Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 5 Issue 11, November-2016

An Improved Design for Association Rule Discovery System for Decision Support System

Macarthy O.[1]
[1]Dept. of Computer Science,
University of Port Harcourt, Nigeria

Onyejegbu L. N.[2]
[2]Dept. of Computer Science,
University of Port Harcourt, Nigeria

***Abstract* - The challenge of associating rules to data set of items has been an important area in data mining research. Recent challenges is how to improve the algorithms that have been used in discovering associations that seem complex but are more useful in decision making and analysis. This paper presents a design for association rule discovery for decision support systems using an improved C4.5 Algorithm. This research work provides an efficient discretization method for the data which aids the discretization of continuous data and improves the rule discovery process from the different data sets and data attributes.**

Keyword: Association Rules, C4.5 Algorithm, Decision Support System, Discretization

I. INTRODUCTION

Mining association rules from large data sets has been a focused topic in recent research into knowledge discovery in stored data. In this project mining association rules may lead to the discovery of more specific and concrete knowledge from data [1]. A method is developed for mining association rules from large transaction databases by extension of some existing association rule mining techniques.

An association relationships rule such as

{Milk,Tea} → {Bread} in the sales data of a store indicates that if a customer buys milk and tea together he is likely to also buy bread. The information can be used for decision about marketing activities such as promotional, pricing or product purchases. Association rules are used in many applications in web data mining, weather forecast, robot control, intrusion detection and bioinformatics.

An association rule is a class of regularities expressed as X Y, where X and Y are sets of items. It implies that the transaction of the database which contain X tend to contain Y. For instance a rule might be that 85% of customers that purchase milk and sugar also purchase tea. The domains of application of association rules include decision support, marketing, business management, analysis of sales data down to diagnosis and prediction.

To study the mining of association rules from a large set of transaction data, we assume that the database contains 1.) a transaction data set, 7, which consists of a set of transactions (Ti, {AP, . . . , A4}), where x is a transaction identifier, Ai E Z (for i = p, . . . , q), and Z is the set of all the data items in the item data set; and (2) the description of the item data set, D, which contains the description of each item in Z in the form of (Ai, description), where Ai E Z. Furthermore, to facilitate the management of large sets of transaction data, our discussion adopts an extended relational model which allows an attribute value to be either a single or a set of values (i.e., in non-first-normal form). Nevertheless, the method developed here is applicable (with minor modifications) to other representations of data, such as a data file, a relational table, or the result of a relational expression.

A pattern, A, is one item Ai or a set of conjunctive items Ai A . . . A Ai, where Ai, . . . ,Aj E 1. The support of a pattern A in a set S, a(A/S), is the number of transactions (in S) which contain A versus the total number of transactions in S. The confidence of A+ B in S, cp(A-, B/S), is the ratio of u(A A B/S) versus o(A/S), i.e., the probability that pattern B occurs in S when pattern A occurs in S. To find relatively frequently occurring patterns and reasonably strong rule implications, a user or an expert may specify two thresholds: minimum support.

.

A. Rule Discovery System

Rule Discovery is data mining method of learning the association rule for interesting relations of variables that occurs in large databases. It finds out the relationships in databases via certain measures of interestingness that are anchored on some strong rules.

The Association rule to be learnt is a class of items displayed as W Z, where W and Z are sets of items and W items is very likely to contain Z items. For example a rule such as "85% of customers that buy milk and sugar in a store also buy tea". The areas of application of association rules include decision support, finance, marketing, business management, analysis of sales data, medical diagnosis and other prediction system[1].

In other to research the mining of association rules from a large set of transaction data, we assumed that the data contains:

- A transaction data set which is made up of a set of transactions. Ti = { d1….di} where d is the set of all the data items in the item data set.
- The description of the item data set, D, which contains the description of each item in D, where di ∈ D.

B. Mining Association Rule

The mining of association rule involve the use of rule extraction algorithm which could be applied to various data in an effort to carry out specific roles which include rule extraction, rule clustering and rule pruning [2].

