Affinity Learning for Mixed Data Clustering

Nan Li and **Longin Jan Latecki**
Department of Computer and Information Sciences
Temple University, Philadelphia, USA
{nan.li,latecki}@temple.edu

Abstract

In this paper, we propose a novel affinity learning based framework for mixed data clustering, which includes: how to process data with mixed-type attributes, how to learn affinities between data points, and how to exploit the learned affinities for clustering. In the proposed framework, each original data attribute is represented with several abstract objects defined according to the specific data type and values. Each attribute value is transformed into the initial affinities between the data point and the abstract objects of attribute. We refine these affinities and infer the unknown affinities between data points by taking into account the interconnections among the attribute values of all data points. The inferred affinities between data points can be exploited for clustering. Alternatively, the refined affinities between data points and the abstract objects of attributes can be transformed into new data features for clustering. Experimental results on many real world data sets demonstrate that the proposed framework is effective for mixed data clustering.

1 Introduction

Clustering is the task of partitioning the data objects into a set of groups (clusters) such that objects in the same group are similar, while objects in different groups are dissimilar. It is one of the most fundamental problems in data mining and machine learning. Numerous algorithms have been developed for clustering. Most of them are designed to handle data with only one type of attributes, e.g. continuous, categorical or ordinal. Mixed data clustering has received relatively less attention, despite the fact that data with mixed types of attributes are common in real applications.

For mixed data clustering, one of the greatest challenges is how to measure the affinities or distances between data points. One of the most straightforward methods for processing mixed data is the so-called 1-hot or 1-of-K encoding. A categorical attribute with K distinct values is encoded to K $0-1$ binary attributes. Each categorical attribute value is transformed into a 1 on its corresponding binary attribute. Then they are treated just like continuous attributes. The more formal Gower's similarity coefficient [Gower, 1971] and its extensions [Legendre and Legendre, 1998; Podani, 1999] compute the partial affinity between two data points on each attribute according to the data type, and then aggregate all of them into a composite similarity measure. Such methods are widely used in practice. However, they essentially compute the affinity or distance "locally" between two data points, without considering the attribute values of other data points. This may result in missing some intrinsic information. For example, in many real world data sets, some values of a categorical attribute are inherently related. Such information would be missed by similarity measures like Gower's coefficient, which simply assume different categories are totally independent and unrelated.

In this paper, we propose a novel affinity learning based framework for mixed data clustering. It includes how to process data with mixed-type attributes, how to learn affinities between data points, and how to exploit the learned affinities for clustering.

First, each original attribute is represented with several abstract objects defined according to the specific data type and values. Each attribute value is then transformed into the initial affinities between the data point and the abstract objects of attribute. For categorical attributes, each category is defined as an abstract object. Its affinities to the data points in this category are initialized to a constant value. For each continuous attribute, two abstract objects are defined to represent its minimum and maximum values. Their initial affinities to each data point are transformed from the individual continuous attribute value with a novel method. For ordinal attributes, all possible values are first ranked and then replaced by their ranks. The new ordinal attributes are processed as continuous attributes.

After the data processing, we obtain a bipartite graph consisting of the data points, the abstract objects of attributes, and the initial affinities between them. The next step is to learn new affinities, including inferring the unknown affinities and refining the known affinities. We adopt the algorithm proposed in [Li and Latecki, 2015], which essentially implements the von Neumann kernel [Kandola *et al.*, 2003] from the perspective of transitive inference confidence. Specifically, the new affinities are learned according to the transitive property of the affinitive relation. All the initial affinities are scaled with a common scaling factor. Any transitive infer-

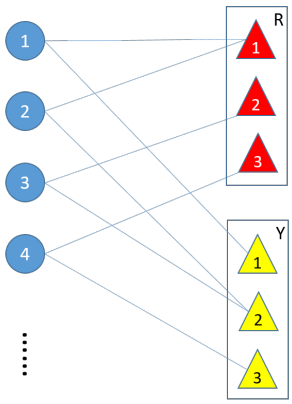

Figure 1: An illustration of data point connections via their attribute values. Blue circles represent data points. Rectangles represent categorical attributes, each has three distinct attribute values.

ence process without self-loops is considered to be effective to reveal the two objects are affinitive. The confidence of such an inference process is quantified as the product of the related scaled affinity values. In general, there can be an infinite number of distinct transitive inference processes between two objects. The confidence of all these inference processes are added up to be the new affinity between two objects. The details of this affinity learning algorithm are presented in Section 3.2.

