Krzysztof Michalik
Uniwersytet Ekonomiczny w Katowicach

VALIDATION AND VERIFICATION OF KNOWLEDGE BASES IN THE CONTEXT OF KNOWLEDGE MANAGEMENT. LOGOS REASONING SYSTEM CASE STUDY

Introduction

For some time the knowledge management has raised interest of many scientists, mainly as regards management studies, but also computer science. The latter is probably caused by the fact, that for many years the majority of information technology applications relate to business, and within its frame to the subject of decision support in the management. At the same time some of the information technologies, mainly those related to the artificial intelligence group, e.g. expert systems, have already matured enough to be able to analyze and process the knowledge. For this reason knowledge has become the subject of interest of both domains: the management and computer science. Thanks to the above, information technology can nowadays support not only the processes of data processing or of making the decisions by conventional methods, but it also provides methods and tools, that can support some knowledge management processes[1].

The aim of this article is to present the chosen aspects of validation and verification of knowledge (V&V) as a vital process of knowledge management, since it is generally omitted in literature concerning knowledge management. Due to the limited volume of this publication it focuses on the validation and verification of knowledge in the rule-based systems, these including expert systems, as being very well predisposed to process knowledge. Attention was also turned to the vulnerability of rule ontology and the rule-based systems to the possibility of automatic validation and verification of knowledge expressed in this way. The importance of validation and verification comes from the fact, that

---

[1] K. Michalik, *Wspomaganie wybranych procesów zarządzania wiedzą przez regułowe systemy ekspertowe*, „Organizacja i Zarządzanie", Zeszyty naukowe nr 53, Politechnika Łódzka, Łódź 2013.

after the process of gaining knowledge and its codification, other processes follow, in which the codified knowledge is no longer subject to major changes and is used in practice. Those processes are, among other things, the retention of knowledge, distribution, its sharing and its practical usage. And so passing over the knowledge, neither validated, nor verified, to the widely understood distribution and finally, its usage may result in serious consequences. In the final stage, decisions in organization may be taken on the basis of such potentially incorrect or even false knowledge. It is obvious, that the incorrect knowledge may lead to the wrong decisions. Thus, the importance of V&V process is difficult to overestimate and, according to the author, the systems processing knowledge should ensure detection of all, automatically detectable, errors of knowledge, before it is being used.

As regards the author's own contribution to this subject and the first attempts of solving this problem in an automatic way, the CAKE[2] (Computer Aided Knowledge Engineering) system can be mentioned, which is a part of the Sphinx artificial intelligence package. The CAKE system aided, among other things, verification of coded knowledge, in the form of knowledge bases for the PC-Shell hybrid expert system shell[3].

The V&V issues discussed have been illustrated by means of the new Logos system, currently being constructed by the author within the frames of the Logos Research Project. Some situations have, by the way, been shown, which might appear in practice of acquisition and coding of tacit knowledge, due to mistakes made either at the stage of acquisition of this knowledge (most frequently expressed in natural/human language), or at the stage of codification, in the case discussed with the use of rules ontology.

1. Some anomalies in knowledge bases

Discussing of validation and verification in this paper has been limited to anomalies, that may appear in the rule knowledge bases. According to Vermessan[4], the following anomalies may appear:

1. Deficiency,
   1.1 Missing rules,

---

[2] K. Michalik, *Business Decision Support Using Hybrid Expert System*, Business Informatics 21(2011), Publishing House of Wrocław University of Economics, Wrocław 2011.

[3] K. Michalik, *PC-Shell/SPHINX jako narzędzie tworzenia systemów ekspertowych* [w:] *Systemy ekspertowe wczoraj, dziś i jutro – Wiedza i komunikacja w innowacyjnych gospodarkach*, red. J. Gołuchowski, B. Filipczyk, Wydawnictwo Uniwersytetu Ekonomicznego, Katowice 2010.

[4] A.I. Vermesan, *Foundation and Application of Expert System Verification and Validation* [in:] *The Handbook of Applied Expert Systems*, ed. J. Liebowitz, CRC Press, New York 1998.

    1.2 Missing values,
2. Circularity (inference loop),
3. Ambivalence,
    3.1 Conflicting set of rules,
4. Redundancy,
    4.1 Redundant rules,
        4.1.1 Duplicate rules,
        4.1.2 Subsumed rules,
    4.2 Unusable rules.

In practice a number of different anomalies or other abnormalities that may appear in knowledge mapped in the rule-based system knowledge base is bigger than those mentioned above. In the CAKE system a considerable subset of anomalies listed has been implemented, defined as follows[5]:

Redundancy

If for any two rules:

$r_i \leftarrow p_{i1} \wedge .. \wedge p_{in}$ and

$r_j \leftarrow p_{j1} \wedge .. \wedge p_{jn}$, and $i \neq j$,

the following is true: $\{ p_{i1}..p_{in}\} = \{ p_{j1}..p_{jn}\}$ and $r_i = r_j$,

we call it as redundancy (duplicate rules).

Subsuming rules

If for any different rules:

$r_i \leftarrow p_{i1} \wedge .. \wedge p_{im}$ and

$r_j \leftarrow p_{i1} \wedge .. \wedge p_{in,}$ where $i \neq j$, and $r_i = r_j$,

the following is true $\{ p_{i1},..,p_{im} \} \subseteq \{ p_{i1},..,p_{in} \}$,

we say that rule r_i subsumes rule r_j.

