International Journal of Advanced Computer Research (ISSN (print): 2249-7277 ISSN (online): 2277-7970)
Volume-2 Number-1 Issue-3 March-2012

Comparative Study between Traditional and Heuristic Approach on Close-degree Concept Lattice for better Attribute Reduction

Ravindra k. Gupta[1], Sapna Sahu[2], Shailendra Singh[3]
HOD, CSE, SSSIST, Sehore[1]
M.Tech Scholar, CSE, SSSIST, Sehore[2]
Professor (CSE), NITTTR, Bhopal[3]

Abstract

Concept lattice is a new mathematical tool for data analysis and knowledge processing. There is an increasing interest on application of concept lattices in the different information systems. The concept lattice may be used for representation of the concept generalization structure generated from the underlying data set. Attribute reduction is very important in the theory of concept lattice because it can make the discovery of implicit knowledge in data easier and the representation simpler. The paper presents a modified lattice building algorithm where the generated concept nodes may contain not only the attributes of the children nodes but some other generalized attributes, too. The generalization structure of the attributes is called attribute lattice. In this approach we find concept lattice, Attribute Concept lattice and, Formal Concept Lattice based on all the deduction we find the Heuristic Result which provide better attribute reduction. We also discuss a comparative study between the traditional and the new heuristic approach.

Keywords

Concept Lattice, Attribute Reduction, Close-degree, Formal concept Lattice

1. Introduction

Formal concept analysis refers to both an unsupervised machine learning technique and, more broadly, a method of data analysis. The approach takes as input a matrix specifying a set of objects and the properties thereof, called attributes, and finds both all the "natural" clusters of attributes and all the "natural" clusters of objects in the input data, where

- a "natural" object cluster is the set of all
- objects that share a common subset of attributes, and
- a "natural" property cluster is the set of all attributes shared by one of the natural object clusters.

Natural property clusters correspond one-for-one with natural object clusters, and a concept is a pair containing both a natural property cluster and its corresponding natural object cluster. The family of these concepts obeys the mathematical axioms defining a lattice, and is called a concept lattice (in French this is called a Treillis de Galois because the relation between the sets of concepts and attributes is a Galois connection).

The field of Formal Concept Analysis [1] introduced in the early 80ies has grown to a powerful theory for data analysis, information retrieval and knowledge discovery. There is nowadays an increasing interest in the application of concept lattices for data mining, especially for generating association rules [3]. One of the main characteristics of this application area is the large amount of structured data to be analyzed. A technical oriented application field of Formal Concept Analysis is the area of production planing where the concept lattices are used to partition the products into disjoint groups during the optimization of the production cost [4]. As the cost of building a concept lattice is a super-linear function of the corresponding context size, the efficient computing of concept lattices is a very important issue, has been investigated over the last decades [5].

We provide here an overview of executing different services based on heuristic approach. The rest of this paper is arranged as follows: Section 2 introduces Concept Analysis; Section 3 describes about Attribute Reduction; Section 4 shows the recent scenario; Section 5 describes the Proposed Work. Section 6 describes Conclusion and outlook.

2. Concept Analysis

Conceptual Knowledge Discovery in Databases (CKDD) has been developed in the field of Conceptual Knowledge Processing. Based on the mathematical theory of Formal Concept Analysis, CKDD aims to support a human-centered process of discovering knowledge from data by visualizing and analyzing the formal conceptual structure of the data.The overall theme and contribution of the

volume "Advances in Knowledge Discovery and Data Mining" [6] is a process-centered view of KDD considering KDD as an interactive and iterative process between a human and a database that may strongly involve background knowledge of the analyzing domain expert. In particular, R. S. Brachman and T. Anand [7] argue in favor of a more human-centered approach to knowledge discovery support referring to the constitutive character of human interpretation for the discovery of knowledge and stressing the complex, interactive process of KDD as being led by human thought. Following Brachman and Anand, CKDD pursues a human-centered approach to KDD based on a comprehensive notion of knowledge as a part of human thought and argumentation.

Natural property clusters correspond one-for-one with natural object clusters, and a concept is a pair containing both a natural property cluster and its corresponding natural object cluster. The family of these concepts obeys the mathematical axioms defining a lattice, and is called a concept lattice (in French this is called a Treillis de Galois because the relation between the sets of concepts and attributes is a Galois connection). The above phenomena are shown in Fig 1.