In comparison to Gower's similarity coefficient and its extensions, our affinity learning method shares the similar idea of aggregating partial affinities on individual attributes into an overall measure. But the significant difference is that our affinities are computed "globally" by taking into account the interconnections among the attribute values of all data points, not just between the two data points. This is illustrated in Figure 1. The numbered blue circles represent data points, i.e. $\{\boldsymbol{x_1}, \boldsymbol{x_2}, \boldsymbol{x_3}, \boldsymbol{x_4}\}$. The two rectangles represent categorical attributes R and Y, each of which has three distinct attribute values. If we compute the affinity S_{ij} just between the two data points $\boldsymbol{x_i}$ and $\boldsymbol{x_j}$, like Gower's coefficient does, then S_{13} and S_{14} are both 0, because they don't have any common attribute values. However, because of the existence of $\boldsymbol{x_2}$, which shares one common attribute value with $\boldsymbol{x_1}$ and $\boldsymbol{x_3}$ respectively, it's intuitive to infer that $\boldsymbol{x_1}$ is more affinitive to $\boldsymbol{x_3}$ in comparison to $\boldsymbol{x_4}$. Our affinity learning method can capture such information by taking into account all transitive inference processes, including $\boldsymbol{x_1} \rightarrow R_1 \rightarrow \boldsymbol{x_2} \rightarrow Y_2 \rightarrow \boldsymbol{x_3}$.

The inferred affinities between data points can be used by many clustering algorithms. Alternatively, the refined affinities between data point and the abstract objects of attribute can be transformed into new data features. With such features, any algorithms can be used for clustering.

The mixed data clustering algorithms derived from the proposed framework achieve superior performance on many real world data sets. The details of the experimental evaluation are presented in Section 4.

2 Related Work

For mixed data clustering, in addition to using 1-hot encoding to obtain continuous features or Gower's coefficient [Gower, 1971] and its extensions [Legendre and Legendre, 1998; Podani, 1999] to measure the similarities between data points, as introduced in Section 1, there are also some specially designed clustering algorithms, including k-prototypes [Huang, 1997; 1998], K-means-mixed [Ahmad and Dey, 2007], CAVE [Hsu and Chen, 2007], M-ART [Hsu and Huang, 2008], INTEGRATE [Böhm *et al.*, 2010], INCONCO [Plant and Böhm, 2011], SCENIC [Plant, 2012] and so on. K-prototypes algorithm [Huang, 1997; 1998], which essentially follows the same idea of k-means algorithm, calculates the dissimilarity between two mixed-type objects as a combination of the squared Euclidean distance measure on the numeric attributes and the simple matching dissimilarity measure on the categorical attributes. K-means-mixed [Ahmad and Dey, 2007], like k-prototypes, is also based on the k-means paradigm and combines distance measures computed separately on numeric attributes and categorical attributes. Unlike k-prototypes, k-means-mixed does not assume a binary or a discrete measure between two distinct categorical attribute values but computes the distance as a function of their overall distribution and co-occurrence with other categorical attributes. This idea of computing distances "globally" is similar to ours, but it's only applied within categorical attributes. CAVE [Hsu and Chen, 2007] uses variance to measure the similarity of the numeric part of the data and computes the similarity of the categorical part based on entropy weighted by the distances in the hierarchies. Similarly, the incremental clustering algorithm M-ART [Hsu and Huang, 2008] also computes the distance between two data points according to distance hierarchies associated with the mixed-type attributes. INTEGRATE [Böhm *et al.*, 2010] applies ideas from information theory to implement the k-means paradigm. It models both numerical and categorical attributes with their probability distributions and minimizes a cost function based on the Minimum Description Length principle for clustering. INCONCO [Plant and Böhm, 2011] and SCENIC [Plant, 2012] process mixed-type attributes in a similar way as INTEGRATE. Their main advantage is the capability of modeling and revealing the cluster-specific dependency patterns among the attributes.

