Contextual Fuzzy Type-2 Hierarchies for Decision Trees (CoFuH-DT) – An Accelerated Data Mining Technique

Kevin McCarty, *Member IEEE*, and Milos Manic, *Senior Member IEEE* [1]
University of Idaho, Idaho, USA
thekev@cableone.net, misko@uidaho.edu

***Abstract*— Advanced data mining techniques (ADMT) are very powerful tools for classification, understanding and prediction of object behaviors, providing descriptive relationships between objects such as a customer and a product they intend to buy. ADMT typically consists of classifiers and association techniques, among them, Decision Trees (DT). However, some important relationships are not readily apparent in a traditional decision tree. In addition, decision trees can grow quite large as the number of dimensions and their corresponding elements increase, requiring significant resources for processing. In either case, rules governing these relationships can be difficult to construct. This paper presents CoFuH-DT, a new algorithm for capturing intrinsic relationships among the nodes of DT, based upon a proposed concept of type-2 fuzzy "contexts". This algorithm modifies a decision tree, first by generating type-1 fuzzy extensions of the underlying DT criteria or "conditions"; combining further those extensions into new abstractions overlaid with type-2 contexts. The resulting fuzzy type-2 classification is then able to capture intrinsic relationships that are otherwise non-intuitive. In addition, performing fuzzy set-based operations simplifies the decision tree much faster than traditional search techniques in order to aid in rule construction. Testing presented on a virtual store example demonstrates savings of multiple orders of magnitude in terms of nodes and applicable conditions resulting in 1) reduced complexity of decision tree, 2) ability to data mine difficult to detect interrelationships, 3) substantial acceleration of decision tree search, making it applicable for 4) real-time data mining of new knowledge.**

I. INTRODUCTION

Organizations make extensive use of data mining techniques in order to define meaningful and predictable relationships between objects [1]. Retailers use these techniques to create recommender systems that seek to bring products and customers together [2]-[4]. Game designers attempt to create worthwhile and realistic adversaries. Zoologists want to create environments in which animals can thrive. One of the most widely employed methods for data mining is the decision tree. The decision tree is created using such methods as ID3 [5]-[7].

Typically a decision tree is viewed as a set of conditions and probabilities that, when combined, represent a node. Examining the tree usually means traversing it in, for example, a depth-first or breadth-first search, looking for nodes to prune, if possible, in order to optimize the search. Instead, consider the tree as a set of elements and filters or "conditions". Each node represents a subset of its parent created by applying one or more conditions to the parent set. The sequence of conditions represents the "path" to a given node.

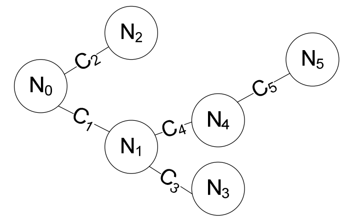

Fig. 1. A basic, horizontal decision tree

Hence, In Fig. 1, the decision tree node, N_1, represents a sample of data for which the condition C_1 applies for N_0: $N_1 = C_1(N_0)$. N_3 then becomes the condition C_3 applied to its parent node, N_1: $N_3 = C_3(N_1) = C_3(C_1(N_0))$.

Generically, any given node N_j is the resulting set derived when applying its "path" condition C_{Nj} to its parent .

$$N_j = C_k(N_{jParent}) \qquad j=1,..,j_{max},\ k=1,..,k_{max} \qquad (1)$$

Where j_{max} is the number of nodes and k_{max} is the number of conditions. For any given node one can determine the conditions, or "path" which lead to it and derive rules to apply this node "knowledge". The structure of a rule for a particular node consists of defining some event (such as a purchase, or the appearance of a threat) combined with the set of node conditions for which some action is appropriate [8]-[10].

