Comparing relational and non-relational algorithms for clustering propositional data

Robson Motta
VICG
ICMC, University of Sao Paulo
Sao Carlos, SP, Brazil
rmotta@icmc.usp.br

Bruno M. Nogueira
LABIC
ICMC, University of Sao Paulo
Sao Carlos, SP, Brazil
brunomn@icmc.usp.br

Alípio M. Jorge
LIAAD - INESC TEC
DCC, FCUP, University of Porto, Portugal
amjorge@fc.up.pt

Alneu de Andrade Lopes
LABIC
ICMC, University of Sao Paulo
Sao Carlos, SP, Brazil
alneu@icmc.usp.br

Solange O. Rezende
LABIC
ICMC, University of Sao Paulo
Sao Carlos, SP, Brazil
solange@icmc.usp.br

Maria Cristina Ferreira de Oliveira
VICG
ICMC, University of Sao Paulo
Sao Carlos, SP, Brazil
cristina@icmc.usp.br

ABSTRACT

Cluster detection methods are widely studied in Propositional Data Mining. In this context, data is individually represented as a feature vector. This data has a natural non-relational structure, but can be represented in a relational form through similarity-based network models. In these models, examples are represented by vertices and an edge connects two examples with high similarity. This relational representation allows employing network-based algorithms in Relational Data Mining. Specifically in clustering tasks, these models allow to use community detection algorithms in networks in order to detect data clusters. In this work, we compared traditional non-relational data-based clustering algorithms with clustering detection algorithms based on relational data using measures for community detection in networks. We carried out an exploratory analysis over 23 numerical datasets and 10 textual datasets. Results show that network models can efficiently represent the data topology, allowing their application in cluster detection with higher precision when compared to non-relational methods.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Clustering; I.5.3 [**Clustering**]: Algorithms

General Terms

Algorithms

Keywords

clustering, network models, community detection

SAC'13 March 18-22, 2013, Coimbra, Portugal.

1. INTRODUCTION

Data clustering is one of the most important activities in data mining. Clustering algorithms discover groups and identify patterns based on object similarity, so that objects within a group have high similarity and objects in different groups have low similarity. These algorithms can be applied mainly to discover underlying structure in data, to perform natural classification of objects or to better understand the structure of the data [8].

There is a large volume of literature on clustering algorithms [26, 8], which may be categorized [29] into six groups: (1) hierarchical clustering; (2) partitional clustering; (3) density-based clustering; (4) grid-based clustering; (5) model-based clustering; and (6) graph-based clustering. The first five groups refer to non-relational algorithms, whereas the last one refers to a class of relational clustering methods. Non-relational methods require a propositional representation of the data, with examples represented as feature vectors. On the other hand, relational methods require a relational data representation, with examples represented as vertices and edges connect related examples.

A relational representation may be created for a non-relational dataset by building similarity-based network models [4, 19], in which network vertices represent data instances and edges connect pairs of highly similar instances. Once a relational representation exists, network-based measures and algorithms may be employed in mining tasks, as an alternative to the traditional (non-relational) approaches.

Both types of clustering algorithms have advantages and disadvantages in specific scenarios. As network-based clustering algorithms require building a relational representation from an originally non-relational dataset, their application incurs in additional cost. Since both kinds of algorithms have similar time complexity, in average, it can be argued that traditional non-relational clustering require less computational effort than the network-based. On the other hand, cluster identification in network models is not biased towards particular shapes or densities, as long as the model represents groups of highly connected examples separated by a few edges, so that community detection methods can be successfully applied to identify relevant clusters.

Recent progress in the analysis of social networks brought about new algorithms for community detection, which also could have some potential for the clustering of standard vectorial data. However, little to nothing is known about their performance, specially when dealing with non-relational data.

In this scenario, we are interested in comparing how these two types of algorithms handle the task of identifying clusters in non-relational data, specially on real-world datasets. In this work we describe an exploratory study on relational and non-relational clustering algorithms when applied to non-relational data. We compared the performance of two popular categories of non-relational algorithms [16], namely hierarchical (Single Link, Average Link, Complete Link and Bisecting K-Means [9]) and partitional algorithms (K-Means [17]), against graph-based algorithms (kNN [30], kDR [1], $PKNN$ [2], HSN [18], MST [29] and $EMST$ [19]).