To learn affinities between heterogeneous objects of data points and attributes, we adopt the algorithm proposed in [Li and Latecki, 2015], which models the new affinities from the perspective of transitive inference confidence. It essentially implements the von Neumann kernel defined in [Kandola *et al.*, 2003]. The idea of learning semantic similarity between terms from a corpus for measuring similarity between text documents in [Kandola *et al.*, 2003] is similar to our idea of capturing the intrinsic information between attribute values. One significant difference, besides the applications are different, is that we explicitly model the interconnections among data points and attribute values together. There are also some other algorithms can be used for affinity learning , such as [Zhou *et al.*, 2003] and [Yang *et al.*, 2013]. The main reasons we do not choose them include: 1. their row or column normalizations on the initial affinity matrix change the original

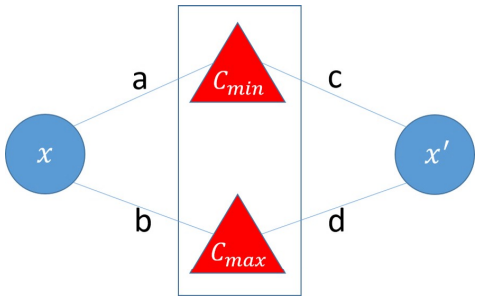

Figure 2: An illustration for explaining the first requirement in equation (1) for transforming a continuous attribute value into initial affinities.

relationships between the heterogeneous objects; and 2. they are not as semantically intuitive and meaningful as the one [Li and Latecki, 2015] we adopt.

3 Our Framework

3.1 Mixed Data Processing

We first transform the data points and their mixed-type attribute values into abstract objects and initial affinities. For categorical attributes, each category is defined as an abstract object. Its affinities to the data points in this category are initialized to 1, while its initial affinities to the rest data points are 0. This is similar to the 1-hot encoding. For each continuous attribute C, two abstract objects are defined to represent its minimum and maximum values, i.e. C_{min} and C_{max}. The attribute value x_C of the data point $\boldsymbol{x}$ is transformed into two initial affinities to the abstract objects of C_{min} and C_{max}. Suppose they are $S_{\boldsymbol{x},C_{min}} = a$ and $S_{\boldsymbol{x},C_{max}} = b$, we have two requirements,

$$\begin{cases} a^2 + b^2 = 1 \\ (C_{min} \times a + C_{max} \times b)/(a+b) = x_C \end{cases} \tag{1}$$

To understand the first requirement, consider the illustration in Figure 2. a, b, c, d on the edges represent the initial affinities of two data points $\boldsymbol{x}$ and $\boldsymbol{x'}$ to the abstract objects of C_{min} and C_{max} respectively. The diffusion based affinity learning algorithms essentially compute the affinity $S_{\boldsymbol{x},\boldsymbol{x'}}$ as

$$S_{\boldsymbol{x},\boldsymbol{x'}} = a \times c + b \times d \tag{2}$$

If $\boldsymbol{x}$ and $\boldsymbol{x'}$ have the same attribute value x_C on C, obviously their affinities to C_{min} and C_{max} should be the same, i.e. $a = c$ and $b = d$. It's also obvious to require that $S_{\boldsymbol{x},\boldsymbol{x'}}$ to be a constant, e.g. 1, no matter what the attribute value x_C is. Therefore, we get the first requirement,

$$S_{\boldsymbol{x},\boldsymbol{x'}} = a \times c + b \times d = a \times a + b \times b = 1 \tag{3}$$

The second requirement makes sure the original attribute value can be restored from the transformed affinities.

Specifically, to transform the attribute value x_C of $\boldsymbol{x}$ into the initial affinities, x_C is first scaled with the Min-Max normalization.

$$x'_C = \frac{x_C - C_{min}}{C_{max} - C_{min}} \tag{4}$$

The scaled attribute value x'_C is in range $[0, 1]$, i.e. $C'_{min} = 0$ and $C'_{max} = 1$. We have

$$\begin{cases} a^2 + b^2 = 1 \\ (0 \times a + 1 \times b)/(a+b) = x'_C \end{cases} \tag{5}$$

Solve the system of equations, we get the affinity transformation formula as

$$\begin{cases} a = \sqrt{(1 - x'_C)^2/(2 \times {x'_C}^2 - 2 \times x'_C + 1)} \\ b = \sqrt{{x'_C}^2/(2 \times {x'_C}^2 - 2 \times x'_C + 1)} \end{cases} \tag{6}$$

In this way, if two data points have the same value on a continuous attribute, their partial affinity inferred by the diffusion based affinity learning algorithm described below based on this agreement is always the same, no matter what the value is.

For ordinal attributes, all possible values are first ranked and then replaced by their ranks. The new ordinal attributes are processed as continuous attributes.

