 5.1, a categorical value in a_j have different distributions $\boldsymbol{\theta}_h^{(j)}$. Therefore, if $h > 1$, the categorical value may have different distributions.
- *A coupling may contribute differently to respective distributions*. The interactions under various distributions may differ. For distribution Θ_h, the attributes are independent (indicated by Θ_h which equals the outer product of attribute distributions). In this case, inter-attribute couplings may not make contribution. On the contrary, for distribution Φ, the attributes interact with each other, which is mainly reflected by inter-attribute couplings.
- *The overall distribution of a categorical value is a mixture of multiple distributions*. In Theorem 5.1, the overall distribution of a categorical value equals the weighted sum of its multiple distributions. The parameter π_h reflects the interactions between the distributions.

The heterogeneity hypotheses fit the above categorical data characteristics and provide a solid foundation for UNTIE to effectively capture the heterogeneity in couplings.

5.2 The Positive Semi-definite Wrapper Kernel

As stated in Section 4.4, $s(\cdot, \cdot)$ has to be a positive semi-definite kernel to enable that $\mathbf{S}$ can be integrated into kernel k-means objective Eq. (20). Before proving the above, we introduce a lemma of kernel properties.

Lemma 1. *If $k_1(\mathsf{o}_i, \mathsf{o}_j)$ and $k_2(\mathsf{o}_i, \mathsf{o}_j)$ are positive semi-definite kernels in $O \times O$, and a constant $a > 0$, then the following $k(\mathsf{o}_i, \mathsf{o}_j)$ functions are positive semi-definite kernels:*
    (1) $k(\mathsf{o}_i, \mathsf{o}_j) = ak_1(\mathsf{o}_i, \mathsf{o}_j)$,
    (2) $k(\mathsf{o}_i, \mathsf{o}_j) = k_1(\mathsf{o}_i, \mathsf{o}_j) + k_2(\mathsf{o}_i, \mathsf{o}_j)$.

This lemma can be found in Section 4.1 of [39].

Theorem 5.2. *The wrapper kernel $s(\mathsf{o}_i, \mathsf{o}_j) = S_{ij}$, which is defined in Eq. (18), is a positive semi-definite kernel.*

Proof. Given coupling spaces and multiple kernel functions, the i-th object o_i corresponds to a real value vector $\mathbf{K}'_{p,\mathrm{i}.} \in \mathcal{R}^{n_o}$ in the p-th kernel space. Therefore, $s_p(\mathsf{o}_i, \mathsf{o}_j) = S_{p,ij} = \mathbf{K}'^{\top}_{p,\mathrm{i}.}\mathbf{K}'_{p,\mathrm{j}.}$ is a well-known linear kernel, which is positive semi-definite. Treating $s_p(\mathsf{o}_i, \mathsf{o}_j)$ as $k_1(\mathsf{o}_i, \mathsf{o}_j)$ in Lemma 1, since $\beta_p \geq 0$, consequently, $\beta_p s_p(\mathsf{o}_i, \mathsf{o}_j)$ is a positive semi-definite kernel according to Formula (1) of Lemma 1. Consequently, as the wrapper kernel $s(\mathsf{o}_i, \mathsf{o}_j) = S_{ij} = \sum_{p=1}^{n_k} \beta_p S_{p,ij}$ is an accumulative summation of $\beta_p s_p(\mathsf{o}_i, \mathsf{o}_j)$, $s(\mathsf{o}_i, \mathsf{o}_j)$ is positive semi-definite by repeatedly adopting Formula (2) of Lemma 1 (treating $\sum_{p=1}^{q} \beta_p S_{p,ij}$ as $k_1(\mathsf{o}_i, \mathsf{o}_j)$, and $\beta_{q+1} S_{q+1,ij}$ as $k_2(\mathsf{o}_i, \mathsf{o}_j)$ for $1 \leq q \leq n_k$). $\square$

Theorem 5.2 guarantees that the kernel matrix $\mathbf{S}$, which is constructed by kernel $s(\mathsf{o}_i, \mathsf{o}_j)$ as Eq. (19), can be incorporated into the kernel k-means objective. This is a fundamental property to support the effective unsupervised learning by UNTIE.

5.3 The Separability of UNTIE-represented Data

The separability of a representation can be measured according to the overlap between object sets (e.g., clusters or classes) w.r.t. the representation. The UNTIE-represented data is with good separability since the resultant representation has the minimum normalized cut, which reflects the minimum overlap. Here, we first define the normalized cut and then prove that the objective of UNTIE is equivalent to learning a representation with the minimum normalized cut.

Given a graph consisting of categorical data $G = <O, \mathbf{A}>$, where O is the set of categorical objects and $\mathbf{A}$ is a non-negative and symmetric affinity matrix that contains the connected strength or similarity between objects:

Definition 1. The *normalized cut* specifies the connection strength between two sets relative to the total connection strengths in a graph. Formally, the normalized cut between sets $O_1, O_2 \subseteq O$ is

$$ normCut(O_1, O_2) = \frac{\sum_{i \in O_1, j \in O_2} \mathbf{A}(i,j)}{\sum_{i \in O_1, j \in O} \mathbf{A}(i,j)}. $$

Definition 1 shows that the normalized cut indicates the overlap between object sets. In other words, the *minimum normalized cut* reflects the maximum separability between clusters, which is essential for most of learning tasks, e.g., clustering.

Theorem 5.3. *The objective of UNTIE in Eq.* (20) *is equivalent to learning a representation with the minimum normalized cut.*

Proof. The objective of minimizing the normalized cuts between clusters can be formalized as,

$$ \text{minimize} \quad \frac{1}{n_c} \sum_{j=1}^{n_c} normCut(O_j, O \setminus O_j), $$

where O_j refers to the set of objects in the j-th cluster, and $O \setminus O_j$ refers to the set of objects in the clusters instead of j-th cluster. This objective function can be converted to a trace maximization problem according to [40] as follows,

$$ \text{maximize} \quad \frac{1}{n_c} \text{Tr}(\mathbf{V}^\top \mathbf{A} \mathbf{V}), $$

where $\mathbf{V} = \mathbf{Z}(\mathbf{Z}^\top \mathbf{D} \mathbf{Z})^{-\frac{1}{2}}$, and $\mathbf{Z} \in \{0,1\}^{n_0 \times n_c}$ is an indicator matrix for the cluster, and $\mathbf{D}$ is the diagonal matrix, in which (i,i)-entry is the sum of the i-th row in $\mathbf{A}$. If further relaxing the matrix $\mathbf{Z}$ to a real value matrix and denoting $\mathbf{H} = \mathbf{D}^{\frac{1}{2}} \mathbf{V}$, the objective function can be converted to

$$ \text{maximize} \quad \frac{1}{n_c} \text{Tr}(\mathbf{H}^\top \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \mathbf{H}) $$
$$ \text{subject to} \quad \mathbf{H} \in \mathcal{R}^{n_o \times n_c} $$
$$ \mathbf{H}^\top \mathbf{H} = \mathbf{I}_{n_c}. $$

Let $\mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}}$ be $\mathbf{S}$ in Eq. (20), by adding a low rank regularization $\text{Tr}(\mathbf{S})$, this objective function is equivalent to the UNTIE objective function Eq. (20). Therefore, the objective of UNTIE in Eq. (20) is equivalent to learning a representation with the minimum normalized cut. □

5.4 Convergence of the UNTIE Algorithm

Theorem 5.4. *The UNTIE algorithm converges to a local minimal solution in a finite number of iterations.*

