ISSN 1691-5402
Environment. Technology. Resources
Proceedings of the 9th International Scientific and Practical Conference. Volume 1I

The Concept of Ontology for Numerical Data Clustering

Peter Grabusts
Rezekne Higher Educational Institution, Research Institute for Regional Studies, Atbrivoshanas alley 76, Rezekne, LV-4601, Latvia, peter@ru.lv

***Abstract*. Classical clustering algorithms have been studied quite well, they are used for the numerical data grouping in similar structures - clusters. Similar objects are placed in the same cluster, different objects – in another cluster. All classical clustering algorithms have common characteristics, their successful choice defines the clustering results. The most important clustering parameters are following: clustering algorithms, metrics, the initial number of clusters, clustering validation criteria. In recent years there is a strong tendency of the possibility to get the rules from clusters. Semantic knowledge is not used in classical clustering algorithms. This creates difficulties in interpreting the results of clustering. Currently, the possibilities to use ontology increase rapidly, that allows to get knowledge of a specific data model. In the frames of this work the ontology concept, prototype development for numerical data clustering, which includes the most important characteristics of clustering performance have been analyzed.**

Keywords – clustering, cluster analysis, ontology.

I Introduction

Cluster analysis as one of the intelligent data analysis tasks is searching for independent groups (clusters), their attributes and performance in the test data [1, 4, 15]. Resolving such a task allows better understand the data, because clustering can be used practically in any application area where data analysis is required.

The author's research interests are related to the cluster analysis and its aspects: application of clustering algorithms, fuzzy clustering, rule extraction from clustered data etc. [6, 7]. Therefore, there is logical desire to pursue the research in ontology inclusion into cluster analysis [5].

To evaluate the clustering performance aspects the following aim was put forward - to analyze and summarize the options of clustering algorithms in order to create an ontology prototype for numerical data clustering. The research tasks are subordinated to the target aim:

- to review clustering algorithms;
- carry out the evaluation of the eligibility of the metrics selection;
- characterize the impact of changes in the number of clusters;
- evaluate the reliability of the results of clustering (clusters validity);
- evaluate the possibility to get rules from clusters;
- develop the ontology concept for numerical data clustering.

II An outline of classical clustering approach

Clustering differs from classification in following: for performing the analysis in clustering process there is no need to distribute a separate variable group. From this point of view, clustering is considered as a "learning without a teacher" and is used in the initial stage of the study.

Cluster analysis is characterized by two features that distinguish it from other methods [4]:

- the result depends on the object or the kind of attributes, they can be clearly
- certain objects, or objects with fuzzy description;
- the result depends on the potential of the cluster and the object relations, that is, we should take into account the possible object belonging to multiple clusters and detection the ownership of the object(strict or fuzzy membership).

Given an important role for clustering in data analysis, object ownership concept was generalized to a class function that defines the belonging of object classes to that particular class. Two classes of characteristic functions were distinguished:

- discrete function that accepts one of two possible values - belong / do not belong to the class (classical clustering);
- a function that accepts values from the interval [0,1]. The closer the values of the function are to 1, the "more" the subject belongs to a certain class (fuzzy clustering).

Clustering algorithms are mainly designed for multi-dimensional data sampling processing, when the data are given in tabular form "object-quality". They allow you to group objects into certain groups, where objects related to each other by a specific rule. It does not matter, how these groups are called – taxons, clusters or classes, the main thing is that it with sufficient precision reflects the characteristics of this object. After clustering the data for further analysis are used with other intelligent data analysis techniques to determine the nature of the resulting regularities and for future uses.

Clustering is typically used during data processing as a first step in the analysis. It identifies groups with similar data that can later be used for the exploration of relationships between the data. Cluster analysis

process formally consists of the following steps (see Fig. 1):

- collecting data necessary for the analysis;
- classes data (clusters) characterizing size and borderline;
- data grouping in clusters;
- definition of classes hierarchy and analysis of the results.

