World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:1, No:12, 2007

A Cumulative Learning Approach to Data Mining Employing Censored Production Rules (CPRs)

Rekha Kandwal, and Kamal K.Bharadwaj

Abstract*—**Knowledge is indispensable but voluminous knowledge becomes a bottleneck for efficient processing. A great challenge for data mining activity is the generation of large number of potential rules as a result of mining process. In fact sometimes result size is comparable to the original data. Traditional data mining pruning activities such as support do not sufficiently reduce the huge rule space. Moreover, many practical applications are characterized by continual change of data and knowledge, thereby making knowledge voluminous with each change. The most predominant representation of the discovered knowledge is the standard Production Rules (PRs) in the form **If P Then D**. Michalski & Winston proposed Censored Production Rules (CPRs), as an extension of production rules, that exhibit variable precision and supports an efficient mechanism for handling exceptions. A CPR is an augmented production rule of the form: **If P Then D Unless C,** where *C* (Censor) is an exception to the rule. Such rules are employed in situations in which the conditional statement 'If P Then D'*** holds frequently and the assertion *C* holds rarely. By using a rule of this type we are free to ignore the exception conditions, when the resources needed to establish its presence, are tight or there is simply no information available as to whether it holds or not. Thus the '***If P Then D'*** part of the CPR expresses important information while the *Unless C* part acts only as a switch changes the polarity of ***D*** to ***~D***. In this paper a scheme based on Dempster-Shafer Theory (DST) interpretation of a CPR is suggested for discovering CPRs from the discovered flat PRs. The discovery of CPRs from flat rules would result in considerable reduction of the already discovered rules. The proposed scheme incrementally incorporates new knowledge and also reduces the size of knowledge base considerably with each episode. Examples are given to demonstrate the behaviour of the proposed scheme. The suggested cumulative learning scheme would be useful in mining data streams.

***Keywords*—**Censored production rules, cumulative learning, data mining, machine learning.

I. INTRODUCTION

KNOWLEDGE is the information that represents long-term relationships, that is, ways of doing things, commonsense, ideas, methods, skills, and so forth. Knowledge is "condensed" information,"squashed" information, an extraction, the "essence" of things. A huge amount of knowledge has been accumulated worldwide during the conscious existence of humanity. This knowledge is changing all the time. There are many issues related to knowledge viz How to make use of existing knowledge in a computer program? How to represent it in such a way that we keep the richness and the depth of knowledge and also make it reasonable to use? How to make compromises with constantly changing knowledge?

Whatever method for representing problem knowledge is used in a computer system, it is usually a way of simplifying the problem to make it reasonable to handle. One of the widely used representations in knowledge Engineering is the Production System. Standard Production rules, of the type IF P THEN D, are not efficient for approximate reasoning and unable to exhibit variable precision in the reasoning process due to rigidity in their structure. Also these rules fragment the knowledge that exists in the data resulting in a large number of discovered rules that makes analysis of the rules very difficult.

As an extension of standard production rule, Michalski and Winston [7] have suggested the **Censored Production Rule (CPR)** as an underlying representational and computational mechanism to enable logic based systems to exhibit variable precision in which certainty varies while specificity stays constant. A CPR has the form, **If P Then D Unless C**, where **C** (Censor) is the exception condition. Such rules are employed in situations in which the conditional statement '**If P Then D**' holds frequently and the assertion **C** holds rarely. By using a rule of this type we are free to ignore the censor (exception) conditions, when the resources needed to establish its presence are tight or there is simply no information available as to whether it holds or does not hold. As time permits the Censor condition **C** is evaluated establishing the conclusion **D** with higher certainty if **C** does not hold or simply changing the polarity of **D** to **~D** if **C** holds. For example, the rule **If** Sunday **Then** John works in the yard **Unless** weather is bad, has the interpretation that if it is Sunday and the weather is good, John will work in the yard; and if it is Sunday and the weather is bad (which occurs rarely), John will not work in the yard.

CONALD (Conference on Automated Learning and Discovery), hosted by CMU came up with a no. of promising research directions for future and posed a trivial problem: Can we devise cumulative learning algorithms that can incrementally incorporate new data & knowledge? [10] In this paper an attempt is made to exploit the inherent

Rekha kandwal is a Ph.D scholar at School of Computer and System Sciences (SC&SS), Jawaharlal Nehru University (JNU), New Delhi-110067, India (e-mail: phone: 91-11-55157029 email: rekha.kandwal@gmail.com).