Proof. Let y be the number of all possible partitions of a categorical data set C, and each partition can be represented by an indicator matrix $\mathbf{H}$. If two partitions are different, their indicator matrices are also different; otherwise, they are identical. We note that y is finite, given C and the number of cluster n_c. Therefore, there are a finite number of $\mathbf{H}$ on C. While applying UNTIE to cluster C, we obtain a series of $\mathbf{H}$, i.e., $\mathbf{H}_1, \mathbf{H}_2, \cdots, \mathbf{H}_{n_i}$, and a series of $\boldsymbol{\omega}$, i.e., $\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \cdots, \boldsymbol{\omega}_{n_i}$, along the iterations. Given a matrix $\mathbf{H}$ and a heterogeneity parameter $\boldsymbol{\omega}$, denote the loss of the UNTIE objective function Eq.(20) as $l_{\mathbf{H}, \boldsymbol{\omega}}$. Since kernel k-means and linear programming for Eq. (22) converge to minimal solutions, $l_{\mathbf{H}, \boldsymbol{\omega}}$ is strictly decreasing, i.e., $l_{\mathbf{H}_1, \boldsymbol{\omega}_1} \geq l_{\mathbf{H}_2, \boldsymbol{\omega}_2} \geq \cdots \geq l_{\mathbf{H}_{n_i}, \boldsymbol{\omega}_{n_i}}$. We assume that the number of iterations n_i is more than $y+1$. That indicates there are at least two same indicator matrices in the sequence, i.e., $\mathbf{H}_i = \mathbf{H}_j$, $1 \leq i \neq j \leq n_i$. For $\mathbf{H}_i$ and $\mathbf{H}_j$, we have the optimized heterogeneity parameter $\boldsymbol{\omega}_i$ and $\boldsymbol{\omega}_j$, respectively. It is clear that $\boldsymbol{\omega}_i = \boldsymbol{\omega}_j$ since $\mathbf{H}_i = \mathbf{H}_j$. Therefore, we obtain $l_{\mathbf{H}_i, \boldsymbol{\omega}_i} = l_{\mathbf{H}_j, \boldsymbol{\omega}_i} = l_{\mathbf{H}_j, \boldsymbol{\omega}_j}$, i.e., the value of objective function does not change. If the value of the objective function does not change, UNTIE stops, and n_i is not more than $y + 1$. UNTIE converges to a local minimal solution in a finite number of iterations. □

5.5 Computational Efficiency

The time complexity of UNTIE is determined by two parts, i.e., building coupling spaces and learning heterogeneities. In building coupling spaces, the time cost depends on what kind of couplings UNTIE captures. In this paper, UNTIE captures the intra- and inter-attribute couplings. The intra-attribute couplings measure the frequency of each value, corresponding to complexity $O(n_v)$. The inter-attribute couplings calculate the relationship between each value pair in each attribute pair, corresponding to the time cost $O(n_{mv}^2 n_a^2)$, where n_{mv} is the maximal number of values in attributes. Consequently, the entire time complexity of building the coupling spaces is $O(n_v + n_{mv}^2 n_a^2)$.

In heterogeneity learning, its time complexity is determined by the time cost of calculating eigenvectors and $\boldsymbol{\omega}$ and the number of iterations. If involving all data in each iteration, it requires $O(n_o^3)$ to calculate eigenvectors and $O((n_o^2)^{3.5} n_\omega^2)$ to solve linear optimization of calculating $\boldsymbol{\omega}$ [8]. Denoting the number of iterations as n_i, the time complexity of heterogeneity learning is $O(n_i n_o^3 + n_i (n_o^2)^{3.5} n_\omega^2)$, where n_ω refers to the number of elements in $\boldsymbol{\omega}$. If taking stochastic optimization, denoting the number of object pairs in each batch as n_b, the time complexity is only $O(n_b^3 n_i + n_b n_\omega n_i)$. Since the batch size $n_b \ll n_o$ holds for large data sets, stochastic optimization is much more efficient if it can converge within a small number of iterations. We thus recommend stochastic optimization to solve the UNTIE objective. Accordingly, the time complexity of UNTIE is $O(n_v + n_{mv}^2 n_a^2 + n_b^3 n_i + n_b n_\omega n_i)$.

The space complexity of UNTIE with stochastic optimization is $O(n_o^2 n_\omega)$. For large categorical data, the space complexity is very high when full data optimization is used, which approaches $O(n_o^2)$. However, conducting stochastic optimization to obtain an approximate solution can largely reduce the space complexity since $n_b \ll n_o^2$. Therefore, we take stochastic optimization in UNTIE to tackle large data.

Overall, UNTIE has the time complexity $O(n_v + n_{mv}^2 n_a^2 + n_b^3 n_i + n_b n_\omega n_i)$ and space complexity $O(n_b n_\omega)$. This means UNTIE is scalable for large data. In addition, UNTIE can be

further sped up by distributed and parallel computing in both building coupling spaces and learning heterogeneities, which will be explored in our future work.

6 EXPERIMENTS

6.1 Experimental Settings

6.1.1 Data Sets

25 data sets[2] are used to evaluate UNTIE. The data includes medical data: Hepatitis, Audiology, Spect, Mammographic, Wisconsin, Breastcancer, Primarytumor, and Dermatology; gene data: DNAPromoter, DNANominal, and Splice; social and census data: Housevotes, and Hayesroth; hierarchical decision-making data: Krvskp, Tictactoe, and Connect4; nature data: Zoo, Soybeanlarge, Flare, and Mushroom; business data: Crx; disaster prediction data: Titanic; and synthetic data with heterogeneous couplings: Mofn3710, Led24, and ThreeOf9. In Table 3, they show strong diversity in terms of data factors: the number of objects (n_o), the number of attributes (n_a), the number of classes (n_c), the average number of nominal values in each attribute (n_{av}), and the maximum number of nominal values in all attributes (n_{mv}). Specifically, the number of objects ranges from 101 to 67,557, and the number of attributes ranges from 4 to 69. The data sets contain both binary and multiple classes with the maximum number of 24 classes. The average and maximal numbers of distinct attribute values range from 2 to 10 and from 2 to 15, respectively.

TABLE 3
Characteristics of 25 Categorical Data Sets

Dataset	n_o	n_a	n_c	n_{av}	n_{mv}
Zoo	101	16	7	2.25	6
DNAPromoter	106	57	2	4	4
Hayesorth	132	4	3	3.75	4
Hepatitis	155	13	2	2.77	3
Audiology	200	69	24	2.23	7
Housevotes	232	16	2	2	2
Spect	267	22	2	2	2
Mofn3710	300	10	2	2	2
Soybeanlarge	307	35	19	3.77	8
Primarytumor	339	17	21	2.47	4
Dermatology	366	33	6	3.91	4
ThreeOf9	512	9	2	2	2
Wisconsin	683	9	2	9.89	10
Crx	690	9	2	5	15
Breastcancer	699	9	2	10.00	11
Mammographic	830	4	2	5	7
Tictactoe	958	9	2	3	3
Flare	1,066	11	6	3.73	8
Titanic	2,201	3	4	2	2
DNANominal	3,186	60	3	4	4
Splice	3,190	60	3	4.78	6
Krvskp	3,196	36	2	2.03	3
Led24	3,200	24	10	2	2
Mushroom	5,644	22	2	4.45	9
Connect4	67,557	42	3	3	3

6.1.2 Representation and Downstream Learning Baselines

We evaluate the UNTIE representation (UNTIE for short) against (1) the categorical distance measure Hamming distance (Hamming); (2) five state-of-the-art categorical data representation

2. More details of these data sets can be found on http://archive.ics.uci.edu/ml and https://www.sgi.com/tech/mlc/db. While we could not find much larger public categorical data, these small sets have rich data characteristics which challenge categorical representation.

methods: CDE [17], COS [4], DILCA [14], Ahmad [13], and Rough [27]; (3) two unsupervised deep representation methods BiGAN [24] and VAE [30] based on the wide-and-deep network [25] (denoted as BiGAN_WD and VAE_WD, respectively).

The various learning tasks *clustering* (two simple but popular clustering methods k-means and k-modes), *classification* (K-Nearest Neighbor-KNN, Support Vector Machine-SVM, Random Forest-RF, and Logistic Regression-LR), and *object retrieval* are undertaken on the UNTIE representations to test their downstream task performance.

6.1.3 Evaluation Perspectives

The following aspects of UNTIE performance are evaluated.

- **Representation quality**: to reveal why UNTIE produces better representation results;
- **Downstream effectiveness**: to evaluate whether the UNTIE representation effectively improves various downstream learning tasks;
- **Efficiency**: to reflect the cost sensitivity of UNTIE in representing data w.r.t. different data characteristics;
- **Flexibility**: to verify whether the UNTIE representation fits and upgrades different learning methods.
- **Stability**: to test UNTIE's parameter sensitivity.

To avoid the impact of class imbalance, we evaluate the UNTIE-enabled clustering and classification performance by F-score (%). The higher F-score indicates better learning performance.

