data mining, medical knowledge bases,
cluster visualization, hierarchical clustering, treemaps

Agnieszka NOWAK-BRZEZIŃSKA[1], Tomasz RYBOTYCKI[1]

VISUALIZATION OF MEDICAL RULE-BASED KNOWLEDGE BASES

In this work the topic of applying clustering as a knowledge extraction method from real-world data is discussed. The authors propose hierarchical clustering method and visualization technique for knowledge base representation in the context of medical knowledge bases for which data mining techniques are successfully employed and may resolve different problems. What is more, the authors analyze the impact of different clustering parameters on the result of searching through such a structure. Particular attention was also given to the problem of cluster visualization. Authors review selected, two-dimensional approaches, stating their advantages and drawbacks in the context of representing complex cluster structures.

1. INTRODUCTION

In the domain of Decision Support Systems and Data Mining, last decade brought along a significant development of new algorithms, tools and applications. The knowledge bases (KB) are constantly increasing in volume, thus the knowledge stored as a set of rules or patterns is getting progressively more complex and much harder to interpret or analyze. Recent advances in the field of artificial intelligence have led to the emergence of expert systems, computational tools designed to capture and make available the knowledge of domain experts. The number of medical expert systems is growing and thanks to progress in key areas such as knowledge acquisition, model-based reasoning and system integration for clinical environments their efficiency is getting better everyday. It is essential for physicians to understand the current state of such research as well as remaining theoretical and logistic barriers before full potential of these systems can be used and new patterns can be discovered. Among many other methods, doctors can use the visualization and analysis of medical data for the purpose of extracting a new and potentially hidden knowledge - common and unusual. The extraction and discovery of knowledge hidden in the data have become particularly important in recent years, especially when taking into consideration the constantly growing amount of information stored in databases and data warehouses. The data is collected because it can potentially be the source of previously unknown and useful correlations, anomalies and trends [4]. However, the discovered patterns denominated in the form of an analytical model, may possess a complicated structure, which hinder the further analysis process. But not only does the excessive amount of available information affect the difficulty of research. A more important factor is their complicated structure, both in terms of high dimensionality, as well as used data types. In this

[1]Institute of Computer Science, University of Silesia, 39 Bedzińska Str., 41-200 Sosnowiec, Poland

paper a specific type of knowledge representation, like rules (denoted as *Horn's* clauses) is considered. Unfortunately, if we use — possibly different — tools for automatic acquisition and/or extraction of rules, the number of them grows rapidly. For modern problems, KB can count up to hundreds or thousands of rules. For such KBs, the number of possible inference paths is enormous. In such cases knowledge engineer can not be totally aware that all possible rule interactions are legal and lead to expected results. The big size of KB causes problems with inference efficiency and interpretation of inference results. Even for domain expert it is difficult to analyze the presented knowledge if the number of elements to analyze is too big. In such cases clustering rules and visualizing resultant structure can be helpful.

That is why the authors propose a method of reorganization of the KB from a set of not related rules to groups of *similar rules* (using cluster analysis methods). Besides the information about the rules in each cluster the visualization of clusters is generated. Such a representation of a KB, especially in specific areas (like medicine), seems to be very helpful for expert in exploring the given domain.

The paper consists of 6 sections. In Section 1 the general information about the authors scientific goals' motivation is presented. The description of the cluster analysis idea for rules in KB is included in Section 2. The following section presents the methods of visualization of a hierarchical data structure. Section 4 contains the description of the software created by authors in order to achieve grouping and graphical representation of data. The experiments with the analysis of their results are considered in section 5. Section 6 contains the summary.

2. HIERARCHICAL CLUSTERING ALGORITHM

Hierarchical clustering (or hierarchical cluster analysis) is one of many methods of cluster analysis. It seeks to build a hierarchical structure of clusters. Most basic hierarchical clustering algorithms merge (or divide) only two (one) clusters during one iteration step and because of that the resultant structure of the algorithm is tree-like. There are two types of hierarchical clustering algorithms:

- agglomerative hierarchical clustering algorithms or AGNES (from agglomerative nesting),
- divisive hierarchical clustering algorithms or DIANA (from divisive analysis).