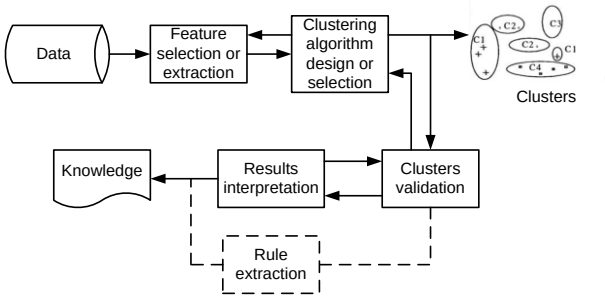

Fig. 1. Clustering procedure [15]

All clustering algorithms have common characteristics, the choice of which characterize the efficiency of clustering (see Fig. 2).

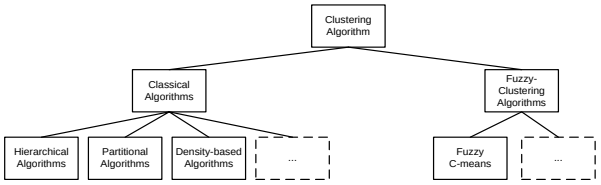

Fig. 2. Hierarchical view of the clustering algorithm class

The most important clustering parameters are following: metric (the distance of cluster element to the cluster center), the number of clusters k, clustering validity, the opportunity to get rules [2, 9, 10].

Metrics. The main purpose of metrics learning in a specific problem is to learn an appropriate distance/similarity function. A metrics or distance function is a function which defines a distance between elements of a set [4, 8, 14]. A set with a metric is called a metric space. In many data retrieval and data mining applications, such as clustering, measuring similarity between objects has become an important part. In general, the task is to define a function Sim(X,Y), where X and Y are two objects or sets of a certain class, and the value of the function represents the degree of "similarity" between the two. Formally, a distance is a function D with nonnegative real values, defined on the Cartesian product X x X of a set X. It is called a metrics on X if for every x, y, z ϵ X:

- D(x,y)=0 if x=y (the identity axiom);
- D(x,y) + D(y,z) ≥ D(x,z) (the triangle inequality);
- D(x,y)=D(y,x) (the symmetry axiom).

A set X provided with a metric is called a metric space.

Euclidean distance is the most common use of distance – it computes the root of square differences between coordinates of a pair of objects:

$$D_{XY} = \sqrt{\sum_{k=1}^{m}(x_{ik} - x_{jk})^2}\ . \qquad (1)$$

Manhattan distance or city block distance represents distance between points in a city road grid. It computes the absolute differences between coordinates of a pair of objects:

$$D_{XY} = \sum_{k=1}^{d} | x_{ik} - x_{jk} |. \qquad (2)$$

Minkowski distance is the generalized metric distance:

$$D_{XY} = (\sum_{k=1}^{d} | x_{ik} - x_{jk} |^p)^{1/p}\ . \qquad (3)$$

Note that when p=2, the distance becomes the Euclidean distance. When p=1, it becomes city block distance.

Cosine distance is the angular difference between two vectors:

$$D_{XY} = \cos(\theta) = \frac{X \bullet Y}{\|X\|\|Y\|} = \frac{\sum_{i=1}^{n} X_i \times Y_i}{\sqrt{\sum_{i=1}^{n}(X_i)^2} \times \sqrt{\sum_{i=1}^{n}(Y_i)^2}} \qquad (4)$$

The distance measure can also be derived from the correlation coefficient, such as the Pearson correlation coefficient. Correlation coefficient is standardized angular separation by centering the coordinates to its mean value. It measures similarity rather than distance or dissimilarity:

$$r_{ij} = \frac{\sum_{k=1}^{d}(x_{ik} - \bar{x}_i)(x_{jk} - \bar{x}_j)}{\sqrt{\sum_{k=1}^{d}(x_{ik} - \bar{x}_i)^2 \sum_{k=1}^{d}(x_{jk} - \bar{x}_j)^2}}, \qquad (5)$$

where $\bar{x}_i = \frac{1}{d}\sum_{k=1}^{d} x_{ik}$.

Noticing that the correlation coefficient is in the range of [-1,1], with 1 and -1 indicating the strongest positive and negative correlation respectively, we can define the distance measure as

$$D_{XY} = (1 - r_{ij})/2. \qquad (6$$

When using correlation coefficients for distance measures, it should be taken into consideration that they tend to detect the difference in shapes rather than determining the magnitude of differences between two objects.

The summary of the metrics is shown in the Figure 3 and Table I.