If an attribute value of $\boldsymbol{x}$ is missing, the related initial affinities are all set to 0.

3.2 Affinity Learning

Now we have a bipartite graph consisting of n data points, m abstract objects of attribute, and the initial affinities between them. We construct a nonnegative symmetric affinity matrix $A = (a_{ij})_{\alpha \times \alpha}$, where $\alpha = m + n$.

$$A = \begin{bmatrix} A_{DD} & A_{DC} \\ A_{CD} & A_{CC} \end{bmatrix} \tag{7}$$

where A_{DD} is a $n \times n$ zero matrix indicating that the affinities between data points are unknown; $A_{DC} = A_{CD}^{\intercal}$ is a $n \times m$ matrix consisting of the initial affinities between data points and the abstract objects of attributes; A_{CC} is a $m \times m$ zero matrix indicating that the affinities between abstract objects of attributes are unknown.

The next step is to scale the nonzero entries in A, i.e. the initial affinities, with a common scaling factor Δ which satisfies

$$\Delta > \max(a_{max}, \rho(A)) \tag{8}$$

where a_{max} is the maximum entry of A; $\rho(A)$ is the spectral radius of A.

Each entry a_{ij} of A is scaled with Δ to obtain another matrix $A' = (a'_{ij})_{\alpha \times \alpha}$ where

$$a'_{ij} = \frac{a_{ij}}{\Delta} \tag{9}$$

Obviously, any entry a'_{ij} of A' is less than 1. Also, the spectral radius of A' is less than 1. Therefore,

$$\lim_{l \to \infty} (A')^l = \mathbf{0} \tag{10}$$

Then we compute a matrix A^* as

$$A^* = (I - A')^{-1} \tag{11}$$

where I is the $\alpha \times \alpha$ identity matrix.

Each entry a^*_{ij} of A^* denotes a value,

$$a^*_{ij} = \sum_{l=0}^{\infty} [(A')^l]_{ij} \tag{12}$$

which is the learned affinity between objects i and j.

The inferred affinities between data points are in A^*_{DD}. The refined affinities between data points and abstract objects of attributes are in A^*_{DC}.

To get the scaling factor Δ, we need to calculate the spectral radius $\rho(A)$ of A. With iterative eigenvalue algorithms, it can be done in $\mathcal{O}(\alpha^2)$. Scaling the nonzero entries of A takes $\mathcal{O}(\alpha^2)$. The straightforward computation for inverting the matrix $I - A'$ takes $\mathcal{O}(\alpha^3)$. Advanced algorithms, such as Strassen algorithm, can further reduce the asymptotic computational complexity. Therefore, the straightforward time complexity of our affinity learning algorithm is $\mathcal{O}(\alpha^3)$. However, A and $I - A'$ are usually very sparse. Consequently, the practical efficiency should be much better. We evaluate it on several real data sets with α up to about $30,000$. The details are presented in Section 4

3.3 Clustering with Learned Affinities

In this paper, we use the complete-linkage algorithm for clustering with the inferred affinities between data points in A^*_{DD}. It is one of the agglomerative hierarchical clustering methods. Specifically, in the beginning, each data point is in a cluster of its own. Then these clusters are iteratively combined until the target cluster number is reached. At each step, the two clusters, whose two members (one in each cluster) have the minimum pair-wise affinity, are combined.

Alternatively, the refined affinities of data points to the abstract objects of attributes can be used as new data features. Specifically, in the $n \times m$ matrix A^*_{DC}, each row is considered as a m-dimensional feature vector of the corresponding data point. In this paper, we choose k-means algorithm and complete-linkage algorithm for clustering with such features.

4 Experimental Evaluation

4.1 Experimental Setup

We evaluate the performance of the proposed clustering framework on several real world data sets from the UCI Machine Learning Repository, including 5 mixed-type (Acute Inflammations, Heart Disease, Credit Approval, Contraceptive Method Choice and Adult) and 2 categorical (Soybean and Tic-Tac-Toe Endgame). The detailed information of these data sets is summarized in Table 1.

Each record of Acute Inflammation data set corresponds to the *yes* or *no* diagnoses of two diseases of the urinary system. We transform the two diagnoses into four classes, i.e. *(yes,yes), (yes,no), (no,yes) and (no,no)*. For Adult data set, we only use the training set, which contains $32,561$ records. For fair comparison, we remove the records with missing attribute values. The final data set contains $30,162$ records. We skip the attribute "education", because it is fully expressed by another attribute "education-num". In Credit Approval data set, 37 (about $\%5$) records have one or more missing values. We simply remove them.