In divisive hierarchical clustering algorithms, at the beginning, all objects are members of one default group. During every iteration step this basic group is divided into smaller groups until the stop condition is met. These methods are used less often than agglomerative methods, because finding an effective way to divide cluster is a nontrivial task [6].

Agglomerative hierarchical clustering (AHC) algorithms presents different approach. During their each iteration step clusters are merged with other clusters. At the beginning each object is considered a cluster itself (or one may say that each object is placed within a cluster that consists only of that object). It can be said that these two types are reverse of one another [5].

In this paper following version of classic (basic) agglomerative hierarchical clustering algorithm [6] was used.

1) Place each object in separate cluster.
2) Build similarity matrix for every cluster pair.
3) Using similarity matrix find most similar pair of clusters and merge them.
4) Update similarity matrix.
5) If stop condition was met end the procedure.
6) Else repeat from step 3.
7) Return structure built this way.

One of the greatest advantages of these kinds of algorithms is that they are independent of how similarity of object is described. There are many methods of specifying resemblance (or

distance) of objects of different types [6]. In some cases complex objects consists of numerical and symbolic data are analyzed and it's impossible to use the most standard similarity measures such as euclidean or Manhattan measures. This led to elaboration of methods that could count resemblance of compound objects with both types of data, such as Gower's measure [3] or Simple Matching Coefficient (SMC). Considering that the objects of grouping were rules, that most often are complex structures and usually consists of numerical data as well as symbolic data, adequate similarity measures had to be selected. In this paper, three inter-object similarity measures were used: Gower's measure [3], simple similarity and weighted similarity [7].

The same situation concerns inter-cluster similarity measures. It does not matter how it is defined for AHC algorithms to work correctly. A great deal of inter-cluster similarity measures were proposed. Four of them were used in this paper: single link, complete link, average link and centroid link [6]. It's important to pick proper inter-cluster similarity measures as some of them are more sensitive to noisy data or undesirable occurrence called cluster chaining (like single link). This can lead to an inadequate interpretation of result. Some of them also works better if grouped objects resemble some kind of shape (e.g. tunnel or small, separate groups). Choosing different inter-cluster and inter-object similarity measures can also lead to (more or less) drastic changes in the form of resultant structure and hence to new patterns being discovered.

3. VISUALIZATION METHODS

As it was mentioned in section 2, the resultant structure of the hierarchical clustering is tree-like. The most common way to visualize this kind of structure is to do it in a dendrogram. Unfortunately, considering increasing size of some, already huge, KBs this solution is often not enough as dendrograms become less clear with size. There are, however, many different ways to visualize large hierarchical structures, one of them being treemaps [10]. Treemaps has been known in literature since 1992, but it's the first attempt to use them to visualize such complex KB. The sole idea of treemaps is to display hierarchical structures using nested rectangles (or different shapes, circles for example [11]) and filling as much space as possible (which in case of rectangular treemap sometimes happens to be all of it). The size of the rectangle (or any other shape used) is usually strongly tied to the size of cluster it represents and thanks to that new patterns can be discovered easier. It's worth mentioning, that colour can be used to achieve the same result. Two treemap algorithms were used in this paper:

- rectangular treemap (with slice-and-dice deployment method) algorithm,
- circular treemap (with deployment method described in [9]) algorithm.

Both of them can be seen on the Fig. 1:

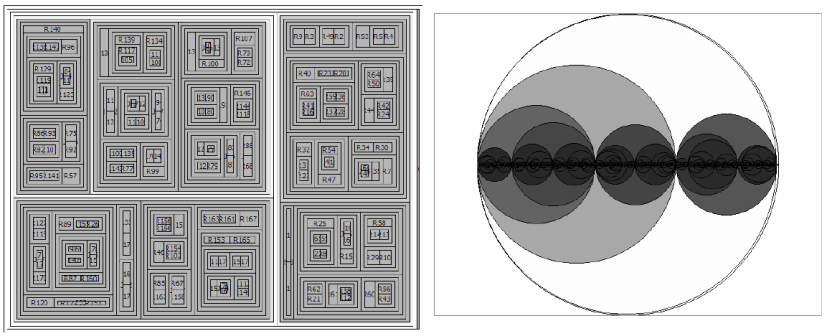

Fig. 1. Two sample treemap visualizations: rectangular treemap (left) and circular treemap (right).