1) Rule Extraction (RE): Its function is to initialize the mined rule list by making it hollow. It sorts the content according to certain frequency, it selects common sample as the base to produce and then add the rule to the list of other extracted

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 5 Issue 11, November-2016

rules. After that, all the samples would be found and removed. The process will be repeated until the example space is exhausted.
2) Rule Clustering: Here, rules are grouped by their various category levels. The rules that belong to same group are clustered together.
3) Rule Pruning: In rule pruning, redundant and repeated rules in a given group are expunged. In a given cluster, many rules may be explained using same example. For instance given the rule "if (color =green) and (height 4) then grass" is captured in a general rule "if (color = green)then grass", this shows that the rule "if (color = green) and (height 4) then grass" is a redundant rule. Rule extraction removes the redundant rules in groups in other to minimize the size of the best rule list. [3].

Several research work on mining association rules have developed from methods for discovery of functional dependencies [4],strong rules[5], classification rules [6],causal rules, clustering, etc. to disk-based methods. It has advanced to better methods for extracting association rules in big data [7] and larger data sets.

II. ASSOCIATION RULE MINING AS A DATA MINING TECHNIQUE

It is clear that data retrieved in huge databases become raw facts for use in checking for knowledge discovery techniques and extracting tools for "gold" were necessary. The present technologies usually depend on users to manually input knowledge into databases. This procedure usually contains errors, and it is extremely time-consuming and costly. Data mining is a strong process to search for, find patterns and relations between data stored in data warehouses. The result of any given mining process, discovery and interpretation of knowledge is provided by the unknown elements in the data set [8]. A datamining process is shown in figure 2 below.

Table 1:Database with transactions (Source [9]).

Transactions	Items
"Customer1"	"BOOKS, CD, VIDEO"
"Customer2"	"CD, GAMES"
"Customer3"	"CD, DVD"
"Customer4"	"BOOKS, CD, GAMES"
"Customer5"	"BOOKS, DVD"
"Customer6"	"CD, DVD"
"Customer7"	"BOOKS, DVD"
"Customer8"	"BOOKS, CD, DVD, VIDEO"
"Customer9"	"BOOKS, CD, DVD"

Analyzing customers transactions in table 1, an Apriori algorithm [9] for each items support count. The item which has a count that is less than the minimum support is needed to be deleted from the pool of candidate items. Then step one of this process is to generate 1-itemset regular occurring Pattern.

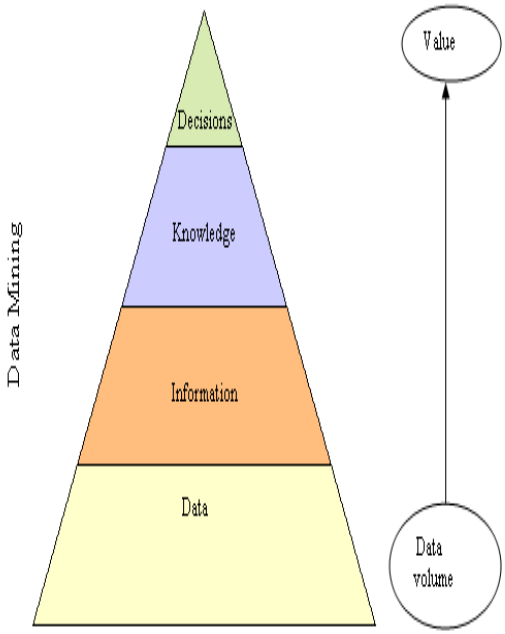

Fig. 1: Data mining process (Source Irina, 2008)

III. ANALYSIS OF EXISTING SYSTEM

There are many existing models in data mining for association rule discovery. We will not analyze all but the system from where our new model will be derived will be analyzed. The model include the C4.5 algorithm.

C4.5 is a standard decision tree generating algorithm that is used to build systems that generate decision trees used in decision making process. It is an improvement and extension of the ID3 algorithm meant for the same purpose but deployed to overcome its disadvantages. C4.5 model, a decision tree generator handles regular, irregular and discrete data patterns. It is the best algorithm for rule discovery and handles large data set giving opportunity for selection. While IDE 3 model can only handle small size data and without selection opportunity. The decision trees generated via C4.5 algorithm can be deployed for classification, making it to also be a statistical classifier. There are certain improvements and changes made to the ID3 algorithm in other to evolve it into C4.5 algorithm [10]. However we still intend to improve the algorithm further by:

i) i)Taking care of situation involving large dataset.
ii) Taking care of missing values of attributes in training data set.
iii) Taking care of varying cost attributes.
iv) Improving the pruning of the tree after its creation and
v) Taking care of attribute that contain discrete as well as continuous values.