The clustering algorithms derived from the proposed framework include: 1. **IA+CL** (**I**nferred **A**ffinities between data points + **C**omplete-**L**inkage algorithm); 2. **FRA+CL** (**F**eature from **R**efined **A**ffinities of the data point to the abstract objects of attributes + **C**omplete-**L**inkage algorithm); 3. **FRA+KM** (**F**eature from **R**efined **A**ffinities of the data point to the abstract objects of attributes + **K**-**M**eans algorithm).

For the three derived clustering algorithms, we vary the scaling factor Δ in equation 9 in the range of $(\max(a_{max}, \rho(A)), 4 \times \max(a_{max}, \rho(A))]$ (see equation (8)) with a step size of 10. The best results achieved by each algorithm in this process are reported. For FRA+CL and FRA+KM, we use the squared Euclidean distance measure.

The comparison algorithms include: 1. **OH+CL** (Feature from **O**ne-**H**ot encoding + **C**omplete-**L**inkage algorithm); 2. **OH+KM** (Feature from **O**ne-**H**ot encoding + **K**-**M**eans algorithm); 3. **GC+CL** (**G**ower's **C**oefficient + **C**omplete-**L**inkage algorithm); 4. **KP** (k-prototypes) [Huang, 1997; 1998]; 5. **KMM** (K-means-mixed) [Ahmad and Dey, 2007]. These algorithms are widely used in practice for mixed data clustering. Some of them are still state-of-the-art in performance. Many recent algorithms, such as [Plant and Böhm, 2011; Plant, 2012] are very complex to be implemented. We are not able to obtain the source code from the authors.

The Gower's coefficient in GC+CL processes ordinal attributes according to Eqs. 2a-b of [Podani, 1999]. For KP (k-prototypes), we scale all numeric attributes to the range of $[0, 1]$ with Min-Max normalization and randomly select k data points without missing values to be the initial prototypes. The parameter γ is varied from 0.5 to 1.5 with a step size of 0.1 for the 5 mixed-type data sets. When using k-means technique, including k-prototypes and K-means-mixed, the maximum number of iterations is set to be 1000 for the Adult data set, which contains much more data, and 100 for the other 6 data sets. Moreover, all the tests are run for 100 times and the average results are reported.

For all the clustering algorithms above, we set the target number of clusters to be the number of classes in each data set. The clustering quality is measured in terms of Jaccard Coefficient, Fowlkes and Mallows Index, and FScore [Jing *et al.*, 2007]. The results are consistent, so only FScore is reported. Suppose k is the class and cluster number, n is the number of data points, n_i and n_j are the numbers of data points in class CLA_i and cluster CLU_j respectively, n_{ij} is the number of data points in both CLA_i and CLU_j, FScore is defined as

$$FScore = \sum_{i=1}^{k} \left(\frac{n_i}{n} \times \max_{1 \le j \le k} \frac{2 \times R_{ij} \times P_{ij}}{R_{ij} + P_{ij}}\right) \tag{13}$$

where $R_{ij} = n_{ij}/n_i$ and $P_{ij} = n_{ij}/n_j$.

All the experiments are implemented in MATLAB R2016a and conducted on a PC with Intel(R) Core(TM) i7 processor up to 3.4 GHz and 16GB RAM.

Table 1: Data Sets for Experimental Evaluation (number of different types of attributes, number of instances and number of classes)

Data set	Continuous	Categorical	Ordinal	#Instance	#Class
Acute Inflammations	1	5	-	120	4
Heart Disease	6	6	1	270	2
Credit Approval	6	9	-	690	2
Contraceptive Method Choice	2	7	-	1,473	3
Adult	6	8	-	48,842	2
Soybean	-	35	-	47	4
Tic-Tac-Toe Endgame	-	9	-	958	2

Table 2: Clustering Results (FScore on AI: Acute Inflammations; HD: Heart Disease; CA: Credit Approval; CMC: Contraceptive Method Choice; Adult; Soybean; TTT: Tic-Tac-Toe Endgame)

