21

Extraction Of Meaningful Rules In A Medical Database

Sang C. Suh, Nagendra B. Pabbisetty and Sri G. Anaparthi
Texas A&M University - Commerce
U. S. A.

1. Introduction

Data Mining has become a prominent approach in recent years for generating rules within databases which concentrates on generating valuable information for decision making. Clustering, in Data Mining is the problem of identifying the distribution of patterns and intrinsic correlations in large data sets by portioning the data points into similarity classes. The traditional clustering algorithms which use distance between points define boundaries in clustering. But, these clustering mechanisms don't apply on boolean and categorical attributes. The clustering is applied on unstructured data to extract knowledge that may take a form of predictive rules.

Clustering enhances the value of existing databases by revealing rules in the data. These rules are useful   for understanding trends, making predictions of future events from historical data, or synthesizing data records into meaningful clusters. Through clustering are similar data items grouped together to form clusters. Clustering algorithms usually employ a distance metric based (e.g., Euclidean) similarity measure in order to partition the database such that data points in the same partition are more similar than points in different partitions. In this chapter, we study clustering algorithms for data with categorical attributes. Instead of using traditional clustering algorithms that use distances between points for clustering which is not an appropriate concept for boolean and categorical attributes, we propose a novel concept of HAC (Hierarchy of Attributes and Concepts) to measure the similarity/proximity between a pair of data points.

Hierarchical clustering is one of the most frequently used methods in unsupervised learning. Given a set of data points, the output is a binary tree whose leaves are the data points and whose internal nodes represent nested clusters of various sizes. The tree organizes these clusters hierarchically, where the hope is that this hierarchy agrees with the intuitive organization of real-world data.

Hierarchical structures are ubiquitous in the natural world. The idea of hierarchical (also known as agglomerative) clustering is to begin with each point from the input as a separate cluster. We then build clusters by, repeatedly merging the two clusters that are closest in features from the initial points. This gives a hierarchy of containment, as each point in the input belongs to a succession of larger clusters. If we keep merging, we end up with a single cluster that contains all points, and the structure of the hierarchy can be represented as a (binary) tree. From this tree we can extract a set of k clusters. For example, terminating the merging when only k clusters remain, or when the closest pair of clusters are at a distance exceeding some threshold. The crucial part of this algorithm is to define a metric to measure the distance between two clusters of multiple points. One kind of definition that uses mathematical model is: the smallest distance between a point in one cluster and a point in another; the greatest distance between such points; or the average distance. Each definition has its own advantages and disadvantages.

This research focuses on hierarchical conceptual clustering in structured, discrete-valued databases. By structured data, we refer to information consisting of data points and relationships between the data points. This differs from a definition of unstructured data as containing free text and structured data containing feature vectors.

Conceptual clustering is an important way of summarizing and explaining data [1, 6]. However, the recent formulation of this paradigm has allowed little exploration of conceptual clustering as a means of improving performance. Furthermore, previous work in conceptual clustering has not explicitly dealt with constraints imposed by real world environments. This chapter presents a clustering using HAC (Hierarchy of attributes and concepts), which is a hierarchical conceptual clustering system that organizes data so as to maximize inference ability. This algorithm uses both the hierarchical and conceptual clustering methods to implement clustering by discovering substructures in database which compress the original data and represent structural concepts in the data. Once a substructure is discovered, the substructure is used to simplify the data by replacing instances of the substructure with a pointer to the substructure definition. The discovered substructures allow abstraction over detailed structures in the original data. Iteration of the substructure discovery process constructs a hierarchical description of the structural data in terms of the discovered substructures. This hierarchy provides varying levels of interpretation that can be accessed based on the specific data analysis goals.

An important property of the conceptual clustering is that, it can enhances the value of existing databases by revealing patterns in the data. These patterns may be useful for understanding trends, for making predictions of future occurrences from historical evidence, or for synthesizing data records into meaningful clusters. A conceptual clustering system accepts a set of object descriptions (events, observations, facts) and produces a classification scheme over the observations. These systems use an evaluation function to determine classes with "good" conceptual descriptions. A learning of this kind is referred to as learning from observation (as opposed to learning from examples). Typically, conceptual clustering systems assume that the observations are available indefinitely so that batch processing is possible using all observations.

In this study, HAC will be used as an aid to represent medical domain knowledge substructures to simplify the generation process of the databases through clustering. As a result, the research will identify interesting relationships and patterns among the data, and represent them in the form of association rules.

2. Related Work

Clustering is the unsupervised classification of patterns (observations, data items or feature vectors) into groups (clusters). The clustering problem has been addressed in many contexts and by researchers in many disciplines; this reflects its broad appeal and usefulness as one of the steps in exploratory data analysis. However, clustering is a difficult problem combinatorial and differences in assumptions and contexts in different communities have made the transfer of useful generic concepts and methodologies slow to occur [2].