DEFINE RULE *rule_name* (2)
ON *event*
IF *condition*
DO *action*

The conditions describe the relationships between node elements whether obvious, such as customers in a store, or more obscure, such as peanut butter and a bottle of cleaner, attempting to draw a meaningful relationship between them; for example:

IF Customer BUYS Computer THEN (3)
Customer BUYS Printer 25%

This condition tells a store manager that a customer that buys a computer is quite likely to want a printer as well and 25% of the time is going to buy one. The manager may choose to act upon this information by bundling printers and computer together in a special to encourage more printer purchases. With a decision tree, he can view the probabilities for any given set of conditions and try to create actions which will improve sales.

The CoFuH algorithm extends traditional fuzzy-1 sets through the use of fuzzy-2 hierarchies called "contexts". In doing so, it both simplifies the underlying data set as well as makes it more semantically precise under the higher-level, polymorphic, implication of its context. This is accomplished using fast, fuzzy-set based operators and rules that remove uninteresting data points that are "out of context" while enhancing what remains. The end result is a smaller, yet more precise and meaningful data set. This paper demonstrates the application of this technique to the decision tree, taking a large tree, fuzzifying it and applying contexts so that the resulting tree is smaller by orders of magnitude and more meaningful. The Contextual Fuzzy Hierarchies algorithm for the Decision Tree (CoFuH-DT) is then used to quickly prune some sample decision trees and create a meaningful relationship between two very different objects – in the example case a jar of peanut butter and a bottle of window cleaner.

This paper is organized as follows: Section II presents background, describing the previous work and issues. Section III presents the problem in detail. Section IV presents the algorithm. Section V applies CoFuH-DT to the example decision tree of the woman buying food items and cleaning supplies and a factory manager trying to decide which production lines to use. Section VI presents conclusions and future work.

II. PROBLEM STATEMENT

For data with many characteristics or non-intuitive ones, it can be difficult to build a manageable and meaningful tree because of the following:

1) For the manager of an online store, for example, understanding the relationships between and among thousands of customers, each with their own tastes and preferences, and products, means having to analyze a decision tree with potentially millions of nodes. Simply creating and managing rules for such a large number of nodes requires substantial computer resources. OnLine Analytical Processing (OLAP) systems [2] help manage huge datasets but do little to address other issues, such as:

2) Semantic differences between experts may lead to disagreements in rule definition [11], [12]. For example, what differentiates a "Good" customer from any other? Is a "bargain shopper" someone who *always* buys items that are on sale or someone who *only* buys items that are on sale?

3) Some relationships between products change within a given context, e.g. Turkey and cranberry sauce are closely associated in the United States during the Thanksgiving holiday but otherwise not closely related.

4) Some relationships among objects may vary over time. For example, in the summer, a sleeping bag might be associated with a swimsuit, bug spray and a fishing pole; while in the winter that same sleeping bag may be more closely associated with a parka, snow shoes and gloves.

5) Decision Trees can be difficult to interpret. Many paths are of no use at all, for instance a node that says ALL BABIES ARE BORN TO PREGNANT WOMAN does not provide much useful information. Other paths may be too obscure to define readily. An example is that of a woman buying certain food items and cleaning supplies. In her mind, these items are closely related in the context of "monthly shopping". The decision tree may reflect this; however, to a retailer such an association may not be so obvious, looking more like an outlier.

In a real world situation involving many products and customers with differing tastes, the number of nodes in a decision tree with n dimensions is determined by the cross product of the number of elements e of each dimension d_i used to branch:

$$\text{Total number of nodes in decision tree} = \prod_{i=1}^{n} d_i \qquad (4)$$

The store manager is probably going to be faced with very large decision tree. Now suppose there is a node on the tree containing the woman's purchase of food and cleaning supplies. The system produces a rule to address the case of the peanut butter to window cleaner relationship:

```
DEFINE RULE PB_Cleaner                                  (5)
ON Customer PURCHASE
IF PURCHASE is PeanutButter
        DO Recommend Window Cleaner
```

This rule does little to describe to the manager the overall context of the purchase and how best to take advantage of this information because there is no natural or obvious relationship between the objects to assess. Simply adding these rules to an already existing rule set means having to manage a substantially larger number of rules and relationships as well as having a more difficult time trying to deriving meaning.