	AI	HD	CA	CMC	Adult	Soybean	TTT
IA+CL	**0.92**	0.78	**0.78**	**0.52**	**0.75**	**1**	0.71
FRA+CL	**0.92**	**0.79**	0.75	0.51	0.73	**1**	**0.76**
FRA+KM	0.80	**0.79**	0.70	0.44	0.73	0.89	0.58
GC+CL	**0.92**	0.71	0.63	0.47	0.58	**1**	0.68
OH+CL	0.76	0.63	0.64	0.48	0.69	**1**	0.68
OH+KM	0.72	0.76	0.69	0.44	0.73	0.88	0.57
KP	0.51	0.76	0.62	0.42	0.68	0.84	0.58
KMM	0.79	0.78	0.77	0.43	0.73	0.91	0.60

4.2 Experimental Results

As shown in Table 2, the three clustering algorithms derived from the proposed framework achieve superior performance (with ties) on all the 7 data sets. Apparently, among these three algorithms, IA+CL is the best. It achieves the best performance on 5 data sets. In comparison to GC+CL, the performance of IA+CL is consistently better (with ties). Since they use the same clustering algorithm, it proves that our inferred affinities between data points, which are computed "globally", capture more useful information than the "locally" computed similarities. We can see FRA+CL is consistently better (with ties) than OH+CL, and FRA+KM is consistently better (with ties) than OH+KM. It means the feature derived from the refined affinities of the data point to the abstract objects of attributes, which is also computed "globally" in our framework, is more effective than the 1-hot encoding feature. On some data sets, the performance of KP and KMM are competitive. But overall, our IA+CL and FRA+CL are superior. Obviously, the algorithms derived from the proposed framework are effective for mixed data clustering.

In order to further prove that it is beneficial to take into account the interconnections among the attribute values of all data points, we compare the performance of IA+CL, FRA+CL and FRA+KM, which are reported in Table 2, with those achieved with "locally" inferred affinities and features from non-refined affinities. Specifically, after scaling the initial affinities with equation 9, we obtain the matrix A'. A'_{DC} contains the non-refined affinities of data points to the abstract objects of attributes. We use them as data feature (**FNRA**: **F**eature from **N**on-**R**efined **A**ffinities) for clustering with the complete-linkage (**CL**) and k-means (**KM**) algorithms. To obtain the "local" affinities between data points, we compute $A^* = A' \times A'$. The affinities in A^*_{DD} are inferred "locally" just between each pair of data points. We call them **LIA** (**L**ocally **I**nferred **A**ffinities) and use the complete-linkage (**CL**) algorithm for clustering. When comparing LIA+CL versus IA+CL, FNRA+CL versus FRA+CL and FNRA+KM versus FRA+KM, on each data set, LIA+CL, FNRA+CL and FNRA+KM use the same scaling factors Δ as IA+CL, FRA+CL and FRA+KM respectively. Table 3 shows the performance comparisons. As we can see, the performance of using "globally" inferred or refined affinities are always better or equal to those of using "locally" inferred or non-refined affinities. It demonstrates that the proposed framework is effective for modeling and exploiting the interconnections among the attribute values of all data points to improve clustering performance.

In order to show that the proposed framework for mixed data clustering is applicable in practice, we evaluate its efficiency on real world data sets. The proposed framework consists of three main components: 1. processing mixed data; 2. learning affinities; 3. clustering with the learned affinities. As introduced in Section 3.1, it takes linear time to process the mixed data. When clustering with the learned affinities, the time complexity totally depends on the selected clustering algorithm. Therefore, in this paper, we only evaluate the efficiency of affinity learning. The average time consumed on this step in the clustering experiments are reported in Table 4.

As shown in Table 4, for small data sets, which contain at most thousands of objects, the time consumed on affinity learning is negligible. For medium data sets, such as Adult, it may take a few minutes. Since these results are obtained on an ordinary PC, we can say, with modern computation technologies and computing power, the proposed framework is

Table 3: Clustering Results with Locally and Globally Learned Affinities (FScore on AI: Acute Inflammations; HD: Heart Disease; CA: Credit Approval; CMC: Contraceptive Method Choice; Adult; Soybean; TTT: Tic-Tac-Toe Endgame)

	LIA+CL	IA+CL	FNRA+CL	FRA+CL	FNRA+KM	FRA+KM
AI	0.92	0.92	0.92	0.92	0.79	0.80
HD	0.71	0.78	0.71	0.79	0.78	0.79
CA	0.60	0.78	0.60	0.75	0.69	0.70
CMC	0.49	0.52	0.49	0.51	0.44	0.44
Adult	0.67	0.75	0.67	0.73	0.69	0.73
Soybean	1	1	1	1	0.88	0.89
TTT	0.68	0.71	0.68	0.76	0.57	0.58

Table 4: Time Consumed on Affinity Learning (sec.)

