Similarity Measures Recommendation for Mixed Data Clustering

Abdoulaye Diop, Nabil El Malki, Max Chevalier, André Péninou, Geoffrey Roman-Jimenez, Olivier Teste, Olivier Teste

Similarity Measures Recommendation for Mixed Data Clustering

Abdoulaye Diop
abdoulaye.diop@irit.fr
Institut de Recherche en Informatique de Toulouse - IRIT
Solution Data Group
Toulouse, France

Nabil El Malki
Max Chevalier
André Péninou
Olivier Teste
firstname.lastname@irit.fr
Institut de Recherche en Informatique de Toulouse - IRIT
Toulouse, France

Geoffrey Roman Jimenez
groman-jimenez@solutiondatagroup.fr
Solution Data Group
Toulouse, France

ABSTRACT

Clustering is an important data mining task which is widely spread in various domains such as biology, finance, marketing, healthcare, and social sciences. It allows the end user to discover, through built clusters, relationships within data. Many non-expert users perceive clustering as an "easy" task because it always produces a result. However, choosing a clustering algorithm at random, without proper parameter tuning, often leads to poor results. In particular, an important choice when applying a clustering algorithm to a specific dataset is the similarity measure. Since clustering algorithms rely on similarities between data points to build clusters, the chosen similarity measure should fit the data as accurately as possible in order to form the best clusters. Mixed Data are data that are characterized by numerical as well as categorical attributes. When clustering mixed data, the same similarity measure cannot be used for the two attribute types. Commonly a pair of similarity measures is used, one dedicated to numerical attributes and one dedicated to categorical attributes. The choice of these two most appropriate similarity measures is very important in mixed data, as it significantly affects the clustering performance.

In this paper, we challenge to recommend the best pairs of similarity measures to end-users, regardless of their experience, when applying a specific clustering algorithm to a mixed dataset to maximize a specific performance measure. The proposed recommendation process relies on knowledge extracted from a meta-model built by an automated machine learning (AutoML) approach. To evaluate the relevance of the recommendation process, experiments are conducted with two well-known clustering algorithms: K-Prototypes and Hierarchical Clustering. Our results show that the recommendations can positively help users select the most appropriate pairs of similarity measures depending on their use cases (i.e. clustering algorithm, dataset, and performance measure). These recommendations outperform the traditionally used similarity measures in the literature, particularly for datasets where the choice of the similarity measures has a significant impact.

SSDBM 2024, July 10–12, 2024, Rennes, France

ACM ISBN 979-8-4007-1020-9/24/07
https://doi.org/10.1145/3676288.3676302

KEYWORDS

Mixed Data Clustering, Similarity Measures Recommendation, Meta-Learning

ACM Reference Format:
Abdoulaye Diop, Nabil El Malki, Max Chevalier, André Péninou, Olivier Teste, and Geoffrey Roman Jimenez. 2024. Similarity Measures Recommendation for Mixed Data Clustering. In *36th International Conference on Scientific and Statistical Database Management (SSDBM 2024), July 10–12, 2024, Rennes, France.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3676288.3676302

1 INTRODUCTION

When performing clustering, we have to make multiple choices about the clustering algorithm, parameter values, similarity measures, etc. Since these choices highly impact the quality of obtained clustering, important efforts have been made in the literature to support data scientists in tasks such as clustering algorithms recommendation for numeric data [8, 13, 14, 24], parameters recommendation such as the number of clusters for K-Means algorithm [25], and similarity measures recommendation for categorical data [3]. These works mainly focus on data with only one type. However, many recent real-world applications generate data that contain mixed numeric and categorical attributes known as mixed data [2]. In this study, we tackle the problem of Similarity Measures Recommendation (SMR) when clustering mixed data.

Defining similarity measures able to deal with mixed numeric and categorical attributes is one of the main challenges for mixed data clustering (MDC) algorithms, remaining an ongoing research challenge because of the diverse nature of the two data types [2, 4]. The adopted strategy in most MDC algorithms is to use a specific similarity measure for each data type (such as *Euclidean distance* for numeric attributes and *Hamming distance* for categorical ones) and then combine the two measures to define a global similarity for mixed data [2]. The selection of an appropriate pair of numeric and categorical similarity measures is fundamental for these algorithms to obtain good clustering results [6, 10]. No need to say there is "*no free lunch*", i.e. there is not a universal similarity measure pair that performs optimally across all datasets. Additionally, the effectiveness of the similarity measure pairs varies depending on the specific MDC algorithm being used. What works well for one algorithm with a particular dataset may not yield satisfactory results with another algorithm [10]. Hence, it is important to select similarity measure pairs based on the considered dataset and its characteristics, as well as the MDC algorithm being used. In practice, the

selection of the similarity measure pair is often done independently of the dataset by using default similarity measures like *Euclidean distance* and *Hamming distance*, or the similarity measures used in the literature for the considered MDC algorithm. However, according to the considered dataset, these measures might not be appropriate, leading to poor clustering performances [10]. Another alternative is to evaluate several possible similarity measure pairs to identify the most suitable ones (trial-and-error). However, the diversity of similarity measures for each data type and the high number of corresponding pairs (number of possible combinations) make this strategy too costly.

To fill this gap, we propose an approach based on meta-learning [27] to create a recommendation system able to recommend, for a given algorithm, suitable similarity measure pairs according to the considered dataset. The idea of this approach is to exploit the knowledge we get from evaluating a given MDC algorithm on various mixed datasets with different similarity measure pairs to train a model named *meta-model* that learns the relationships between the datasets' characteristics called *meta-features* and the performances of the similarity measure pairs. So, for a new dataset, the proposed system only computes its meta-features and uses the trained meta-model to recommend suitable similarity measure pairs for this dataset. The main contributions of this work are as follows:

(1) This is the first attempt, to the best of our knowledge, to automatically recommend couples of similarity measures for **mixed data clustering**.
(2) We extend meta-learning to the context of mixed data while previous approaches focus on algorithm selection or homogeneous datasets (i.e. numerical or categorical).
(3) Motivated by the context of MDC, in order to capture specific geometry of mixed datasets, we propose additional meta-features to complete those existing in the literature.
(4) We validate the proposed approach on two commonly used MDC algorithms, *K-Prototypes* and *Hierarchical Clustering*. Our experiments show that the similarity measure pairs recommended for these two algorithms perform better than the baseline pairs, especially for datasets highly impacted by the choice of the similarity measure pair.
(5) Finally, we provide the following resources at https://github.com/AbdoulayeDiop/simrec/tree/simrec-v1
    - The code for reproducing the experiments.
    - The code for using the created SMR systems with the already trained meta-models, for K-Prototypes and Hierarchical Clustering.
    - We also provide the meta-datasets used to train the meta-models so they can be continuously extended with new datasets faced in practice.