Kamal K.Bharadwaj is a Professor at SC&SS, JNU, New Delhi, India (phone: 91-9810196636, e-mail: kbharadwaj@gmail.com)

structural properties of CPRs to accommodate cumulative learning scenario. A dynamic system which comprehends the knowledge with each episode is developed. Results on the behaviour of the proposed scheme are also included.

II. Background

The major shortcomings of an ordinary logic based reasoning system based on standard production rule ***If P Then D***, is that you cannot tell much about the way you want it to perform its task. For example, you can not give the following instructions [7]:

- Give me a reasonable answer immediately even if it is somewhat general and if there is enough time then give me a more specific answer.
- Give me a reasonable answer immediately. If there is enough time tell me you are more confident in the answer or change your mind and give me a better answer.
- Give me only a highly certain answer even if it is somewhat general, and if there is enough time then give me a more specific answer.
- Give me a reasonable answer immediately even if it is somewhat less certain and if you have enough time then give me a more specific answer.

A system having represented real-world knowledge should also be capable of handling these types of requirements for natural and efficient reasoning. In the real world, both humans and computers often have to reason using insufficient, incomplete, or tentative premises. Moreover, both are subject to constraints of time and memory. Variable precision logic (VPL) [7] is concerned with problems of reasoning with incomplete information, subject to resource constraints and the problem of reasoning efficiently with exceptions. VPL offers mechanisms for handling trade-off between the precision of inferences and computational efficiency of driving them. Specificity and certainty are two aspects of precision. Certainty refers to the degree of belief in a statement, whereas specificity refers to the degree of detail of a description. A system that gives more specific answers given more time (or resources in general) is called a "variable specificity system". A system that gives more certain answers given more time is called a "variable certainty system". There can be various combinations of the two systems, reflecting that specificity and certainty are inversely related; we can gain specificity at the expense of certainty, or vice-versa.

Let us now give a more quantitative definition of a censored production rule:

$$P \Rightarrow D \llcorner C. \tag{1}$$

Where P is a premise, D is a decision, and C is a censor. Although the unless operator, $\llcorner$, is logically equivalent to the commutative exclusive-or operator, the unless operator has an expositive aspects which is not commutative. In order to capture the asymmetry precisely, let us associate two parameters, γ_1 and γ_2, with rule (1)

$$P \Rightarrow D \llcorner C: \gamma_1, \gamma_2. \tag{2}$$

Both γ_1 and γ_2 are point probabilities, one indicating the strength of the relationship between P and D, and the other, between P and C. Now consider the following sets: Ω is a finite sample of events; Ω_P is the set of events for which P holds; Ω_{PD} is the subset of events for which both P and D hold; Ω_{PC} is a subset of events for which both P and C hold. Given these sets, the parameters γ_1 and γ_2 are defined as follows:

$$\gamma_1 = \frac{\mathbf{Pr[P, D]}}{\mathbf{Pr[P]}} = \mathbf{Pr\ [D|P]} \approx \frac{|\mathbf{\Omega}_{\mathbf{PD}}|}{|\mathbf{\Omega}_{\mathbf{P}}|},$$

$$\gamma_2 = \frac{\mathbf{Pr[P, C]}}{\mathbf{Pr[P]}} = \mathbf{Pr\ [C|P]} \approx \frac{|\mathbf{\Omega}_{\mathbf{PC}}|}{|\mathbf{\Omega}_{\mathbf{P}}|},$$

Where $|\Omega_i|$ denotes the cardinality of Ω_i. Also we assume that $\Omega_D \cap \Omega_C = \varnothing$ and
$\Omega_D \cup \Omega_C = \Omega_P$; thus $\Pr[P|D] + \Pr[C|P] = 1$.
A CPR exhibits variable precision in which certainty varies while specificity remains constant.

III. Cumulative Learning based on CPRs

CPR system helps us to collect fragmented knowledge and representing these as collective one significantly reducing the knowledge base. This representation scheme reduces the complexity of the discovered knowledge substantially, makes knowledge base easy to understand and efficient for future processing.