6.1.4 Implementation Settings

The default settings for implementing UNTIE are as follows. The kernels used in UNTIE are 11 Gaussian kernels with width from 2^{-5} to 2^5 and three Polynomial kernels with order from 1 to 3. We use the stochastic optimization method Adam [37] to solve the UNTIE objective function with the initial learning rate 10^{-3}, the batch size 20, and the max number of iterations $1,000$. For the parameters of the baseline methods, we take their recommended settings. UNTIE is implemented in Python 3.5 and Tensorflow r1.2, all experiments are conducted in a Windows 10 workstation with Intel i5-5300 U CPU@2.30GHz and 8GB memory.

6.2 Evaluating the UNTIE Representation Quality

6.2.1 Reducing Heterogeneous Couplings Inconsistency

Here, we evaluate how UNTIE effectively reduces the inconsistency during learning heterogeneous couplings. First, we quantitatively measure the inconsistency by the intra- and inter-coupling heterogeneity indicators. Then, we analyze the relations between UNTIE-enabled clustering results and the inconsistency indicated by these indicators.

The intra-coupling heterogeneity indicator (I_{intra}) measures the degree of heterogeneity in value distributions. It assigns a higher heterogeneity degree to couplings if each value in the couplings has more significant diverse distributions. Intuitively, if a value has multiple distributions, its representations in different distributions may be inconsistent with each other. The larger difference its distributions have (i.e., higher heterogeneity degree), the stronger inconsistency may exist. In our experiment, I_{intra}

compares the difference between distributions of a value per each ground-truth cluster, which is formalized as follows:

$$I_{intra} = \frac{\sum\limits_{j=1}^{n_a} \sum\limits_{i=1}^{n_v^{(j)}} \mathrm{NM}_{n_c} \left(\sqrt{\sum\limits_{k=1}^{n_c} \left(\frac{|g^{(j)}(\mathsf{v}_i^{(j)}) \cap g^{(c)}(\mathsf{c}_k)|}{|g^{(j)}(\mathsf{v}_i^{(j)})|} \right)^2} \right)}{n_a},$$
(23)

where c_k refers to the k-th cluster, n_c is the number of ground-truth clusters, function $g(\cdot)$ has the same definition as in Eq. (1), and function NM_{n_c} normalizes a value to $[0, 1]$ w.r.t. n_c clusters, which is defined as

$$\mathrm{NM}_{n_c}(x) = 1 - \frac{1 - x}{1 - \sqrt{\sum\limits_{k=1}^{n_c} \left(\frac{1}{n_c} \right)^2}}.$$
(24)

I_{intra} reflects the inconsistency within couplings caused by heterogeneous data distributions. I_{intra} is large if each value has diverse distributions in each cluster, otherwise small. A larger I_{intra} indicates a stronger inconsistency within a coupling. In extreme cases, I_{intra} is 1 when each value only appears in a ground-truth cluster, and I_{intra} is 0 when every value has the same distribution in every ground-truth cluster.

To show the distribution of intra-coupling inconsistency, we illustrate the probability density of I_{intra} on 25 testing data sets in Fig. 2 per kernel density estimation (KDE)[3]. The KDE smoothly estimates the probability density of I_{intra} within its range. A larger density indicates a higher probability of an inconsistency degree. As shown in Fig. 2, most of data sets have strong inconsistency within couplings, which indicates the necessity of eliminating inconsistency in learning heterogeneous couplings.
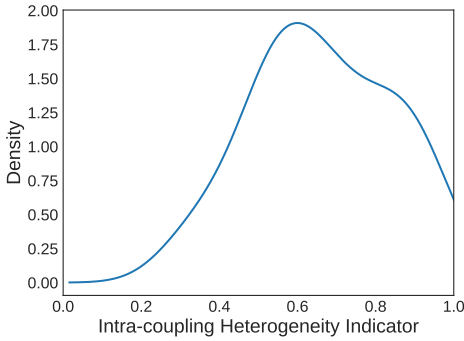

Fig. 2. The Probability Density of Intra-coupling Heterogeneity Indicator per Kernel Density Estimation.

The inter-heterogeneity indicator (I_{inter}) represents the degree of difference between couplings. Intuitively, the increase of difference between couplings (i.e., higher heterogeneity degree) may increase the inconsistency between categorical data representations. In our experiment, I_{inter} calculates the distance between coupling matrices to reflect the inter-coupling inconsistency. Here, a coupling matrix is a similarity matrix of the values in an attribute, which is defined by a coupling learning method. For UNTIE, a coupling matrix is calculated by the Euclidean distance w.r.t. a coupling vector representation of values (e.g., the intra-attribute coupling value representation as Eq. (1) or the inter-attribute

3. Here we treat the intra-coupling inconsistency of the 25 testing data sets as the random variable to calculate its KDE.

coupling value representation as Eq. (4)). For CDE, a coupling matrix is calculated by the Euclidean distance w.r.t. a value clustering representation. For COS and DILCA, the coupling matrices are their value similarity matrices before weighted integration. Denoting the k-th coupling matrix of the z-th attribute as $\mathbf{C}^{(z,k)}$, the inter-heterogeneity indicator is defined as

$$I_{inter} = \sqrt{\frac{\sum\limits_{k=1}^{n_m^{(z)}} \sum\limits_{l=1}^{n_m^{(z)}} \frac{\sum\limits_{i=1}^{n_v^{(z)}} \sum\limits_{j=1}^{n_v^{(z)}} \left(\mathbf{C}_{ij}^{(z,k)} - \mathbf{C}_{ij}^{(z,l)} \right)^2}{n_v^{(z)2}}}{n_m^{(z)2}}},$$
(25)

$n_m^{(z)}$ is the number of couplings for the z-th attribute, and $\mathbf{C}_{ij}^{z,k}$ is the (i, j)-th entry of $\mathbf{C}_{ij}$. I_{inter} differs for different coupling learning methods. A larger I_{inter} indicates a stronger inconsistency between couplings. I_{inter} is large if each coupling matrix is largely different from the others, and otherwise small. In an extreme case, I_{inter} is 0 when all coupling matrices are the same.

The probability density of I_{inter} on 25 testing data sets is shown in Fig. 3 per KDE, which shows UNTIE and CDE involve a higher degree of inter-coupling heterogeneity compared to DILCA and COS. On one hand, the higher degree of inter-coupling heterogeneity provides richer information for representation. On the other hand, the higher degree of inter-coupling heterogeneity may contain inconsistency representation with higher probability.
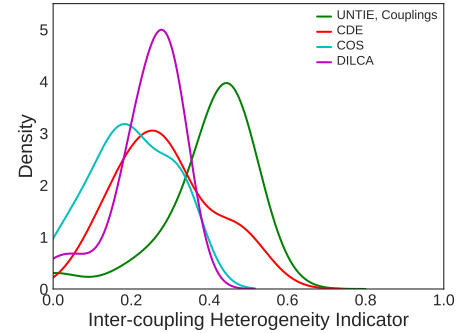

Fig. 3. The Probability Density of Inter-coupling Heterogeneity Indicator per Kernel Density Estimation.

To evaluate whether UNTIE can effectively resolve the coupling inconsistency problem, we analyze the UNTIE-enabled clustering performance at different inconsistency levels indicated by the intra-/inter-coupling heterogeneity indicators. Considering the imbalanced distributions of these indicators as shown in Fig. 2 and Fig. 3, we set 5 inconsistency levels, each of them contains the same number of data sets, according to the intra-/inter-coupling heterogeneity indicators. For example, the first level contains 20% data sets with the smallest values of I_{intra}/I_{inter}, while the fifth level contains 20% data sets with the largest values of I_{intra}/I_{inter}. We calculate the performance of a heterogeneous coupling learning method at an inconsistency level by averaging its enabled clustering rank in Table 4 on the data sets with the inconsistency level. We illustrate the relations between the UNTIE-enabled clustering performance and the inconsistency level in Fig. 4 and Fig. 5 per I_{intra} and I_{inter}, respectively.