2 RELATED WORK

Mixed data clustering. One of the main challenges of MDC algorithms is to find innovative ways to define novel similarity measures for mixed data [2]. Since most existing similarity measures are defined for data with only one type, a common strategy is to define similarity for mixed data as a combination of two numeric and categorical similarity measures. In [23], Philip and Ottaway extend the hierarchical clustering algorithm to mixed data by using the Gower similarity [5]. The Gower similarity is a similarity measure for mixed data defined as a weighted combination of the *Manhattan distance* on normalized numeric attributes and the *Hamming distance* for categorical attributes. In [18], Huang introduced K-Prototypes which extends the K-Means algorithm with a new representation of cluster centers and a new definition of similarity. The similarity is defined as a weighted sum of the *squared Euclidean distance* for numeric attributes and the *Hamming distance* for categorical ones. Based on the same idea of combining two numeric and categorical similarity measures, different weighting strategies and different choices of similarity measures have been proposed in later studies based on K-Prototypes [1, 17, 20] or using different clustering algorithms such as K-Medoids [6, 15], Fuzzy C-Medoids [12], Spectral Clustering [22], and Density-Based Clustering [9, 11].

Other strategies to define similarity for mixed data have also been considered such as distance hierarchies [16] and graph-based dissimilarity [31]. However, creating distance hierarchies for categorical attributes requires domain knowledge and computing graph-based dissimilarity can be time intensive. Recently, strategies based on deep representation learning [33] and clustering with deep neural networks [21] have been proposed. However, despite their representation abilities, the inherent weakness of deep learning models in terms of interpretability may limit their applications, especially for clustering-based data exploration, data understanding, knowledge acquisition, and so on.

Meta-Learning. Meta-learning methods have been widely studied in the field of algorithm selection as support tools for machine learning practitioners [8, 25]. These methods use prior learning experience to learn to predict algorithm performances according to datasets' meta-features [29] (in our case we are interested in similarity measures selection instead of algorithm selection, but the principle remains the same). The first step is to identify meta-features that may impact the performances of considered algorithms. The second step is to accumulate the knowledge we get from running the considered algorithms on various datasets. This knowledge is represented as a new dataset, named *meta-dataset*. Each record of the meta-dataset corresponds to one dataset and contains its meta-features (which represent the predictive attributes) and the performances of the different algorithms on that dataset (which represent the target attributes). Finally, a standard machine learning model, named *meta-model*, is trained on the meta-dataset to learn to predict the performances of the different algorithms according to the meta-features of the datasets.

The first application of meta-learning to clustering is from [8], in the context of clustering algorithm selection. Given a dataset, the proposed approach provides a ranking of the 7 candidate algorithms. They used 8 meta-features based on statistical measures and evaluated their framework using 32 micro-array datasets about cancer gene expression. Their results suggest that the proposed approach performs better than a baseline based on average ranking. Later studies have mainly worked on designing new meta-features to enhance predictive capability. In [13], a new set of meta-features based on the distribution of similarity between observations is proposed to extract more information about the internal structure of the datasets. Vukicevic *et al.* [29] introduce meta-features based on

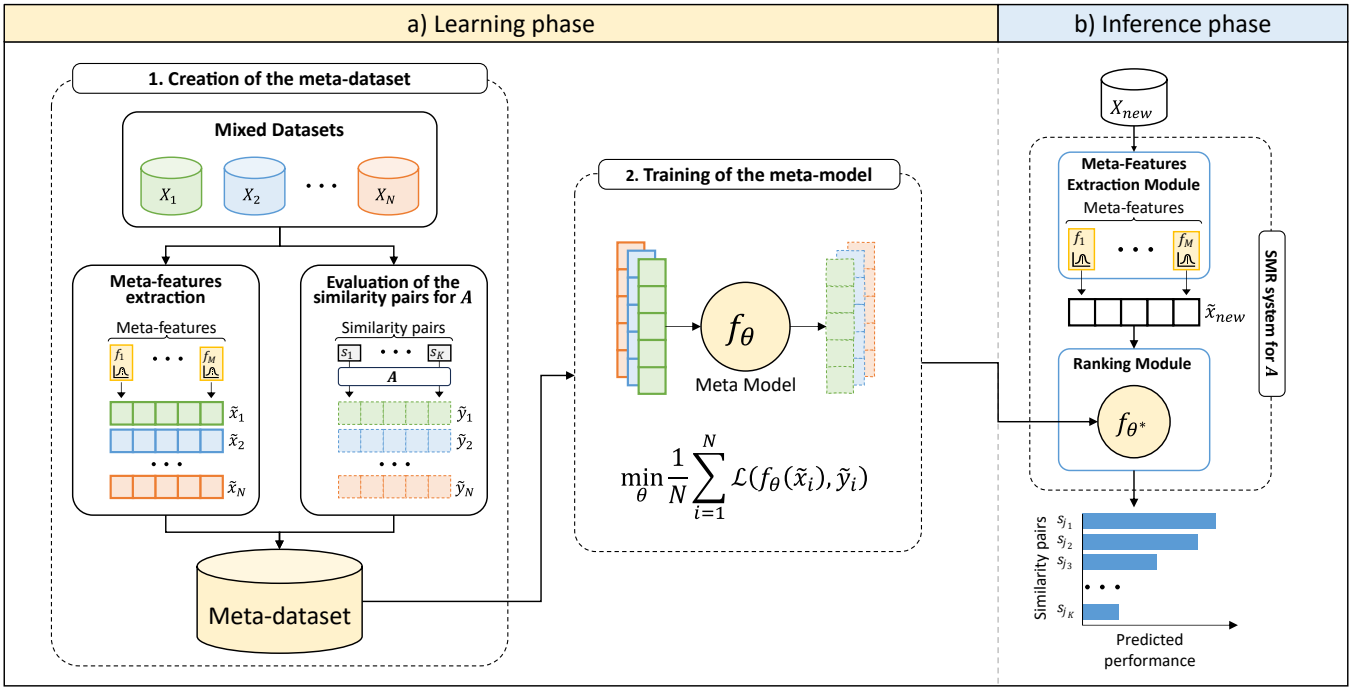


Figure 1: Overview of the proposed meta-learning approach for the recommendation of similarity measures for MDC

internal cluster evaluation metrics. Pimentel and de Carvalho [24] proposed new meta-features based on correlation and dissimilarity measures. In [25], meta-learning is used for recommending the number of clusters for the K-Means algorithm. New meta-features based on density distribution are introduced since they may convey information about the number of clusters.

