Implementation of Knowledge-Based Expert System Using Probabilistic Network Models

Mr. Mayank
Technical Specialist, Avekshaa Technologies Pvt. Ltd.
Mumbai, Maharashtra, India
mayank.rnath@gmail.com

Abstract—The latest development in machine learning techniques has enabled the development of intelligent tools which can identify anomalies in the system in real time. These intelligent tools become expert systems when they combine the algorithmic result of root cause analysis with the domain knowledge. Truth maintenance, fuzzy logic, ontology classification are just a few out of many techniques used in building these systems. Logic is embedded in the code in most of the traditional computer program, which makes it difficult for domain experts to retrieve the underlying rule set and make any changes. These system bridge the gap by making information explicit rather than implicit. In this paper, we present a new approach for developing an expert system using decision tree analysis with probabilistic network models such as Bayes-network. The proposed model facilitate the process of correlation between belief probability with the unseen data by use of logical flowcharting, loopy belief propagation algorithm, and decision trees analysis. The performance of the model will be measured by evaluation and cross validation techniques.

Keywords*-Probabilistic network models, Expert System, Logical flowcharting, loopy belief propagation algorithm*

*****

I. INTRODUCTION*(HEADING 1)*

Recent years have seen a revolution in both the content and the methodology of work in artificial intelligence. Traditionally, a general purpose search mechanism is set up to string together elementary reasoning steps to find a complete solution but they do not scale up to difficult and large problem instances. The alternative approach is to use more powerful, domain expertize that allows several reasoning steps and can easily handle recurring cases in a narrow area of that particular domain[1]. Expert system merges this approach with probability based network models like Bayes-network, which in turn combines the domain knowledge with belief propagation to overcome the availability and over dependence on data[2]. The current systems, in particular, operate on scientific properties i.e. locality, detachment, and truth-functionality. These properties are governed by the domain expert through knowledge acquisition module, knowledge base, explanation module and inference engine rule set before presenting to the end user as illustrated in Fig.1[3]. But the bad news is that the properties of detachment, truth functionality and locality fails to manage uncertain knowledge[1].

This paper deals with the elimination of such difficulties and proposing a model which induces a minor variation on the certainty-factor inference which is equivalent to bayesian inference. As this factor can yield a disastrously incorrect degree of belief through over counting of evidence[4], we eliminated the undesirable interactions between rule through decision tree analysis and used the bayesian network in one layer top it. The remainder of the paper is structured is as follows. Section 2 presents the proposed system in detail. Section 3 presents experimental results while section 4 concludes the paper.

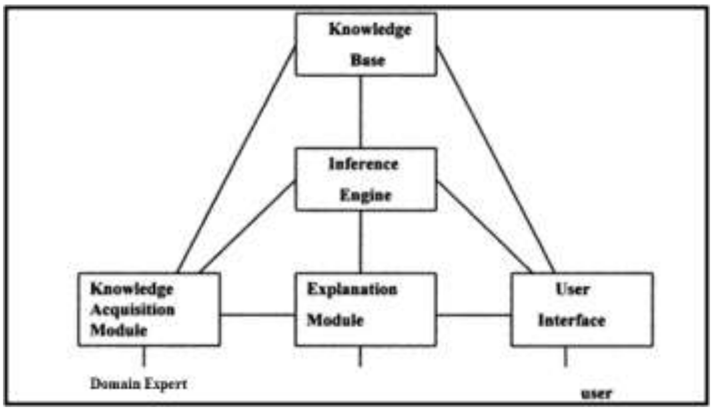

Fig. 1 Typical setup of knowledge based system

II. PROPOSED SYSTEM

Data patterns will form a Polytree as shown in Fig. 2 and can be modeled as an Acyclic Directed Graph in combination with decision tree analysis. There can be multiple events triggering the same uncertain pattern chains and hence the system can have multi-roots.