Fuzzy-1 decision trees were created in an attempt to address some of these issues [13], [14] but run into difficulty dealing in areas where even the semantics themselves are called into question [11]. In fuzzy-1 form, decision trees simplify sets of nodes but do little to address the overall complexity of the tree itself.

Hybrid approaches [1], behavioral abstractions [3], [15], Online Analytical Mining (OLAM) [2], [8], [9] and multi-level association rules [7], [16] have also been devised to deal with these issues. While successful, these approaches consume significant computing resources and can end up creating numerous, multi-layer and often difficult to understand conditions. On the above rule with a multi-level association, PB _Cleaner with a monthly shopping hierarchy might end up looking like the following:

```
DEFINE RULE PB_Cleaner                                  (6)
ON Customer PURCHASE
IF PURCHASE is Peanut Butter
AND SHOPPING_TYPE IS MONTHLY
AND DAY IS First Saturday of Month THEN
        DO Recommend Window Cleaner
OR
IF PURCHASE is Window Cleaner
AND SHOPPING_TYPE IS MONTHLY
AND DAY IS First Saturday of Month THEN
        DO Recommend Peanut Butter
```

Does that now mean the manager has to stock bread and window cleaner together? When does a customer shop monthly? Adding more conditions to make sense of the relationship means increasing the number of dimensions, dimension elements, and corresponding nodes, as well as the difficulty in coming up with a semantically viable and accurate description of the relationship. Exponential growth occurs without a corresponding growth in usefulness.

Suppose the virtual store manager wishes to give his customers the best shopping experience possible. He has lots of statistics about purchases already made and uses decision trees to breakdown the types of purchases his customers make. There are lots of things he must take into account such as how often they shop, what sort of things they buy when they come in, what sorts of other products might they be interested in and so on. His initial decision tree might consist of the following, Customer Type (CT), Product Type (PT), Relative Product Price (RPP), Day of Week (DW), Time of Year (TY), Customer Age (CA), Geographic Location (GL)

TABLE I. DIMENSIONS FOR A VIRTUAL STORE MANAGER

Dimension	Sample Values
CT	normal, bargain, premium, bulk, impulsive
PT	food, cleaning, household…
RPP	bargain, normal, sale, premium
DW	Sunday, Monday, … Saturday
TY	Jan 1, Jan 2…Dec 31
CA	1, 2, 3, …100
GL	address, city, postal code

Even for a small average number of elements (say 10) per dimension the total number of nodes of this configuration would run into the millions. In addition, a large percentage of these nodes (such as day of year) contain very little useful information much of the time, but not all of the time. Even removing those uninteresting nodes still leaves a very large tree with a correspondingly large number of rules to manage. By fuzzifying the tree and overlaying strategic contexts according to the algorithm persented, the manager will greatly reduce the complexity of his rules to a more manageable state.

IV. CoFuH-DT – CONTEXTUAL FUZZY TYPE-2 HEIRARCHIES FOR DECISION TREES ALGORITHM

The CoFuH-DT algorithm presented in this paper consists of the following two phases: 1) deconstruction of a decision tree into datasets and filters, then 2) fuzzification of both datasets and filters resulting in a series of fuzzy sets. Fuzzy type-2 membership functions, representing one or more newly introduced "contexts" are applied to the sets, separating via fuzzy arithmetic those elements that are in context from those out of context. From the remaining fuzzy sets a smaller, in context decision tree is constructed.