In [3], meta-learning is used for similarity measures recommendation for clustering categorical data. The authors used statistical meta-features about the datasets and their attributes. They considered 10 similarity measures for categorical data and validated their approach using one clustering algorithm (Hierarchical Clustering) and 60 synthetic datasets. Zhu *et al.* [34] propose a meta-learning approach to recommend similarity measures for clustering numeric data. Besides statistical meta-features, they also use structural information-based and distance-based (distribution of the Euclidean distance between pairs of observations) meta-features. They considered 9 similarity measures for numeric data and validated their approach using two clustering algorithms (K-Means and CURE) and 199 datasets.

More recently an interesting survey [26] provides a taxonomy of existing works on automated machine learning methods for clustering. It underlines that cited work are applied on datasets where all attributes are homogeneous (i.e. either numerical or categorical).

Furthermore, when dealing with mixed data, we show in [10] that existing clustering algorithms for mixed data are mainly based on two strategies: the **homogenization strategy** where all attributes are converted to a single type and the **mixed strategy** where similarity measures for the different data types are combined to define a similarity measure for heterogeneous data. Furthermore, the experiments showed that the mixed strategy outperforms the homogenization strategy.

In this context, we introduce in next sections, our proposal for recommending similarity measure pairs for MDC algorithms based on the mixed strategy.

3 META-LEARNING BASED SIMILARITY MEASURES RECOMMENDATION FOR MDC

3.1 Overview

Let $\mathcal{X}$ be the set of mixed datasets and $\mathcal{A}$ be the set of MDC algorithms that define similarity for mixed data by combining two numeric and categorical similarity measures. Let $\mathcal{S}^n$ and $\mathcal{S}^c$ be the sets of existing numeric and categorical similarity measures respectively. Let $\mathcal{S} = \{s_j\}_{j=1}^{K} \subset \mathcal{S}^n \times \mathcal{S}^c$ be a finite subset of similarity measure pairs. Given an algorithm $A \in \mathcal{A}$, the proposed meta-learning-based SMR system contains two modules (figure 1.b): a **Meta-Features Extraction Module** and a **Ranking Module**. The meta-features extraction module extracts a vector representation of the input dataset. This vector contains the meta-features of the dataset. The ranking module contains a machine learning model (the meta-model) that takes as input the vector representation of the dataset and predicts from this vector the ranking of the similarity measure pairs $s_j \in \mathcal{S}$ according to their performances on the input dataset when using A. To build such a system, we follow two main steps (figure 1.a - Learning phase):

(1) **Creation of the meta-dataset.** The meta-dataset is a knowledge database containing results of prior evaluations of the considered MDC algorithm A on various datasets with the different similarity measure pairs $s_j \in \mathcal{S}$. Let us consider

that we have access to a set of mixed datasets $\{X_i\}_{i=1}^N$. For each dataset X_i, the meta-dataset stores a tuple $(\tilde{x}_i, \tilde{y}_i)$ such that $\tilde{x}_i = (f_m(X_i))_{m=1}^M$ is the meta-features vector of X_i with $\{f_m : \mathcal{X} \rightarrow \mathbb{R}\}_{m=1}^M$ being the considered meta-features, and $\tilde{y}_i = (\tilde{y}_{i,j})_{j=1}^K$ is a performance vector containing the performances of the similarity measure pairs s_j on X_i. Meta-features are described in section 3.2

(2) **Training of the meta-model.** In this step, the created meta-dataset is used to train the meta-model in the ranking module. The meta-model is described in section 3.3.

Once the training is done, for a new dataset X_{new}, the system first computes its meta-features vector $\tilde{x}_{new}$. Then the meta-features vector is given as input to the learned meta-model, which predicts the corresponding ranking of the similarity measure pairs.

3.2 Meta-features

Our main hypothesis is that the performances of the similarity measure pairs on a given dataset depend on the meta-features of the dataset. So, it is crucial to define meta-features that describe well the datasets and embed useful information for accurate prediction of the performances of the similarity measure pairs.

3.2.1 Meta-features selected from the literature. Although several meta-features have been proposed in the literature [8, 13, 24, 25, 29], we consider only meta-features based on statistical measures about the datasets and their attributes (table 1). This is because other meta-features based on similarity, density, and clustering evaluation have been designed for homogeneous (numerical) data and cannot be directly applied to mixed data. Furthermore, they are computationally more intensive and need a predefined similarity measure, which is not trivial in the context of SMR.

3.2.2 Proposed meta-features. To complete the meta-features selected from the literature, we propose 30 new meta-features (in table 2) that extract information about diverse notions exploited by the similarity measures.

Let X be a mixed dataset with p numeric and q categorical attributes. We denote $A_j^n, 1 \leq j \leq p$ the j^{th} numeric attribute and $A_l^c, 1 \leq l \leq q$ the l^{th} categorical attribute. The first 10 meta-features are based on squared numeric attributes since several similarity measures for numeric data like *Euclidean distance* and *squared Euclidean distance* use squared attribute values. For each numeric attribute A_j^n of the considered dataset, we compute the mean and standard deviation of its squared values: $\{u^2 : u \in A_j^n\}$. Then, the meta-features are defined using the *min*, q_1, *mean*, q_3, and *max* values of the computed mean and standard deviation over all numeric attributes. Based on the same idea, the next 10 meta-features consider the internal products of numeric attribute values ($\{u.v : u, v \in A_j^n\}$).

The next 5 meta-features are based on the frequency of categorical attribute values. The aim is to provide some information about the balance between categories within the same attribute. This can give important insights about frequency-based similarity measures. Given a categorical attribute A_l^c, we compute the frequency of each category (u) within the attributes $\{\frac{\#u}{card(A_l^c)} : u \in set(A_l^c)\}$, where $\#u$ is the number of occurrences of u in A_l^c . Then, we use the standard deviation of these frequencies to estimate the balance between the categories. Finally, the meta-features are defined as the *min*, q_1, *mean*, q_3, and *max* values of the standard deviations across all categorical attributes. The last 5 meta-features are based on the mutual information between categorical attributes. They provide information about the relationships between the categorical attributes (in terms of shared information) and can give important insights for co-occurrence based similarity measures. We compute the mutual information [7], I, between all pairs of categorical attributes $\{I(A_k^c, A_l^c) : 1 \leq k < l \leq q\}$. Then, the meta-features are defined as the *min*, q_1, *mean*, q_3, and *max* of the computed mutual information values.

3.3 Meta-Model

Our objective is to learn a meta-learning model able to predict the ranking of the similarity measure pairs according to the datasets' meta-features. This problem is known as *label ranking* in the literature [32]. We implemented two types of meta-models.

