Data Marks for Uncertainty Management in Expert System Knowledge Bases

Antoni Niederliński *antoni.niederlinski@ue.katowice.pl*
Economic University in Katowice
Katowice, Poland

Abstract

A way to circumvent the awkwardness and constraints encountered while using Certainty Factors (CF) for modelling uncertainty in uncertain Knowledge Bases has been proposed. It is based on introducing Data Marks for askable conditions and Data Marks for conclusions of relational models, followed by choosing the best suited way to propagate those Data Marks into Data Marks of rule conclusions. This is done in a way orthogonal to the inference using Aristotelian Logic. Using Data Marks instead of Certainty Factors removes thus the intelectual discomfort caused by rejecting the Aristotelian law of excluded middle while using the CF methodology.
Keywords: Expert systems, Certainty Factors, Knowledge Bases, Uncertainty, Data Marks.

1. Introduction

Expert systems most often use standard Aristotelian Logic to infer conclusions out of a set of rules, models and facts, containing appropriate logical variables and arithmetical variables. Aristotelian logic operates with two logical constants (*truth* and *falsehood*), and relies upon the *law of excluded middle*, that states that every conclusion or condition is either *true* or *false*. Dichotomizing the reality into true - false or black - white categories, using the law of excluded middle, is the cornerstone of traditional expert systems, see e.g. [2], [7] and [8].
However, there are countless decision problems containing some form of uncertainty, either in the very rules to be used for inference, in their askable conditions (like "good reputation") or in their relational models (like "lower limit <= good collateral <= upper limit"). Taking this uncertainty into account may well contribute towards better, more realistic decisions be produced by expert systems.
The main tool used traditionally for managing uncertainty in expert systems knowledge bases are Certainty Factors. Certainty Factors are reals from the range [-1,1], assigned to uncertain rules, conditions and conclusion, which characterize the subjective confidence of some expert that this rule, condition or conclusion is true. CF=1 means absolute truth (of a rule, condition or conclusion respectively), CF=-1 means absolute falsehood (of a rule, condition or conclusion values of CF from the interval $-1 < CF < 1$ denote different degrees of confidence (of a rule, condition or conclusion respectively).
They were developed in the mid 1980s by David McAllester at MIT, who proposed a Certainty Factor Algebra for using them in expert systems. The first important application of certainty factors to expert systems was the MYCIN expert system, see [1] and [11]. Up til now they are a well-established tool for modelling uncertainty of knowledge bases, as examplified by a number of textbooks, see e.g. [7], [8] and [12]. Their usage was modified many times to accommodate real-world constraints. One of the recent modifications (see [4] and [5]) was to distinguish two types of rules with same conclusion: cumulative rules (with independent lists of conditions), and disjunctive rules (with dependent lists of conditions). The main advantages of the basic and of the modified Certainty Factor Algebra are:

- their ability to propagate numerical values characterising uncertainty of rules, of relational models and conditions into numerical values characterising the uncertainty of conclusions;
- the possibility to generate numerical values of certainty factors for conclusions of rules, conclusions of relational models and conditions using data mining for historical records of decision outcomes; A number of data mining tools provide results, which enable a calculation of certainty factors;
- the possibility to use them for expressing the preferences of decision makers.

The main drawback of all existing Certainty Factor modfication is their rejection of standard Aristotelian Logic. Knowledge bases build with Certainty Factors contain no logical variables and need no logical inferences. They ignore the law of excluded middle, the result being that any condition or conclusion may be partially true (with some Certainty Factor smaller than 1), and partially false
(with another Certainty Factor smaller than 1). Other drawbacks are given by propagating Certainty Factors for nested rules and the one-and-only-one fixed way Cetrtainty Factors for rule conditions or conclusions of relational models are transformed into Certainty Factor of rule conclusion, which may on occasion contradict common sense. The drawbacks are perhaps the reason modern developments on expert systems, created under the heading of *Business Rule Management Systems*, don't even mention Certainty Factors (see e.g. [3], [10] and [13]), although business application knowledge bases most often do need some form of uncertainty modelling.

2. Contribution

The aim of the paper is to present a simple and intuitively obvious approach towards modelling uncertainty of Knowledge Bases, which does not reject the Aristotelian Logic background of rules, relational models, conditions and conclusions, but supplements them with an *orthogonal* mechanism of marking the values of conditions and conclusions of relational models with Data Marks and may use any conceivable way of propagating these values into Data Marks of rule conclusions.