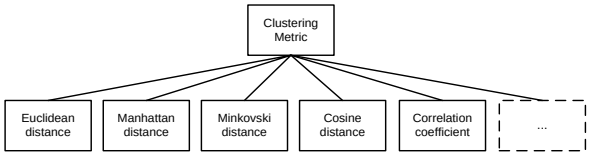

Fig. 3. Hierarchical view of the clustering metrics class

TABLE I
DISTANCE MEASURES AND THEIR APPLICATIONS

Measure	Examples and applications
Euclidean distance	K-means with its variations
Manhattan distance	Fuzzy ART, clustering algorithms
Cosine distance	Text Mining, document clustering
Pearson correlation	Widely used as the measure for microarray gene expression data analysis

Traditionally Euclidean distance is used in clustering algorithms, the choice of other metric in definite cases may be disputable. It depends on the task, the amount of data and on the complexity of the task.

Cluster numbers. An important role in the realization of clustering algorithms is the number of clusters and initial centers determination. Generally is assumed that it is priori known the number of clusters and as values of the initial cluster centers m are offered to take the first points of training set m.

Clustering validity. Cluster validity is a method to find a set of clusters that best fits natural partitions (number of clusters) without any class information. There are three fundamental criteria to investigate the cluster validity: external criteria, internal criteria, and relative criteria [4]. In this case only external cluster validity index was analyzed.

Given a data set X and a clustering structure C derived from the application of a certain clustering algorithm on X, external criteria compare the obtained clustering structure C to a pre-specified structure, which reflects *apriori* information on the clustering structure of X. For example, an external criterion can be used to examine the match between the cluster labels with the category labels based on a priori information.

Based on the external criteria, there is the following approach: comparing the resulting clustering structure C to an independent partition of the data P, which was built according to intuition about the clustering structure of the data set.

If P is a pre-specified partition of data set X with N data points and is independent of the clustering structure C resulting from a clustering algorithm, then the evaluation of C by external criteria is achieved by comparing C to P. Considering a pair of data points x_i and x_j of X, there are four different cases based on how x_i and x_j are placed in C and P.

- Case 1: x_i and x_j belong to the same clusters of C and the same category of P.
- Case 2: x_i and x_j belong to the same clusters of C but different categories of P.
- Case 3: x_i and x_j belong to different clusters of C but the same category of P.
- Case 4: x_i and x_j belong to different clusters of C and different category of P.

Correspondingly, the numbers of pairs of points for the four cases are denoted as a, b, c and d. Because the total number of pairs of points is N(N-1)/2, denoted as M, we have:

$$M = a + b + c + d = \frac{n(n-1)}{2}, \qquad (7)$$

where n is the number of data points in the data set. When C and P are defined, one can choose one of the many clustering quality criteria. Some popular clustering quality criteria are following (see also Fig. 4) [4].

Rand index is calculated by using the following formula:

$$R = \frac{a+d}{M}. \qquad (8)$$

Rand index suggests an objective criterion for comparing two arbitrary clusterings based on how pairs of data points are clustered. Given two clusterings, for any two data points there are two cases:

- The first case is that the two points are placed together in a cluster in each of two clusterings or they are assigned to different clusters in both clusterings.
- The second case is that the two points are placed together in a cluster in one clustering and they are assigned to different clusters in the other.

Hubert index is calculated by using the following formula:

$$\Gamma = \frac{1}{M}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} X_{ij} Y_{ij}. \qquad (9)$$

The value of both index ranges between 0 and 1. A higher index value indicates greater similarity between C and P.

Jaccard coefficient:

$$J = \frac{a}{a+b+c}. \qquad (10)$$

Fowlkes and Mallows index:

$$FM = \sqrt{\frac{a}{a+b}\frac{a}{a+c}}. \qquad (11)$$

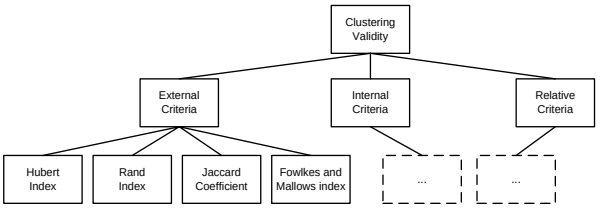

Fig. 4. Hierarchical view of the clustering validity class

Rule extract. The possibility to convert clustering information directly into symbolic knowledge form is through the rules (rule extraction). These assumptions

are formulated as IF ... THEN ... rules [3]. The benefits of the mining rules are as follows:

- the opportunity to verify the rules on different variants of the input data is given;
- failures of training data can be identified, thus clustering operation can be improved by introducing new or removing additional clusters;
- determination of a previously unknown regularities in the data that currently have a growing importance in Data Mining industry;
- the resulting rules can be set up in a base of rules, which might also be used for similar types of applications.