3.3.1 Regression-based meta-model. For this meta-model, we transform the label ranking task into a multi-output regression task where the goal is to predict the performances of the similarity measure pairs. The predicted performances are then used to create the ranking. Let $f_\theta : \mathbb{R}^M \rightarrow \mathbb{R}^K$ be the meta-model and $\{(\tilde{x}_i, \tilde{y}_i)\}_{i=1}^N$ be the meta-dataset. We recall that M is the number of meta-features and K is the number of similarity measure pairs. The regression-based meta-model is trained by solving the following problem:

$$\min_\theta \frac{1}{N} \sum_{i=1}^{N} \|f_\theta(\tilde{x}_i) - \tilde{y}_i\|_2^2 \tag{1}$$

f_θ can be any regression model that supports multiple outputs. We tested several models including k-Nearest Neighbors (KNN), ELasticNet, Decision Tree, Random Forest, and Neural Networks with different architectures. In this paper, results are shown only for the KNN model which gave the best results.

3.3.2 Pairwise-Preference meta-model. The main drawback of the regression loss is that a prediction can have an important loss while preserving the ranking of the target vector. Inversely, a prediction can have a small loss and not align with the ranking of the target vector. In the pairwise-preference approach [19], the label ranking task is transformed into multiple binary regression tasks. For each couple of similarity measure pairs (s_k, s_l), a specialised model $f_{\theta_{k,l}}^{k,l} : \mathbb{R}^M \rightarrow [-1, 1]$ is trained to predicts the difference between the performances of s_k and s_l:

$$\min_{\theta_{k,l}} \frac{1}{N} \sum_{i=1}^{N} \left\| f_{\theta_{k,l}}^{k,l}(\tilde{x}_i) - (\tilde{y}_{i,k} - \tilde{y}_{i,l}) \right\|_2^2 \tag{2}$$

Finally, the global model f_θ is obtained using a voting strategy:

$$f_\theta = \left(f_{\theta_k}^k\right)_{k=1}^K \text{ with } f_{\theta_k}^k(\tilde{x}_i) = \sum_{l \neq k} f_{\theta_{k,l}}^{k,l}(\tilde{x}_i) \tag{3}$$

For the specialized regression models $f_{\theta_{k,l}}^{k,l}$, we use Decision Trees since they gave the best results among all tested models. The meta-model is called Pairwise Decision Tree (PR-DTree) in the following.

Table 1: Meta-features extracted from the literature

Name	Description	Variants
Samples	Number of samples (N)	-
Attributes	Number of attributes (d)	-
Dim	Dimensionality (d/N)	-
NumAtt	Number of numerical attributes (p)	-
CatAtt	Number of categorical attributes (q)	-
NumOnCat	p/q	-
MeansNumAtt	Means of numerical attributes	$min, q_1, mean, q_3, max$
StdNumAtt	Standard deviations of numerical attributes	$min, q_1, mean, q_3, max$
Covariance	Covariance between numerical attributes	$min, q_1, mean, q_3, max$
CardCatAtt	Cardinal of categorical attributes	$min, q_1, mean, q_3, max$
EntropyCatAtt	Entropy of categorical attributes	$min, q_1, mean, q_3, max$

Table 2: Proposed Meta-features

Name	Description	Variants
MeansSqNumAtt	Means of squared numerical attributes	$min, q_1, mean, q_3, max$
StdSqNumAtt	Standard deviations of squared numerical attributes	$min, q_1, mean, q_3, max$
MeansIntProdNumAtt	Means of internal product of numerical attributes	$min, q_1, mean, q_3, max$
StdIntProdNumAtt	Std of internal product of numerical attributes	$min, q_1, mean, q_3, max$
StdFreqCatAtt	Std of frequencies of categorical attribute values	$min, q_1, mean, q_3, max$
MutualInfoCatAtt	Mutual information between categorical attributes	$min, q_1, mean, q_3, max$

4 EXPERIMENTS

4.1 Clustering algorithms and clustering evaluation

To validate the proposed approach, we consider the K-Prototypes [18] and H-AVG [23] (Hierarchical Clustering with average linkage) algorithms. We chose these two algorithms since they are well known and are among the most impacted by the choice of the similarity measure pair according to [10]. For H-AVG, as suggested in [10], we replace the Gower similarity by the following similarity measure:

$$s(x_i, x_j) = (1 - w) \cdot s^n(x_i^n, x_j^n) + w \cdot s^c(x_i^c, x_j^c) \text{ with } w \in [0, 1] \quad (4)$$

Where x_i and x_j are two observations. x_i^n and x_i^c are the numeric and categorical parts of x_i respectively. s^n and s^c are two numeric and categorical similarity measures respectively. The main parameters of these algorithms are the number of clusters and the combination weight of the numeric and categorical similarity measures. For the number of clusters, we use the number of classes in the ground truth. The combination weight is determined using a grid search strategy. Clustering performance is evaluated with the clustering accuracy (CA) [2], one of the most commonly used clustering evaluation metrics. Let $(\hat{y}_1, ..., \hat{y}_{N_X})$ and $(y_1, ..., y_{N_X})$ be respectively the obtained cluster labels and the ground truth labels for a given dataset X, with $N_X = |X|$. Let k be the number of clusters. The clustering accuracy is defined by:

$$CA = \max_{\sigma} \frac{1}{N_X} \sum_{i=1}^{N_X} 1\left(\sigma(\hat{y}_i) = y_i\right) \quad (5)$$

Where σ is a permutation of $\{1, ..., k\}$ that maps each cluster label to a corresponding class label in the ground truth. $1\left(\sigma(\hat{y}_i) = y_i\right) = 1$ if $\sigma(\hat{y}_i) = y_i$, 0 otherwise. The accuracy score takes values in $[0, 1]$, and greater values indicate a better match between the found clusters and the ground truth labels.

4.2 Similarity measures

We consider the same similarity measures used by Diop *at al.* [10] in their study on the impact of similarity measures on MDC. There are ten similarity measures for numeric data and twelve for categorical data resulting in 120 similarity measure pairs. The **similarity measures for numeric data** are *Euclidean distance*, *Manhattan distance*, *Chebyshev distance*, *Squared Euclidean distance*, *Canberra distance*, *Mahalanobis distance*, *Cosine dissimilarity*, *Pearson dissimilarity*, *Loretzian distance* and *Divergence distance*. The **similarity measures for categorical data** are *Hamming distance* or *Overlap* similarity, *Eskin*, *Occurrence Frequency*, *Inverse Occurrence Frequency*, *co-occurence* based similarity, *Jaccard*, *Dice*, *Klusinski*, *Rogerstanimoto*, *Russellrao*, *Sokalmichener*, and *Sokalsneath*. Please refer to [10] for a more detailed description of these measures.