Hierarchical clustering is one of the most frequently used methods in unsupervised learning. Given a set of data points, the output is a binary tree (dendogram) whose leaves are the data points and whose internal nodes represent nested clusters of various sizes. The tree organizes these clusters hierarchically, where the hope is that this hierarchy agrees with the intuitive organization of real-world data. Hierarchical structures are ubiquitous in the natural world [5].

There are two general approaches to hierarchical clustering: top-down and bottom-up. The top-down approach starts with a cluster containing all points that is recursively split until enough sub clusters have been created. Heckel et al. [10, 11] use this approach on vector fields where each point has a position and a vector. The cluster with the highest error value will split into two clusters recursively so that the error values of the remaining clusters decrease with each split.

The bottom-up approach starts with all points as individual clusters and merges the two clusters with least difference, until one big cluster has been formed from all clusters. Telea and van Wijk [10] use this approach to simplify complex vector fields using an elliptic similarity function. They merge the pair of vectors with the least position, magnitude and direction differences until all vectors have been merged into one single vector.

Conceptual clustering is used to summarize the result. It enhances the value of existing databases by revealing patterns in the data. These patterns may be useful for understanding trends, for making predictions of future occurrences from historical evidence, or for synthesizing data records into meaningful clusters. The hybrid conceptual clustering [3] is used to handle both the incremental and non incremental problems for clustering successfully but it is computationally expensive moreover can be applied on small data sets. In past decades, many conceptual clustering algorithms have been proposed which can automatically acquire knowledge or concepts from large amounts of information acquired from experience or observation [1, 7, 8, 9]. Concepts in COBWEB are represented by probabilistic expressions and are acquired by using four learning operators and an evaluation function called category utility. But the category utility used in original COBWEB has a bias to prefer larger size classes in concept hierarchy. This bias produces some spurious intermediate nodes in concept hierarchy (classification tree). These nodes make tree deeper and complex, so we can’t understand concepts within the nodes of tree easily [9].

This chapter presents an efficient non-metric measure called HAC (Hierarchy of Attributes and Concepts) for clustering of categorical as well as non-categorical(quantitative) data through which the proximity and relationships between data items can be identified.

3. Information

3.1 Hierarchy of Attributes and Concepts

The present paper introduces a hierarchical description of concepts by attributes, mathematical formalization presents the concepts as matrices whose columns represents terms constructed by attributes. A spherical model is developed to present a vocabulary (spanned space) of concepts. In the beginning of this section the following definitions are introduced.

Attribute: Is a basic characteristic or a feature of a term.

Term: Is considered as a set of connected attributes.

Concept: Is a language independent meaning associated with at least one term, or set of terms.

Vocabulary: Is a set of terms and concepts.

HAC is both a hierarchical and conceptual clustering system that organizes data to maximize inference ability. This algorithm implements clustering by discovering substructures in database which compress the original data and represent structural concepts in the data. Once a substructure is discovered, it is used to simplify the data by replacing instances of the substructure with a pointer to the substructure definition. The discovered substructures allow abstraction over detailed structures in the original data. Iteration of the substructure discovery process constructs a hierarchical description of the structural data in terms of the discovered substructures. This hierarchy provides varying levels of interpretation that can be accessed based on the specific data analysis goals. HAC accepts database of structured data (concepts) as input. This type of data is naturally represented using a graph or diagrammatical form. The graph representation includes labeled vertices with vertex ID (identification) numbers, attributes on X-axis and downward directed edges (see Fig 1). Each vertex represents a concept and the value of that concept is given by the directed edges (usually map to the attribute's value on X-axis or to some other concept).