This paper is organized as follows. In Section 2 we present some related work on comparing relational and non-relational clustering algorithms. In Section 3 we introduce some methods and background knowledge involved on relational and non-relational data clustering. In Section 4 we present the results of our empirical evaluation. Finally, in Section 5 we conclude and point out some perspectives for further work.

2. RELATED WORK

Many papers address non-relational data clustering. The contributions by Jain et al. [10, 8] provide a useful summary of state-of-the-art non-relational data clustering algorithms. Xu and Wunsch [26] compared different non-relational clustering algorithms in five different applications (two benchmark datasets, two bioinformatics datasets and on the Traveling Salesman Problem dataset). They concluded that there is no clustering algorithm that can correctly solve all problems, pointing out that appropriate preprocessing and post-processing steps in clustering tasks would improve clustering results.

Many contributions that compare non-relational algorithms focus on their performance in specific applications. Jiang et al. [11] present a survey of clustering algorithms in gene expression data, pointing out when each method can suitably solve a given clustering problem in this context. The authors consider several algorithms that have been applied to cluster gene expression data, such as K-Means, SOM, Hierarchical Clustering and Model Based Clustering. They categorize their application in gene-based clustering, sample-based clustering and subspace clustering. Again, the conclusion was that there is no standard clustering method or evaluation criteria for every application in gene data clustering and the choice of the most suitable one relies on the user's experience. Zhao and Karypis [28] have compared different criterion functions on partitional document clustering. They used modifications of K-Means and Bisecting K-Means algorithms, adopting seven different criterion functions in their clustering processes, four of which have been proposed by themselves. The results show that the performance difference observed by the different criterion functions can be attributed to the extent to which these functions are sensitive to clusters of different degrees of tightness, and the extent to which they can lead to reasonably balanced solutions. Liao [15] discusses a variety of clustering algorithms applied to the time series data clustering problem. Three components of time series clustering are discussed: the clustering algorithm, the similarity measure and the evaluation criterion. Some applications are discussed in business, engineering, science, medicine and entertainment.

As far as community detection in networks is concerned, relevant contributions are distinguished mainly by analysing networks with different attributes. Danon et al. [3] compared 16 community detection methods on artificial data sets, analyzing the precision and computational cost, and concluding that the non-parametric Fast Greedy [20] method achieves a good compromise of both measures. Lancichinetti and Fortunato [12] have evaluated multiple community identification methods in real-world networks with different properties. The method based on random walk by Rosvall and Bergstrom [23] did particularly well regarding performance and versatility, being applicable to weighted and directed graphs. Leskovec et al. [14] have also compared different methods regarding their relative performance and the systematic biases in the clusters identified. Orman et al. [21] applied five community detection methods on artificial networks with real-world network properties, analysing their precision.

Nonetheless, the previous contributions considered relational data only. The approach of obtaining a network from non-relational data has already been employed by Oliveira et al. [4], who derive a network model and partition it into communities for cluster identification. In their model each example is assigned a random initial angle that is gradually updated considering their neighboring angles until reaching a stable state. The authors proposed a hierarchical model for cluster identification based in this network, and applied this algorithm on artificial datasets and two real-world datasets, showing it performed better than traditional clustering algorithms (KMeans and Hierarchical Clustering). A major drawback is the high dependence on two parameters, the number of neighbors to be considered and the angle updating rate. In a different approach, Granell et al. [6] use a data similarity matrix to detect communities in real-world non-relational data. This similarity matrix is interpreted as a complete weighted graph and a multi-resolution scheme [5] is employed for community detection. The method was applied solely over the Iris dataset, and no comparisons were performed with traditional clustering methods.

3. BACKGROUND

In this section we present some methods and background knowledge involved on relational and non-relational data clustering. These methods are used in the comparisons described in Section 4.

3.1 Non-relational clustering

Non-relational clustering algorithms may be divided in two major categories: partitional and hierarchical clustering [9]. Partitional algorithms subdivide the dataset into a set of mutually independent clusters, i.e., generates a single partition of the dataset. Hierarchical clustering, on the other hand, aims at obtaining a nested sequence of partitions.