Table 3: Original datasets description: for each statistic, we show its minimum and maximum values as well as its three quartiles

	# of attributes	# of samples	# of numeric attributes	# of categorical attributes	# of classes
min	3	50	1	2	2
25%	9	235	3	3	2
50%	17	1.3K	7	6	2
75%	33	9.8K	17	15	5
max	1.6K	1.5M	1.6K	137	48

4.3 Datasets

We have selected various mixed datasets from the OpenML platform [28]. We only considered mixed datasets without missing values that have already been used for clustering and having ground truth labels. It is important to note that labels are not used during the clustering process. They are only used to evaluate the clustering results. We manually filtered redundant datasets and ended with 84 datasets. Table 3 presents some descriptive statistics about them. Due to computational constraints, original datasets with a large number of observations are down-sampled to a maximum of 2000 observations. Furthermore, to have a more representative meta-dataset and improve the generalization performances of the meta-models on new and unknown datasets, we generated for each of the 84 original datasets, 10 augmented datasets by down-sampling on row and columns (while keeping at least one numeric and one categorical attribute).

It is important to note that the augmented datasets are used only for training the meta-models not for testing. Moreover, when a dataset is in the test set, all its augmentations are removed from the training set to avoid overfitting.

Finally, since it is not relevant to evaluate clustering performances using a ground truth that does not reflect any structural information about the considered dataset or fails to align with any achievable partitioning by the considered clustering algorithm, we consider for each clustering algorithm, only datasets for which at least one similarity measure pair achieves high accuracy ($CA \geq 0.7$). As a result, we have 404 datasets for H-AVG (36 original + 368 augmented) and 342 for K-Prototypes (31 original + 311 augmented).

4.4 Baselines

We compare the proposed SMR system to the following baselines:

- **Random Baseline (RB)**: This baseline uses a random similarity measure pair.
- **Literature Baseline (LB)**: This baseline uses the similarity measure pair used in the literature for the considered clustering algorithm and can be considered as the default choice of a data scientist when there is no tool to select suitable similarity measures automatically. For K-Prototypes [18], the pair (*Squared Euclidean*, *Hamming*) is used. For H-AVG, the Gower similarity used in [23] is equivalent to the pair (*Manhattan*, *Hamming*).
- **Average Ranking Baseline (ARB)**: This baseline ranks the similarity measure pairs according to their average accuracy on all datasets in the meta-dataset.

4.5 Evaluation Metrics

Let $s_{\hat{\pi}_1} \succ ... \succ s_{\hat{\pi}_K}$ and $s_{\pi_1} \succ ... \succ s_{\pi_K}$ be respectively the ranking predicted by the SMR system and the true ranking of the similarity measure pairs on a given dataset X, where $\hat{\pi}$ and π are two permutations of the set $\{1, .., K\}$. $s_k \succ s_l$ indicate that the similarity measure pair s_k is better than s_l. We consider two evaluation metrics:

- **top-k accuracy**: it evaluates the quality of the k top-ranked similarity measure pairs.

$$\text{top-}k = \frac{\max_{j=1}^{k} CA(X, s_{\hat{\pi}_j})}{CA(X, s_{\pi_1})} \tag{6}$$

  Where $CA(X, s_j)$ denotes the accuracy of the similarity pair s_j on dataset X for the considered clustering algorithm.
- **NDCG** (Normalized Discounted Cumulative Gain): A metric based on the notion of Discounted Cumulative Gain (DCG), which evaluates the quality and the ranking of the top-ranked similarity measure pairs. The DCG at rank k is defined by:

$$DCG@k = \sum_{j=1}^{k} \frac{rel(X, s_{\hat{\pi}_j})}{log_2(j+1)}, rel(X, s_j) = \left(\frac{CA(X, s_j)}{CA(X, s_{\pi_1})}\right)^{\alpha}, \alpha > 0 \tag{7}$$

  $rel(X, s_j)$ is the relevance of s_j for X. α is a positive number that controls how the relevance decreases when the performance of s_j decreases relative to the performance of the best pair. We use $\alpha = 4$ in the experiments. The NDCG is then defined by normalizing the DCG with the Ideal DCG (IDCG), which corresponds to the DCG of the true ranking:

$$NDCG@k = \frac{DCG@k}{IDCG@k} \tag{8}$$

4.6 Results

To train and evaluate the meta-models, we consider a leave-one-out (LOO) procedure. We realize N_{OD} (the number of original datasets) iterations, such that at each iteration, one original dataset is selected for testing while all remaining datasets (original and augmented) except the augmentations of the selected dataset are used for training. During the training, the hyper-parameters of the different meta-models are defined using a grid search cross-validation strategy and keeping the set of hyper-parameters that maximize the mean top-1 accuracy.

Table 4 shows the mean and standard deviation (std) of the top-k accuracy on the original datasets for different values of k. The meta-models outperform the baselines for the different values of k. For the H-AVG algorithm, the KNN model outperforms the PR-DTree model for $k = 1$, while the latter performs better for higher values of k. For the K-Prototypes algorithm, the PR-DTree model yields the best performances for the different values of k. Interestingly, for $k = 10$ both models yield a mean top-k accuracy close to or higher than 0.95 for the two algorithms, showing the ability of the meta-models to identify top-performing similarity measure pairs.

Table 5 shows for different values of k, the $NDCG@k$ (mean and std) of the ranking predicted by the meta-models compared to the $NDCG@k$ of the ARB. The two meta-models outperform the ARB for the different values of k indicating that the model

Table 4: Mean and std of the top-k accuracy for different values of k

	H-AVG			K-Prototypes		
Method	top-1	top-5	top-10	top-1	top-5	top-10
LB	0.937±0.08	-	-	0.837±0.13	-	-
RB	0.929±0.08	-	-	0.853±0.09	-	-
ARB	0.924±0.1	0.933±0.1	0.945±0.09	0.874±0.12	*0.930±0.11*	0.936±0.11
KNN	**0.960±0.06**	*0.969±0.06*	*0.977±0.05*	*0.894±0.11*	0.922±0.11	*0.944±0.08*
PR-DTree	*0.953±0.06*	**0.971±0.06**	**0.987±0.03**	**0.905±0.1**	**0.936±0.09**	**0.956±0.08**

Table 5: Mean and std of the $NDCG@k$ metric for different values of k

	H-AVG			K-Prototypes		
Method	NDCG@5	NDCG@10	NDCG@20	NDCG@5	NDCG@10	NDCG@20
ARB	0.786±0.26	0.787±0.24	0.801±0.21	0.703±0.25	0.717±0.24	0.724±0.21
KNN	*0.831±0.2*	*0.837±0.19*	*0.856±0.17*	*0.717±0.25*	**0.741±0.23**	**0.760±0.19**
PR-DTree	**0.840±0.18**	**0.849±0.17**	**0.864±0.15**	**0.721±0.22**	*0.723±0.21*	*0.738±0.18*

Table 6: *pvalues* of the Wilcoxon signed-rank tests

	H-AVG				K-Prototypes			
	top-1 accuracy			$NDCG@10$	top-1 accuracy			$NDCG@10$
	LB	RB	ARB	ARB	LB	RB	ARB	ARB
KNN	0.012	0.004	0.024	0.014	0.029	0.007	0.084	0.383
PR-DTree	0.434	0.016	0.052	0.005	0.015	0.011	0.124	0.307

better identifies the top-performing pairs and better predicts their corresponding ranks.