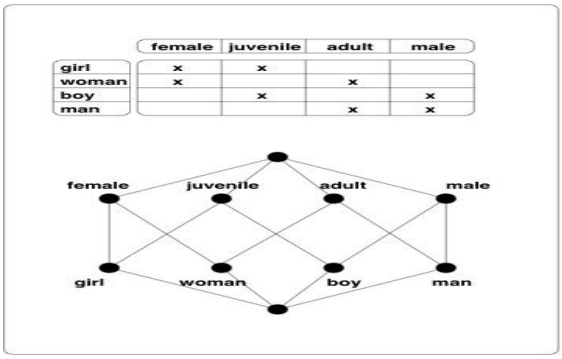

Fig 1: Concept Lattice

3. Attribute Reduction

Attribute Reduction process is shown in Fig 2. In this phase the data values are analyzed first. Then we apply the separation by using formal concept. After applying formal concept we analyze the concept that particular phase is called concept analysis. The last phase is attribute reduction which is based on all the above phenomena for which the underlying conditions is applicable to the heuristic result. Then we get that result where less redundant attributes are present.

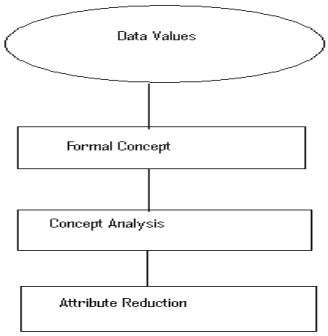

Fig 2: Attribute Reduction

4. Recent Scenario

In 2008, Jen-Wei Huang et al. [8] proposed about a progressive algorithm Pisa, which stands for Progressive mIning of Sequential patterns, to progressively discover sequential patterns in defined time period of interest (POI). The POI is a sliding window continuously advancing as the time goes by. Pisa utilizes a progressive sequential tree to efficiently maintain the latest data sequences, discover the complete set of up-to-date sequential patterns, and delete obsolete data and patterns accordingly. The height of the sequential pattern tree proposed is bounded by the length of POI, thereby effectively limiting the memory space required by Pisa that is significantly smaller than the memory needed by the alternative method, Direct Appending (DirApp).

In 2010, Shiow-yang Wu et al. [9] proposed about a complex activity is modeled as a sequence of location movement, service requests, the co-occurrence of location and service, or the interleaving of all above. An activity may be composed of sub activities. Different activities may exhibit dependencies that affect user behaviors. They argue that the complex activity concept provides a more precise, rich, and detail description of user behavioral patterns which are invaluable for data management in mobile environments. Proper exploration of user activities has the potential of providing much higher quality and personalized services to individual user at the right place on the right time.

In 2010, Huili Meng et al. [10] proposed about the reduction of the concept lattice. First, they present a close degree of concept to measure the close-degree of two concepts with the attributes reduction. Based

on the close-degree of two concepts, we propose the close-degree of concept lattice. Then they use the close-degree of concept lattice as heuristic information and design an attribute reduction algorithm. The reduction algorithm attempts to get one reduction of the attribute set of concept lattice. Last, they give an application example for proving the validity of the algorithm.

5. Proposed Work

In this section we proposed a Heuristic approach for better attribute reduction.

Our approach is divided into four parts
1) Concept Lattice
In this phase we calculate all the true values based on 1 in the data set. We calculate all the rows attribute whose value is 1 and left those attribute that are not in use.

2) Attribute Lattice
In this phase we integrate those rows value which are Homogeneous according to the concept lattice.
For example:
1- abe
2,4- abc
3-d
4,2 –abc

3) Formal Concept Lattice
In this phase we select only those attributes which are homogeneous in nature and left all other related values so that the memory size is reduced for our approach.
For example:
2,4-abc
4,2-abc

4) Heuristic Result
In this phase we have the privilege for finding reduction based on attribute values and also on the based on occurrences. This gives us the privilege to take only those attributes in the account which are currently being used or may be used in the services.

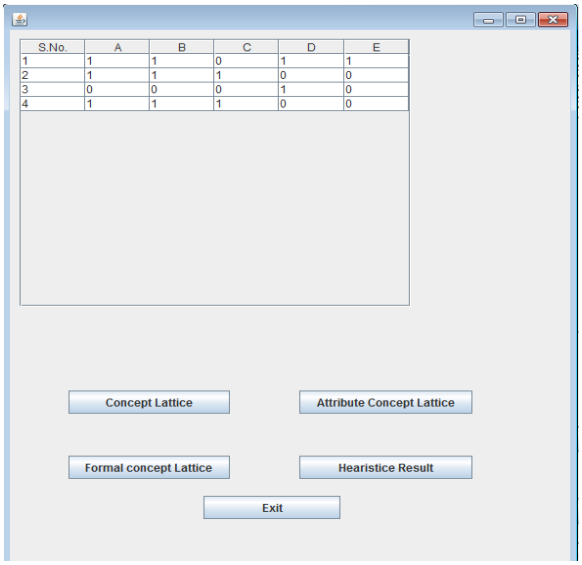

Fig 3: Data Value for attribute Reduction

Comparision
A heuristic evaluation is a usability inspection method for computer software that helps to identify usability problems in the user interface (UI) design. It specifically involves evaluators examining the interface and judging its compliance with recognized usability principles (the "heuristics").