K-Means [17] is possibly the best known partitional algorithm, and operates seeking for an optimal partition of the dataset by minimizing the sum-of-squared-error criterion. K-Means obtains k clusters from the data, where k is a predefined parameter. The algorithm starts by randomly choosing k cluster prototypes (centroids). Then, each example x_i in the dataset is added to the cluster C_j with the nearest centroid. After this process is done for the n exam-

ples, the prototypes of each cluster C are updated by calculating the mean of all examples belonging to that cluster. The assignment of examples to clusters and the centroid updates are iteratively done until clusters become stable. The time complexity of this algorithm is approximately linear ($O(nkdt)$, where n is the number of examples, k is the desired number of clusters, d is the number of dimensions and t is the number of iterations until the algorithm stabilizes).

On the class of hierarchical clustering algorithms, two subcategories can be found: the agglomerative and the divisive algorithms [9]. In agglomerative hierarchical clustering, each example is considered as a singleton. At each cluster step, agglomerative algorithms merges the nearest pair of clusters, until one single cluster remains. The divisive algorithms, on the other hand, start with all examples in a single cluster and successively divide them until all clusters are singletons. The agglomerative algorithms are the most popular, due to their smaller complexity ($O(n^2)$ per step) as compared with the divisive methods ($O(2^n)$ per step).

The best-known agglomerative algorithms are the Single Link, Complete Link and Average Link, distinguished by the distance matrix updating process in each clustering step. When a new cluster C_{new} is formed by merging two clusters C_i and C_j, the algorithm updates the distance matrix by calculating the distance $d_{k,new}$ of all other point C_k to C_{new}. In the Single Link algorithm, $d_{k,new}$ is the smallest value among $d_{i,k}$ and $d_{j,k}$ ($d_{a,b}$ denotes the distance between two examples a and b); in the Complete Link algorithm, $d_{k,new}$ is the highest value among $d_{i,k}$ and $d_{j,k}$; and in the Average Link algorithm, $d_{k,new}$ is the average of $d_{i,k}$ and $d_{j,k}$.

On the divisive category, Bisecting K-Means is very popular (see [24]). It consists in iteratively applying the K-Means algorithm to split a cluster in two subclusters, thus generating nested partitions of the dataset. This process is repeated until each cluster has just one element.

In applying non-relational clustering algorithms, a user must choose the proper number of clusters to partition the dataset. A partitioning can be obtained either by a direct application of partitional algorithms or by cutting the dendrogram obtained by hierarchical clustering. Cluster structure quality measures may be employed to assist users on deciding the number of clusters to search for. Considering that the only information about the examples are the attributes (i.e., no class information is given), relative validation measures allow comparing the partitions obtained by different clustering partitions [25]. Several measures can be applied in this process. Here we have selected three well-known measures [25]: Calinski-Harabasz Index, Dunn's Index and Silhouette Width Criterion. The three of them evaluate the partitions by making geometrical considerations about compactness and separation of the obtained clusters. They allow evaluating different configurations of a particular algorithm that lead to different numbers of clusters. The configuration that maximizes the evaluation measure can be taken as the best partition for that algorithm.

3.2 Similarity-based networks and community structure

Tabular data may be transformed into a connected graph, e.g., based on criteria established by similarity relations amongst data examples, so that relational clustering algorithms can be applied. Generally, network vertices represent examples and edges represent dissimilarity (or distance) relations. Different graph models may emphasize different connection patterns and data distributions, e.g., groups of highly similar examples may be strongly connected whereas highly dissimilar ones are loosely connected.

A well-known example is the Minimum spanning tree graph (MST), which is a sub-graph (a tree) of a complete graph that has the minimum number of edges. For weighted graphs, MST has the edge set with minimum total cost. An MST may be computed from a complete graph built from the data taking as edge weights the corresponding pairwise distances between examples. Recently, Zhong et al. [29] used an MST graph to introduce a novel split-and-merge hierarchical clustering method.

The kNN graph [30] is also a traditional model. It connects each example to its k nearest-neighbors according to a given distance function, where input parameter k defines how many neighbors to connect. Usually, obtaining connected graphs requires adopting a high value for k.