Main difference between the two is the shape used to represent a cluster. As one can perceive, the colour and size of the shape represents size of the cluster. Larger and lighter the shape, the bigger the group. As it can be noticed on the Fig. 1, using full available space may sometimes not be the best approach. On the circular treemap it's obvious that structure consists of two large clusters and one tiny (to the right), but in case of rectangular treemap a quick glance at the picture may lead to false conviction that there are three large clusters (which isn't necessarily wrong, but it may lead to different conclusions).

When visualizing a lot of clusters one may notice that visualizations becomes less and less clear and hence it's harder to determine which clusters are inside examined group. One way to bypass this problem is to implement an responsive visualization (such as the one used in this paper). It allows user to (e.g.) dive deeper into the hierarchy and thus visualizing less, but more precise objects. Thanks to that more space becomes available for presenting examined group and readability of presented data becomes a lot better. While examining huge knowledge bases (that consists of e.g. hundreds or thousands of rules), that describe some complex disease case, it's obvious that sole visualization would not be enough. It would certainly be hard to examine a structure (just like in case of dendrogram) when only thing one could see was an accumulation of hundreds or thousands of small shapes. There is, however, a way to support exploration of such knowledge base by additional description statistics, which, with proper implementation, will allow user to get detailed data about selected cluster.

4. MEDICAL KNOWLEDGE BASES REPRESENTATION

One of the most popular ways of storing knowledge in the intelligent information systems are still rules, regardless of the development of different knowledge representations — semantic networks, object-oriented representations and frame systems, probabilistic methods of knowledge representation and processing. Recent years have brought a renaissance of applications of rules' representation. The rules are considered as standard result form of data mining methods, e.g. decision rules bind the values of conditional attributes with decision attribute, describing in this way the relation between attributes in the decision tables. Medical knowledge bases are very often built using the rules' representation mostly because it is the simplest method of achieving knowledge from domain experts (specialists). If the domain is thoroughly examined, the number of rules can be huge and their description can be complex. That is why it is worth to find some possibilities to represent characteristics of similar cases of a given sickness. Visualization of rules clusters may contain many small clusters (each representing a group of similar rules) or few big clusters (a few groups, in each of them many similar rules that differs only in some small aspects may be placed). Such a visualization may also find and present the outliers (single rules which is not similar to any of other) on the figure, which means that beside the most of rules centered around the phenomenon studying the disease are also isolated rules describing the distinct (too often unique) cases. Without the graphical representation it would be difficult to explore such information from large and complex rules set.

4.1. CLUVIS AS A TOOL FOR KNOWLEDGE BASES REPRESENTATION

Before experiments were performed, an overview of literature was made in search of a software capable of effectively visualizing large KBs. Tools to visualize hierarchical ontologies (specific model of data representation) were found, but the authors did not find tools that allow to visualize a group of rules (clauses in the form IF ... THEN ...) generated using one of many possible clustering algorithms.

The overview also revealed that treemaps haven't yet been used to visualize groups of rules in

KBs. In [8], for example, the authors selected the DBSCAN density-based algorithm as a basis for discovering trends and relations between objects in databases and implement rectangular treemap visualization technique to present groups of objects in relational databases, not such complex structures as decision rules. These two were direct arguments that made the authors try to implement selected methods of visualization for hierarchical clustering algorithms for such specific datasets as KBs. The result of that attempt is CluVis.

CluVis (Cluster Visualizator) [9] is an application used to group rules found inside input KBs and then visualize resultant structure of the grouping. It is meant to work with KBs built by Rough Set Exploration System ($RSES$) [1], however users may create own bases and CluVis will work just fine (as long as custom base resembles RSES knowledge bases format at least to some level).

CluVis implements basic AHC algorithm mentioned in section 2 (along with every inter-cluster and inter-object similarity measure mentioned in this paper) and two treemap visualization algorithms described in section 3. Some of its other features are abilities to:

1) generate screen shots of visualizations,
2) generate report of current grouping,
3) generate report of chosen cluster,
4) dive deeper into the hierarchy (responsive visualization),
5) count modified Dunn Index (MDI) which is clustering quality measure [2], [9].