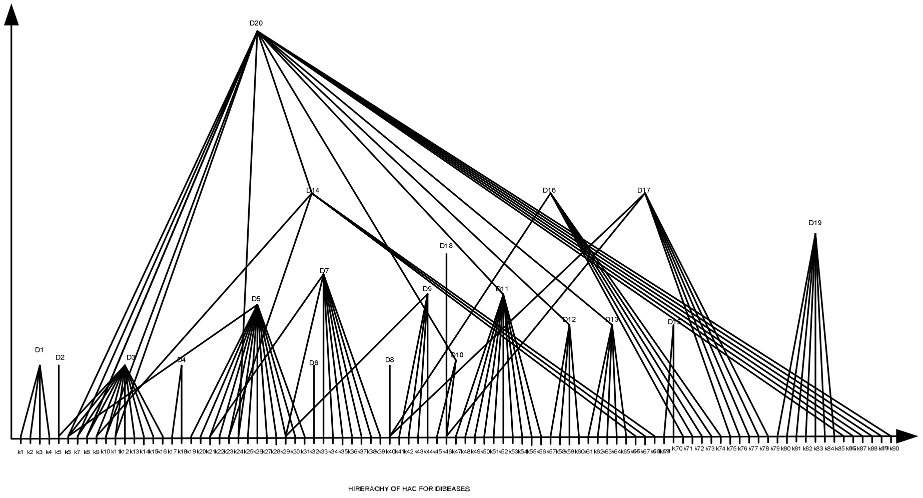

D20
D14
D16
D17
D19
D18
D7
D9
D11
D5
D12
D13
D15
D1
D2
D3
D4
D6
D8
D10
k1 k2 k3 k4 k5 k6 k7 k8 k9 k10 k11 k12 k13 k14 k15 k16 k17 k18 k19 k20 k21 k22 k23 k24 k25 k26 k27 k28 k29 k30 k31 k32 k33 k34 k35 k36 k37 k38 k39 k40 k41 k42 k43 k44 k45 k46 k47 k48 k49 k50 k51 k52 k53 k54 k55 k56 k57 k58 k59 k60 k61 k62 k63 k64 k65 k66 k67 k68 k69 K70 k71 k72 k73 k74 k75 k76 k77 k78 k79 k80 k81 k82 k83 k84 k85 k86 k87 k88 k89 k90
HIRERACHY OF HAC FOR DISEASES

Fig. 1. The Graphical representation of HAC

ID	Disease Name	Causes Id
1	Acute sinusitis	K1,K2,K3,K4
2	Anaphylaxis	K5,K6,K7,K8,K9,K10,K11,K12,K13,K14,K15,K16
3	Penicillin allergy	K6
4	Latex allergy	K17,K18
5	Peanut allergy	K6,K19,K20,K21,K22,K23,K24,K25,K26,K27,K28, K29,K30,K31
6	Mold allergy	K32
7	Nickel allergy	K21,K29,K33,K34,K35,K36,K37,K38,K39
8	Dust mine allergy	K40
9	Asperigillosis	K29,K41,K42,K4,K44
10	Soy allergy	K45,K46
11	Shellfish allergy	K47,K48,K49,K50,K51,K52,K53,K54,K55,K56
12	Wheat allergy	K57,K58,K59,K60
………	……	………… .
20	Food allergy	K6,K7,K8,K9,K10,K11,K23,K24,K45,K46,K47,K4, K49,K50,K51,K52,K53,K54,K56,K57,K58,K59,K60 ,K62,K63,K64,K65K66,K67,K68,K88,K89,K90,K91 ,K92,K93

Table 1. The HAC attribute Table based on attribute allergies.

Using the graph we can create a concept table for every attribute in the database and this table will have all the concepts and their value which are relevant according to the attribute. Ultimately with the help of these concept tables we can get a table where each record will have an attribute name, its value and relationship with other concepts. Each attribute in such a table represents a cluster.

For example Table1, which is formed from set of documents within the domain of education, consist of three columns. The second column specifies the attribute name and the third column specifies the attribute values. Now from table 1 we form another table called Concept Table (Table 2). Every row in this table (Table 2) consists of three fields. The first field shows the concept name whereas the second and third attributes represents concept value and concept attributes.

For example, the Concept Name C1 whose value is sense is a concept formed from the attributes F1,F2,F12,F17,F21,F36,F54 which represents various Food ID’s. Recall that HAC can be represented as a closed diagramatical entity shown in Fig 2.

Food Category ID	Food Category	Food ID
C1	Dairy	F1,F2,F12,F17,F21,F36,F54
C2	Seafood	F3,F4,F73,F74,F75,F76,F77,F78
C3	Poultry	F18,F40,F48,F26,F10,F30
C4	Grains	F7,F16,F29,F33,F50,F68,F69,F70,F8
C5	Nuts	F5,F90,F84,F91,F93
C6	Fruits	F32,F37,F71,F72,F81,F82,F85,F86,F87,F89,F92,F95
C7	Vegetables	F6,F53,F55,F58,F67,F80,F83
C8	Bakery	F11,F13,F14,F20,F43,F45,F51
C9	Drinks	F9,F15,F24,F25,F35,F39
C10	Fat items	F22,F28,F41,F5
C11	Seeds	F42,F60,F61
C12	Leafyand Salad vegetables	F52,F57,F64,F65,F79,F88
C13	Junk foods	F19,F23,F27,F31,F38,F49,F44,F46,F47,F94
C14	Spices	F34,F56,F62,F63,F66

Table 2. The HAC concept Table based on the attribute Food Category.

3.2 Approach

HAC is both a hierarchical and conceptual clustering system that organizes data so as to maximize inference ability. This algorithm implement clustering by discovering substructures in database which compress the original data and represent structural concepts in the data. Once a substructure is discovered, the substructure is used to simplify the data by replacing instances of the substructure with a pointer to the substructure definition. The discovered substructures allow abstraction over detailed structures in the original data. Iteration of the substructure discovery process constructs a hierarchical description of the structural data in terms of the discovered substructures. This hierarchy provides varying levels of interpretation that can be accessed based on the specific data analysis goals.

HAC accepts database of structured data (concepts) as input. This type of data is naturally represented using a graph. The graph representation includes labeled vertices with vertex id numbers, attributes on X-axis and downward directed edges. Each vertex represents a concept and the value of that concept is given by the directed edges (usually map to the attribute's value on X-axis or to some other concept). With this graph we can make a concept table for every attribute in the database and this table will have all the concepts and their value which are relevant according to the attribute. Ultimately with the help of these concept tables we can get a table in which each record will have an attribute name, its value and relationship with different concepts. Each attribute in this table represents a cluster.