Based on the kNN network, Aoyama et al. [1] proposed the k degree-reduced nearest neighbor graph (kDR) as a fast approximate similarity search method. Unlike kNN, the kDR graph does not include an edge between x and $y \in N_k(x)$ (the k-neighborhood of x) without which a greedy search algorithm can reach x from y along the existing edges. Then kDR has a smaller average degree than kNN. Constructing the kDR graph requires building a kNN network with $k = 1$ and then adopting an incremental procedure on k, until $k = k_{max}$ (k_{max} provided in advance). Initially, the kDR graph is exactly a kNN graph with k equal to 1. Then, for each kNN graph, the edges are verified and maybe inserted in the kDR graph. An edge from x to y is inserted if in the current kDR graph the distance between x and every adjacent to y is higher than the distance between x and y.

Bayá and Granitto [2] build similarity-based network models from kNN graphs to cluster gene expression data. Connectedness is an essential property of those networks, ensuring that a finite distance path exists between any pair of examples. Network construction departs from a kNN graph, which is transformed into a connected graph – Penalized K-Nearest-Neighbor-Graphs (PKNN). The rationale is that edges linking nodes in different components are assigned weights significantly lower than those edges internal to a component, and the best strategy to obtain a connected graph was merging the kNN and MST graphs.

Motta et al. [18] proposed the Hierarchical Similarity Network model (HSN), with few connections linking isolated examples and many connections linking dense data regions. This model, however, requires an input parameter that controls the average degree of the resulting network. Moreover, the agglomerative construction strategy seeks to optimize the network's modular structure, even though modularity is not necesarily a property of the input dataset.

A network model named Extended Minimum Spanning Tree (EMST), introduced by [19], expands a graph's Minimum Spanning Tree (MST) by connecting each vertex to its most similar vertices, employing a criterion that considers the MST connections. The EMST network is built from the data in two stages: first a complete weighted graph is created, with edge weights given by the pairwise dissimilarity values between vertices. Departing from the graph's MST, edges are added based on connection patterns identified in the MST. The resulting network preserves the original data distribution, with vertices in dense data regions highly con-

Table 1: Data sets for experimental evaluation.

dataset	# ex.	# att.	# cl.
balance	625	4	3
blood-transf.	748	4	2
cleveland	298	13	5
diabetes	768	8	2
ecoli	336	7	8
glass	214	10	6
habermans	306	3	2
heart-statlog	270	13	2
ionosphere	351	34	2
iris	150	4	3
isolet	1559	617	26

dataset	# ex.	# att.	# cl.
libras	360	90	15
madelon	600	500	2
mult-features	2000	649	10
musk-v1	476	166	2
satimg	500	36	6
sonar	208	60	2
spectf	267	44	2
vehicle	846	18	4
vertebral	310	6	3
vowels	990	10	11
wine	178	13	3

dataset	# ex.	# att.	# cl.
zoo	101	16	7
Amazon	1500	10000	50
CNAE-9	1080	856	9
re0	1504	2886	13
re1	1657	3758	25
tr23	204	5832	6
tr31	927	10128	7
tr41	878	7454	10
cbr-ilp-ir-son	675	1423	4
KDVis	1624	520	4
News2011	1771	3731	23

nected and isolated vertices sparsely connected. It does not artificially enforce a modular structure, because the number of edges incident to a vertex varies according to the region of its corresponding example in multidimensional data space.

Once a network is formed, it is possible to obtain clusters by detecting community structure. Several algorithms have been proposed to identify subgroups of densely connected vertices sparsely connected to other subgroups. Newman [20] introduced Fast Greedy, an agglomerative hierarchical algorithm that does not require the number of communities as input. At each stage it computes a quality measure, modularity Q, that is maximized when the "ideal" number of communities is reached. Distinguishing features of this algorithm are the proposed modularity Q and the low computational cost as compared to other methods ($O(nlog^2n)$).

Community detection algorithms may adopt the modularity Q to establish the ideal number of communities to search for, as in the Adaptive Clustering agglomerative method [27]. It runs in two stages: the first one groups highly connected vertices, using the Fast Greedy algorithm, and the second reassigns vertices to groups based on their local connections and network neighborhood, seeking to maximize Q. It is shown to produce better decompositions of the network into communities as compared to Newman's algorithm [20], at an equivalent computational cost.