A. Existing C4.5 Algorithm

The C4.5 algorithm [11] before modification presented below, clearly shows that the C4.5 algorithm constructs the decision tree using a divide and conquer strategy.

Step1: ComputeClassFrequency(T);

Step2: if OneClass or FewCases
return a leaf;

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 5 Issue 11, November-2016

Create decision node (N);

Step3: For Each Attribute A
ComputeGain(A);

Step4: N.test = AttributeWithBestGain;

Step5: if N.test is continuous
find Threshold;

Step6: For Each T in the splitting of T

Step7: if T is Empty
Child of N is a leaf
else

Step8: Child of N = FormTree(T)

Step9: ComputeErrors of N;
return N

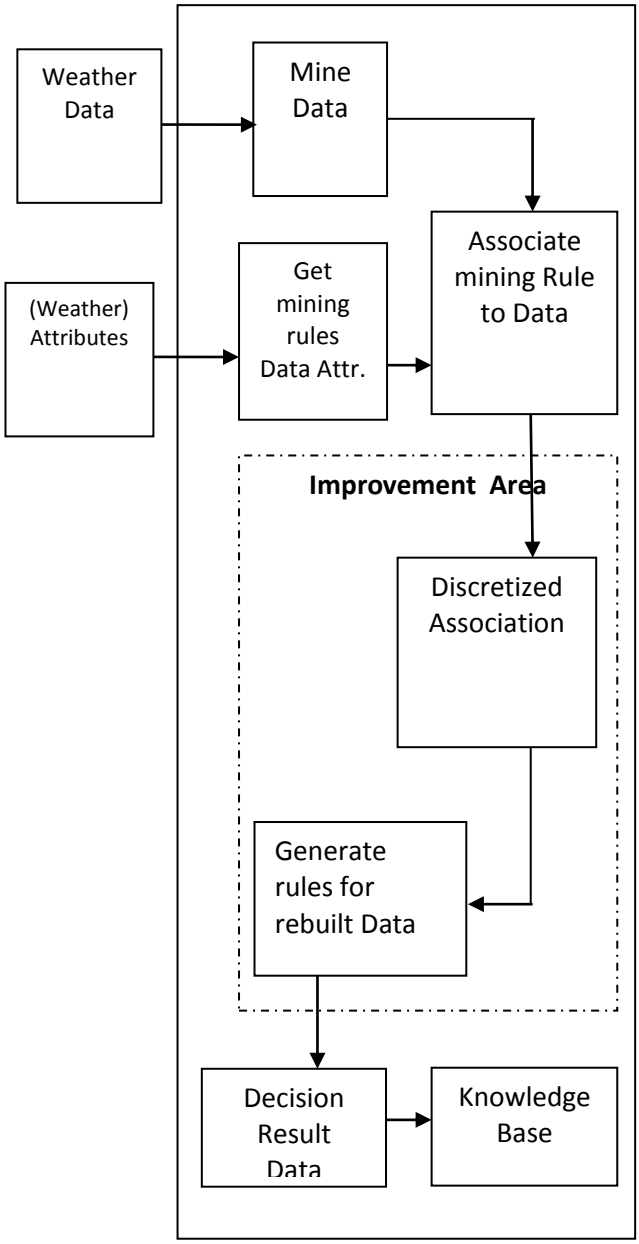

Fig. 2: The Proposed Architecture of the Rule Discovery System

IV. IMPROVED C4.5 ALGORITHM

The algorithm for the system is presented below:

1. Select dataset as an input to the algorithm for processing.
2. Create a given root node for the given tree
3. If all dataset are positive return the single-node tree root, with label = +
4. If all dataset are negative return the single-node tree root, with label = -
5. If attribute is empty, return the single-node tree Root, with label = most common value of Target attribute in dataset
6. Select the classifiers
7. A the attribute from Attributes that best classifies the dataset
8. The decision attributes for Root A
9. Calculate entropy, information gain, gain ratio of attributes.
10. For each given value vi,
11. Insert a new branch of the tree below the Root that corresponds to the test A = vi
12. Let dataset vi be subset of dataset that have value vi for A
13. If dataset vi is empty
14. Then below the new branch, add a leaf node with label = most common value of Target_attribute in Dataset
Else
15. Below this new branch add the subtree
Goto Step 2
16. Tree generator generates the decision tree.