Several artificial neural network algorithms use clustering during learning process, that leads to a hidden neurons (hidden units) deriving, which are actually centers of clusters [9, 13].

The nature of each hidden unit enables a simple translation into a single rule:

IF $Feature_1$ is TRUE AND IF $Feature_2$ is TRUE ... AND IF $Feature_n$ is TRUE THEN $Class_x$. (12)

where a *Feature* is composed of upper and lower bounds calculated by the center μ_n positions, width σ and feature steepness S. The value of the steepness was discovered empirically to be about 0.6 and is related to the value of the width parameter. The values of μ and σ are determined by the training algorithm. The upper and lower bounds are calculated as follows:

$$X_{lower} = \mu_i - \sigma_i + S \text{ and } X_{upper} = \mu_i + \sigma_i - S. \quad (13)$$

Then rule extraction RULEX process can be seen below in Table II [9].

TABLE II

RULE EXTRACTION ALGORITHM

Procedure:

For each hidden unit:

For each μ_i

$X_{lower} = \mu_i - \sigma_i + S$

$X_{upper} = \mu_i + \sigma_i - S$

Build rule by:

antecedent=[X_{lower}, X_{upper}]

Join antecedents with AND

Add class label

Write rule

Consequently, a base for the rules has been obtained.

III POSSIBILITIES OF ONTOLOGY

In recent years the development of ontologies is formal description of the terms in the domain and the relationships between them that moves from the world of artificial intelligence laboratories to desktops of domain experts [5]. In the World Wide Web ontologies have become common things. Ontologies on the net range from large taxonomies, categorizing Web sites, to categorizations of products sold and their characteristics. In many disciplines nowadays standardized ontologies are being developed that can be used by domain experts to share and annotate information in their fields.

Informally, an ontology is a description of the view of the world in relation to a particular area of interest. This description consists of the terms and rules for the use of these terms, limiting their roles within a specific area. Formally, ontology is a system consisting of a set of concepts and a set of statements about the concepts on the base of which you can build up classes, objects, relations, functions, and theories.

The main components of the ontology are: classes or concepts, relationships, functions, axioms, examples.

There are various definitions of ontology, but recently the generally recognized is the following definition: "An ontology is a formal explicit specification of a shared conceptualization" [11]. Ontologies are often equated with taxonomic hierarchies of classes. Thus, the aim of ontology is to accumulate knowledge in general and formal way.

Ontologies can be classified in different forms. One of the most popular types of classification is proposed by Guarino, who classified types of ontologies according to their level of dependence on a particular task or point of view [12]:

- *Top-level ontologies*: describe general concepts like space, time, event, which are independent of a particular problem or domain.
- *Domain-ontologies*: describe the vocabulary related to a generic domain by specializing the concepts introduced in the top-level ontology.
- *Task ontologies*: describe the vocabulary related to a generic task or activity by specializing the top-level ontologies.
- *Application ontologies*: they are the most specific ones. Concepts often correspond to roles played by domain entities. They have a limited reusability as they depend on the particular scope and requirements of a specific application.

Ontologies are widely used in Semantic Web and document clustering, but there is very little information about the use of ontology in numerical data clustering.

Thus, an ontology is an explicit representation of knowledge. It is a formal, explicit specification of shared conceptualizations, representing the concepts and their relations that are relevant for a given domain of discourse [11].

IV ONTOLOGY CONCEPT OF CLUSTERING NUMERICAL DATA

Worked out numerical data clustering ontology concept is composed of the following classes:

Clustering Task. It is an abstract class. It is connected with the clustering algorithm class. Depending on the purpose and the clustering area (domain) clustering algorithm, the number of clusters and the data sample are chosen.

Clustering_Algorithm. This class represents a list of available clustering algorithms and their features (see Fig. 1).