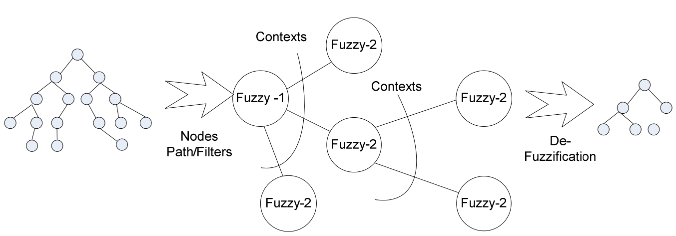

Fig. 2. CoFuH-DT reduction of decision tree

The steps of the CoFuH-DT algorithm is as follows:

1. Condition creation

Let $N_1..N_n$ be the set of nodes generated through data mining techniques such as ID3 [15], creating a decision tree for the original data set D.

$$N = \{N_1, N_2, \ldots, N_n\} \qquad (7)$$

Now let R1..Rn be the set of rules generated by applying individual paths to each node to its data.

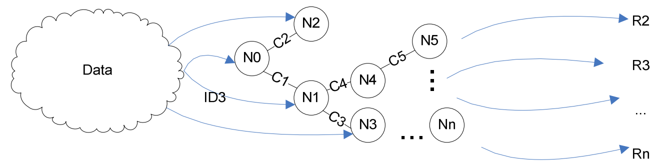

Fig. 3. Rule Creation using decision tree

2. Condition Normalization

Create a function f to normalize a set of conditions and corresponding rules C_R by mapping each Ci to the range [0,1] translating those values to a normalized set C_{norm}:

$$C_{norm} = \{f(C_i), C_i \in C_R, f(C_i) \in [0,1]\} \qquad (8)$$

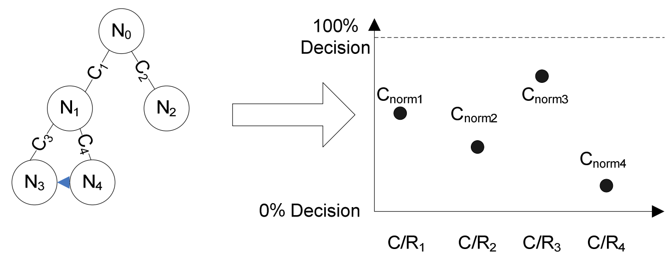

Fig. 4. Normalization of a decision tree

3. Condition fuzzification

Fuzzification of the normalized values occurs by extending those values using fuzzy type-1 membership functions and fuzzy hedges (ensuring appropriate representation, if necessary, across the entire set) to generate the fuzzy type-1 set μ_{Cnorm}.

Discrete points (e.g. a decision whether to recommend purchases of certain foods such as bread, ham, etc.) now become a series of fuzzy triangles as demonstrated in Fig. 5 with the original crisp conditions represented as a series of ranges at the base of each triangle.

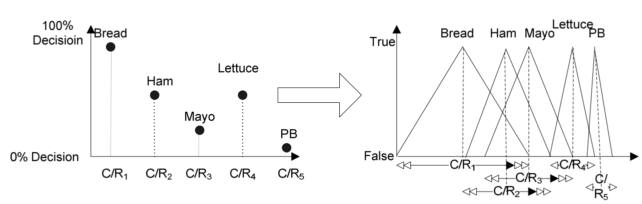

Fig. 5. Fuzzifying customer's decision tree

In cases where there are multiple Boolean conditions for a node we can apply Zadeh's operators AND and OR for fuzzy unions and intersections (for conditions $C_1 \ldots C_n$)

$$\begin{aligned} \cup\, \mu_{C_i} &= \min\,(\mu_{C_1}, \mu_{C_2}, .., \mu_{C_n}) \\ \cup\, \mu_{C_i} &= \max(\mu_{C_1}, \mu_{C_2}, .., \mu_{C_n}) \end{aligned} \tag{9}$$

More extreme examples can make use of mean and weighted mean or other general algebraic operators [17].

4. Context creation

Create fuzzy sets describing "contexts" which group items which may or may not have a natural association but which do relate within a given broader context. Contexts can also bring together elements of different clusters while at the same time preserving cluster identity.