To ensure that the right cluster is being targeted, shape representing targeted cluster changes its colour to red on mouse hover. That feature is presented on Fig. 2.

Fig. 2. Highlight feature of CluVis.

CluVis is an open source application written using QT 5.3 and C++ 11. It's source code and any additional information are currently available at : https://github.com/Tomev/CluVis.

5. EXPERIMENTS

Different kinds of medical knowledge bases were analyzed during the experiments. It was already mentioned at the beginning of the article that such real data sets often consists of complex data structures (that contain different data types). Very often there are a lot of features used to describe the objects (rules). The most interesting information to be presented about the analyzed domain, is the data set which concerns the Krukenberg tumor. It refers to a malignancy in the ovary that metastasized from a primary site, classically the gastrointestinal tract, although it can arise in other tissues such as the breast. Gastric adenocarcinoma, especially at the pylorus,

is the most common source. Krukenberg tumors can be seen in all age groups, with an average age of 45 years. The optimal treatment of Krukenberg tumors is unclear. That is why it is so crucial to know the specific symptoms of it before it's too late for a patient. All symptoms are nonspecific and can also arise with a range of problems other than cancer, and a diagnosis can only be made following confirmatory investigations such as computed tomography (CT) scans, laparotomy and/or a biopsy of the ovary. The original dataset contains 200 rules, each of them described by some of the given set of 23 features like: age, localization, type of the surgery, the information if the patient still alive etc. For such a data set CluVis software was used to group the rules, present the visualization of this specific KB and make the exploration of patterns easier. Different parameters were set (similarity measures, number of clusters) and the results of them were analyzed. Few of the records are presented in table 1.

Table 1. Table containing experiments data.

	Inter-object	Inter-cluster	Ungr.R.	smCl	bgCl	MinRinCl	MaxRinCl
1,2	G	S	29	29	1	1 (0%)	171 (81%)
3,4	G	Co	13	23	7	1 (0%)	29 (13%)
5,6	G	A	29	29	1	1 (0%)	171 (85%)
7,8	G	C	29	29	1	1 (0%)	171 (82%)
9,10	S	S	29	29	1	1 (0%)	171 (84%)
11,12	S	Co	7	26	4	1 (0%)	47 (23%)
13,14	S	A	29	29	1	1 (0%)	171 (85%)
15,16	S	C	28	29	1	1 (0%)	170 (84%)
17,18	W	S	27	29	1	1 (0%)	167 (82%)
19,20	W	Co	0	24	6	2 (0%)	15 (7%)
21,22	W	A	29	29	1	1 (0%)	171 (85%)
23,24	W	C	17	29	1	1 (0%)	134 (66%)

The meaning of the columns are as follows: Inter-object (object similarity measure: G - Gower, S - Simple Similarity, W - Weighted Similarity), Inter-cluster (clusters similarity measure: S - Single, Co - Complete, A - Average, C - Centroid), Ungr.R.(number of ungrouped rules), smCl (number of small clusters), bgCl (number of big clusters), MinRinCl (number of rules in the smallest cluster and the % of KB which it covers), MaxRinCl (number of rules in the biggest cluster with % covering). It's important to notice, that all coverages were rounded down. One quite interesting case (row named 3,4) is presented in the Fig. 3.