As shown in Fig. 4, UNTIE significantly outperforms its competitors on data sets at the inconsistency levels 2 to 5 in terms of I_{intra}, which have strong intra-coupling heterogeneity

as shown in Fig. 2. While other methods ignore the intra-coupling heterogeneity, UNTIE captures it by learning value weights in kernel spaces w.r.t. Eq. (8). As a result, UNTIE reduces the inconsistency and achieves better performance. In contrast, on the data sets with the inconsistency level 1, UNTIE dose not show superiority. This is because these data sets do not have much coupling inconsistency, which further demonstrates that UNTIE gains better performance by reducing the inconsistency between heterogeneous couplings.
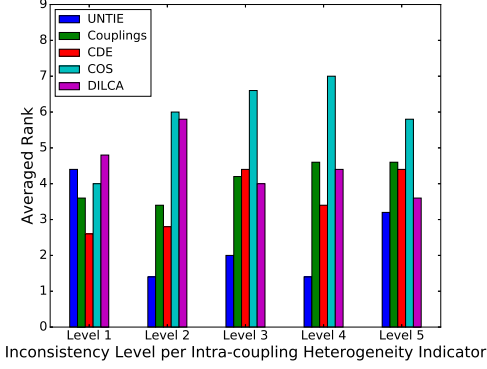

Fig. 4. The UNTIE-enabled Clustering Performance on Data Sets with Different Inconsistency Levels per the Intra-heterogeneity Indicator.

In Fig. 5, the UNTIE-enabled clustering performs much better than that enabled by existing heterogeneous coupling learning methods, especially on the data with higher inconsistency levels. Since the couplings represented by UNTIE may also incur inconsistency as shown in Fig. 3, UNTIE well reduces the inconsistency to guarantee a good representation. This reduction is mainly contributed by the multiple kernels used in UNTIE, which enables to capture the fitness of couplings for different distributions. In Fig. 5, UNTIE shows similar performance by simply combining multiple couplings on data sets with the inconsistency levels 2 and 3. This phenomenon indicates the heterogeneous couplings learned in our method capture much richer data information compared to other methods and enable better performance when the inconsistency between these couplings is not so substantial. However, with the inconsistency increase, simply combining these couplings may worsen the results, as shown on the data sets with the inconsistency levels 4 and 5 in Fig. 5.
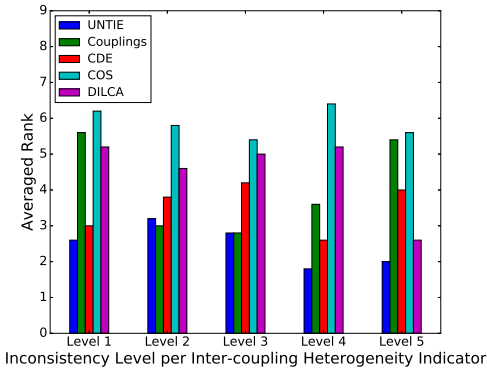

Fig. 5. The UNTIE-enabled Clustering Performance on Data Sets with Different Inconsistency Levels per the Inter-heterogeneity Indicator.

6.2.2 Goodness of the UNTIE-enabled Metric

Per Definition 4 in [41], a similarity function Q (e.g., the similarity function $s(\cdot, \cdot)$ in Eq. (18)) is strongly (ϵ, γ)-good for a learning problem P if at least a $1 - \epsilon$ probability mass of objects o satisfies:

$$E_{\mathsf{o},\mathsf{o}' \sim \Phi}[Q(\mathsf{o}, \mathsf{o}')|c(\mathsf{o}) = c(\mathsf{o}')]$$
$$\geqslant E_{\mathsf{o},\mathsf{o}' \sim \Phi}[Q(\mathsf{o}, \mathsf{o}')|c(\mathsf{o}) \neq c(\mathsf{o}')] + \gamma,$$

where E refers to the expected value on distribution Φ in the learning problem P, and $c(\mathsf{o})$ refers to the class of o. In this criterion, ϵ indicates the proportion of objects whose averaged intra-class similarity is not γ degrees larger than their averaged inter-class similarity value. With the same γ, the smaller ϵ reflects the better similarity function for the learning problem P. Here, γ indicates to what extent the intra-class similarity is larger than the inter-class similarity on the $1 - \epsilon$ proportion of data which is best separated. When ϵ is fixed, the larger γ reflects the better similarity function. The intuition of this criterion is that a good similarity measure should effectively differ data in the same class from those in other classes. More importantly, a (ϵ, γ)-good similarity can induce a classifier with a bounded error (see more details in Theorem 1 of [41]).

In this experiment, we compare the UNTIE's similarity function $s(\cdot, \cdot)$ (defined in Eq. (18)) with the similarity functions in the other similarity-based representation methods per the (ϵ, γ)-good criterion. For the CDE-learned vector representation, we reverse the Euclidean distance to measure the similarity between objects. Since different ϵ values may correspond to different γ values, we draw the (ϵ, γ)-curves to demonstrate the quality of the learned metric and the compared methods. With the same ϵ, the better metric incurs a greater γ. In other words, the better metric yields a higher curve in the (ϵ, γ)-curve. In this experiment, we draw the (ϵ, γ)-curves on four data sets, i.e., Mofn3710, Dermatology, Crx and Breastcancer. The results are shown in Fig. 6. It should be noted that, we only focus on ϵ that can guarantee a non-negative margin, i.e., $\gamma \geqslant 0$. Therefore, Fig. 6 only displays a part of the (ϵ, γ)-curve, in which $\gamma \geqslant 0$.

The results illustrate that UNTIE is better than its competitors in terms of the (ϵ, γ)-good criterion. The results also reveal the insight behind the clustering performance in Table 4. For data Dermatology and Crx, the UNTIE-enabled clustering has much higher F-score than others since UNTIE yields larger margins between different classes, which is reflected by the (ϵ, γ)-good in Figs. 6(b) and 6(c). For Mofn3710, all methods obtain low F-score, and the UNTIE-enabled clustering achieves the same result as CDE which is only slightly better than the other competitors. The reason is shown in Fig. 6(a), where nearly 20% of the Mofn3710 data cannot be well separated by UNTIE (γ is 0 when ϵ is smaller than 0.2) while nearly 30% of that cannot be well separated by its competitors. For other data sets, e.g., Breastcancer, all methods achieve good results since the (ϵ, γ)-good criterion indicates that these methods can well separate the data sets.

6.2.3 Visualization of UNTIE Representations

We visualize the separability of different representations. The UNTIE, CDE and BiGAN_WD-represented data is converted from high-dimensional representation to two-dimensional embedding by the t-Distributed Stochastic Neighbor Embedding (t-SNE) [42]. The basic idea of t-SNE is to minimize the Kullback-Leibler divergence between the joint probabilities of the low-dimensional embedding and the high-dimensional representation.
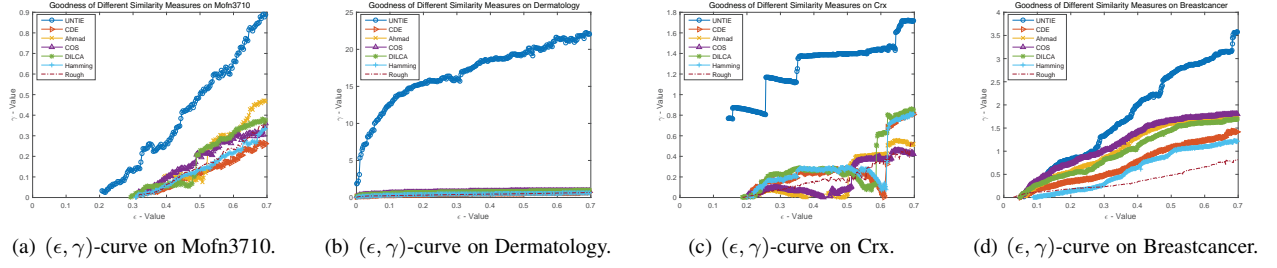
(a) (ϵ, γ)-curve on Mofn3710. (b) (ϵ, γ)-curve on Dermatology. (c) (ϵ, γ)-curve on Crx. (d) (ϵ, γ)-curve on Breastcancer.

Fig. 6. The (ϵ, γ)-curves of Different Transformed Similarity Measures: A better metric yields a better result.