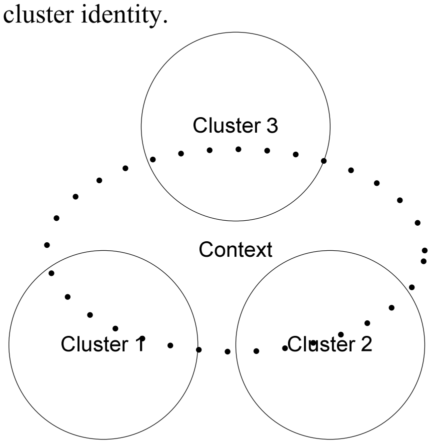

Fig. 6. Context unifying 3 clusters

For the decision tree, this has the effect of "pruning" all those nodes which fall out of context (Fig 7.).

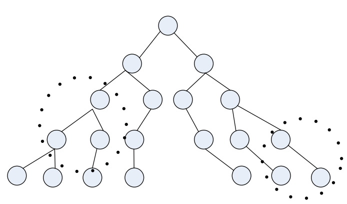

Fig. 7. Nodes pruned by context

Using fuzzy, new dimensions of uncertainty are added, allowing new specifications to exist and altering existing ones. In the example of the woman doing her monthly shopping, the context and new dimension of uncertainty "monthly shopping" alters the notion of both "food" and "cleaning supplies" by increasing membership in "food" for those items which are bought only occasionally while reducing it for others. At the same time, the context draws a link between food and cleaning supplies imposing a hierarchy of "monthly shopping" on top of both. Hence the resulting fuzzy-2 set, "monthly shopping" produces a new set consisting of "monthly food" and "monthly cleaning supplies" whose original primary sets locally are still regarded as "food" and "cleaning supplies". The membership of any item in any base set, e.g. "food" now assumes a more polymorphic representation dependent upon the one or more contexts in which it happens to find itself.

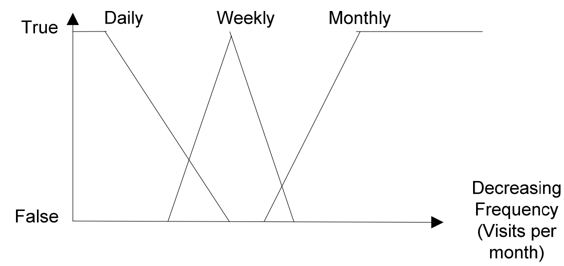

Fig. 8. Context of shopping type

The context contains the values "daily", "weekly" and "monthly".

Adding additional dimensions is a matter of creating and applying other contexts. For example, suppose the manager wanted to take into account various holiday periods. Now new contexts such as "Thanksgiving" or "St. Patrick's Day" get overlaid onto the decision tree to create a potentially different representation for the nodes underneath.

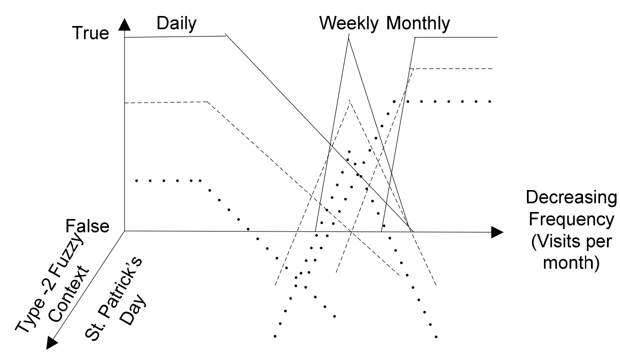

Fig. 9. Fuzzy deformation under a context

5. Fuzzy type-2 application of contexts to fuzzified conditions

Fuzzy type-2 contexts extend the newly created fuzzy type-1 set by adding an additional dimension of uncertainty. The context creates a fuzzy-2 set [11] $\hat{C}$, whose members are the combination of the context functions over the original fuzzy-1 membership functions over the original conditions shown in Eq 8. Applying the Zadeh product operator across the domain of $\hat{C}$ eliminates those sets and the underlying conditions which are "out of context". Setting appropriate minimum memberships thresholds can serve to further reduce the final result space R_C:

$$R_C = \cap \hat{C} \tag{10}$$

This has the desired effect of pruning those nodes completely out of context as well as marginalizing those elements which are only of minimal interest.