Under the Dempster-Shafer Interpretation of VPL, four belief values are associated with each Censored Production Rule [3, 8]:

$$P \rightarrow D \llcorner C: \alpha,\beta,\gamma,\delta \tag{3}$$

such that

(i) $P\wedge \sim C \rightarrow D: \alpha$
(ii) $P\wedge C \rightarrow \sim D: \beta$
(iii) $P \rightarrow D: \gamma$
(iv) $P \rightarrow \sim D: \delta$

That is a CPR (3) can be factorized into four PRs, (i) - (iv). This interpretation suggests a mechanism for discovering CPRs from the set of PRs. That is a CPR (3) can be discovered from a group of already discovered four PRs ((i)-(iv)).

For example, the following rule set generated as a result of mining on certain data:

(i) Weekday_morning ∧~Oversleep → Read_paper : 0.9

the '0.9' states that on weekday mornings when I do not oversleep I read the paper at least 0.9 of the time because there are other factors which would keep me from reading paper; such as the paper boy throwing it on the roof, which are not being considered;

(ii) Weekday_morning ∧Oversleep → ~Read_paper : 1

the '1' states that on weekday mornings when I oversleep I certainly do not read the paper.

(iii) Weekday_morning → Read_paper : 0.6

the '0.6' states that I read the paper at least three out of five weekday mornings as I oversleep at most twice a week.

(iv) Weekday_morning → ~Read_paper : 0.2

the '0.2' states that I do not read the paper at least one out of five weekday mornings because I oversleep at least once a week.

Combining all the above four flat rules, we can discover a single Censor Production rule which expresses all the above four rules:

Weekday_morning → Read_paper $\lfloor$ ~ Oversleep : 0.9, 1, 0.6, 0.2,

stating that I read the paper before going to work unless I oversleep, which occurs once or twice a week

This is the first step towards comprehension of knowledge where a set of four rules is converted into a single CPR thereby reducing the knowledge base considerably (depending on the nature of data the maximum reduction could be even one fourth). As machine learning techniques are increasingly being used to solve real life problems, post analysis of rules will become increasingly important. In dynamic domain where rules may change over time, it is important to know what the changes are. Here it is presumed that Flat production rules are made available by using a suitable data mining algorithm and these flat rules are reduced to CPRs as above. The proposed system would act on such rules to generate comprehensive knowledge with each episode. This activity handles change in knowledge incrementally without revisiting the original database.

Following Michalski and Winston [7], the following combination of rules can be used for considerable reduction of knowledge base:

$$\left.\begin{array}{l} P1 \rightarrow D \lfloor C \\ P2 \rightarrow D \lfloor \end{array}\right| > P1 \vee P2 \rightarrow D \lfloor C \qquad (4)$$

$$\left.\begin{array}{l} P \rightarrow D1 \lfloor C \\ P \rightarrow D2 \lfloor C \end{array}\right| > P \rightarrow D1 \wedge D2 \lfloor C \qquad (5)$$

$$\left.\begin{array}{l} P \rightarrow D \lfloor C1 \\ P \rightarrow D \lfloor C2 \end{array}\right| > P \rightarrow D \lfloor (C1 \vee C2) \qquad (6)$$

$$\left.\begin{array}{l} P \rightarrow D \lfloor C1 \\ D \rightarrow D1 \lfloor C2 \\ D \rightarrow D1 \end{array}\right| > P \rightarrow D1 \lfloor (C1 \vee C2) \qquad (7)$$

A. *Cumulative Learning Algorithm*

Input: Knowledge base KB1 and Knowledge base KB2.
/* For each rule, **'*IF*'** part is represented as **'Pset',** **'*UNLESS*'** part as **'**Cset**'** and **'*decision part*'** as 'Dset' */