Clustering_Metric. This class represents a list of available distance metrics for clustering algorithms (see Fig. 2).

Clustering_Validity. This class represents a list of cluster validity methods (see Fig. 3).

Clustering_Rule. This class represents a list of rule extraction methods from clusters (if it is possible).

Based on such analysis of classes the following approach is offered for ontology-based clustering, as shown in Fig. 5.

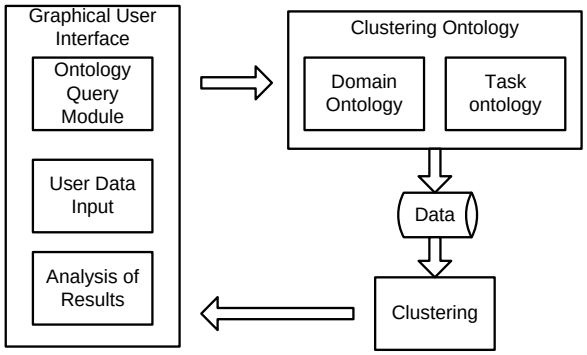

Fig. 5. The framework concept of ontology-based numerical data clustering

Developing framework Protégé OWL tool is used for construct this prototype [16].

Protégé is a special tool, which is thought to create and edit ontology, but OWL (Web Ontology Language) is a language through which it is possible to define the ontology. OWL ontology may include descriptions of classes, their characteristics and their instances. OWL formal semantics describes how, using these data get information which was not openly described in ontology, but which follows from the data semantics. Protégé is a free open-source platform, which contains special tool kit which makes it possible to construct domain models and knowledge-based applications based on ontologies. In Protégé environment a number of knowledge-modeling structures and actions that support ontology creation, visualization and editing of different display formats are implemented.

Protégé is an extensible knowledge model. The internal representational primitive in Protégé can be redefined declaratively. Protégé's primitive - the component of its knowledge model - provide classes, instances of these classes, frame representing attributes of classes and instances.

Ontology development with the help of Protégé starts with the definition and description of classes hierarchy, and then instances of these classes and different types of relationships (properties in Protege) in order to put more meaningful information within the ontology are assigned.

For demonstration ontology development two classes are chosen: *Clustering_Algorithm* and *Clustering_Metric* (see Fig. 6 and Fig. 7).

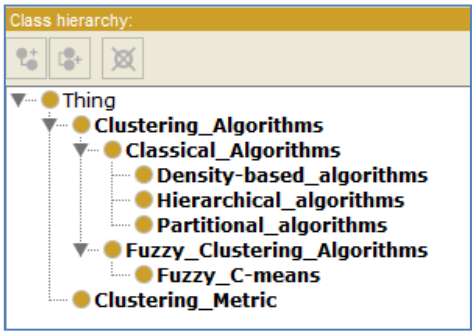

Fig. 6. Clustering domain subclasses in the "Class hierachy" tab of Protégé

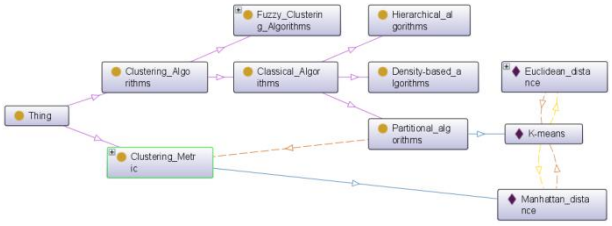

Fig. 7. A visualization of the domain Clustering subclasses in OntoGraf tab

Since clustering algorithm relates to partitional algorithms class, then in class *Partitional_algorithms* was included member *K_means*. K-Means algorithm can use metrics Euclidean distance or Manhattan distance, then in class *Clustering_Metric* were included members: *Euclidean_distance* and *Manhattan_distance*.

K_means member was defined by following characteristics (properties):

K_Means – use -> Euclidean_distance

K_Means – use -> Manhattan_distance

In turn, *Clustering_Metric* object *Euclidean_distance* was assigned following property:

Euclidean_distance - isUsedBy -> K_means (see Fig. 8).