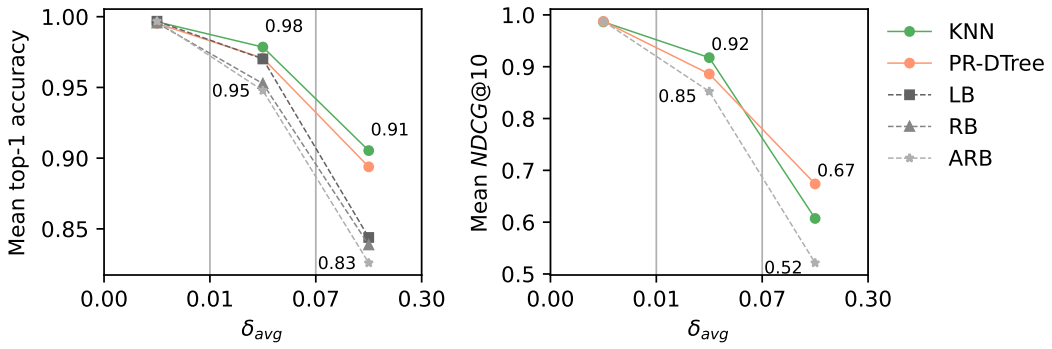
To confirm the previous observations, we use the Wilcoxon signed-rank test [30] to test if the superiority of the meta-models over the baselines is statistically significant. The Wilcoxon signed-rank test is a non-parametric statistical hypothesis test used to compare two related paired samples. For clarity, we only consider the top-1 accuracy and the $NDCG@10$ metrics. For each pair (M, B) of a meta-model M and a baseline B, we test the alternative hypothesis H_1 = `"M is better than B"` at the 0.05 level of significance. The *p_values* are shown in table 6. For H-AVG, the *p_values* obtained with the KNN meta-model are less than the significance level (0.05) for all baselines. So, we can conclude that the KNN meta-model is significantly better than all baselines for the H-AVG algorithm. For K-Prototypes, the two meta-models perform significantly better than the literature and random baselines ($p\_value \leq 0.05$). However, for the ARB, the *p_values* are greater than 0.05.

To further analyze the results, we introduce the notion of the "difficulty" of a dataset. We consider a dataset X as difficult when a randomly chosen pair has a low probability of being close in accuracy to the best pair. This means that the average value δ_{avg} (equation 9) of the accuracy difference δ_k between any given pair s_k and the best pair for that dataset is large. Inversely, the dataset is considered easy when δ_{avg} is small (i.e. all similarity measure pairs have similar accuracy to the best pair). The more the dataset is difficult, the more it is important to choose the similarity measure pair correctly.

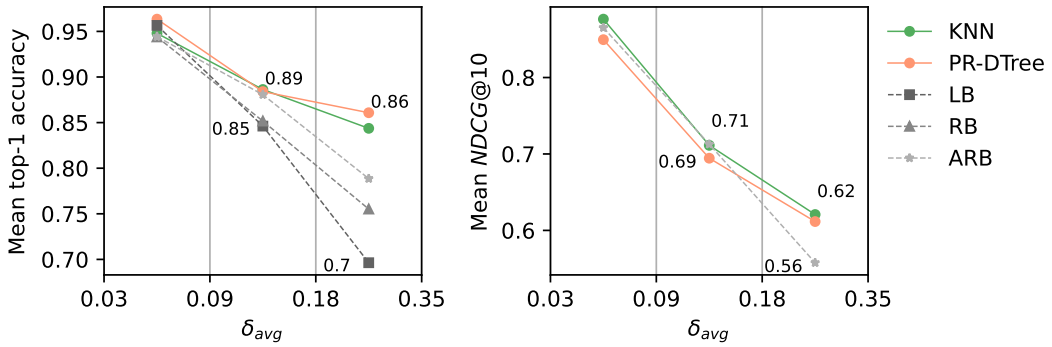
$$\delta_{avg} = \frac{1}{K}\sum_{k=1}^{K}\delta_k, \text{ where } \delta_k = 1 - \frac{CA(X, s_k)}{\max_{j=1}^{K} CA(X, s_j)} \tag{9}$$

For each clustering algorithm, we divide the values of δ_{avg} into three intervals of equal number of datasets. Figure 2 shows, for each interval, the mean top-1 accuracy and the mean $NDCG@10$ of the datasets in the interval. As expected, for small values of δ_{avg}, the meta-models and the baselines yield similar results. However, when δ_{avg} increases, i.e. when the choice of the similarity measures pair is more important, the meta-models outperform the baselines and the difference also increases with δ_{avg}. So, the proposed approach seems even more interesting for datasets for which the choice of the similarity measure pair really matters.

Finally, to analyze the impact of the additional meta-features, we evaluate the performance of the meta-models when using the meta-features from the literature only compared to the performances when we add the proposed meta-features to those used in the literature. Tables 7 and 8 show the obtained results for KNN and PR-DTree meta-models respectively. For PR-DTree, adding the proposed meta-features improves the mean top-1 accuracy and the mean $NDCG@10$ for the two clustering algorithms compared to using the literature meta-features only. For KNN, adding the proposed meta-features leads to better or worse results depending on the clustering algorithm and the evaluation metric. This indicates that the proposed meta-features extract useful information about



(a) H-AVG



(b) K-Prototypes

Figure 2: Mean top-1 accuracy and mean *NDCG*@10 according to δ_{avg}. The values of δ_{avg} are divided into 3 intervals. For each interval, the mean top-1 accuracy and mean *NDCG*@10 of the datasets in this interval are represented.