For the advocated approach, all conditions and conclusions are still logical variables, being either true or false and respecting the law of excluded middle,
all rules are generating true or undetermined conclusions (for the Open World Assumption) and true or false conclusions (for the Closed World Assumption), and all relational models are generating true or undetermined conclusions (for the Open World Assumption) and true or false conclusions (for the Closed World Assumption).
However, to some logically grounded askable conditions or conclusions of relational models, Data Marks may be attached additionally by the knowledge-base designer. The Data Mark is a real from the range [1,..,10], which reflects the degree with which the grounded condition or conclusion of relational model is considered to be true or preferable to the decision maker. The Data Marks attached to conditions and conclusions of relational models may be used to reflect the uncertainty of those conditions as well as the preferences of the decision maker. The may also by calculated (as Certainty Factors) by using data mining procedures.
This is presented by Fig.1 for the case of askable conditions. The orthogonality of Aristotelian Logic and Data Marks means that none of them is interfering with the other: chaining is proceeding with Aristotelian Logic; after it is done, Data Marks are attached to some askable conditions and a way to propagate them into Data Marks for conclusions of some rules is freely choosen.
This is were another advantage of Data Marks makes itself apparent: the designer is no longer forced to stick to one and only one established way of propagating uncertainty of askable conditions to uncertainty of conclusions, as is the case while using Certainty Factors. Now it is possible to freely choose a way to propagate Data Marks of conditions into Data Marks of rule conclusions that is best suited for the problem under consideration. E.g. the designer may

decide to make the Data Marks for some rule conclusion equal to the mean of the Data Marks of its conditions, or equal to the maximum or minimum value of the Data Marks of its conditions.

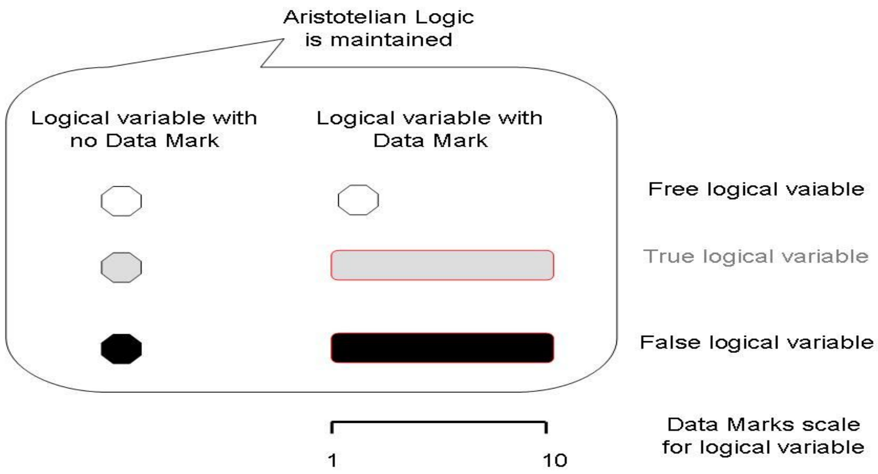

Fig. 1. The concept of Data Marks for askable logical conditions.

The same applies to Data Marks for conclusions of relational models, as exemplified by Fig. 2.

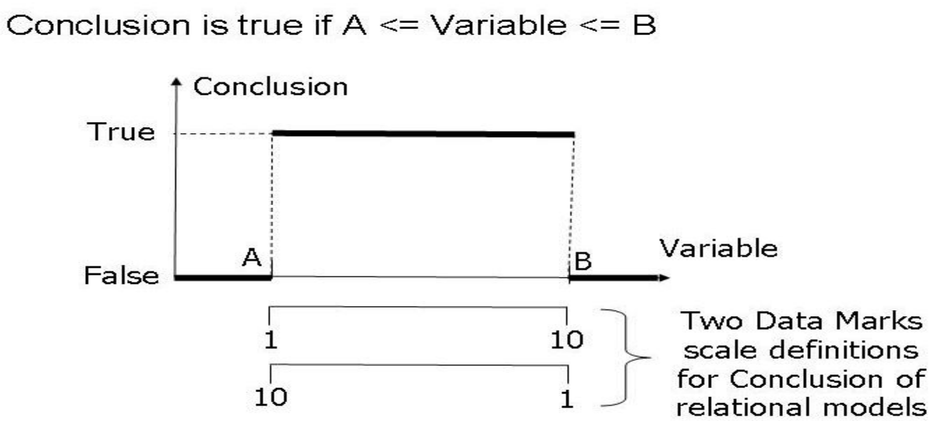

Fig. 2. The concept of Data Marks for conclusions of relational models.

For some instantiated variable, the relation is either true or false. If it is true, a Data Mark may be attached to the model conclusion using one of two possible Data Mark scales:

- the increasing “from left to right” scale may be used e.g. for a conclusions like “Good collateral” and the corresponding variable “Collateral size”;
- the increasing “from right to left” scale may be used e.g. for a conclusion like “Good WiFi Propagation” and the corresponding variable “Distance from router”.

3. Implementation

A detailed presentation of the Data Marks methodology and its applications may be found in [3]. It was implemented so far in the Polish version of the *rmse* expert system shells named *rmse_ED* and *rmse_RD*, see [14], and is prepared to be introduced into the *rmes* system [9].
To make Data Marks user friendly, some new models have to be introduced to the set of models described in [4]:

- The first set of models are models urging the user to declare Data Marks for non-askable conditions. Using the model structure presented it rmse, it may look e.g. like this:

```
model(108,"student gets grant", "Mark value of
   students scientifc activity", "Data_Mark of
      students scientific activity","=","0.0",1)
```

The model – while activited for ‘`student gets grant`’ being true – asks the user to declare a value for the ‘students scientific activity’ Data Mark from the range [1,…,10] and unifies this value with the variable ‘Mark value of students scientific activity’.