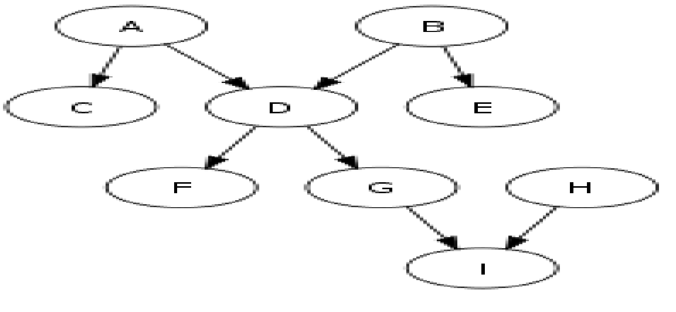

Fig. 2 Polytree representation of data for flow of events

ISSN: 2321-8169
65 – 68

Analyzing preceding patterns is important for establishing the causal relationships. 3 variables determine the relevance of a Pattern under analysis with its preceding pattern. The proposed system is illustrated in Fig. 3.

Variable 1 – Temporal proximity.
Score_1 = ∑(1/timeElapsedSinceLastAnomaly)
Inverse value ensures that higher the elapsed time lower will be the score.

Variable 2 – Probability of an event A occurring in the system with S samples–
Score_2=P(A)=n(A)/n(S) ,where P(A) is the possibility of the event A, n(A) is the number of ways in which the event can occur, n(S) = Total number of possible outcomes

Variable 3 – Probability of an event A occurring before event B. (Same as sequence Fidelity used in the last approach) [Note: 7th Nov 2016 - This variable has been removed since each event has to be compared with every other – this is process intensive and hence dropped.]
Score_3 = P(A before B) = n(A before B) / n(Anomalies)
P(A before B) is the probability of A occurring before B, n(A before B) is the number of cases in which the event A has occurred before B, n(Anomalies) is the total number of detected anomalies.

Variable 4 – Co-relation coefficient between A and B.
A strong correlation increases the probability of A causing an impact on B. The co-relation coefficients will be calculated for all the parameters in the system based on historical data. Some of the parameters that are physically not co-related for instance CPU on a web server and CPU on App server need not be co-related to decrease the dimensionality.

Variable 5 – Variable 2 can have a tendency to skew the results as it is based on a long-term temporal analysis. To reduce its effect, it is important to consider event probabilities in the 'near past' in addition to the 'over-all past'. Hence, the probabilities over a shorter duration (past 10-20 records) will be considered in the equation.

Final Correlation Score = Probability(Score_1 X Score_2 X Score_3)
This will be a positive whole number between 0 and 1.

We will use Bayes network to model this problem. Variable_3 may *probably* get adjusted in the Bayes network. Other 2 variables will have to be adjusted.

Other important metrics –
Uncertain pattern chain life – indicates the time taken for the chain to be formed. Reducing time-to-form indicates that the anomalies are manifesting faster than before. This can occur due to increased volume or effect of other anomalies on this chain.

Uncertain pattern life – this indicates the time for which the pattern was observed before it transformed into another pattern. This data along with the Chain life can be used to find which pattern in the chain is manifesting faster.

Impact on response time – Chains that lead to response time anomalies need to be monitored meticulously given their importance on the end user experience. Likewise, there may be other important parameters like response time defined by the domain

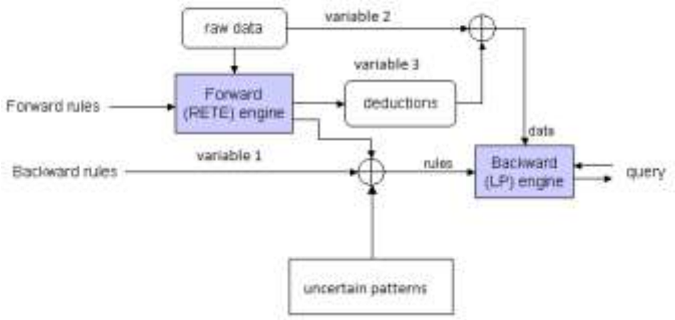

Fig. 3 Proposed System

III. Implementation and Performance evaluation

Simulations are conducted using python machine learning libraries, large data set from banking domain. Dell Vostro 3000 machine with 8 Gb RAM, i3 Processor is used for experimentation purposes. For plotting purpose matplotlib, mpld3, ggplot is used.