Table 7: Performances of KNN according to the considered meta-features

	H-AVG		K-Prototypes	
Meta-features	top-1 accuracy	NDCG@10	top-1 accuracy	NDCG@10
Literature	0.954±0.07	**0.84±0.18**	**0.91±0.11**	0.717±0.21
Literature + Proposed	**0.960±0.06**	0.837±0.19	0.894±0.11	**0.741±0.23**

Table 8: Performances of PR-DTree according to the considered meta-features

	H-AVG		K-Prototypes	
Meta-features	top-1 accuracy	NDCG@10	top-1 accuracy	NDCG@10
Literature	0.949±0.07	0.834±0.17	0.869±0.12	0.691±0.23
Literature + Proposed	**0.953±0.06**	**0.849±0.17**	**0.905±0.1**	**0.723±0.21**

the datasets, but a meta-feature selection may be needed to identify the optimal subset of meta-features from the proposed ones and those taken from the literature.

5 DISCUSSION

Our experiments demonstrate that the proposed approach can be used as an effective solution for similarity measures recommendation in the context of MDC. We discuss here some important aspects of this work.

First, one might initially perceive meta-learning as a costly solution since the construction of the meta-dataset during the learning phase needs prior evaluations of the considered clustering algorithm on all datasets using all similarity measure pairs (with a search of the optimal combination weight). For indication, the total computation time for these evaluations is about 8 hours for the Hierarchical Clustering algorithm and 5 days for the K-Prototypes algorithm using parallel execution over the datasets on a platform with 16 CPUs Intel Xeon Gold 6226R CPU@2.90GHz. Note that this

difference between the two algorithms is because distance matrices are used for the Hierarchical Clustering algorithm. So even if each numeric or categorical similarity measure is present in several similarity measure pairs, its distance matrix can be computed only once. It is essential to note that these evaluations are computed only once and the associated computational cost is not a major problem since they only concern the learning phase. Moreover, once the meta-models are trained, their use in practice (inference phase) helps avoid expensive trial-and-error strategies, leading to significant time and energy savings.

Second, this study complements existing studies to support machine learning practitioners in their classical tasks such as algorithm selection, algorithm parameter setting, similarity measures selection, and so on. Our work focuses on this latter task. Once the user has selected a clustering algorithm, our recommendations allow to drastically reduce the needed effort to find suitable similarity measure pairs. Nonetheless, the search for optimal algorithm parameters remains necessary.

It is also important to note that despite utilizing labeled datasets during training (labels were used for evaluation purposes to train the meta-models to identify suitable similarity measure pairs), our approach is designed for an unsupervised setting, as the meta-features used by the meta-models do not incorporate labels.

Finally, we think important improvements can be made, particularly for the meta-features by understanding how they are involved in the meta-models' predictions and identifying the most important ones. Also, there is a lack of meta-features adapted to the context of SMR for mixed data in the literature. Future works in this direction will be of high interest.

6 CONCLUSION

In this paper, we proposed a meta-learning approach for similarity measures recommendation in mixed data clustering. This approach exploits the experience we get from evaluating a given clustering algorithm on various mixed datasets with different similarity measure pairs to learn to predict the ranking of the similarity measure pairs according to the datasets' characteristics. We validated the approach on two commonly used mixed data clustering algorithms: K-Prototypes and Hierarchical Clustering. Our experiments show that the recommended similarity measure pairs for these two algorithms, perform better than the considered baselines (including classically used similarity measure pairs in the literature), especially for datasets highly impacted by the choice of the similarity measure pair.

For future works, we plan to conduct more in-depth studies about the meta-features to identify and design meta-features that better characterize mixed datasets or end-user tasks.

ACKNOWLEDGMENTS

This work was supported by a grant overseen by the French National Research and Technology Association (ANRT) as part of the CIFRE (2020/0868) with the SolutionData Group company (France).

REFERENCES

[1] Amir Ahmad and Lipika Dey. 2007. A k-mean clustering algorithm for mixed numeric and categorical data. *Data & Knowledge Engineering* 63, 2 (Nov. 2007), 503–527. https://doi.org/10.1016/j.datak.2007.03.016

[2] Amir Ahmad and Shehroz S. Khan. 2019. Survey of State-of-the-Art Mixed Data Clustering Algorithms. *IEEE Access* 7 (2019), 31883–31902. https://doi.org/10.1109/ACCESS.2019.2903568

[3] Guilherme Alves, Miguel Couceiro, and Amedeo Napoli. 2019. Similarity Measure Selection for Categorical Data Clustering. (Dec. 2019). https://hal.archives-ouvertes.fr/hal-02399640 preprint.

[4] Fatima Barcelo-Rico and Jose-Luis Diez. 2012. Geometrical codification for clustering mixed categorical and numerical databases. *Journal of Intelligent Information Systems* 39, 1 (Aug. 2012), 167–185. https://doi.org/10.1007/s10844-011-0187-y

[5] Sudha Bishnoi and B. K. Hooda. 2020. A survey of distance measures for mixed variables. *International Journal of Chemical Studies* 8 (July 2020), 338–343. https://doi.org/10.22271/chemi.2020.v8.i4f.10087

[6] Weksi Budiaji and Friedrich Leisch. 2019. Simple K-Medoids Partitioning Algorithm for Mixed Variable Data. *Algorithms* 12, 9 (Sept. 2019), 177. https://doi.org/10.3390/a12090177 Number: 9 Publisher: Multidisciplinary Digital Publishing Institute.

[7] Thomas M Cover. 1999. *Elements of information theory.* John Wiley & Sons.

[8] Marcilio de Souto, Ricardo Prudêncio, Rodrigo Soares, Daniel Araujo, Ivan Costa, Teresa Ludermir, and Alexander Schliep. 2008. Ranking and Selecting Clustering Algorithms Using a Meta-Learning Approach. In *Proceedings of the International Joint Conference on Neural Networks*. 3729–3735. https://doi.org/10.1109/IJCNN.2008.4634333

[9] Shifei Ding, Mingjing Du, Tongfeng Sun, Xiao Xu, and Yu Xue. 2017. An entropy-based density peaks clustering algorithm for mixed type data employing fuzzy neighborhood. *Knowledge-Based Systems* 133 (Oct. 2017), 294–313. https://doi.org/10.1016/j.knosys.2017.07.027

[10] Abdoulaye Diop, Nabil El Malki, Max Chevalier, Andre Peninou, and Olivier Teste. 2022. Impact of similarity measures on clustering mixed data. In *Proceedings of the 34th International Conference on Scientific and Statistical Database Management (SSDBM '22)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3538712.3538742