Heuristic evaluations usually are conducted by a small set (one to three) of evaluators. The evaluators independently examine a user interface and judge its compliance with a set of usability principles. The result of this analysis is a list of potential usability issues or problems.

The usability principles, also referred to as usability heuristics, are taken from published lists. Ideally, each potential usability problem is assigned to one or more heuristics to help facilitate fixing the problem. As more evaluators are involved, more true problems are found.

The method can provide some quick and relatively inexpensive feedback to designers. Feedback can be obtained early in the design process. Assigning the correct heuristic can help suggest the best corrective measures to designers.

In this phase we have the privilege for finding reduction based on attribute values and also on the based on occurrences. This gives us the privilege to take only those attributes in the account which are

currently being used or may be used in the services. Where we also apply frequency based deduction.

The result analysis is based on frequency of Heuristic result and traditional algorithm. It possibly takes a very long time on large inputs until the program has completed its work and gives a sign of life again. Sometimes it makes sense to be able to estimate the running time before starting a program. Obviously, the running time depends on the number n of the strings to be sorted.

There is often a time-space-tradeoff involved in a problem, that is, it cannot be solved with few computing time and low memory consumption. One then has to make a compromise and to exchange computing time for memory consumption or vice versa, depending on which algorithm one chooses and how one parameterizes it.

Sometimes we find the statement in the manual that an operation takes amortized time O(f(n)). This means that the total time for n such operations is bounded asymptotically from above by a function g(n) and that f(n)=O(g(n)/n). So the amortized time is (a bound for) the average time of an operation in the worst case. The special case of an amortized time of O(1) signifies that a sequence of n such operations takes only time O(n). One then refers to this as constant amortized time. Such statements are often the result of an amortized analysis: Not each of the n operations takes equally much time; some of the operations are running time intensive and do a lot of "pre-work" (or also "post-work"), what, however, pays off by the fact that, as a result of the pre-work done, the remaining operations can be carried out so fast that a total time of O(g(n)) is not exceeded. So the investment in the pre-work or after-work amortizes itself.

6. Conclusion

Concept lattice is a new mathematical tool for data analysis and knowledge processing. Attribute reduction is very important in the theory of concept lattice because it can make the discovery of implicit knowledge in data easier and the representation simpler. The paper presents a modified lattice building algorithm where the generated concept nodes may contain not only the attributes of the children nodes but some other generalized attributes, too. The generalization structure of the attributes is called attribute lattice. In this approach we find concept lattice, Attribute Concept lattice and, Formal Concept Lattice based on all the deduction we find the Heuristic Result which provide better attribute reduction. The comparison study shows that the method is good in comparison to the previous one.

References

[1] B. Ganter and R. Wille: Formal Concept Analysis, Mathematical Foundations, Springer Verlag, 1999.

[2] R. Godin and R. Missaoui and H. Alaoui: Incremental concept formation algorithms based on Galois lattices,Computational Intelligence, 11(2), 1995, 246-267 .

[3] K. Hu and Y. Lu and C Shi: Incremental Discovering Association Rules: A Concept Lattice Approach, Proceedings of PAKDD99, Beijing.

[4] C. Lindig: Fast Concept Analysis, Proceedings of the 8 th ICCS, Darmstadt, 2000 .

[5] L. Nourine and O. Raynaud: A Fast Algorithm for Building Lattices, Information Processing Letters, 71, 1999, 197-210.

[6] U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, R. Uthurusamy (eds.): Advances in Knowledge Discovery and Data Mining. AAAI/MIT Press, Cambridge 1996.

[7] R. J.Brachman, T.Anand: The process of knowledge discovery in databases.

[8] Jen-Wei Huang, Chi-Yao Tseng, Jian-Chih Ou, and Ming-Syan Chen," A General Model for Sequential Pattern Mining with a Progressive Database",IEEE VOL. 20, NO. 9, SEP. 2008.

[9] Shiow-yang Wu, and Hsiu-Hao Fan, "Activity-Based Proactive Data Management in Mobile Environments",IEEE Transactions on Mobile Computing, VOL. 9, NO. 3, March 2010.

[10] Huili Meng, Jiucheng Xu, "The Close-degree of Concept Lattice and Attribute Reduction Algorithm Based on It", 2010 IEEE International Conference on Granular Computing.