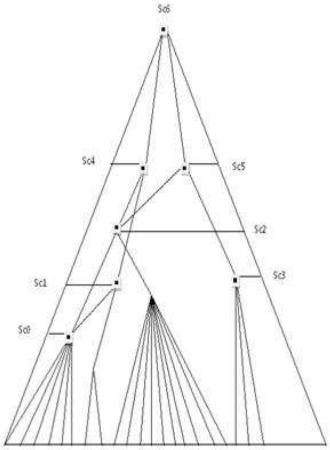


Fig. 2. The Triangular representation of the HAC.

4. Spherical Model

The graph model developed in the previous section is useful for connection purposes. To provide an opportunity to generate and manipulate concepts a matrix form is considered here after.

$$C_{kxn} = \begin{pmatrix} a_{11} \;\; a_{12} \;\ldots\ldots\ldots\ldots\ldots\; a_{1n} \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ a_{k1} \;\; a_{k2} \;\ldots\ldots\ldots\ldots\ldots\; a_{kn} \end{pmatrix}$$

Every column in C_{kxn} represents a term and every a_{ij} represents an attribute. Thus every concept could be decomposed to set of independent terms and every term be generated by pivot element. Therefore every concept (matrix) could be transformed to set of linearly independent terms (columns), created by linearly independent attributes (enteries). As a consecuence every concept will have an invertable minor and the dimensions of this minor will be called a rank of the concept. The set of linearly independent terms T_i where i = 1, 2,…, n over the field of real numbers may span the entire set of concepts.

$$T = \text{span}\,\{T_i\}\; i=1,\ldots..,m \tag{1}$$

If the dimension of the set T is too large the matrix form of concepts presentation would happen to be memory inefficient. To overcome this obstacle a spherical model is developed to represent terms and concepts generation. The central section of the sphere represents the entire set of linearly independent attributes used to build up the basis T_i where i = 1, 2,…,

m. The next part forms the terms and is followed by concepts. Finally the model is ending with only one highest point as shown in figure 3.

If the total number of terms, coming from a certain domain, is n then the maximum number of concepts to be generated by (n-1) terms over a numerical field of 1 element (number) is

$$\binom{n}{n-1}.$$

But if *n-2* elements are used over the same numeric field the number of composed concepts would be $\binom{n}{n-2}$.

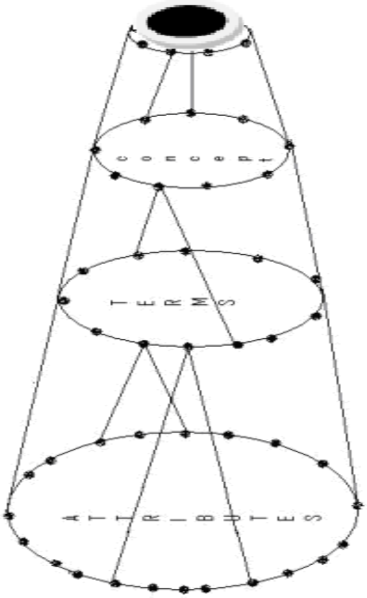


Fig. 3. The Spherical Model

Therefore the total number of concepts to be generated over a field with 1 element and n terms is 2^n. If the numeric field consists of *j* elements (numbers), and the maximum size of a term (column) is k then the total number of concepts to be generated is $j^k 2^n$. From theoretical view point it could be a very big number, but in practice the number of concepts would be smaller because some terms could happen to be mutually contradictive and cannot be linked in a single concept.

5. Algorithm

On the basis of the theoretical concepts an algorithm is developed to form an HAC.
Algorithm Forming_of_HAC_Clusters (A_Name [], A_Values [])

1. Make concept tables for different attributes using HAC. In this basically we have to make such that each level or concept is meaningful.
2. Now, make a “Cluster Point Table” which will have attribute name, attribute value and relationship with different concepts. Attribute name is basically the name of cluster and each and every attribute name is associated with an attribute value
3. Concept will give the value of cluster which is a combination of different attributes and concepts.
4. Concept values can be generated by using following steps:

```
5.  Let total_num= Total number of attributes values   in given table;
6.  While I <= total_num
    Begin:
     Num=1;
          AC_1=total number of Attribute or concepts in  concept table;
          AC _2=total number of Attribute or concepts in concept table;
        If AC_1 and AC_2 forms a cluster
              C_num= AC _1 * AC _2;
               Num++;
        End if
        Increment I;
    End.
7.   For i=1 to num
          $V_i$ = total number of records under different
          combination of $C1_{(j, j+1---m)}$ and $C2_{(k, k+1----n)}$
          Here, $V_{(i, i+1----total\_num)}$ are different clusters or attributes in Cluster Point Table.
          $C1_{(j, j+1---m)}$ are different concepts of concept table1
          $C2_{(k, k+1---m)}$  are different concepts of concept table2
```

From this Cluster Point table and Concept tables (made through HAC) we can generate rules.