[11] Mingjing Du, Shifei Ding, and Yu Xue. 2017. A novel density peaks clustering algorithm for mixed data. *Pattern Recognition Letters* 97 (Oct. 2017), 46–53. https://doi.org/10.1016/j.patrec.2017.07.001

[12] Pierpaolo D'Urso and Riccardo Massari. 2019. Fuzzy clustering of mixed data. *Information Sciences* 505 (Dec. 2019), 513–534. https://doi.org/10.1016/j.ins.2019.07.100

[13] Daniel Gomes Ferrari and Leandro Nunes de Castro. 2015. Clustering algorithm selection by meta-learning systems: A new distance-based problem characterization and ranking combination methods. *Information Sciences* 301 (April 2015), 181–194. https://doi.org/10.1016/j.ins.2014.12.044

[14] Itay Gabbay, Bracha Shapira, and Lior Rokach. 2021. Isolation forests and landmarking-based representations for clustering algorithm recommendation using meta-learning. *Information Sciences* 574 (Oct. 2021), 473–489. https://doi.org/10.1016/j.ins.2021.06.033

[15] Sandhya Harikumar and Surya Pv. 2015. K-Medoid Clustering for Heterogeneous DataSets. *Procedia Computer Science* 70 (Jan. 2015), 226–237. https://doi.org/10.1016/j.procs.2015.10.077

[16] Chung-Chian Hsu, Chin-Long Chen, and Yu-Wei Su. 2007. Hierarchical clustering of mixed data based on distance hierarchy. *Information Sciences* 177, 20 (Oct. 2007), 4474–4492. https://doi.org/10.1016/j.ins.2007.05.003

[17] J.Z. Huang, M.K. Ng, Hongqiang Rong, and Zichen Li. 2005. Automated variable weighting in k-means type clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27, 5 (May 2005), 657–668. https://doi.org/10.1109/TPAMI.2005.95

[18] Zhexue Huang. 1998. Extensions to the k-Means Algorithm for Clustering Large Data Sets with Categorical Values. *Data Mining and Knowledge Discovery* 2, 3 (Sept. 1998), 283–304. https://doi.org/10.1023/A:1009769707641

[19] Eyke Hüllermeier, Johannes Fürnkranz, Weiwei Cheng, and Klaus Brinker. 2008. Label ranking by learning pairwise preferences. *Artificial Intelligence* 172, 16 (Nov. 2008), 1897–1916. https://doi.org/10.1016/j.artint.2008.08.002

[20] Jinchao Ji, Tian Bai, Chunguang Zhou, Chao Ma, and Zhe Wang. 2013. An improved k-prototypes clustering algorithm for mixed numeric and categorical data. *Neurocomputing* 120 (Nov. 2013), 590–596. https://doi.org/10.1016/j.neucom.2013.04.011

[21] Yonggu Lee, Chulwung Park, and Shinjin Kang. 2023. Deep Embedded Clustering Framework for Mixed Data. *IEEE Access* 11 (2023), 33–40. https://doi.org/10.1109/ACCESS.2022.3232372

[22] Felix Mbuga and Cristina Tortora. 2022. Spectral Clustering of Mixed-Type Data. *Stats* 5, 1 (March 2022), 1–11. https://doi.org/10.3390/stats5010001 Number: 1 Publisher: Multidisciplinary Digital Publishing Institute.

[23] G. Philip and B. S. Ottaway. 1983. Mixed Data Cluster Analysis: An Illustration Using Cypriot Hooked-Tang Weapons. *Archaeometry* 25, 2 (1983), 119–133. https://doi.org/10.1111/j.1475-4754.1983.tb00671.x

[24] Bruno Almeida Pimentel and André C. P. L. F. de Carvalho. 2019. A new data characterization for selecting clustering algorithms using meta-learning. *Information Sciences* 477 (March 2019), 203–219. https://doi.org/10.1016/j.ins.2018.10.043

[25] Bruno Almeida Pimentel and André C. P. L. F. de Carvalho. 2020. A Meta-learning approach for recommending the number of clusters for clustering algorithms. *Knowledge-Based Systems* 195 (May 2020), 105682. https://doi.org/10.1016/j.knosys.2020.105682

[26] Yannis Poulakis, Christos Doulkeridis, and Dimosthenis Kyriazis. 2024. A Survey on AutoML Methods and Systems for Clustering. *ACM Trans. Knowl. Discov. Data* 18, 5, Article 120 (feb 2024), 30 pages. https://doi.org/10.1145/3643564

[27] Joaquin Vanschoren. 2018. Meta-Learning: A Survey. https://doi.org/10.48550/arXiv.1810.03548

[28] Joaquin Vanschoren, Jan N. van Rijn, Bernd Bischl, and Luis Torgo. 2013. OpenML: networked science in machine learning. *SIGKDD Explorations* 15, 2 (2013), 49–60.

[29] Milan Vukicevic, Sandro Radovanovic, Boris Delibasic, and Milija Suknovic. 2016. Extending meta-learning framework for clustering gene expression data with component-based algorithm design and internal evaluation measures. *International Journal of Data Mining and Bioinformatics* 14, 2 (Jan. 2016), 101–119. https://doi.org/10.1504/IJDMB.2016.074682 Publisher: Inderscience Publishers.

[30] Frank Wilcoxon. 1945. Individual Comparisons by Ranking Methods. *Biometrics Bulletin* 1, 6 (1945), 80–83. https://doi.org/10.2307/3001968 Publisher: [International Biometric Society, Wiley].

[31] Yiqun Zhang and Yiu-Ming Cheung. 2023. Graph-Based Dissimilarity Measurement for Cluster Analysis of Any-Type-Attributed Data. *IEEE Transactions on Neural Networks and Learning Systems* 34, 9 (Sept. 2023), 6530–6544. https://doi.org/10.1109/TNNLS.2022.3202700

[32] Yangming Zhou, Yangguang Liu, Jiangang Yang, Xiaoqi He, and Liangliang Liu. 2014. A Taxonomy of Label Ranking Algorithms. *Journal of Computers* 9 (March 2014). https://doi.org/10.4304/jcp.9.3.557-565

[33] Chengzhang Zhu, Qi Zhang, Longbing Cao, and Arman Abrahamyan. 2020. Mix2Vec: Unsupervised Mixed Data Representation. In *2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA)*. 118–127. https://doi.org/10.1109/DSAA49011.2020.00024

[34] Xiaoyan Zhu, Yingbin Li, Jiayin Wang, Tian Zheng, and Jingwen Fu. 2020. Automatic Recommendation of a Distance Measure for Clustering Algorithms. *ACM Transactions on Knowledge Discovery from Data* 15, 1 (Dec. 2020), 7:1–7:22. https://doi.org/10.1145/3418228