(a) UNTIE-represented Distributions. (b) CDE-represented Distributions. (c) BiGAN_WD-represented Distributions.
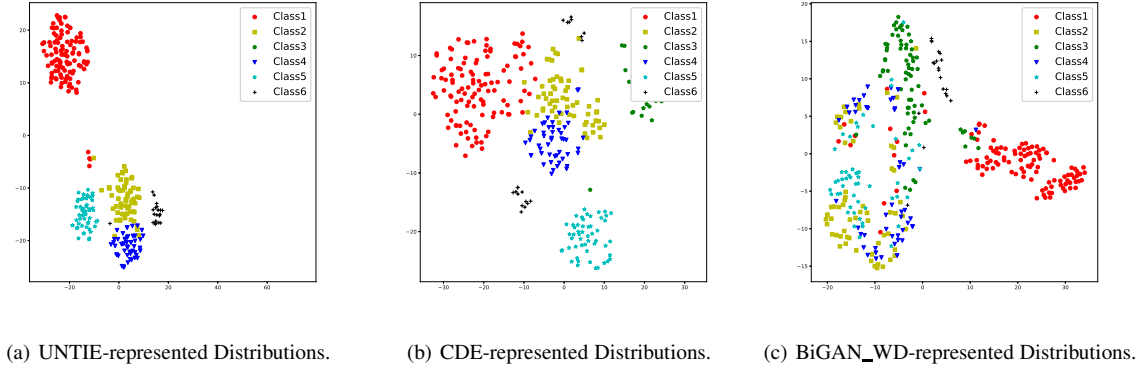
Fig. 7. The Visualization of Different Representation Methods on Dermatology. The UNTIE-represented data shows clearer boundaries between different clusters. The plotted two-dimensional embedding is converted from high-dimensional representation by t-SNE. Different symbols refer to different data clusters per the ground truth.

Fig. 7 shows the visualization of these different representation methods on Dermatology. The UNTIE-represented data has a more compact distribution and leads to clearer boundaries between different clusters, compared to that from CDE and BiGAN_WD-represented data. It qualitatively demonstrates that the UNTIE representation is more separable and suitable for downstream tasks such as clustering and classification. This is because UNTIE learns the representation by optimizing the objective function Eq. (20), i.e., by minimizing the distance between objects within a cluster and maximizing the distance between objects in different clusters.

(a) Training Loss on Hepatitis. (b) Training Loss on Housevotes. (c) Training Loss on Mofn3710.

(d) Training Loss on Dermatology. (e) Training Loss on Crx. (f) Training Loss on Breastcancer.

Fig. 8. The UNTIE's Training Loss on Different Data Sets. The stochastic optimization method for UNTIE is Adam [37] with the initial learning rate 10^{-3} and the batch size 20. The X-axis refers to the number of iterations, and y-axis refers to the loss value of UNITE's objective function Eq. (22).

6.3 Evaluating the UNTIE Effectiveness

6.3.1 UNTIE-enabled Clustering

The vector-based representations learned by methods UNTIE, CDE, BiGAN_WD and VAE_WD are incorporated into K-means, which is probably the most popular clustering method and is sensitive to a distance measure. The representations learned by similarity-based representation methods, including COS, DILCA, Ahmda, Rough and Hamming, are fed into K-modes, which is the most commonly used clustering method for categorical data. To evaluate the performance of heterogeneity learning, we compare UNTIE with its variant which concatenates the representations in the coupling spaces without heterogeneity learning, denoting as Couplings, which is incorporated into K-means.

The comparison results are shown in Table 4. The best results are highlighted in bold and Δ is the ratio of UNTIE's improvement over the best results of other measures. On half of the data sets, UNTIE performs significantly better than the compared methods. For example, the F-score improves 51.72% on DNAnominal and 30.75% on Dermatology compared to the best-performing methods DILCA and COS. On the other half of the data sets, UNTIE achieves the same or comparable results to other methods. For example, the F-scores of UNTIE and CDE are both 56.56% on Mofn3710 and 54.8% on Tictactoe. UNTIE effectively captures the intrinsic categorical data characteristics by revealing the value and attribute couplings and the heterogeneity within and between these couplings to induce the representation. These embedded characteristics guarantee the effectiveness of UNTIE representations, ensuring that the UNTIE-enabled clustering can generally achieve better results than others. It is noted that BiGAN_WD, and VAE_WD perform badly on most of data sets except Mofn3710
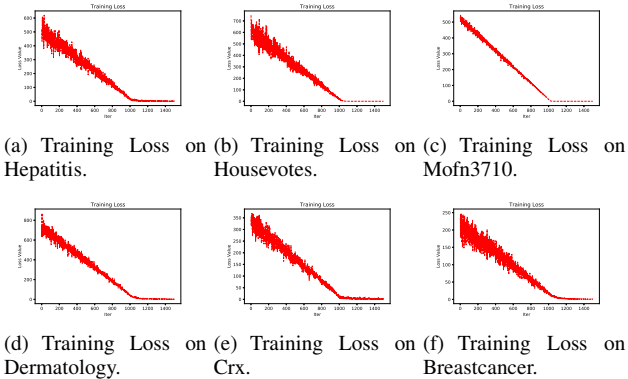
(a) Time Cost w.r.t. Number of Objects.

(b) Time Cost w.r.t. Number of Attributes.

(c) Time Cost w.r.t. Number of Attribute Values.
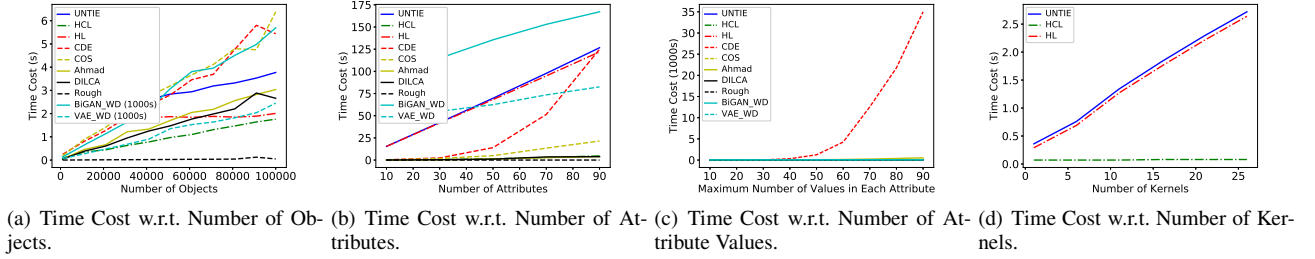
(d) Time Cost w.r.t. Number of Kernels.

Fig. 9. The UNTIE Computational Cost w.r.t. Data Factors: Object Number n_o, Attribute Number n_a, and Maximum Number of Attribute Values n_{mv}. The solid line refers to the total time cost of UNTIE. The dotted line refers to the time cost of building the coupling spaces. The star line refers to the time cost of the heterogeneity learning.

where BiGAN_WD achieves the highest.

To statistically compare UNTIE's performance with the above categorical representation methods, we calculate their averaged ranks by the Friedman test and Bonferroni-Dunn test [43]. The χ_F^2 of Friedman test is 83.21 associated with p-value $3.71e^{-14}$. This result indicates that the performance of all the compared methods is not equal. Further, the Bonferroni-Dunn test evaluates the critical difference (CD) between UNTIE and other methods, and shows the CD at p-value < 0.1 is 2.17. As shown in Table 4, UNTIE achieves an overall averaged rank 2.82, which is better than other measures. For example, it is 0.8 better than that of the best state-of-the-art method CDE (3.62), and 5.94 better than VAE_WD (8.76). Regarding the CD, the UNTIE's performance is significantly better than most of state-of-the-art methods except CDE, DILCA, and Adam. Although UNTIE and CDE do not show significant difference under the Bonferroni-Dunn test at p-value < 0.1, UNTIE captures the heterogeneity in couplings which cannot be learned by CDE. Therefore, the performance of UNTIE is better than CDE in most cases, especially on data sets with complex structures and heterogeneous distributions. For example, on DNAnominal, UNTIE achieves 89.79% while CDE only achieves 51.14% in terms of F-score. All the comparison results are shown in Fig. 10, which reveals UNTIE is significantly ($p < 0.1$) better than almost all the compared categorical representation methods.
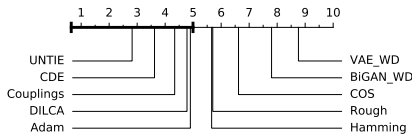

Fig. 10. Comparison of UNTIE vs. the Other Representation Methods per the Bonferroni-Dunn Test. All methods with ranks outside the marked interval are significantly different (p < 0.1) from UNTIE.

The results also show that UNTIE and Couplings achieve the overall performance of 2.82 and 4.34 averaged rank, respectively, in comparison with 7.8 and 8.76 for BiGAN_WD and VAE_WD which rank the worst. This shows that heterogeneity learning contributes to an additional 1.52 averaged rank over the representation of Couplings. UNTIE does not consistently beat Couplings over all data sets, showing that not all data sets involve strong heterogeneity. For example, on Hepatitis, Mammographic and Tictactoe, the couplings-enabled representations show better results, while both UNTIE and Couplings do not make significant improvement over other methods, which also demonstrates that the clustering labels in these data sets are not sensitive to the captured

couplings and heterogeneity. This may indicate other unknown complexities in these data sets that could be further explored.