C4.5 also contains a mechanism to re-express decision trees as ordered lists of if-then rules. Each of the path gives the condition that must be satisfied if a case is to be classified by that leaf. C4.5 generalizes this prototype rule by dropping any conditions that are irrelevant to the class, guided again by the heuristic for estimating true error rates. The set of rules is reduced further based on the MDL principle described above. There are usually substantially fewer final rules than there are leaves on the tree, and yet the exactness of the tree and the derived rules is similar. Rules have the added advantage of being more easily understood by people.

A. Use Case Design of Rule Discovery System for Improved C4.5 model

Figure 3 shows the use case design in this dissertation. It has four actors and eight corresponding actions. The first actor is the data source admin who ensures that data gotten from the metrological station is entered correctly into the system. The data source admin generates the data file used in rule discovery and data mining process. The rule discovery system is an actor which performs certain actions such as discretization of data, reading of the data into the attribute data store from where processing activities are expected to take place. The aggregation of the processing data in the system is also handled by the rule discovery system which equally takes care of the mining of the processed data after the C4.5 must have processed the data.

The C4.5 Application Programming Interface (API) an actor, uses its internal modules to handle discretization of data and also read attributes data file into the attribute data store for matching with the dot data file containing raw data. The processing of data proper is also handled by the API using the C4.5 algorithm in the API.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 5 Issue 11, November-2016

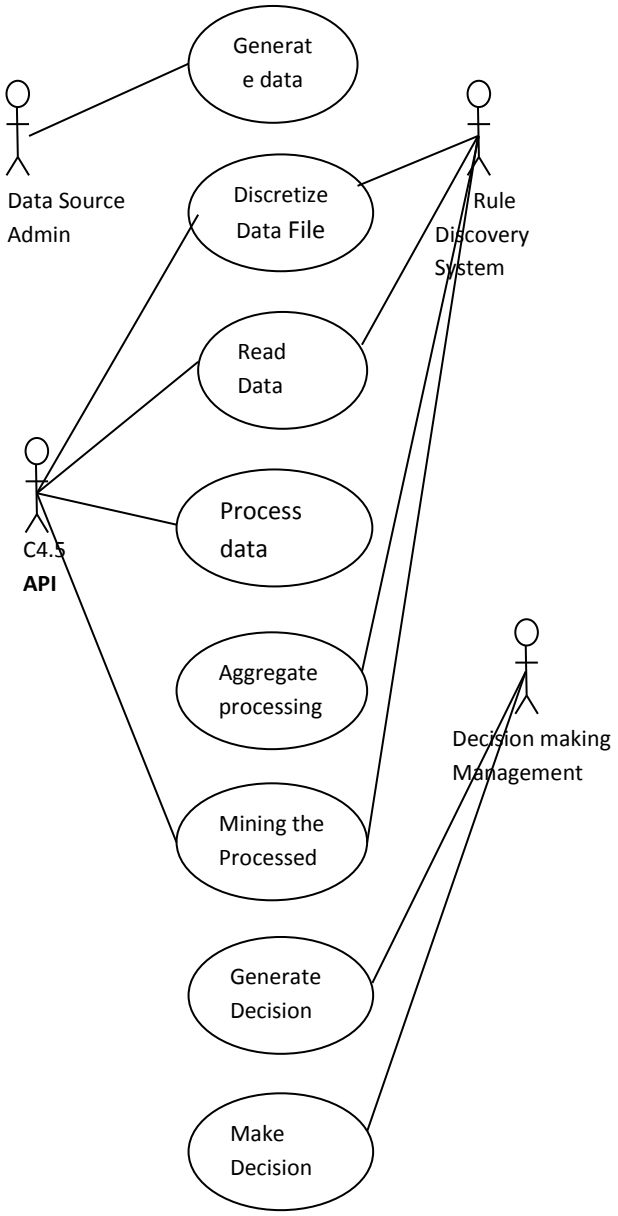

Fig. 3: Use case design of the Rule Discovery System for Improved C4.5 model

The data mining process is also handled with the rule discovery system from where we expect the system to produce decision tree to be used as a decision support system for management decisions. The last actor is the decision making management saddled with the responsibility of generating understandable decision tree structure which it also uses in taking informed decision on areas needed and on which data has been collected and processed.