The above algorithm is used to form the concepts and form an HAC from the given attribute values. The same algorithm is used if an update of an HAC is needed. The algorithm is capable of adding new concepts, terms and attributes.

6. Case Study: Clustering Using the HAC

This is an example between various allergy diseases and food categories in an allergy database. First a chart of various concepts is formed in which each concept will have allergy disease name as its value. Note that it could include multiple entries.

6.1 HAC1

In Table 1 which is an attribute table for allergy disease, the attribute names are mapped with its values for reference purposes. The concepts D1 through D20 in the TACR for the attribute “Allergy diseases” shown in Table 3 summarize the concepts in the HAC hierarchy of Figure 1 that hold among different attributes and concepts of diseases.

Note that the root concept D20 embraces all concepts and attributes of “Food related allergy diseases.”

Cause ID	Cause
K1	cold virus
K2	bacterial or fungal
K3	deviated nasal septum
K4	nasal polyps
K5	Drugs
K6	Peanuts
K7	tree nuts
K8	Milk
K9	Eggs
K10	Fish
K11	Shellfish
K12	Insect stings from bees
……….	………..
K93	Mushrooms

Table 3. TACR for Allergy Diseases

6.2 HAC2

Similarly, we can build the HAC structure for another attribute “Food category” as shown in Figure 4. The hierarchy shows a conceptual relationship among the values and concepts related to the attribute “Foods”. For this purpose the attribute table in Table 2 is used.

Food ID	Food values
F1	Eggs
F2	Milk
F3	Fish
F4	Shellfish
F5	Tree nut
F6	Beans
F7	Wheat
F8	Gluten
F9	Alcoholic drinks
F10	Beef
F11	Bread
F12	Butter
…	………
F95	Cherry

Table 4. TACR for Food Category

The concepts C1 through C14 in the TACR for the attribute “Food Category” shown in Table 4 summarize the concepts in the HAC hierarchy of Figure 4 showing the relationships among different values and concepts associated with Food Category.

Using the TACRs for the selected attributes, Diseases and Food Category in this case, a graph can be constructed to represent clusters at concept levels. The graph is shown in Figure 5. Each point in the graph represents a concept that is formed by a combination of attribute values. Let’s call this point in the graph a Concept Cluster Point (CCP). Since the cluster point represents a high level concept, it naturally converts to a rule that matches with a concept. Furthermore, the support of each rule can be given by calculating the number of contributing entries in the original relational table to form the concept.

Conceptual Points	Support	Concepts (diseases)	Concepts (food category)
V1	5	D5	C5
V2	2	D10	C5
V3	6	D11	C2
V4	6	D12	C4
V5	2	D13	C1
------	---	-----	----
V15	1	D20	C6
V16	1	D20	C7

Table 5. Cluster Point Table

In Table 5 above, the CCPs V1 through V9 each represent a concept cluster representing a high level concept associated with departments and degrees with a support value. Hence the next step is to convert these CCPs into characteristic rules. To help this process, the CCP graph in Figure 5 is used. Note that each CCP is associated with an appropriate support value which represents the weight to support the converted rule representing the concept.

Each CCP in Figure 5 represents rules regarding how many people affected with specific allergy affected by food category and concepts. A rule can directly be generated from the cluster point table with the support from the attribute tables and TACRs. The cluster point table is shown as cluster point graph, each point in the graph represents rule associated with support. For example, if we want to query how the milk allergy is affected by dairy products.

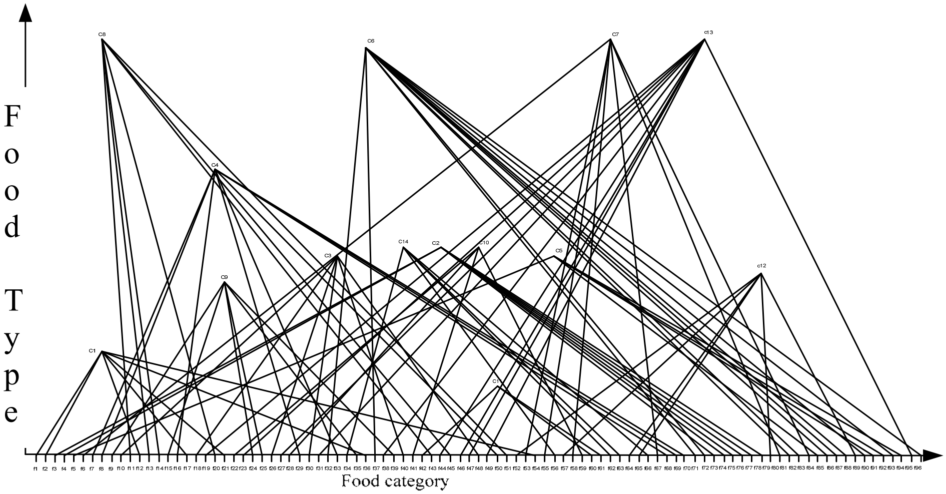


Fig. 4. HAC for Food and Foodtypes

In this example, the milk allergy has D13 and the dairy product is C1, both the combination of D13 and C1 represents V3 that has support 5.