Output: Knowledge base KB3

```
1. for (i=0;i<=no. of rules in KB1; i++)
     for (j=0;j<=no. of rules in KB2; j++)
      {If (Pset of R_i== Pset of R_j)
        {If (Dset of R_i==Dset of R_j)
          {If (Cset of R_i==Cset of R_j)
            { a. construct a new rule 'R_k' where
                  Pset ← Pset of R_i ;
                  Dset ← Dset of R_i ;
                  Cset ← Cset of R_i ;
              b. Mark R_i and R_j
              c. store R_k in KB3;k++;}
           else
            {a. construct a new rule 'R_k' where
                  Pset ← Pset of R_i ;
                  Dset ← Dset of R_i ;
                  Cset ← ( Cset of R_i V Cset of R_j) ;
              b. Mark R_i and R_j
              c. store R_k in KB3;k++;}
          }
          If (Cset of R_i==Cset of R_j)
            {  a. construct a new rule 'R_k' where
                  Pset ←  Pset of R_i ;
                  Dset ← Dset of R_i ∧ Dset of R_j ;
                  Cset ← Cset of R_i ;
               b. Mark R_i and R_j
               c. store R_k in KB3;k++;}
        }

      else{If (Dset of R_i==Dset of R_j)
            { a. construct a new rule 'R_k' where
                  Pset ← (Pset of R_i ∨ Pset of R_j);
                  Dset ← Dset of R_i;
                  Cset ←(Cset of R_i ∨ Cset of R_j);
              b. Mark R_i and R_j
              c. store R_k in KB3;k++;} }
}

2. for (all Unmarked R_i in KB1)
     {  a. construct a new rule 'R_k' where
           Pset ←  Pset of R_i ;
           Dset ← Dset of R_i ;
           Cset ← Cset of R_i ;
        b. Mark R_i
        c. store  R_k in KB3;k++; }

3. for (all Unmarked R_j in KB2)
     { a. construct a new rule 'R_k' where
           Pset ← Pset of R_j ;
           Dset ← Dset of R_j ;
           Cset ← Cset of R_j ;
       b. Mark R_j
       c. store  R_k in KB3;k++; }
```

By applying algorithm in each episode, the size of knowledge base reduces significantly giving us a good summary of the knowledge accommodating new knowledge suitably and removing all redundant information. For example, suppose KB1 and KB2 are knowledge base containing the set of rules collected over different point of time.

KB1:

1.obstacle_ahead∧speed_distance_ratio high → use_brakes

2.obstacle_ahead∧speed_distance_ratio high → use_brakes ⌊ tire_traction poor

3. on ice_road→ tire_traction poor ⌊ using_chain

KB2:

1.obstacle_ahead∧speed_distance_ratio high → use_brakes ⌊ brake_condition poor

2.on gravel_road → tire_traction poor

Applying our algorithm to these different knowledge bases, we can have a single knowledge base KB3 which contains the comprehensive but complete set of rules as:

KB3:

1.obstacle_ahead ∧ speed_distance_ratio high → use_brakes ⌊ tire_traction poor ∨ brake_condition poor

2. on ice_road ∨ on gravel_road → tire_traction poor ⌊ using_chain

IV. Proposed Scheme

In the proposed cumulative learning scheme transactions over a long duration are divided into sets of consecutive episodes. In every episode the knowledge base generated depends not only on the current set of knowledge but also efficiently accommodate knowledge gained during the previous episode. This new knowledge base will act as 'knowledge of previous episode' in future processing. Instead of making mining a finite, closed ended process which produces a well defined knowledge, the idea is to make it a continuous process thereby generating continually improved knowledge. The proposed scheme is summarized diagrammatically in Fig.1 and explained through an example in Fig. 2.

V. Conclusion

The problem of too many rules has been studied by many researchers in data mining. The main approaches used are: using some interestingness measures to filter out those uninteresting rules; using the user's domain knowledge to help him identify unexpected rules, using some user defined measure to prune the rule space. Limited research has been done on what happens after a set of rules has been induced.
In this paper, we have exploited the inherent properties of CPRs system to implement cumulating learning approach as post processing scheme. As a preprocessing of the proposed learning algorithm, possible CPRs are discovered from the already discovered set of flat PRs. The transformation of PRs into CPRs results in considerable reduction in the number of already discovered PRs. The proposed cumulative learning scheme incrementally incorporates new knowledge with each episode. Thereby generating continually, improved set of CPRs with minimum redundancy.

As data streams are gaining prominence in a growing number of emerging applications, advanced analysis and mining of data streams is becoming increasingly important. [14,15]. One of the most important applications of the proposed cumulative learning scheme would be in mining data streams. Development of cumulative learning schemes based on Hierarchical Censored production rules (HCPRs) system [2, 3, 4] is under progress