4. EMPIRICAL EVALUATION

We now compare network-based clustering strategies with the traditional ones, focusing on how homogeneous are the clusters considering the previously known classes. Evaluation has been conducted on 23 numerical data sets from the UCI repository[1] plus 10 textual data sets (2 from UCI, 5 from the CLUTO project[2] and 3 from an Internet repository[3]). A summary of these datasets is presented in Table 1, informing the number of examples, attributes and classes.

We conducted evaluations considering non-relational algorithms based on hierarchical agglomerative clustering (Single, Average and Complete), on hierarchical divisive clustering (Bisecting K-Means) and on partitional clustering (K-Means). These methods require a desired number of clusters as input, and their performance was thus measured in all datasets varying the number of clusters from 2 to 50. Three relative measures of cluster quality, namely Sillhouette Coefficient, Dunn's Index and the Calinski-Harabasz Index have been employed to identify the best cluster structure for each method, generating 15 combinations to compare.

[1]http://archive.ics.uci.edu/ml/

[2]http://glaros.dtc.umn.edu/gkhome/views/cluto

[3]http://vicg.icmc.usp.br/infovis2/DataSets

We have also evaluated relational clustering approaches considering the following network models: kNN, kDR, PKNN, HSN, MST and EMST. For the first four models, which are parametric, the input values (k for the first three; degree for HSN) considered were 3, 5, 7 and 11. For each network, we employed the community detection methods Fast Greedy [20] and Adaptive Clustering [27], and the modularity measure Q [20] to identify the best community structure. Then, we have 18 networks and 2 community detection methods, thus producing 36 possible combinations.

After the cluster identification, validation was conducted with three state-of-the-art external cluster evaluation measures: Rand Index [22], Adjusted Rand Index [7] and F-Score Measure [13]. These measures assign a score to the cluster result according to the clusters degree of purity, i.e., according to the methodt's capability of generating clusters that represent classes, establishing a direct relation between clusters and classes. For this evaluation to be possible we only considered datasets with a class attribute (which, of course, has been ignored in the clustering process).

A comprehensive comparison among so many methods from different categories requires a preliminary filtering of the alternatives. In a first analysis we compared results obtained within the same category (hierarchical agglomerative, hierarchical divisive, partitional and graph-based clustering) to identify the most representative methods. For each category it was obtained the average ranking for each evaluation measure, as shown in Table 2. In relational approaches, the kNN network model was not considered since its major drawback of limiting the minimum number of clusters that can be generated is solved in the PKNN, which is very similar to kNN. Among the two network detection methods, the Adaptive Clustering was selected, as in general it presented better results than Fast Greedy.

The analysis pointed to the Average Link as the best agglomerative hierarchical clustering algorithm. For this algorithm, the partition obtained using the Sillhouette Coefficient was superior to the one obtained using the Dunn's Index and the Calinski-Harabasz Index. For the hierarchical divisive algorithm (Bisecting K-Means), partitions obtained using Calinski-Harabasz Index presented a better evaluation score. For the partitional algorithm (K-Means), partitions obtained using the Dunn's Index had a better evaluation value. Finally, the PKNN model presented the better results compared to the other parametric network models and the best results were achieved for $k = 11$. The EMST model obtained better results as compared to the other non-parametric network model (MST).

In summary, we ended with the following methods to com-

Table 2: Comparison of the average rank for each cluster detection method, separated by their category.

non-relational agglomerative									
	HC-AL (Sil)	HC-AL (CH)	HC-AL (Dunn)	HC-SL (Sil)	HC-SL (CH)	HC-SL (Dunn)	HC-CL (Sil)	HC-CL (CH)	HC-CL (Dunn)
Rand	**2,94**	3,64	3,58	4,70	3,94	4,76	3,88	4,00	3,85
Adj.Rand	**2,36**	2,85	2,67	5,21	4,30	5,24	4,12	4,52	4,03
F-Score	4,39	5,42	3,91	3,97	3,73	3,45	3,79	3,48	**3,12**
Average	**3,23**	3,97	3,38	4,63	3,99	4,48	3,93	4,00	3,67