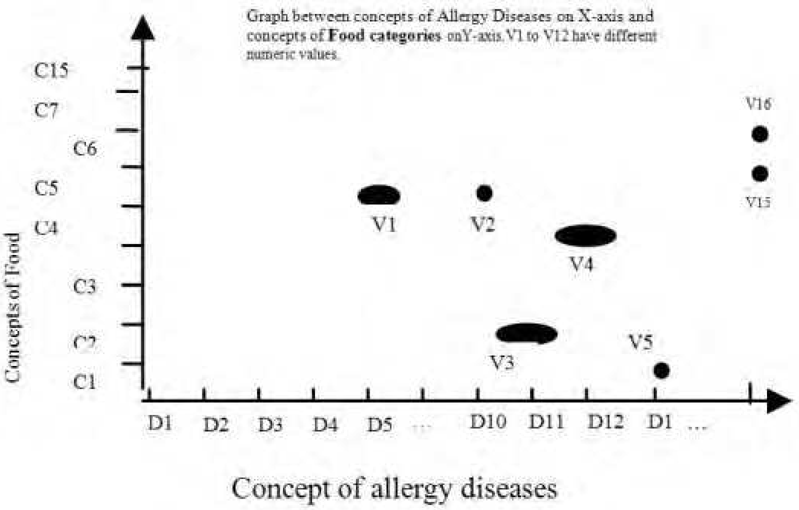


Fig. 5. Cluster Point Graph

7. Implementation

We have used a medical database which contains information about patient causes and food types. This is generated syntactically based on the case studies found in the websites [13, 14].

We used 2 concept tables for implementation of our algorithm; first concept table is for attribute 'Allergy diseases' and second concept table is for attribute 'Food categories'.

Each concept table will have a different hierarchy level for respective attributes, example in Concept table for attribute 'Allergy disease' concept D1 (Acute sinusitis) represents hierarchy level of cold virus, bacterial or fungal, deviated nasal septum, nasal polyps. Similarly concept table for the attribute 'Food categories' represents different hierarchy level of degrees. Using these two concept tables we made 'Cluster Point Table' which has all the possible clusters, each cluster is a combination of 'Allergy disease' attribute and 'Food categories' attribute.

We used non-numeric or categorical data for our clustering algorithm. This algorithm is implemented in 'JAVA' with a simple user interface which makes it very easy to use. User just needs to select different concepts of different attributes, concepts are dynamically generated from concept tables in database. After selecting the concepts the user needs to press the 'Submit' button that will display the result or you can say rules for our 'Cluster Point Graph' on an additional results page. 'Cluster Point Graph' represents all the clusters possible from different combinations of different concepts.

8. Result

Depending on the concept tables the results are generated in the form of rules. Each row in the 'cluster point table' represents a cluster that takes a form of rule. The total number of rules depends on the number of concept tables. Here we are using two concept tables that have generated rules has follows:

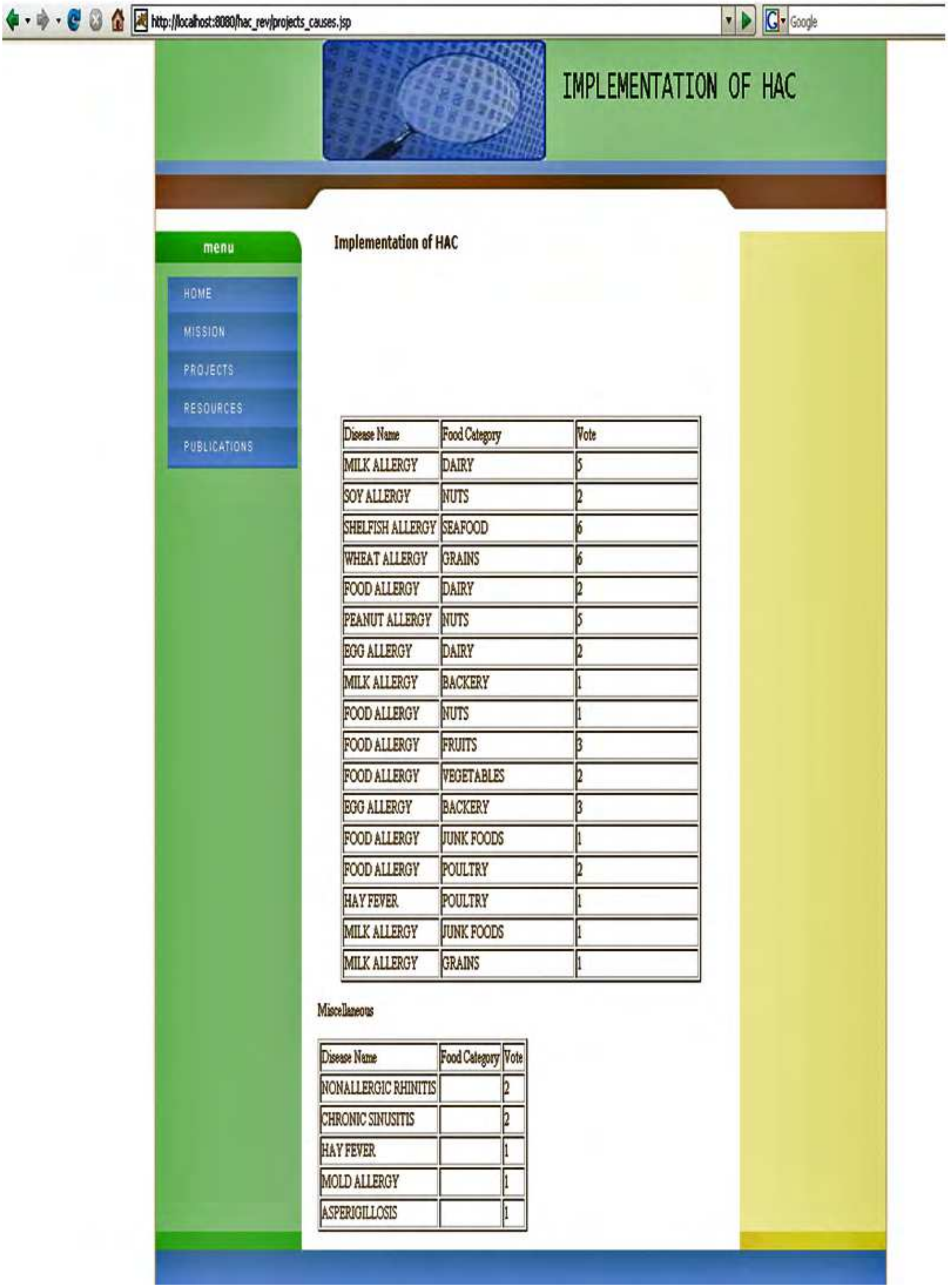
1. The two concept tables used are Allergy Diseases and Food Categories. The sample rules generated such as:

a) Most likely, the persons are affected by specific allergy due to specific food category.

b) The total number of persons with specific values of different concepts.

From the user perspective the user selects the disease and food category then it shows the resulted cluster value from cluster point table. If the user select Concept 'Milk Allergy' and Concept 'Dairy Products' if you submit the query then it will generate the rule as 'There are 5 persons affected by 'Milk Allergy' due to 'Dairy Products' or either way we can generate the rule or there are 45 persons with different symptoms are affected. The sample result is shown in Figure 6.

2. Furthermore, the algorithm can be implemented on three concept tables. These generates rules, based on the concept tables and input tables. The HAC generates different rules based on the concept tables with different combinations of concepts. Each concept table represents conceptual hierarchies of categorical attributes. This algorithm specifies the clustering on categorical attributes on the concept tables to derive the association rules in the form of cluster point table.



Implementation of HAC

Disease Name	Food Category	Vote
MILK ALLERGY	DAIRY	5
SOY ALLERGY	NUTS	2
SHELFISH ALLERGY	SEAFOOD	6
WHEAT ALLERGY	GRAINS	6
FOOD ALLERGY	DAIRY	2
PEANUT ALLERGY	NUTS	5
EGG ALLERGY	DAIRY	2
MILK ALLERGY	BACKERY	1
FOOD ALLERGY	NUTS	1
FOOD ALLERGY	FRUITS	3
FOOD ALLERGY	VEGETABLES	2
EGG ALLERGY	BACKERY	3
FOOD ALLERGY	JUNK FOODS	1
FOOD ALLERGY	POULTRY	2
HAY FEVER	POULTRY	1
MILK ALLERGY	JUNK FOODS	1
MILK ALLERGY	GRAINS	1

Miscellaneous

Disease Name	Food Category	Vote
NONALLERGIC RHINITIS		2
CHRONIC SINUSITIS		2
HAY FEVER		1
MOLD ALLERGY		1
ASPERIGILLOSIS		1

Fig. 6. Sample output screen of Cluster Point table

9. Conclusion

In this paper, we studied the hierarchical conceptual clustering applied on structured databases. We have given a new method of HAC with conceptual clustering to explore categorical data. The main contributions of the paper are to develop HAC algorithm which is applied on categorical attributes instead of traditional algorithms which apply the distance metric measures. The results show that the data is categorized using hierarchical conceptual clustering with HAC. There are numerous types of clustering techniques most of these techniques are applicable only on the unstructured data. Sometimes, there is a need to apply clustering on categorical attributes which is not suitable to apply on it. So, the non-metric measures are used to perform clustering on the categorical attributes which represents the closest proximity between the data attributes. HAC is used to represents databases in the form concept tables for categorical data which contains the concepts formed on the domain. This technique can be applied on any of the fields which have structure data. The information is extracted using the HAC algorithm from structured data.