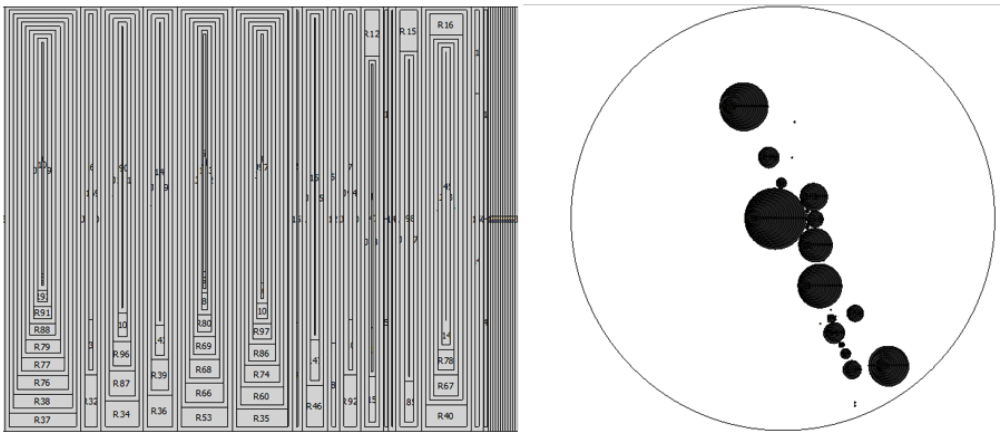

Fig. 3. 30 clusters of rules presented in treemaps structure.

It presents the results of using two methods described in section 3. On the visualization one may see 23 small clusters and 7 big clusters. It also produces 13 of ungrouped rules (these rules were not similar to any of the created clusters and left as outliers). In the biggest cluster

there are 29 rules (it covers 13% of whole KB). CluVis lets user click on selected group and obtain the characteristics of it. Representative of the biggest cluster is following:

(age=50-70) $\&$ (weight=decrease$<10\%$) $\&$ (localization=stem) $\&$ ... $\&$
(schema=45+6c5Fu+LV) $\&$ ... $=>$ (status=ALIVE)

It means that 29 rules describes the cases with some of these features. There are a lot of interesting factors to investigate, one of them, according to the authors, being influence of similarity measure (inter-object and inter-cluster) on number of small clusters generated by algorithm. Interesting results can be noticed in the Fig. 4 (a) where for three different similarity measures it is possible to compare four inter-cluster similarity measures in accordance to the number of big clusters. The figure shows result of grouping using complete linkage method (Co). It's the only method for which the number of big cluster is few times larger than for all other.

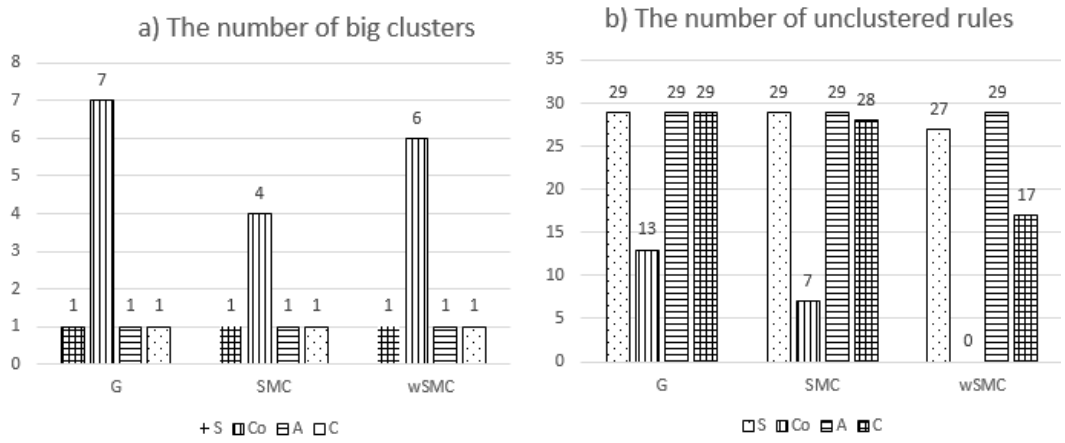

Fig. 4. a)The number of big clusters generated using different similarity measures.b)The number of ungrouped rules generated using different similarity measures.

Looking at the Fig. 4 (b) it is possible to see that for complete linkage method, the number of ungrouped rules is few times smaller than for all other methods. It seems that complete linkage method generates many large clusters and minimalizes outliers number.

What can be also seen in the table 1 is that there is a correlation between the number of small clusters created during the agglomerative hierarchical algorithm in accordance to used method of inter and intra-objects. No matter what similarity measure we use (G, SMC or wSMC) the number of small clusters is quite similar. The only difference is noticed that the Co method provides a minimum number of small aggregates (probably due to the fact that it allows at the same time to create a greater number of large clusters).