non-relational divisive			
	BKM (Sil)	**BKM (CH)**	BKM (Dunn)
Rand	1,91	**1,88**	1,94
Adj.Rand	2,03	**1,85**	1,88
F-Score	**1,85**	1,91	2,00
Average	1,93	**1,88**	1,94

non-relational partitional			
	KM (Sil)	KM (CH)	**KM (Dunn)**
Rand	1,97	2,03	**1,85**
Adj.Rand	2,03	2,03	**1,79**
F-Score	**1,82**	2,12	1,91
Average	1,94	2,06	**1,85**

relational non-parametrical		
	MST (AC)	**EMST (AC)**
Rand	1,61	**1,39**
Adj.Rand	1,85	**1,15**
F-Score	1,91	**1,09**
Average	1,79	**1,21**

relational parametrical												
	KDR 3(AC)	KDR 5(AC)	KDR 7(AC)	KDR 11(AC)	HSN 3(AC)	HSN 5(AC)	HSN 7(AC)	HSN 11(AC)	PKNN 3(AC)	PKNN 5(AC)	PKNN 7(AC)	**PKNN 11(AC)**
Rand	6,97	6,30	6,03	6,03	8,12	7,61	6,94	7,18	6,76	6,39	**4,76**	4,85
Adj.Rand	8,58	5,94	4,88	4,88	9,64	8,45	6,33	6,91	8,18	6,39	4,33	**3,45**
F-Score	9,45	5,91	4,70	4,45	10,15	8,48	6,76	5,36	8,79	6,39	4,76	**2,76**
Average	8,33	6,05	5,20	5,12	9,30	8,18	6,68	6,48	7,91	6,39	4,62	**3,69**

Table 3: Comparison of the average rank for the best cluster detection method from each category.

	HC-AL (Sil)	BKM (CH)	KM (Dunn)	**PKNN 11(AC)**	EMST (AC)
Rand	3,24	3,85	3,36	**2,03**	2,48
Adj.Rand	3,30	3,88	3,64	**1,73**	2,45
F-Score	3,18	3,12	3,55	**2,24**	2,91
Average	3,24	3,62	3,52	**2,00**	2,62

Table 4: Results from the statistical comparisons (method in the line vs. method in the column). Green symbols indicate positive values of statistical comparison, red colors indicate negative values. Filled symbols indicate significant difference (p-value lower than 0.01).

	PKNN 11(AC)	EMST (AC)	HC-AL (Sil)	KM (Dunn)	BKM (CH)
Rand Index					
PKNN11(AC)	–	△	▲	▲	▲
EMST(AC)	▽	–	▲	▲	▲
HC-AL(Sil)	▼	▼	–	△	△
KM(Dunn)	▼	▼	▽	–	△
BKM(CH)	▼	▼	▽	▽	–
Adjusted Rand Index					
PKNN11(AC)	–	▲	▲	▲	▲
EMST(AC)	▼	–	▲	▲	▲
HC-AL(Sil)	▼	▼	–	△	△
KM(Dunn)	▼	▼	▽	–	△
BKM(CH)	▼	▼	▽	▽	–
F-Score					
PKNN11(AC)	–	▲	△	▲	△
EMST(AC)	▼	–	△	△	△
HC-AL(Sil)	▽	▽	–	△	△
KM(Dunn)	▼	▽	▽	–	▽
BKM(CH)	▽	▽	▽	△	–

pare: (i) HC-AL (Sil), indicating the Hierarchical Clustering using Average Link and validation through Sillhouette Coefficient; (ii) KM (Dunn) as the K-Means with maximum Dunn's Index; (iii) BKM (CH) as the Bisecting K-Means with maximum Calinski-Harabasz Index; (iv) PKNN11 (AC) as the network PKNN with $k = 11$ and the network detection method Adaptive Clustering; and (v) EMST (AC) as the non-parametric network model.

We then compared the methods selected in each category and the results are shown in Table 3. Relational methods produced the best results, with PKNN11 presenting a slight advantage when compared to EMST. We applied the Wilcoxon's paired test with $\alpha = 0.01$ to check whether there is significant difference in the performance of the algorithms. The results are given in Table 4.