Data set	#Instance	#Attribute	#Object	Time Consumed
Acute Inflammations	120	6	132	0.005
Heart Disease	270	13	300	0.01
Credit Approval	653	15	705	0.03
Contraceptive Method Choice	1473	9	1493	0.09
Adult	30,162	13	30,256	40
Soybean	47	35	105	0.005
Tic-Tac-Toe Endgame	958	9	985	0.04

applicable in practice.

5 Conclusions

The main contributions of this paper include: 1. we develop a novel framework for mixed data clustering; 2. our approach to mixed data processing, especially the way we transform continuous attribute values into initial affinities, is novel; 3. it's novel to transform the refined affinities between data points and the abstract objects of attributes into new data features. Experimental results on several real world data sets demonstrate the proposed framework is effective.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant IIS-1302164.

References

[Ahmad and Dey, 2007] Amir Ahmad and Lipika Dey. A k-mean clustering algorithm for mixed numeric and categorical data. *Data & Knowledge Engineering*, 63(2):503–527, 2007.

[Böhm *et al.*, 2010] Christian Böhm, Sebastian Goebl, Annahita Oswald, Claudia Plant, Michael Plavinski, and Bianca Wackersreuther. Integrative parameter-free clustering of data with mixed type attributes. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 38–47. Springer, 2010.

[Gower, 1971] John C Gower. A general coefficient of similarity and some of its properties. *Biometrics*, pages 857–871, 1971.

[Hsu and Chen, 2007] Chung-Chian Hsu and Yu-Cheng Chen. Mining of mixed data with application to catalog marketing. *Expert Systems with Applications*, 32(1):12–23, 2007.

[Hsu and Huang, 2008] Chung-Chian Hsu and Yan-Ping Huang. Incremental clustering of mixed data based on distance hierarchy. *Expert Systems with Applications*, 35(3):1177–1185, 2008.

[Huang, 1997] Zhexue Huang. Clustering large data sets with mixed numeric and categorical values. In *Proceedings of the 1st pacific-asia conference on knowledge discovery and data mining,(PAKDD)*, pages 21–34. Citeseer, 1997.

[Huang, 1998] Zhexue Huang. Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data mining and knowledge discovery*, 2(3):283–304, 1998.

[Jing *et al.*, 2007] Liping Jing, Michael K Ng, and Joshua Zhexue Huang. An entropy weighting k-means algorithm for subspace clustering of high-dimensional sparse data. *IEEE Transactions on Knowledge and Data Engineering*, 19(8):1026–1041, 2007.

[Kandola *et al.*, 2003] Jaz Kandola, Nello Cristianini, and John S Shawe-taylor. Learning semantic similarity. In *Advances in Neural Information Processing Systems*, pages 673–680, 2003.

[Legendre and Legendre, 1998] Pierre Legendre and Louis Legendre. Numerical ecology, volume 24, (developments in environmental modelling). 1998.

[Li and Latecki, 2015] Nan Li and Longin Jan Latecki. Affinity inference with application to recommender systems. In *Web Intelligence and Intelligent Agent Technology (WI-IAT), 2015 IEEE/WIC/ACM International Conference on*, volume 1, pages 393–400. IEEE, 2015.

[Plant and Böhm, 2011] Claudia Plant and Christian Böhm. Inconco: interpretable clustering of numerical and categorical objects. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1127–1135. ACM, 2011.

[Plant, 2012] Claudia Plant. Dependency clustering across measurement scales. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 361–369. ACM, 2012.

[Podani, 1999] János Podani. Extending gower's general coefficient of similarity to ordinal characters. *Taxon*, pages 331–340, 1999.

[Yang *et al.*, 2013] Xingwei Yang, Lakshman Prasad, and Longin Jan Latecki. Affinity learning with diffusion on tensor product graph. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):28–38, 2013.

[Zhou *et al.*, 2003] Dengyong Zhou, Olivier Bousquet, Thomas Navin Lal, Jason Weston, and Bernhard Schölkopf. Learning with local and global consistency. In *NIPS*, volume 16, pages 321–328, 2003.