For the retailer with the customer doing monthly shopping, de-fuzzification of the remaining conditions yields a much smaller decision tree. In addition, by using the context applied over the remaining conditions, they take on new meaning within that context. The rule developed previously in (6) can now be generalized to:

```
DEFINE RULE ShoppingType                    (11)
ON Customer PURCHASE
IF PURCHASE IS MonthlyContextItem THEN
      DO Recommend Other MonthlyContextItems
```

This new rule is both simpler to implement as well as more descriptive and intuitive. It also takes into account the contextual components of the shopping trip (such as the regular monthly shopping day). In the case of the peanut butter and window cleaner, while distinct and very different types initially, they are united under the context of "monthly shopping".

IV. TEST EXAMPLES

These test examples were used to demonstrate the effectiveness of the algorithm when applied to real world situations. Developing appropriate contexts and then applying them to the underlying dimension elements results in a significant decrease in the number of "in context" elements as well as the resulting decision tree.

Example 1. Trivial Case

In the trivial case where the context has no affect on the underlying fuzzy conditions, for example "monthly shopping" on a list of only monthly shopping items, no deformation occurs and any set operations and the set of rules reduces to that described in (2).

Example 2. Woman in store

Suppose the woman customer comes into the virtual store to buy some groceries. The decision tree with each dimension containing the number of elements listed in Table 2. The traditional decision tree would consist of 1.4 million potential nodes, depending upon the available data. Pruning the tree using standard methods requires traversing a large number of nodes, investigating each for applicability. However, creating a context of "Monthly Shopping" (MS) and applying the fuzzification processes a number of things occur:

1) The "impulsive" customer type (CT) falls out of context as MS is considered planned reducing the size of CT from 5 to 4.

2) Many of the product types (PT) that are considered impulse (e.g. books, candy) or quickly perishable items (e.g. bread, lettuce) or irregular purchases (e.g nails), daily purchases, weekly purchases and holiday items fall out of context reducing the size of the PT from 10 to 4.

3) Relative Product Price is unaffected by MS.

4) Since MS occurs on the weekend, Day of Week (DW) values Monday through Friday fall out of context reducing the DW dimension from 7 to 2.

5) Time of Year (TY) is unaffected

6) Customer Age (CA), the context MS usually involves heads of household which eliminates certain age categories such as "Under 10", "Young Adult 10-20", bringing the CA category from 10 to 8.

7) Geographic Location (GL) is unaffected.

Even more dramatic would be a context such as "Holiday - St. Patrick's Day". As the types of products shoppers celebrating St. Patrick's Day require is very small and the type of individual celebrating the holiday is likewise limited, the resulting decision tree is reduced considerably. The final node totals of customer decision trees for "Monthly Shopping" and "Holiday – St. Patrick's Day" are shown in Table 2.

	Traditional Decisioin Tree	Context Monthly Shopping	Context St. Patrick's Day
Dimensions In Context	7	7	7
Elements In Context	51	42	24
Potential Nodes	1.4×10^6	9.6×10^4	1024