B. UML Design of Rule Discovery for Improved C4.5 model

The UML design visualizes and specifies the several class that are interrelated and function together as separate modules for the system to be implemented. It also models the various modules of the system in rule discovery analysis as well as the data mining API fixed in the system. The UML model design will assist in understanding the system at the implementation level and equally decrease the difficulty in the implementation of the system for developers and users. In figure 4, the design clearly describe the behavior of the RuleDisPro class in terms of its state, stimuli and its transition and the other classes that interact with it in the program directly and by inheritance. In the design, Calculate class extends the RuleUsageData class allowing some methods such as trainUsage and testUsage to be available for Calculate class in equation handling that is required in the data mining process. The Discritisation class and the C4.5 File Filter class as well as the RuleUsageData class equally extends the RuleDisPro class making the RuleDisPro method to be readily available for use by all the class that extended it. It is equally clear that the main(args) method in the RuleDisPro class automatically makes the class to be dependent to the Frame1 class defined in figure 5 and to other classes defined in figure 4. The frame object defined as a field in RuleDisPro class as an object of Frame1 a class designed as an extension of JFrame class in the Java standard classes.

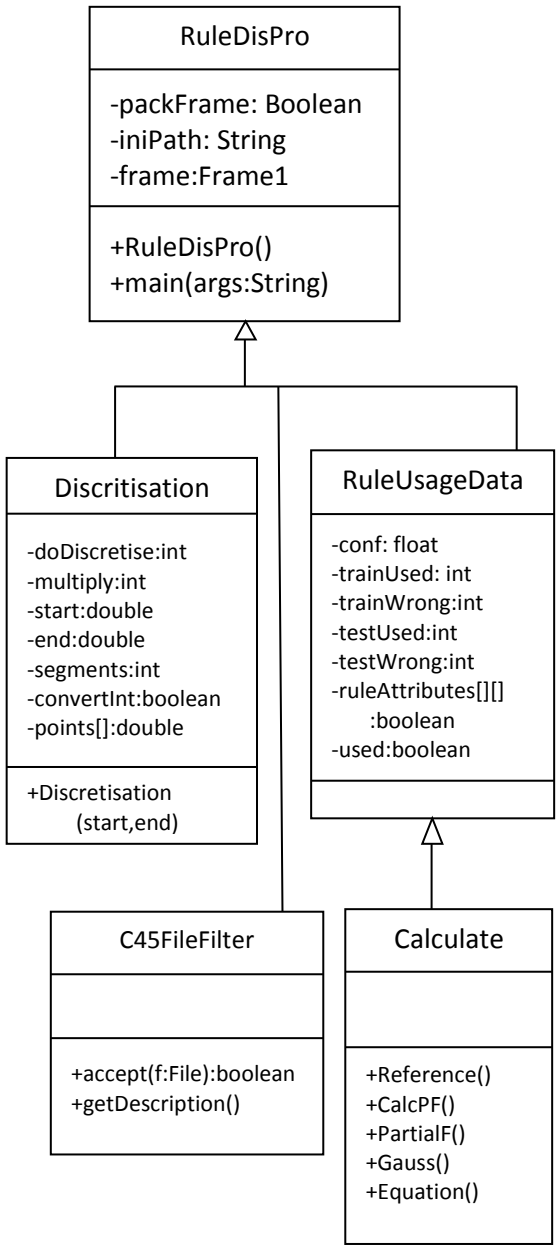

Fig. 4: A UML Design of the Rule Discovery System for Improved C4.5

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 5 Issue 11, November-2016

The Java standard class relationship with some classes defined in Rule discovery system designed in this chapter are clearly illustrated in figure 5 Though there are other Interfaces Rule discovery system may have implemented, the major classes they inherited where the once presented in the UML design in the dissertation design work present in figure 5. The root is the Swing package which have most of the GUI classes which can be reused by simply extending the classes. The major classes are JFrame and its contents, JDialog, JCheckBox and a ListCellRenderer interface.