One observes that regarding the measures Rand Index and Adjusted Rand Index, network-based methods presented better results with statistical significance as compared to non-relational methods. Both approaches presented improved performance on the F-Score measure, with PKNN (AC) presenting high significant difference. Among the network--based methods, PKNN (AC) was superior to the EMST (AC) method in all evaluation measures. When comparing the non-relational clustering methods, no predominance was oberved of a particular method over the others.

5. CONCLUSIONS

Relational and non-relational clustering algorithms have been successfully applied to a broad set of data mining problems. Non-relational methods deal with propositional representations of the data, while relational methods require a graph model that represents examples as vertices and edges connect similar examples. Similarity-based network models allow the creation of a relational representation for a non-relational dataset which allows relational algorithms to be applied over originally non-relational data.

This paper presented a comparison of relational and non-relational clustering algorithms in a non-relational context. We considered three categories of non-relational clustering methods: hierarchical agglomerative (Single Link, Complete Link and Average Link); hierarchical divisive (Bisecting K-Means); and partitional (K-Means). These methods were compared with two categories of relational clustering methods: parametric models (kNN, kDR, $PKNN$ and HSN); and non-parametric models (MST and $EMST$). We compared the performance of these methods over 33 datasets (23 numeric and 10 textual datasets) and evaluated the partitions using three different external evaluation measures (Rand Index, Adjusted Rand Index and F-Score).

In our first evaluation, we compared the performance of methods within the same category and methods from different categories. On the non-relational category, hierarchical agglomerative clustering presented better performance as compared to the other algorithms. When analysing the group of relational algorithms, both $PKNN$ and $EMST$ presented good evaluation values, with a slight advantage to the $PKNN$ method. Finally, we selected the relational and non-relational methods with best evaluations and compared them. The relational methods presented better results with high statistical significance (superior to 0.99). This indicates that it may be worth using relational methods in non-relational clustering problems. A possible reason for this result is the ability of relational-clustering algorithms

to deal with datasets with different topological features. As real-world datasets contain clusters with different shapes, sizes and densities, relational clustering algorithms can obtain better results than non-relational algorithms in detecting cluster in this context.

A major drawback in applying relational algorithms in a propositional context is the additional cost of obtaining a relational data representation for the propositional data. On the other hand, relational methods automatically detect the number of clusters in the dataset, while non-relational methods require the number of clusters as an input parameter.

As future work we intend to perform a controlled analysis of the performance of the methods when handling topological features of data such as cluster shape and density, using artificially generated datasets.

6. ACKNOWLEDGEMENTS

The authors acknowledge the financial support of FAPESP (Project 2011/19850-9), CAPES, CNPq (Brazil), ERDF / Program COMPETE and the Portuguese Government/FCT (project FCOMP-01-0124-FEDER-022701).

7. REFERENCES

[1] K. Aoyama, K. Saito, H. Sawada, and N. Ueda. Fast approximate similarity search based on degree-reduced neighborhood graphs. In *Proc. the 17th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, pages 1055–1063, NY, USA, 2011. ACM.

[2] A. E. Bayá and P. M. Granitto. Clustering gene expression data with a penalized graph-based metric. *BMC Bioinformatics*, 12:2, 2011.

[3] L. Danon, A. D. Guilera, J. Duch, and A. Arenas. Comparing community structure identification. *J. Stat. Mech. Theor. Exp.*, 2005(9):P09008–09008, 2005.

[4] T. B. S. de Oliveira, L. Zhao, K. Faceli, and A. C. P. L. F. de Carvalho. Data clustering based on complex network community detection. In *IEEE Congress on Evolutionary Computation*, pages 2121–2126, 2008.

[5] S. Gómez, P. Jensen, and A. Arenas. Analysis of community structure in networks of correlated data. *Phys. Rev. E*, 80(1):016114+, July 2009.

[6] C. Granell, S. Gómez, and A. Arenas. Data clustering using community detection algorithms. *Int. J. of Complex Systems in Science*, 1:21–24, 2011.

[7] L. Hubert and P. Arabie. Comparing partitions. *J. of Classif.*, 2:193–218, 1985. 10.1007/BF01908075.