6.3.2 UNTIE-enabled Retrieval

We further test the UNTIE representation performance of object retrieval, which also heavily depends on data representation. Every object is used as a query, and its k-closest objects are retrieved per a distance measure. The precision@k, i.e., the fraction of the retrieved k objects that are the same-class neighbors, is reported. We use the Euclidean distance for UNTIE-, CDE-, BiGAN_WD- and VAE_WD-represented data to compare with the distance measured by COS, DILCA, Ahmda, Rough and Hamming for retrieval. Three data sets Dermatology, DNAnominal and Splice are tested to evaluate the UNTIE-enabled retrieval performance.

Different from the clustering results, the precision@k of retrieval can demonstrate the quality of learned representation from local (when k is small) to global (when k is large). The results are shown in Fig. 11, in which the precision of UNTIE-enabled retrieval consistently outperforms the others. It reflects that UNTIE can capture more details of data distributions than other representation methods, which is powered by learning hierarchical value-to-attribute couplings and heterogeneities.

6.4 Evaluating the UNTIE Efficiency

The efficiency of UNTIE is affected by the number of iterations to achieve convergence in Algorithm 1 and different data factors. In this section, we first empirically evaluate the convergence speed of UNTIE, and then evaluate the computational cost of UNTIE under different data factors.

6.4.1 The UNTIE Convergence

Due to space limitation, we randomly select six real data sets to demonstrate the convergence of UNTIE. The optimization method for UNTIE is Adam with the same setting as in Section 6.1. The training loss of objective function Eq. (22) on these data sets is shown in Fig. 8, the loss value converges rapidly at around 1,000 iterations. Since the batch size in each iteration is only 20, the time cost of 1,000 iterations is very low.

6.4.2 The UNTIE Computational Cost w.r.t. Data Factors

We further generate synthetic data to evaluate the computational cost of UNTIE in terms of the following data factors [12]: the number of objects n_o, the number of attributes n_a, and the maximum number of values in each attribute n_{mv}. The default settings of these factors are as follows: n_o is $1,000$, n_a is 10, and n_{mv} is 3. We generate three groups of data and tune one of these

TABLE 4
Clustering F-Score (%) with Different Embedding Methods: The value-based representations are fed into k-means and the similarity-based representations are fed into k-modes to get the clustering results. The best results are highlighted in bold. Δ indicates the UNTIE's improvement over the best results of the other measures. The averaged rank of a method over all data sets with significant difference from others w.r.t. the Bonferroni-Dunn test (p-value < 0.1) is labelled by *.

Dataset	UNTIE	Couplings	CDE	COS	Ahmad	DILCA	Rough	Hamming	BiGAN_WD	VAE_WD	Δ
Zoo	**76.12**	74.85	75.04	72.10	71.34	71.34	62.79	73.27	56.93	24.41	1.44%
DNAPromoter	**95.28**	92.45	61.61	49.24	49.92	85.85	63.20	52.68	51.99	50.87	10.98%
Hayesroth	**54.17**	**54.17**	52.85	38.98	33.76	32.87	38.92	33.06	44.91	37.14	2.50%
Hepatitis	70.40	**73.64**	69.82	46.29	66.72	65.13	59.21	59.21	61.08	51.24	0.00%
Audiology	34.99	34.48	32.18	27.71	**35.38**	31.77	22.36	29.05	20.00	19.97	0.00%
Housevotes	**90.51**	88.36	89.65	88.36	88.36	88.79	87.04	86.64	83.64	53.84	0.96%
Spect	55.04	55.04	52.55	36.26	34.93	34.76	**57.63**	35.94	34.71	48.38	0.00%
Mofn3710	56.65	44.69	56.65	50.18	50.22	48.68	50.62	50.98	**60.34**	49.00	0.00%
Soybeanlarge	**69.29**	64.88	62.19	60.10	56.84	59.42	46.41	55.31	48.38	14.83	11.42%
Primarytumor	24.62	24.87	23.43	19.81	23.65	21.76	22.38	**26.19**	22.17	14.68	0.00%
Dermatology	**97.51**	72.78	73.10	74.58	72.87	72.61	57.99	66.60	38.54	23.82	30.75%
ThreeOf9	34.86	34.86	54.63	35.32	35.32	35.32	**65.19**	54.22	50.03	54.64	0.00%
Wisconsin	93.91	95.58	**96.20**	94.28	95.12	95.49	94.44	89.98	74.26	81.45	0.00%
Crx	**85.49**	52.65	52.65	36.99	52.65	79.29	63.47	79.29	51.81	51.69	7.82%
Breastcancer	93.27	94.75	95.20	93.56	94.89	**95.25**	94.37	93.27	65.94	79.15	0.00%
Mammographic	82.77	**82.89**	81.66	80.06	81.66	82.65	80.67	81.50	60.48	70.59	0.00%
Tictactoe	54.80	**62.61**	54.80	51.88	50.87	52.97	50.19	53.59	54.38	50.24	0.00%
Flare	37.08	31.20	32.44	35.79	34.20	35.59	38.85	**39.22**	31.98	22.30	0.00%
Titanic	33.72	29.77	33.72	29.77	33.72	33.72	**36.27**	33.72	31.58	28.61	0.00%
DNAnominal	**89.79**	67.70	51.14	41.91	46.68	59.18	43.28	41.44	35.18	32.21	51.72%
Splice	79.73	42.29	**87.12**	31.31	47.34	45.87	42.79	42.48	26.60	32.55	0.00%
Krvskp	51.09	51.09	51.03	46.72	**55.17**	**55.17**	53.73	53.86	42.94	50.36	0.00%
Led24	**69.50**	45.82	48.03	53.91	51.83	61.08	32.65	28.82	18.38	13.12	13.79%
Mushroom	82.69	82.76	82.83	**82.91**	82.86	82.39	78.18	82.29	71.48	60.78	0.00%
Connect4	**33.20**	31.14	31.91	27.23	32.88	33.14	30.34	31.43	30.53	29.18	0.18 %
Averaged Rank*	**2.82**	4.34	3.62	6.62	4.9	4.78	5.7	5.66	7.8	8.76	0.8

(a) Precision on Dermatology.   (b) Precision on DNAnominal.   (c) Precision on Splice.
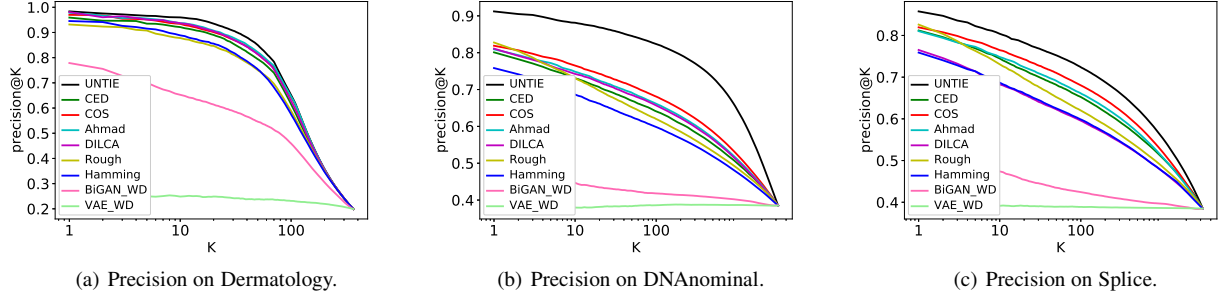
Fig. 11. The Precision@k of Different Categorical Data Representation Methods: A better metric yields a higher value.

factors for each group. For the first group of data, the number of objects is adjusted from $1,000$ to $100,000$. For the second group of data, the number of attributes is tuned from 10 to 100. For the third group of data, the maximum number of values in attributes is changed from 10 to 100. The time cost of UNTIE under each data factor is shown in Fig. 9.