There are two main classes that are inheriting the JFrame class, this classes include the Frame1 and the HelpWindow class. The Frame1 class has serial version field and other fields that are used to define the objects that instantiate several other classes deployed in the system. The methods in the classes also include Frame1 a constructor method, the method that initiates the variable, test Rules that examines the discovery rules, readDataFile that reads the data into the C4.5, runC4.5 that loads the C4.5 API and run C4.5 Rules method executes the rule discovery, enumDecisionValues that enumerate the decision values, runC45Rules system using the C4.5 improved algorithm.

The HelpWindow class contains the process WindowEvent which is used to display the items necessary for help information on a special and separate window. The JDialog class was extended by the Frame AboutBox which used dialogBox to provide brief information on the system application.

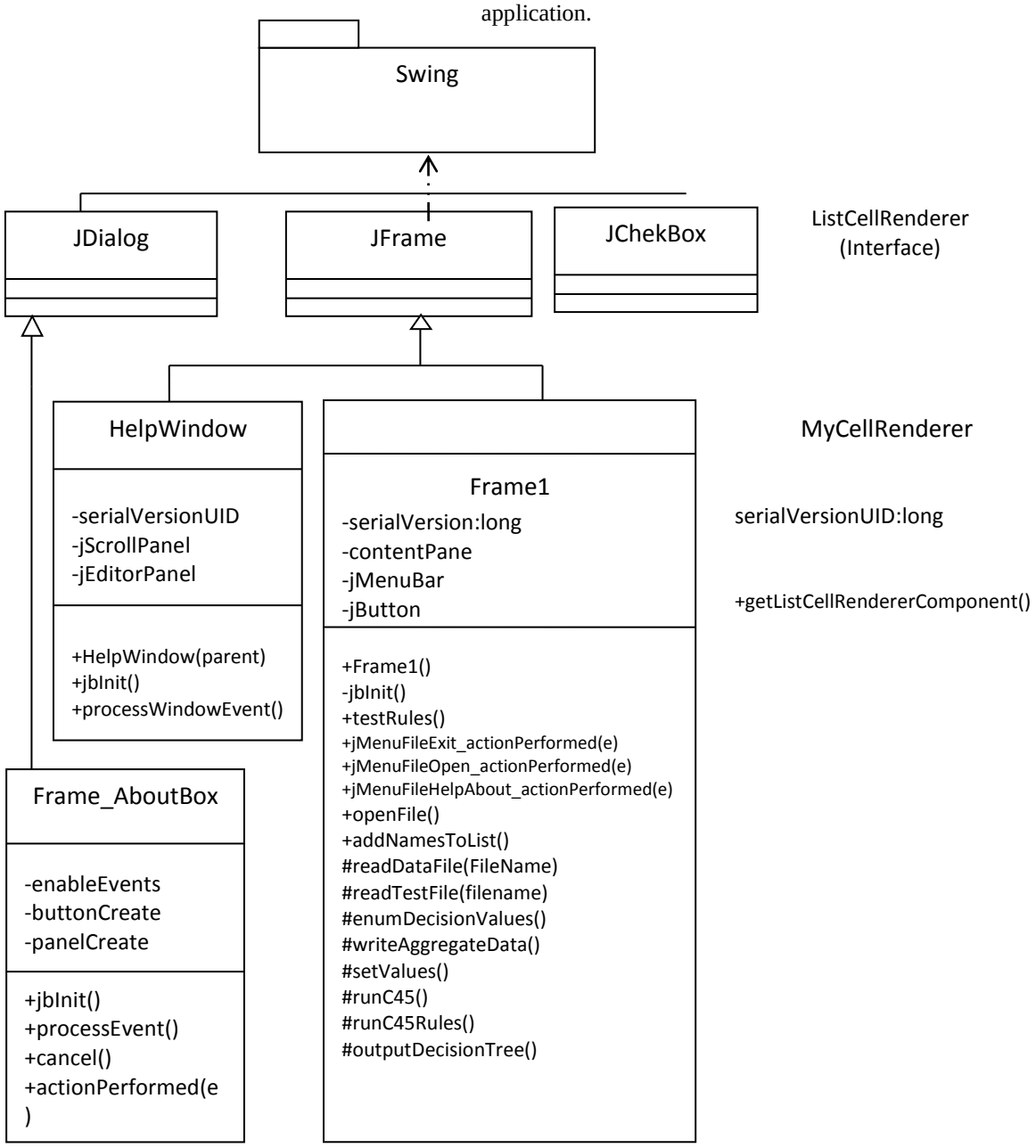

Fig. 5: A UML Design showing Swing Inheritance of Improved C4.5 User Interface

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 5 Issue 11, November-2016

V. CONCLUSION

The design of a data mining system for association rule discovery in building decision support system is presented in this paper. This was done by designing the model for association rule discovery using Object-Oriented methodology. The designed system model used UML case diagram and class diagram to represent the classes and the process flow of the system. This design provides the benefit of stepwise development and also adding scalability features to systems as at when the users needed them in building the decision support system from the data provided. This makes the system mining process more accurate and more predictable and rule discovery process more reliable. This is coupled with the need to meet the need for consistency in software development in areas such as precision in mining the data provided for the system. Finally the Java class hierarchy design and its link to our class implementations was presented to ease the implementation process.

VI. RECOMMENDATION

We recommend the system design principles to all developers and in object-oriented work requiring data mining activities and rule discovery use in developing system. By using the design system, each developer will be able to implement the system easily. Researchers working on project on rule discovery processing will need the use of design classes for deploying the principle of reusability. The increased usage of the design will further improve the