[8] A. K. Jain. Data clustering: 50 years beyond k-means. *Pattern Recogn. Lett.*, 31(8):651–666, 2010.

[9] A. K. Jain and R. C. Dubes. *Algorithms for clustering data*. Prentice-Hall, Inc., NJ, USA, 1988.

[10] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Computing Surveys*, 31(3):264–323, Sept. 1999.

[11] D. Jiang, C. Tang, and A. Zhang. Cluster analysis for gene expression data: a survey. *IEEE Trans Knowl Data Eng*, 16(11):1370 – 1386, 2004.

[12] A. Lancichinetti and S. Fortunato. Community Detection Algorithms: A Comparative Analysis. *Phys. Rev. E*, 80(5):056117+, 2009.

[13] B. Larsen and C. Aone. Fast and effective text mining using linear-time document clustering. In *Proc. of the 5th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, KDD '99, pages 16–22, New York, NY, USA, 1999. ACM.

[14] J. Leskovec, K. J. Lang, and M. Mahoney. Empirical comparison of algorithms for network community detection. In *Proc. of 19th Int. Conf. on World wide web*, WWW '10, pages 631–640, USA, 2010. ACM.

[15] T. W. Liao. Clustering of time series data - a survey. *Pattern Recog.*, 38(11):1857 – 1874, 2005.

[16] C.-R. Lin and M.-S. Chen. Combining partitional and hierarchical algorithms for robust and efficient data clustering with cohesion self-merging. *IEEE Trans. on Knowl. and Data Eng.*, 17(2):145–159, Feb. 2005.

[17] J. B. MacQueen. Some methods for classification and analysis of multivariate observations. In L. M. L. Cam and J. Neyman, editors, *Proc. of the V Berkeley Symp. on Math. Stat. and Prob.*, volume 1, pages 281–297. University of California Press, 1967.

[18] R. Motta, A. Andrade Lopes, and M. C. F. Oliveira. Centrality measures from complex networks in active learning. In *DS'09: Proc. 12th Int. Conf. on Discovery Science*, pages 184–196. Springer-Verlag, 2009.

[19] R. Motta, A. de Andrade Lopes, and M. C. F. de Oliveira. Similarity-based network models and how evaluate them - technical report 384. Technical report, Institute of Mathematics and Computer Science - University of Sao Paulo, 2012.

[20] M. Newman. Fast algorithm for detecting community structure in networks. *Phys. Rev. E*, 69:066133, 2004.

[21] G. K. Orman, V. Labatut, and H. Cherifi. Qualitative comparison of community detection algorithms. *CoRR*, abs/1207.3603, 2012.

[22] W. M. Rand. Objective Criteria for the Evaluation of Clustering Methods. *J. of the American Stat. Assoc.*, 66(336):846–850, 1971.

[23] M. Rosvall and C. T. Bergstrom. Maps of random walks on complex networks reveal community structure. *Proc. of the Nat. Acad. of Sciences*, 105(4):1118–1123, Jan. 2008.

[24] S. M. Savaresi and D. L. Boley. A comparative analysis on the bisecting k-means and the pddp clustering algorithms. *Intell. Data Anal.*, 8(4):345–362, 2004.

[25] L. Vendramin, R. J. G. B. Campello, and E. R. Hruschka. Relative clustering validity criteria: A comparative overview. *Stat. Anal. Data Min.*, 3(4):209–235, Aug. 2010.

[26] R. Xu and I. Wunsch, D. Survey of clustering algorithms. *Neural Networks, IEEE Transactions on*, 16(3):645 –678, may 2005.

[27] Z. Ye, S. Hu, and J. Yu. Adaptive clustering algorithm for community detection in complex networks. *Phys. Rev. E*, 78(4):046115, 2008.

[28] Y. Zhao and G. Karypis. Empirical and theoretical comparisons of selected criterion functions for document clustering. *Mach. Learn.*, 55(3):311–331, 2004.

[29] C. Zhong, D. Miao, and P. Fränti. Minimum spanning tree based split-and-merge: A hierarchical clustering method. *Inf. Sciences*, 181(16):3397–3410, Aug. 2011.

[30] X. Zhu. Semi-Supervised Learning Literature Survey. Technical report, U. Wisconsin-Madison, 2005.