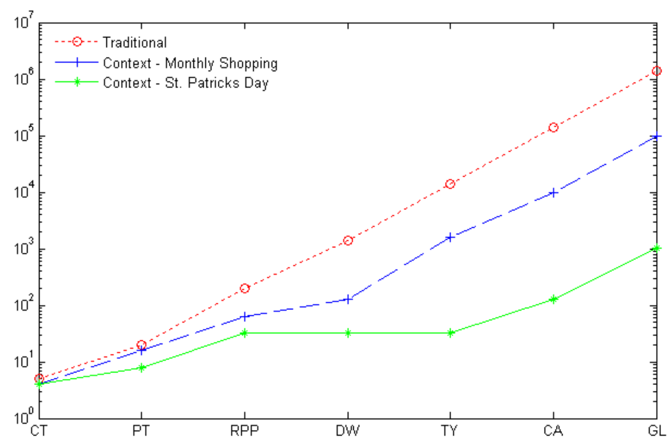

Fig 10. Node growth under normal conditions and contexts

Example 3. Plant manager

The manager of a plant uses a decision tree to decide how to set up the production line, taking into account inventory, backlog, capacity, other dimensions (limited for simplicity to 10 elements per dimension). Creating holiday contexts allows the manager to tailor production to meet the changing demands as holidays come and go. Other contexts such as "Preferred customer" or "Holiday schedule" quickly reduce the number of possibilities to a small number of in-context production options for example, the "Preferred customer" context eliminates all low priority, non-customer components, while "Holiday schedule" eliminates those components not purchased or shipped during the holiday.

TABLE III. EXAMPLE 3 NODE REDUCTION UNDER CONTEXTS

	Traditional Decsn. Tree	Context Pref. Customer	Context Holiday Sched.
Dimensions In Context	7	7	7
Elements In Context	70	27	33
Potential Nodes	1×10^7	4400	3.6×10^4

V. CONCLUSION

As shown in examples 2 and 3, use of contexts reduces significantly the number of in context dimension elements. In example 2 the original number dropped from 51 to 42 to 24 for "St. Patrick's Day". The reductions are even more dramatic when applied to the number of potential nodes of the decision tree (dropping from 1.4×10^6 down to 1024) affecting a reduction of approximately 3 orders of magnitude.

Whether an e-commerce retailer, behavioral scientist, the manager of a production or of a sports team;,each can rely upon decision trees to formulate rules for actions. However, outliers and large combinations of conditions can create difficult and confusing sets of rules that have limited applicability. Current solutions attempt to alleviate this problem through clever techniques or sheer brute force to derive meaning but have difficulty if relationships are numerous or non-intuitive.

The fuzzy methods demonstrated in this paper improve upon these techniques by introducing new dimensions of uncertainty serving to both reduce the number and complexity of rules as well as tie non-intuitive relationships together within a larger meaningful context. The examples demonstrate many orders of magnitude improvement of subsequent decision tree construction over traditional methods.

Future work involving the use of artificial neural networks (ANN) needs to be done. ANNs are well-suited for learning and adaptive control and can be used to automatically generate meaningful higher-level contexts to serve as the basis for new rule creation. In addition generalizing the CoFuH algorithm to provide a more complete and generic framework for other advanced data mining techniques is a goal.

ACKNOWLEDGMENT

The authors would like to thank the National Science Foundation Idaho EPSCoR Research Infrastructure Improvement (RII) Grant, Grand Challenge Initiative, 2005-2008 for their support.

REFERENCES

[1] D. Liu, C. Lai, W. Lee; *A Hybrid of Sequential Rules and Collaborative Filtering for Product Recommendation*; 9th IEEE International Conference on E-Commerce Technology and the 4th IEEE International Conference on Enterprise Computing, E-Commerce, and E-Services, July 2007

[2] C. Qiming; U. Dayal, M. Hsu; *A distributed OLAP infrastructure for e-commerce*; IFCIS International Conference on Cooperative Information Systems, 1999

[3] M. Kitayama, R. Matsubara, Y. Izui; *Application of data mining to customer profile analysis in the power electric industry*; IEEE Power Engineering Society Winter Meeting, 2002. IEEE, Jan. 2002

[4] K. Fu, D. Geng; *Using the Data Mining Approach to Determine the Product Preferences of Target Customers*; 2007 International Conference on Service Systems and Service Management, June 2007