Very preliminary consultation with experts allow us to hope that the rules discovered in the visualization of medical data will contribute to the effective induction of knowledge in the study area. Detection of cases of abnormal (for visualization are shown as small circles (squares) separated from the rest of the data) indicates that there are cases of quite rare and descriptive different from most of the accumulated knowledge. This usually means the need for a deeper exploration of the studied areas especially in the area recognized as unusual.

6. SUMMARY

The aim of this paper was to discuss the topic of applying visualization techniques for medical KBs. A hierarchical agglomerative algorithm and treemap visualization techniques were introduced. The authors claim that clustering large set of objects (rules in this case) is

not enough when taking into account exploration such enormous amount of data in order to find some hidden knowledge in it. The extraction of valuable knowledge from large data sets, grouped at first, can be difficult or even impossible. Modularization of KBs help to manage domain knowledge stored in systems using described method of knowledge representation because it divides rules into groups of similar forms, context etc. Cluster analysis produce groups of rules naturally, using the similarity concept. The authors propose to use clusters of rules and visualize them using treemap algorithms. The authors hope that this two-phase way of rules representation allows the domain experts to explore the knowledge hidden in these rules quicker and more efficiently than before. Using this solution in such a specific domain like medicine brings hope that it will be easier to find some characteristics in presented diseases or to discover unusual symptoms which will lead to predicting some serious diseases and preventing the development of distressing symptoms, often saving human lives. Experiment verified that the proposed technique allows a clear and comprehensible presentation medical knowledge hidden in the data. In future the authors plan to extend software's functionality, especially in the context of parameters using in clustering and visualizing procedures, as well as importing other types of data sources. It would be easier then to use the created software (CluVis) in many expert systems and human experts in their everyday work.

Results from the experiments confirmed, that the parameters like inter-object similarity measures or inter-cluster similarity methods (single, complete linkage etc.) influence on clusters size, structure and number. It also confirmed (it is known in literature) that single-link clustering can produce straggling clusters, called chaining, where clusters may be forced together due to single elements being close to each other, even though many of the elements in each cluster may be very distant to each other. Complete linkage tends to find compact clusters, however it suffers from a different problem. It pays too much attention to outliers, points that do not fit well into the global structure of the cluster.

ACKNOWLEDGEMENT

This work is a part of the project „Exploration of rule knowledge bases" founded by the Polish National Science Centre (NCN: 2011/03/D/ST6/03027).

BIBLIOGRAPHY

[1] BAZAN J. G., SZCZUKA M. S., WROBLEWSKI J. A new version of rough set exploration system. Rough Sets and Current Trends in Computing, 2002. Springer-Verlag, Berlin, pp. 397–404.
[2] DUNN J. C. Well separated clusters and optimal fuzzy partitions. Journal of Cybernetica, 1974, Vol. 4. pp. 95–104.
[3] GOWER J. C. A general coefficient of similarity and some of its properties. Biometrics, 1971, Vol. 27. International Biometric Society, Washington, pp. 857–871.
[4] HAN J., KAMBER M., PEI J. Data mining: Concepts and techniques. 2011. Morgan Kaufmann Publishers Inc.
[5] JAIN A. K., DUBES R. C. Algorithms for clustering data. 1988. Prentice Hall, New Jersey.
[6] MORZY T. Eksploracja danych. metody i algorytmy. 2013. Wydawnictwo Naukowe PWN, Warszawa.
[7] NOWAK-BRZEZINSKA A., JACH T. Wnioskowanie w systemach z wiedza niepena. Studia Informatica, 2011. Wydawnictwo Politechniki lskiej, Gliwice.
[8] NOWAK-BRZEZINSKA A., XIESKI T. Exploratory clustering and visualization. 18th International Conference in Knowledge Based and Intelligent Information and Engineering Systems, KES 2014, Gdynia, Poland, 15-17 September 2014, 2014. pp. 1082–1091.
[9] RYBOTYCKI T. Wizualizacja struktur hierarchicznych dla regulowych baz wiedzy. 2015. Sosnowiec.
[10] SHNEIDERMAN B. Tree visualization with tree-maps: 2-d space-filling approach. Transactions on Graphics (TOG), 1992. Association for Computing Machinery, New York.
[11] WETZEL K. pebbles - using circular treemaps to visualize disk usage. 2004.