A. Data Readiness and Analysis

Table 1 Data Table Structure

Pattern	Previous observed pattern	Avg. time between previous and current (A)	Pattern resemblance (B)	Avg. Sequential fidelity (C)	Score $\frac{1XBXC}{A}$
AANNN	ANNNN	4 seconds	0.9	0.7	0.1575
AANNN	A'NNNN	8 seconds	0.2	0.3	0.075
AAANN	AANNN	3 seconds	0.9	0.8	0.2399
AAANN	AA'NNN	10 seconds	0.6	0.2	0.012

In the table 1 note than A and C have Average values. Hence, ALL the previous similar patterns must be considered to arrive at the average value. B i.e. Pattern resemblance will not change and hence average need not be considered.

ISSN: 2321-8169
65 – 68

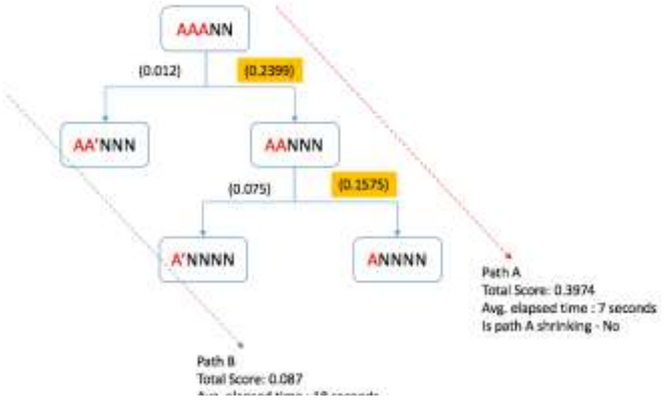

Fig. 4 Decision Tress analysis on the data

We can identify the entire chain having a problem or individual nodes that are impacting the system once the graph is ready through decision tree analysis. Result of the entire setup simulated under four method patterns is presented in next section,

B. *Results*

As can be seen using the proposed system all four methods are uniformly capturing data patterns with minimal loss of information. Fig. 5 shows that forward engine in the proposed model works on noise reduction as well as minimizing the uncertainty using belief probabilities form domain experts.

In Fig. 6, it is evident that method A time which calculated by using traditional system was uncertain and in presence of unseen data, belief propagation was quite dispersed. But method, methodCD and method time which is calculated by the proposed system is quite stable. Uncertain patterns were identified with greater accuracy and overall time is also uniform as well as stable.