[5] J. Zhao, Z. Chang, *Neuro-Fuzzy Decision Tree by Fuzzy ID3 Algorithm and Its Application to Anti-Dumping Early-Warning System*, International Conference on Information Acquisition, April 2006

[6] F. Li, J. Sun, X. Wang, *Analysis on the Fuzzy Filter in Fuzzy Decision Trees*, 2nd International Conference on Machine Learning and Cybernetics, Nov. 2003

[7] N. Li, Y. Xiao-Wei, Z. Cheng-Qi, Z. Shi-Chao; *Product hierarchy-based customer profiles for electronic commerce recommendation*; International Conference on Machine Learning and Cybernetics, Nov. 2002

[8] Adomavicius, G.; Tuzhilin, A.; *Using data mining methods to build customer profiles*; Computer, Volume 34, Issue 2, Feb 2001

[9] I.S.Y. Kwan; *A mental cognitive model of Web semantic for e-customer profile*; 13th International Workshop on Database and Expert Systems Applications, Sept. 2002

[10] M. Dai, Y. Huang, *Data Mining Used in Rule Design for Active Database Systems*; 4th International Conference on Fuzzy Systems and Knowledge Discovery, June 2007

[11] J. M. Mendel, R. I. Bob John; *Type-2 Fuzzy Sets Made Simple*, IEEE Transactions on Fuzzy Systems, April 2002

[12] M. Hanss, M.; *Applied Fuzzy Arithmetic*, New York: Springer-Verlag, 2005, pp 58-85

[13] J. Lee, J. Sun, L. Yang, *A Fuzzy Matching Method of Fuzzy Decision Trees*; 2nd International Conference on Machine Learning and Cybernetics, Nov. 2003

[14] X-Z. Wang, D.S. Yeung, E.C.C. Tsang, *A Comparitive Study on Heuristic Algorithms for Generating Fuzzy Decision Trees*; Transactions on Systems, Man and Cybernetics, April 2001

[15] C. Haruechaiyasak, C. Tipnoe, S. Kongyoung, C. Damrongrat, N. Angkawattanawit; *A dynamic framework for maintaining customer profiles in e-commerce recommender systems*; The 2005 IEEE International Conference on e-Technology, e-Commerce and e-Service, 2005. EEE '05. Proceedings; April 2005

[16] D. van de Vlag, A. Stein, *Incorporating Uncertainty via Hierarchical Classification Using Fuzzy Decision Trees*; Transactions on Geoscience and Remote Sensing, Jan. 2007

[17] E. Cox, *Fuzzy Modeling and Genetic Algorithms for Data Mining and Exploration*, Morgan Kaufmann, January 2005, pp 76-94

[18] S. Robak, A. Pieczynski; *Employing fuzzy logic in feature diagrams to model variability in software product-lines*; 10th IEEE International Conference and Workshop on the Engineering of Computer-Based Systems, April 2003

[19] W. Ke, Z. Senqiang, J.M.S. Yeung, Y. Qiang; *Mining customer value: from association rules to direct marketing*; 19th International Conference on Data Engineering, March 2003

[20] G. Yutao, J.P. Muller; *A personalized product recommendation algorithm based on preference and intention learning*; Seventh IEEE International Conference on E-Commerce Technology, July 2005

[21] K. McCarty, M. Manic; *Line-of-Sight Tracking Based Upon Modern Heuristics Approach*; International Conference on Industrial Electronics and Applications, June 2008

[22] W. Sung-Shun, L. Mei-Ju; *Personalized product recommendation in e-commerce*; 2004 IEEE International Conference on e-Technology, e-Commerce and e-Service, March 2004

[23] K. Lee, K. Lee, J. Lee, *A Fuzzy Decision Tree Induction Method for Fuzzy Data; International Fuzzy Systems Conference Proceedings*, August 1999