The structured data is applied on input and the results formed are rules extracted from the data. Clustering is important for both types of data. The modern data mining mechanisms are used to apply on the data. From a machine learning standpoint, this research has been greatly influenced by work in conceptual clustering. HAC seeks classifications that maximize a heuristic measure (as in conceptual clustering systems) and uses a search strategy abstracted from incremental systems such as UNIMEM [8].

10. Future Work

Our algorithm can be effective in the medically related areas to allergic disease. Implementing this algorithm with multiple concept tables can be a potential extension of the approach in our case we have implemented only on two concept tables. Another important enhancement of this algorithm can be its implementation on other domains in medicine and also domains apart from medical areas.

11. Acknowledgement

This research has been partially supported by L-3 Communication Corporation, ComCept Division under Project Corvus and TAMU-C research grant #140854-20300.

12. References

[1] Douglas H. Fisher, “Knowledge Acquisition Via Incremental Conceptual Clustering”, University of California, Machine Learning, Volume 2, Issue 2, pp. 139-172, CA, USA, 1987.

[2] A.K. Jain, M.N. Murty, P.J. Flynn , “Data Clustering: A Review”, ACM Computing Surveys (CSUR), Volume 31, Issue 3, pp. 264 - 323 , September 1999

[3] Jungsoon Park Yoo, Chrisila,C.P., & Sung Yoo. "A Hybrid Conceptual Clustering System", Proceedings of the 1996 ACM 24th Annual Conference on Computer science, pp. 105-114, 1996

[4] Fei Wu., Georges,G., “Gradual clustering Algorithms”, Seventh International Conference on Database Systems for Advanced Applications,2001.

[5] Sudipto Guha1, Rajeev Rastogi and Kyuseok Shim3, “ROCK: A Robust Clustering Algorithm for Categorical Attributes”, IEEE International Conference on Data Engineering, "Information Systems", Volume 25, Number 5, pp. 345 – 366, 2000.

[6] Sang C.Suh, Sam I. Saffer, Nikil Goel, “Generating Meaningful Rules Using Attribute Concept Hierarchy “, Artificial Neural Networks in Engineering, pp. 1-6, 2006.

[7]. E. A. Freigenbaum, H. Simon, “EPAM-like models of recognition and learning”, Cognitive Science, Volume 8, pp. 305 - 336, 1984.

[8] M. Lebowitz, “Experiment with incremental concept formation: UNIMEM”, Machine Learning, Volume: 2, 1987, pp. 103 -138

[9] Pyo Jae Kim, Jin Young Choi, “Incremental Conceptual Clustering Using a Modified Category Utility”, http://iccl.snu.ac.kr/ilab/paper/dist_file/ 124ITC_pjkim.pdf

[10] Carl J. Granberg, Lingi, “Hierarchical clustering of large volumetric datasets”, Computer graphics and interactive techniques in Australasia and South East Asia, Proceedings of the 3rd international conference on Computer graphics and interactive techniques in Australasia and South East Asia, pp. 425 – 428,2005

[11] Heckel B.,Weber G., Hamann B., Joy K, ”Construction of vector field hierarchies”, Proceedings of Visualization’99, pp. 19–25,1999

[12] Telea A., van Wijk, J, “Simplified representation of vector fields”, Proceedings of Visualization’99, pp.35–42,1999

[13] www.mayoclinic.com

[14] http://www.wrongdiagnosis.com

Machine Learning
Edited by Yagang Zhang

ISBN 978-953-307-033-9
Hard cover, 438 pages
Publisher InTech
Published online 01, February, 2010
Published in print edition February, 2010

Machine learning techniques have the potential of alleviating the complexity of knowledge acquisition. This book presents today's state and development tendencies of machine learning. It is a multi-author book. Taking into account the large amount of knowledge about machine learning and practice presented in the book, it is divided into three major parts: Introduction, Machine Learning Theory and Applications. Part I focuses on the introduction to machine learning. The author also attempts to promote a new design of thinking machines and development philosophy. Considering the growing complexity and serious difficulties of information processing in machine learning, in Part II of the book, the theoretical foundations of machine learning are considered, and they mainly include self-organizing maps (SOMs), clustering, artificial neural networks, nonlinear control, fuzzy system and knowledge-based system (KBS). Part III contains selected applications of various machine learning approaches, from flight delays, network intrusion, immune system, ship design to CT and RNA target prediction. The book will be of interest to industrial engineers and scientists as well as academics who wish to pursue machine learning. The book is intended for both graduate and postgraduate students in fields such as computer science, cybernetics, system sciences, engineering, statistics, and social sciences, and as a reference for software professionals and practitioners.

Sang C. Suh, Nagendra B. Pabbisetty and Sri G. Anaparthi (2010). Extraction Of Meaningful Rules In A Medical Database, Machine Learning, Yagang Zhang (Ed.), ISBN: 978-953-307-033-9, InTech, Available from: http://www.intechopen.com/books/machine-learning/extraction-of-meaningful-rules-in-a-medical-database

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821