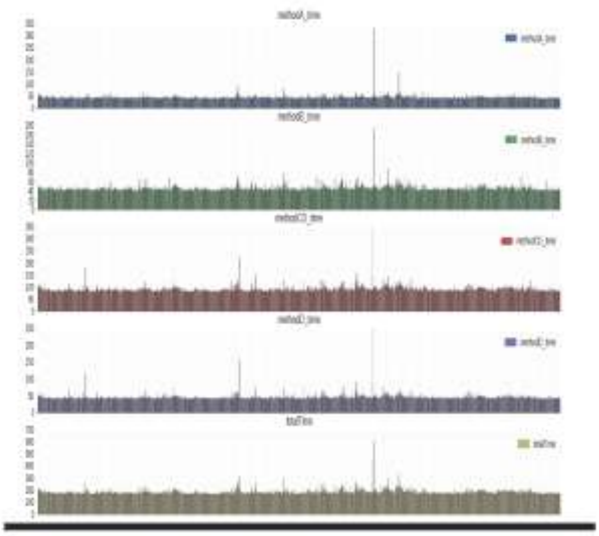

Fig. 5 Forward engine inference from raw data

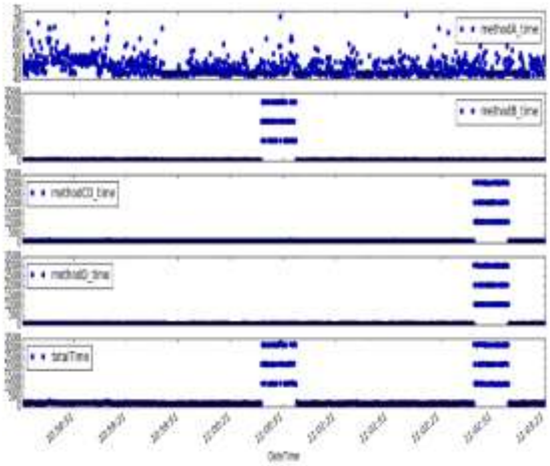

Fig. 6 Uncertain pattern identification in four methods using the proposed model

Below is the result of simulation on a raw sample of data generated from banking domain as shown in Fig. 7. A number of advisories captured are plotted against the traditional system and proposed system. It is evident that proposed model is performing far better than the traditional system.

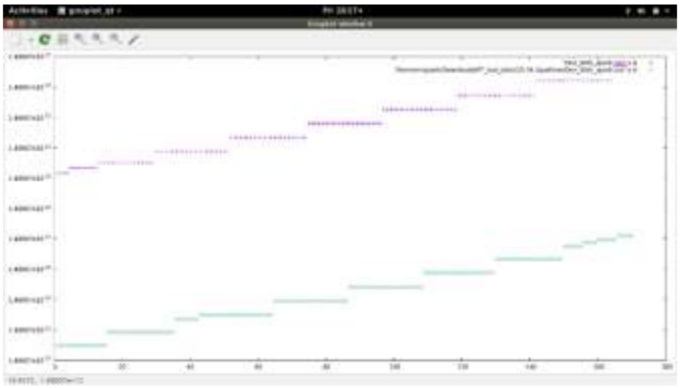

Fig. 7 Comparison of number of advisory generated Traditional systems(green) vs Proposed system(blue)

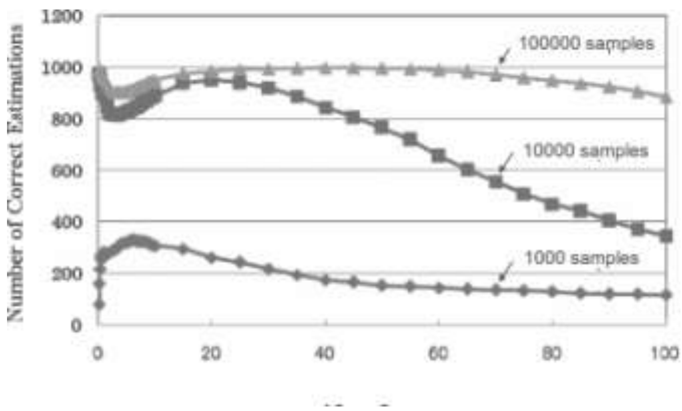

Fig. 8 Model prediction correctness over different sample range

ISSN: 2321-8169
65 – 68

IV. CONCLUSION

In this paper, we studied the limitation of traditional expert systems in terms of locality, detachment and truth-functionality. At the same time, proposed a new approach which is flexible in terms of usability, flexibility, and parameterization. Proposed model utilizes the full potential of decision tree analysis and probabilistic network models while dealing with unseen data. Additionally, results show that proposed model has a better algorithmic approach in terms handling belief probabilities, and propagation. We presented the probabilistic test of independence between two continuous variables. Our solution will benefit the determination of any number of continuous, ordinal discrete or/and categorical attributes.

REFERENCES,.

[1] Russel S., and Norgiv., Artificial Intelligence : A modern approach, 2nd Edition: 2003.

[2] Cooper, Gregory F. and Herskovits, Ed- ward, A Bayesian Method for the Induction of Probabilistic Networks from Data, in Machine Learning, 9, pp. 309-11, 1992.

[3] Barnett GO, Cimino JJ, Hupp JA, Hoffer EP. DXplain. An evolving diagnostic decision-support system. JAMA. 1987 Jul 3;258(1):67-74

[4] Jaakkola, T., Sontag, D., Globerson, A., & Meila, M. (2010). LearningBayesian Network Structure using LP Relaxations. InProceedings of the 13th International Conference on ArtificialIntelligence and Statistics, AISTATS'10, pp. 358–365