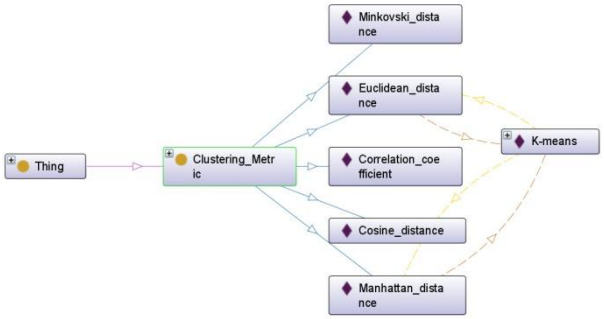

Fig. 8. K-means property visualization in *Clustering_Metric* class

The demonstration example clearly shows that with a help of Protégé we can create an effective description of the ontology, but it is sufficiently laborious process. Author will continue his work on numerical data clustering ontology and its further improvement.

V Conclusion

There are no directly formalized criteria in cluster analysis, so different clustering parameters are chosen in subjective assessment. This refers to the clustering algorithm selection, the choice of number of clusters in each case, the cluster validation criteria determination. Also very important is the knowledge extracted from clusters in the form of rules. All this leads to problems in interpreting the results of clustering. In recent decades, cluster analysis has evolved from one of the data analysis section into a separate direction, which is closely related to knowledge support systems. Partly, this happened due to the introduction of the ontology concepts in the description of clustering characteristics. The use of clustering ontology in documents and semantic web applications is developing very rapidly, but the numerical data clustering is undeservedly neglected. The author has taken the attempt to define and develop an ontology-based prototype for numerical data clustering. This conception contains several concept classes: clustering algorithms, cluster numbers, cluster validity and other characteristics features. In further studies these classes refinement and a real model development according to data clustering purpose will be carried out.

VI References

[1] B. S. Everitt, *Cluster analysis*. John Viley and Sons, London, 1993, 170 p.

[2] L. Kaufman and P. J. Rousseeuw, Finding groups in data. An introduction to cluster analysis. John Wiley & Sons, 2005.

[3] S. Russel and P. Norvig, *Artificial Intelligence: A Modern Approach*. Prentice Hall, 2010, 1132 p.

[4] R. Xu and D. C. Wunch, *Clustering*. John Wiley & Sons,2010, pp. 263-278.

[5] D. Gašević, D. Djurić and V. Devedžić, Model driven architecture and ontology development. Springer-Verlag, 2006.

[6] F. Hoppner, F. Klawonn, R. Kruse and T. Runkler, *Fuzzy Cluster Analysis*. John Whiley and Sons, New York, 1999,289 p.

[7] M. Crawen and J. Shavlik, Using sampling and queries to extract rules from trained neural networks. Machine Learning: Proceedings of the Eleventh International Conference, San Francisco, CA, 1994.

[8] P. Vitanyi, *Universal similarity*. *ITW2005*, Rotorua, New Zealand, 2005.

[9] R. Andrews and S. Gewa, "RULEX and CEBP networks as the basis for a rule refinement system," in *J. Hallam et al, editor, Hybrid Problems, Hybrid Solutions*. IOS Press, 1995.

[10] G. Gan, C. Ma and J. Wu, "Data clustering: Theory, algorithms and applications," ASA-SIAM series on Statistics and Applied Probability, SIAM, Philadelphia, ASA, Alexandria, VA, 2007.

[11] T. R. Gruber, "A translation approach to portable ontologies," Knowledge Acquisition, 5(2),199-220, 1993.

[12] N. Guarino, "Formal Ontology in Information Systems," 1st International Conference on Formal Ontology in Information Systems, FOIS, Trento, Italy, IOS Press, 3-15, 1998.

[13] D. R. Hush and B. G. Horne, "Progress in Supervised Neural Networks. What's new since Lippmann?" IEEE Signal Processing Magazine, vol.10, No 1.,p.8-39, 1993.

[14] M. Li, X. Chen, B. Ma and P. Vitanyi, "The similarity metric," IEEE Transactions on Information Theory, vol.50, No. 12, pp.3250-3264, 2004.

[15] X. Rui and D. Wunsch II, "Survey of clustering algorithms," Neural Networks, IEEE Transactions on, 16(3):645–678, May 2005.

[16] "Protégé project homepage," [Online]. Available: http://protege.stanford.edu/index.html [Accessed: March 13, 2013].