system when the detail components are further explored. We equally recommend the work to researchers who will use it in object-oriented implementation using OOP languages for data mining program development. Advanced research will expand the scope of the work which will make the project useful as a launching pad for development of more advanced work involving decision support system development.

ACKNOWLEDGEMENTS

The main author wish to acknowledge her parents Chief and Mrs. E.O. Macarthy for the moral support provided during the research. Her lovely husband, Mr. Hutchinson Marcus O. is also acknowledged for his prayers and financial support which led to the actualization of the research. I would also appreciate the positive contributions from Apostle Diri Dandison T. and others who assisted in one way or the other to get the work done.

REFERENCES

[1] Jiawei H. and Yongjian F. (1995) Discovery of Multiple-Level Association Rules from Large Databases, Proceedings of the 21st VLDB Conference Zurich, Swizerland.
[2] Gaurab T. (2015) Effective Data Mining For Proper Mining Classification Using Neural Networks, International Journal Of Data Mining & Knowledge Management Process (IJDKP) Vol.5, No.2. 112-125
[3] AgrawaI R. and Srikant R. (1995) Mining sequential patterns. In Proceeding International Conference on Data Engineering, Taipei, Taiwan.
[4] Mannila H. and Raiha K. J. (1987) Dependency Inference. In Proc. 1987 Int. Conf. Very Large Data Bases, pp. 155-158, Brighton, England.
[5] Piatetsky-Shapiro G. (1991) Discovery, analysis, and presentation of strong rules. In G. Piatetsky-Shapiro and W. J. Frawley, editors, Knowledge Discovery in Databases, 229-238. AAAI/MIT Press.
[6] Han J., Cai Y., and Cercone N. (1993) Data-driven discovery of quantitative rules in relational databases. IEEE Tmns. Knowledge and Data Engineering, 5:29-40.
[7] OMICS(2015) Big Data Analysis and Data Mining International Conference on Big Data Analysis,at Lexington, USA.
[8] Irina T. (2008) Association Rule Mining as a Data Mining Technique, Journal Seria Matematică - Informatică – Fizică Vol. LX No. 1 49 - 56
[9] Wasilewska, A.(2007) APRIORI Algorithm, Lecture Notes, http://www.cs.sunysb.edu/~cse634/lecture_notes/07apriori.pdf, accessed 2015
[10] Kalpesh A., Aditya G., Amiraj D., Rohit J. and Vipul H.(2013) Predicting Students' Performance Using ID3 and C4.5 Classification Algorithms, International Journal Of Data Mining and Knowledge Management Process (IJDKP) Vol.3, 5.
[11] Quinlan J. R.(1996), Improved use of continuous attributes in C4.5, Journal of Artificial Intelligence Research, Vol. 4, 77-90.