Fig. 9(a) shows that the time cost of UNTIE is almost stable (from 1.8(s) to 3.6(s)), which demonstrates it has good scalability w.r.t. the amount of data n_o. Our analysis shows that the minor time cost increase is caused by the Python built-in functions when identifying categorical value location in the data. Since this cost increases with an extremely small proportion w.r.t. the amount of data, we can ignore it when applying UNTIE. Fig. 9(b) and Fig. 9(c) demonstrate the time cost has approximately linear relation with both n_a and n_{mv}, which is consistent with the time complexity of UNTIE analyzed in Section 5.5. These results also show that the main cost of UNTIE is on the heterogeneity learning (HL), which has the linear relation with both n_a and n_{mv}.

Meanwhile, the cost of building hierarchical coupling learning (HCL) has quadratic relation with n_a and n_{mv}. The reason is that UNTIE calculates the pairwise value relations when learning inter-attribute couplings. However, it only slightly affects the cost of UNTIE when n_a and n_{mv} are small. For categorical data with high dimensionality, a trade-off between sufficiently capturing couplings and preserving efficiency is required.

The computational costs of UNTIE and state-of-the-art methods are at the same level in terms of data factor n_o. It indicates all of these methods can handle large amount of data. We can see that UNTIE has a higher computational cost compared to other methods in terms of n_a. As shown in 9(b), the higher cost is brought by heterogeneity learning, which has linear relation with n_a. For the hierarchical coupling learning, the cost of UNTIE is at the same level as the state-of-the-art methods. As for n_{mv}, UNTIE is much more efficient than CDE, which is the state-of-the-art method with the best representation performance as shown in the previous experiments.

To evaluate the relation between time cost and the number of kernel functions, we set the number of kernels used in UNTIE from 1 to 30 and test the computational cost of UNTIE on the synthetic data set with the default data factors. The UNTIE time cost with a different number of kernels is shown in Fig. 9(d). This shows that the UNTIE time cost is linear to the number of kernels with a very small slope. Increasing the number of kernels only slightly affects the computational time of UNTIE. This is consistent with our theoretical analysis, which indicates n_ω is linear to the time complexity of UNTIE. Here, n_ω has a linear relation with the number of kernels.

6.5 Evaluating the UNTIE Flexibility

We further demonstrate the UNTIE flexibility when fed into the classifiers KNN, SVM, RF and LR. We randomly select 90% of objects in each data set for training and the remainder for testing. To reduce the impact of noise and randomness, 20 sampling iterations generate 20 sets of training and test data for the experiments. The averaged classification performance and standard deviation are reported w.r.t. F-score (%). The vector representations learned by UNTIE and CDE are used as the input of these classifiers. The results comparing with CDE-enabled classifiers are shown in Table 5 and illustrate that UNTIE representations can fit different classifiers and enhance their performance on categorical data, as compared with the results of CDE-enabled classifiers.

6.6 Evaluating the UNTIE Stability

To evaluate the stability of UNTIE per the kernel functions in heterogeneity learning, we adopt three groups of kernel functions where each has a varying number of functions. The first group only contains Gaussian kernels, the second group only contains Polynomial kernels, and the third group mixes Gaussian and Polynomial kernels. The kernel functions in each set are shown in Table 6. The clustering F-score enabled by UNTIE w.r.t. different kernel function sets on two data sets DNAPromoter and Monfn3710 are illustrated in Fig.12.

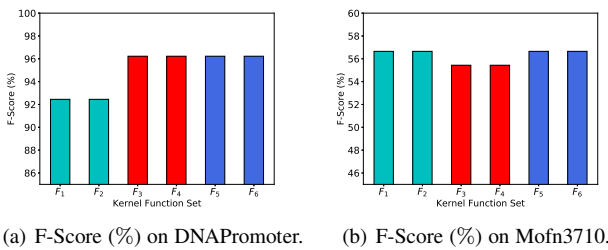(a) F-Score (%) on DNAPromoter.　(b) F-Score (%) on Mofn3710.

Fig. 12. The Clustering F-Score (%) with UNTIE w.r.t. Different Kernel Function Sets: The same color indicates the same kernel function group.

UNTIE is stable in terms of the number of kernel functions. As shown in Fig. 12, UNTIE achieves the same clustering F-score w.r.t. kernel function sets in the same group, where kernel functions are of the same type but different numbers. Although different kernel functions may generate different effects of UN-TIE on different data sets (e.g., the Gaussian kernel family in group 1 enables better performance on DNAPromoter, and the Polynomial kernel family in group 2 enables better performance on Mofn3710), UNTIE can comprehensively learn information from multiple kernels while eliminates their redundancy and inconsistency. Accordingly, it always enables the best clustering

performance with the kernel function sets F_5 and F_6 in group 3, which involves kernel functions in both groups 1 and 2.

7 CONCLUSIONS

Categorical data representation is critical yet challenging as complicated coupling relationships and heterogeneities are often embedded in complex categorical values, attributes and objects. Existing work including deep learning is troubled by unsupervised categorical representation learning. This paper introduces a heterogeneous coupling learning method UNTIE for unsupervised categorical data representation. By modeling value-to-object hierarchical couplings and their complementary and inconsistent influence on representations, UNTIE reveals the nonlinear relations between couplings and discloses the heterogeneous distributions within couplings. Both theoretical and empirical analyses show the effectiveness and efficiency of UNTIE.

An important lesson learned in UNTIE is to select appropriate kernels w.r.t. specific data characteristics and domain knowledge of the underlying problems. This work shows the need and potential of shallow learners in handling complex data characteristics in particular couplings, heterogeneities, and inconsistency. The poor results on some data in Table 4 also show the challenge and open issues on categorical representation of complex data characteristics even in small data. In addition, modeling more complex couplings, such as very high-dimensional couplings, may represent more complicated relations and interactions embedded in high-dimensional data.

8 ACKNOWLEDGMENT

REFERENCES

[1] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[2] S. Boriah, V. Chandola, and V. Kumar, "Similarity measures for categorical data: A comparative evaluation," in *SDM'08*, 2008, pp. 243–254.

[3] L. Bai, J. Liang, C. Dang, and F. Cao, "The impact of cluster representatives on the convergence of the k-modes type clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 6, pp. 1509–1522, 2013.

[4] C. Wang, X. Dong, F. Zhou, L. Cao, and C.-H. Chi, "Coupled attribute similarity learning on categorical data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 4, pp. 781–797, 2015.

[5] S. Seth and M. J. Eugster, "Archetypal analysis for nominal observations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 5, pp. 849–861, 2016.

[6] G. Tutz and J. Gertheiss, "Regularized regression for categorical data," *Statistical Modelling*, vol. 16, no. 3, pp. 161–200, 2016.

[7] Y. Qian, F. Li, J. Liang, B. Liu, and C. Dang, "Space structure and clustering of categorical data," *IEEE transactions on neural networks and learning systems*, vol. 27, no. 10, pp. 2047–2059, 2016.

[8] C. Zhu, L. Cao, Q. Liu, J. Yin, and V. Kumar, "Heterogeneous metric learning of categorical data with hierarchical couplings," *IEEE Transactions on Knowledge and Data Engineering*, 2018.

[9] G. Pang, L. Cao, and L. Chen, "Outlier detection in complex categorical data by modeling the feature value couplings." in *IJCAI'16*, 2016, pp. 1902–1908.

[10] L. Cao, "Coupling learning of complex interactions," *Information Processing & Management*, vol. 51, no. 2, pp. 167–186, 2015.

[11] P. Brémaud, *The Coupling Method*. Springer International Publishing, 2017, pp. 397–415.

[12] L. Cao, *Data Science Thinking: The Next Scientific, Technological and Economic Revolution*. Springer International Publishing, 2018.

TABLE 5
KNN, SVM, RF and LR Classification F-score (%) with UNTIE and CDE. The best results are highlighted in bold.