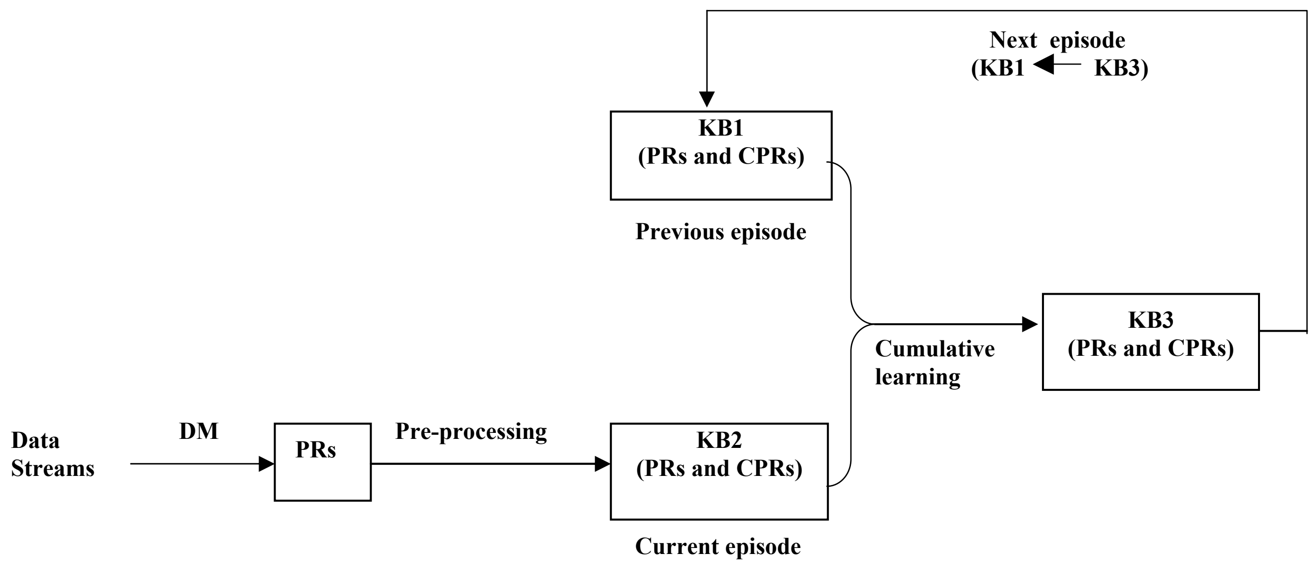

Fig. 1 Cumulative learning scheme

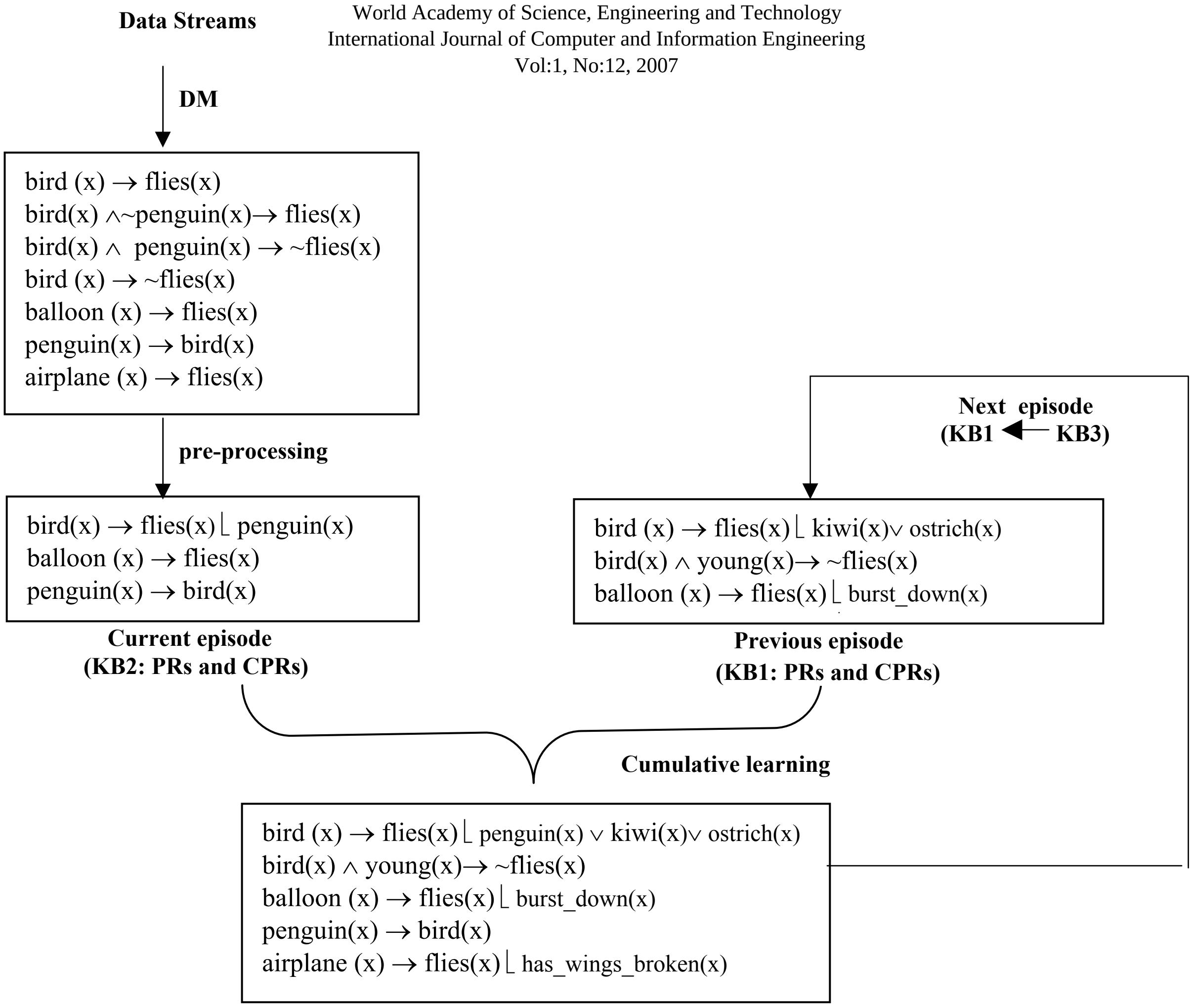

Fig. 2 Cumulative learning for flying objects

REFERENCES

[1] Han, J., Kamber, M. "*Data mining: Concepts and Techniques*" Academic Press (2001).

[2] Bharadwaj, K.K., Jain, N.K.: *Hierarchical Censored Production Rules (HCPRs) System*, Data and Knowledge Engineering, vol.8 (North Holland), 1992.

[3] Bharadwaj, K.K., Neerja, *Goel, G.C.: Hierarchical Censored Production Rules (HCPRs) Systems Employing the Dampster-Shafer Uncertainty Calculus*, Information and Software technology, Butterworth-Heinemann Ltd. (U.K.) Vol. 36 No., 155-164, 1994.

[4] Jain, N.K. ,Bharadwaj, K.K.,: *Some Learning Techniques in Hierarchical Censored Production Rules(HCPRs) System*, International Journal of intelligent systems, John Wiley & sons, Inc.,vol. 13,pp 319-344, 1997.

[5] Quinlan, J.R. (1986): *Induction of Decision tress*: Machine learning;1(1);81-106,1986.

[6] Adriaan, P., Zantingre, D. "*Data Mining*", Addison Wesley, 1999.

[7] Michalski, R.S., Winston, P.H., *Variable Precision Logic*, Artificial intelligence,29,121-146,1986.

[8] Jain,N.K., Bharadwaj K.K. and, Norian Marrengallo " *Extended Hierarchical Censored Production Rules System*", vol. 9, no 3-4, journal of Intelligence Systems, UK ,1999.