- The next set of models are models urging the user to declare Data Marks for arguments constrained by relational models, e.g.:
- `model_e(100,"No condition","Good collateral", "<=,<=",["A", "collateral", "B"],1).`

The indeterminism of this model amounts to ‘`Good collateral`’ being “more” good for collaterals being nearer to the upper limit ‘`B`’ and “less” good for collaterals being nearer to the lower limit ‘`A`’. To model this indeterminism the user may put into the model base the following model:

```
model(101," Good collateral","Mark value of Good
 Collateral","Data_Mark Good Collateral","=", "0.0",1)
```

where `Data_Mark` is a key word. The model – while activited for ‘`Good collateral`’ being true – asks the user to declare a value for the ‘`Good collateral`’ Data Mark from the range [1,…,10] and unifies this value with the variable ‘`Mark value of Good Collateral`’.

- The uncertainty of rule conclusion has to be modeled by choosing any way to propagate the Data Marks of conditions into Data Marks of conclusions. Now we need models producing Data Marks for conclusions of rules with conditions having Data Marks already declared or computed.

Consider first the case of unique rules, i.e. rules with a unique conclusion. This (depending upon the nature of the knowledge base) can be done in many ways, the most often needed are the following:
1) The conclusion Data Mark is equal to the arithmetic mean of condition Data Marks. That means that all conditions are influencing the conclusion Data Mark.
2) The conclusion Data Mark is equal to the smallest condition Data Mark. This may reflect the risk involved with the Conclusion: the choice is a cautious one.
3) The conclusion Data Mark is equal to the largest condition Data Mark. This may reflect a rather infrequent situation where the most certain condition is all that matters: the choice is a risky one.

Consider next the case of two rules with the same conclusion, e.g.

```
rule(1,"Conclusion",["Condition_11", "Condition_12"],1)
rule(2,"Conclusion",["Condition_21", "Condition_22"],1).
```

The rules are disjunctive: at most one of them may be fulfilled, and for these rule the Data Mark for its conclusion should be established. Because we do not know which rule will be fulfilled and the calculations for the conclusion Data Mark needs different model for different rules, we have to write the rules in a slightly different way:

```
rule(1,"Conclusion_1",["Condition_11","Condition_12"],1)
rule(2,"Conclusion_2",["Condition_21", "Condition_22"],1)
```

while adding the following rules:

```
rule(11, "Conclusion",["Conclusion_1",
   "Mark value of Conclusion_1 > 0"],1)
rule(12, "Conclusion",["Conclusion_2",
   "Mark value of Conclusion_2 > 0"],1).
```

At most only one of the rules 1 and 2 will be fulfilled and for this rule the model base has to establish a Data Mark value for its conclusion. So the corresponding rule from the set 11, 12 is fulfilled and '`Conclusion`' is true with the Data Mark established for either '`Conclusion_1`' or '`Conclusion_2`'.

4. Conclusion

Data Marks together with the rmes expert system shell have been used for three years for Real-World application and student instructions, demonstrating their versatility in designing uncertain Knowledge Bases, their simplicity of reasoning and the ensuing obviousness for users. They seem to be a worthwhile replacement for the Certainty Factor approach, capable of dealing with all kinds of uncertainty managed so far by Certainty Factors, with no their disadvantages.

References

1. Buchanan B. i E. Shortliffe, Rule-Based Expert Systems: The MYCIN Experiments, Addison-Wesley, Reading, MA, 1984.
2. Leondes, C. T. (Ed), Expert Systems. The Technology of Knowledge Management and Decision Making for the 21-st Century, vol. 1, Academic Press, San Diego, 2002.
3. Morgan, T., Business Rules and Information Systems, Addison Wesley, Boston, 2008.
4. Niederliński A., A Modification of the Stanford Certainty Factor Algebra for Uncertain Expert Systems, in AI-METH 2004, Methods of Artificial Intelligence, Gliwice, 2004.
5. Niederliński, A., rmes Rule-and Model-Based Expert Systems, pkjs, Gliwice, 2008.
6. Niederliński, A., Systemy ekspertowe dla automatyzacji zarządzania, pkjs, Gliwice, 2015.
7. Nikolopoulos C., Expert Systems, Marcel Dekker, New York, 1997.
8. Parsaye K. and M. Chignel, Expert Systems For Experts, John Wiley and Sons, New York, 1998.
9. Rule- and Model-Based Expert Systems, http://www.rmes.pl, 2013.
10. Ross, R. G., Principles of the Business Rule Approach, Addison Wesley, Boston, 2003.
11. Shortliffe E., Computer Based Medical Consultations: MYCIN, American Elsevier, New York, 1976.
12. Turban W., R. Sharda i D. Delen, Decision Support and Business Intelligence Systems, Pearson, Boston, 2011.
13. von Halle, B., Business Rules Applied, John Wiley and Sons, New York, 2002.
14. Witryna regułowo-modelowych systemów ekspertowych rmse, http://www.rmse.pl, 2016.