Data set	UNTIE-SVM	CDE-SVM	UNTIE-KNN	CDE-KNN	UNTIE-RF	CDE-RF	UNTIE-LR	CDE-LR
zoo	**100**	88.00±18.33	**100**	**100**	**100**	**100**	**100**	**100**
DNAPromoter	**94.42±6.81**	91.37±7.41	**87.19±10.79**	76.06±10.62	**89.32±8.81**	87.58±11.37	**94.41±6.03**	90.35±8.20
hayesroth	**82.23±6.22**	80.92±6.96	60.15±10.52	**62.38±10.33**	**82.48±9.00**	82.11±8.26	**82.08±7.29**	**82.08±7.29**
lymphography	**87.06±12.02**	85.68±11.26	**82.22±11.08**	79.03±15.07	82.17±14.59	**84.16±11.31**	**83.12±12.90**	81.67±13.84
hepatitis	**48.13±8.65**	46.85±6.04	70.16±13.37	**70.47±14.52**	**66.61±11.12**	64.30±12.25	**70.39±16.85**	**70.39±16.85**
audiology	**73.41±7.29**	73.25±5.90	**48.19±9.47**	47.79±8.20	**63.80±10.24**	59.78±11.83	47.70±7.11	**66.29±11.26**
housevotes	96.71±3.88	**96.92±3.69**	**95.58±3.69**	92.54±4.18	**96.02±4.82**	95.13±4.61	93.74±5.00	**93.98±5.15**
spect	67.60 ± 11.92	**68.46±10.88**	55.41±9.95	50.54±8.56	66.61±10.93	**67.13±12.08**	**69.48±12.52**	**69.48±12.52**
mofn3710	**100**	**100**	**87.18±6.29**	86.04±7.77	76.64±10.51	**78.31±12.16**	**100**	**100**
soybeanlarge	90.94±3.83	**93.61±4.34**	93.70±4.26	**96.03±3.85**	**92.88±5.81**	92.69±6.12	**89.57 ±5.98**	88.57±6.97
Averaged Rank	**1.35**	1.65	**1.35**	1.65	**1.35**	1.65	1.45	**1.55**

TABLE 6
Three Groups of Kernel Functions for Testing the UNTIE Stability

Group	Set	Kernel Functions
Group 1	F_1	Gaussian kernels with width $\{2^{-3}, 2^{-2}, \cdots, 2^3\}$
	F_2	Gaussian kernels with width $\{2^{-5}, 2^{-4}, \cdots, 2^5\}$
Group 2	F_3	Polynomial Kernels with order $\{1, 2\}$
	F_4	Polynomial Kernels with order $\{1, 2, 3\}$
Group 3	F_5	Gaussian kernels with width $\{2^{-3}, 2^{-2}, \cdots, 2^3\}$ Polynomial Kernels with order $\{1, 2\}$
	F_6	Gaussian kernels with width $\{2^{-5}, 2^{-4}, \cdots, 2^5\}$ Polynomial Kernels with order $\{1, 2, 3\}$

[13] A. Ahmad and L. Dey, "A method to compute distance between two categorical values of same attribute in unsupervised learning for categorical data set," *Pattern Recognition Letters*, vol. 28, no. 1, pp. 110–118, 2007.

[14] D. Ienco, R. G. Pensa, and R. Meo, "From context to distance: Learning dissimilarity for categorical data clustering," *ACM Transactions on Knowledge Discovery from Data*, vol. 6, no. 1, pp. 1–25, 2012.

[15] H. Jia, Y.-m. Cheung, and J. Liu, "A new distance metric for unsupervised learning of categorical data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 5, pp. 1065–1079, 2016.

[16] K. Zhang, Q. Wang, Z. Chen, I. Marsic, V. Kumar, G. Jiang, and J. Zhang, "From categorical to numerical: Multiple transitive distance learning and embedding," in *SDM'15*, 2015, pp. 46–54.

[17] S. Jian, L. Cao, G. Pang, K. Lu, and H. Gao, "Embedding-based representation of categorical data by hierarchical value coupling learning," in *IJCAI'17*, 2017, pp. 1937–1943.

[18] T. D. T. Do and L. Cao, "Gamma-poisson dynamic matrix factorization embedded with metadata influence," in *NeurIPS'18*, 2018, pp. 5829–5840.

[19] M. K. Ng, M. J. Li, J. Z. Huang, and Z. He, "On the impact of dissimilarity measure in k-modes clustering algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 3, pp. 503–507, 2007.

[20] L. Cao, "Non-iidness learning in behavioral and social data," *The Computer Journal*, vol. 57, no. 9, pp. 1358–1370, 2014.

[21] G. Pang, L. Cao, L. Chen, and H. Liu, "Learning homophily couplings from non-iid data for joint feature selection and noise-resilient outlier detection," in *IJCAI'17*, 2017, pp. 2585–2591.

[22] L. Ralaivola, M. Szafranski, and G. Stempfel, "Chromatic pac-bayes bounds for non-iid data: Applications to ranking and stationary β-mixing processes," *Journal of Machine Learning Research*, vol. 11, no. Jul, pp. 1927–1956, 2010.

[23] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 11, pp. 3371–3408, 2010.

[24] J. Donahue, P. Krähenbühl, and T. Darrell, "Adversarial feature learning," *arXiv:1605.09782*, 2016.

[25] H. Cheng, L. Koc, J. Harmsen, T. Shaked, T. D. Chandra, H. Aradhye, G. Anderson, G. S. Corrado, W. Chai, M. Ispir *et al.*, "Wide & deep learning for recommender systems," in *1st workshop on deep learning for recommender systems*, 2016, pp. 7–10.

[26] S. Jian, L. Hu, L. Cao, and K. Lu, "Metric-based auto-instructor for learning mixed data representation," in *AAAI'18*, 2018, pp. 3318–3325.

[27] F. Cao, J. Liang, D. Li, L. Bai, and C. Dang, "A dissimilarity measure for the k-modes clustering algorithm," *Knowledge-Based Systems*, vol. 26, pp. 120–127, 2012.

[28] S. Q. Le and T. B. Ho, "An association-based dissimilarity measure for categorical data," *Pattern Recognition Letters*, vol. 26, no. 16, pp. 2549–2557, 2005.

[29] C. Wang, Z. She, and L. Cao, "Coupled attribute analysis on numerical data." in *IJCAI'13*, 2013, pp. 1736–1742.

[30] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv:1312.6114*, 2013.

[31] S. S. Bucak, R. Jin, and A. K. Jain, "Multiple kernel learning for visual object recognition: A review," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1354–1369, 2014.

[32] C. Wang, C.-H. Chi, W. Zhou, and R. Wong, "Coupled interdependent attribute analysis on mixed data," in *AAAI'15*, 2015, pp. 1861–1867.

[33] C. Wang, L. Cao, M. Wang, J. Li, W. Wei, and Y. Ou, "Coupled nominal similarity in unsupervised learning," in *CIKM'11*, 2011, pp. 973–978.

[34] I. S. Dhillon, Y. Guan, and B. Kulis, "Kernel k-means: spectral clustering and normalized cuts," in *KDD'04*, 2004, pp. 551–556.

[35] S. Jegelka, A. Gretton, B. Schölkopf, B. K. Sriperumbudur, and U. Von Luxburg, "Generalized clustering via kernel embeddings," in *Annual Conference on Artificial Intelligence*, 2009, pp. 144–152.

[36] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research*, vol. 12, no. Jul, pp. 2121–2159, 2011.

[37] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv:1412.6980*, 2014.

[38] D. B. Dunson and C. Xing, "Nonparametric bayes modeling of multivariate categorical data," *Journal of the American Statistical Association*, vol. 104, no. 487, pp. 1042–1051, 2009.

[39] I. Steinwart and A. Christmann, *Support Vector Machines*. Springer Science & Business Media, 2008.

[40] X. Y. Stella and J. Shi, "Multiclass spectral clustering," in *ICCV'03*, 2003, pp. 313–319.

[41] M.-F. Balcan, A. Blum, and N. Srebro, "A theory of learning with similarity functions," *Machine Learning*, vol. 72, no. 1-2, pp. 89–112, 2008.

[42] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[43] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine learning research*, vol. 7, pp. 1–30, 2006.

Chengzhang Zhu was a PhD student in the Faculty of Engineering and IT, University of Technology Sydney, Australia when this work was done. His research interests include metric learning, non-IID learning, and data representation, in addition to general interest in broad data science especially machine learning.

Longbing Cao (SM06) received a PhD in Pattern Recognition and Intelligent Systems from the Chinese Academy of Science and a PhD in Computing Sciences from the University of Technology Sydney. He is a professor and an ARC Future Fellow (Level 3). His current research interests include data science, artificial intelligence, behavior informatics, and their enterprise applications.

Jianping Yin received a PhD in computer science and technology from the National University of Defense Technology in 1990. He is a distinguished professor at the Dongguan University of Technology. His research interests include theoretical computer science, artificial intelligence, pattern recognition, and networking algorithms.

This figure "Chengzhang.jpg" is available in "jpg" format from:

http://arxiv.org/ps/2007.10720v1