[9] Ananthanarayana, V.S., Murty, M.N., Subramanian, D.K.: *Dynamic Data Mining*, Proceedings of the International Conference, KBCS-2002.

[10] Sebastian Thrun, Christos Faloutsos, Tom Mitchell, Larry Wasserman: *Automated Learning and Discovery: State-Of-The Art and Research Topics in a Rapidly Growing Field*, CMU_CALD-98-100, September 1998.

[11] Ryszard S. Michalski, Pavel Brazdil: *Introduction, Special Issue on Multistrategy learning*, Machine Learning, vol 50, pp 219-222, 2003.

[12] Bing Liu , Minqing Hu and Wynne Hsu, "*Intuitive Representation of Decision Trees Using General Rules and Exceptions*" American Association for Artificial Intelligence,2000.

[13] Nikola K.Kasabov. "*Foundation of Neural Networks, Fuzzy systems, and Knowledge Engineering*" The MIT Press (2001).

[14] Brian Babcock, Shivnath Babu, Mayur data, Rajeev Motwani, and Jennifer Widom: *Models and Issues in data Stream Systems*, Proceeding of 21st ACM Symposium on Principles of Database Systems (PODS 2002).

[15] Guozhu Dong, Jiawei Han, laks V.S. Lakshmanan, Jian Pei, Haixun Wang, Philip S. Yu: *Online Mining of changes from data Streams: Research Problems and Preliminary Results*, In Proceedings of the 2003 ACM SIGMOID Workshop on Management and Processing of data Streams.