Contradiction of rules

Two rules:

$r_i \leftarrow p_1 \wedge .. \wedge p_n$ and

$\neg r_j \leftarrow p_1 \wedge .. \wedge p_n$, where $i \neq j$ and $r_i = r_j$,

are contradictory (conflicting rules). In practice the notion of contradictory rules may be understood in a much broader way.

Incoherence of rules

Two rules:

$r_i \leftarrow p_1 \wedge .. \wedge p_n$ and

$r_j \leftarrow p_1 \wedge .. \wedge p_n$, where $i \neq j$ and $r_i \neq r_{j,}$

are descibed in CAKE system as incoherent.

---

5 K. Michalik, M. Kwiatkowska, K. Kielan, *Application of Knowledge Engineering Methods in Medical Knowledge Management* [in:] *Fuzziness and Medicine: Philosophical Reflections and Application Systems in Health Care*, eds. R. Seizing, M. Tabacchi, Springer, Berlin, Heidelberg 2013.

Missing rules

If there are no rules for the specified attributes, one usually refers to this as absence of rules. Such a situation may either mean a specified kind of error, or may be a consequence of creating an application based on the approach defined as rapid prototyping.

Unused attributes

The appearance of non-used attributes in the modeled knowledge may mean an error e.g. its literal kind (the so called 'typo') in the identifier of the attribute. It may lead to a very serious consequences, if the system will not detect this type of anomalies. The final effect may be not reaching the defined set of solutions, which may remain not recognized for a long time, after the knowledge base is implemented. The system discussed automatically protects the knowledge engineer from such a situation by means of attributes declaration in the *facets block*[6]. In certain situations information concerning this subject may be just a word of warning and does not have to be connected with the error in the knowledge model in the form of rules base. Such a situation, similarly to the previous case, may be more a result of rapid prototyping and the earlier specification of attributes, than defining the rules based on them. In every situation the knowledge engineer should be notified, by the system, about this anomaly in the knowledge base. Missing rules and unused attributes link relations proving that these notions aren't so obvious as they seem.

Unused values of attributes

This is a situation very similar, as regards interpretation, to the one discussed earlier in relation to attributes. The knowledge engineer must also be informed about it. Information on this subject may indicate an error, but may result from the stage of the knowledge base construction. In case when such a disturbance has a character of an error, its direct consequence may also be a loss of the ability to infere/derive some of the conclusions. More detailed discussion on some theoretical aspects connected with V&V in rules systems can be found, among other sources, in works by Ligęza[7], Niederliński[8] and Owoc[9].

---

[6] K. Michalik, *PC-Shell/SPHINX…*, op. cit.

[7] A. Ligęza, *Logical Foundations for Rule-Based Systems*, Springer, Berlin 2006.

[8] A. Niederliński, *Regułowo-modelowe systemy ekspertowe rmse*, Wydawnictwo Pracowni Komputerowej Jacka Skalmierskiego, Gliwice 2013.

[9] M.L. Owoc, *Wartościowanie wiedzy w inteligentnych systemach wspomagających zarządzanie*, Wydawnictwo Akademii Ekonomicznej, Wrocław 2004.

2. The example of V&V in Logos system

Logos reasoning system is constructed by the author within the frame of the research project under the same name. The author has already finished the research (R&D) regarding the Sphinx project, including, among other things, the PC-Shell – hybrid, expert system shell, as well as CAKE system (this including V&V functions). Taking advantage of some of the experience gained during the construction and the use of those systems, the author tries to implement, in a fuller and more advanced form, the V&V processes in the Logos system. The entire functionality within the frame of V&V is now included in one system, and not in two, as it was the case of PC-Shell and CAKE of the Sphinx package.

As it has already been mentioned, according to assumptions, in Logos system strong emphasis is placed on V&V functions (Fig. 1), as well as automatic detection of broad spectrum, theoretically possible to predict anomalies present in knowledge bases. For example, already in presently being finished 1.0 version, in comparison to CAKE system, there is, among other things, in addition to the previously mentioned anomalies implemented detection of recursion and inconsistency. While this paper is being written, both theoretical and development research is being carried out, concerning the conflicting rules, as well as checking of the completeness of the knowledge base. Other aspect of changes implemented in V&V algorithms is to make them possibly the most general in character. Due to the limited volume of present paper some more attention was devoted to recursion in rules. For example in Logos system (as opposed to PC-Shell and CAKE) in case of recursion rules no limit is set to the number of intermediary rules causing a loop, that is harmful to the process of reasoning. In other words, both the situation of direct and indirect recursion references are incorporated, with the discretionary number of levels. This is well illustrated by example (Fig. 2), where the additional complication is visible, which can appear in the process of detecting of this kind of anomalies, when the same rules may be engaged in forming of different recursion references through several different levels of rules. The algorithm applied is generalized to such an extent, that it is making it possible to detect this kind of harmful loop at the level equal to the number of all rules in the knowledge base. As distinguished from terminology used by Vermessan (circularity, inference loop), and other, the author uses the notion of recursion. This is because the notion of the loop has, in computer science, strong connotation, relating to the organization of the control flow, connected with structural programming within the frame of algorithmic paradigm. The detection of loops, this including their nesting – as distinguished from recursion – is relatively simple, due to, among other things, a clear separation of the beginning and ending of nested loops. While discussing the rule-based systems, this including expert systems, one has to clearly separate the prolog systems. While in prolog systems recursion is indispensable tool to design any kind of serious program, in case of e.g. expert systems, it always has a harmful character. The differences may be illustrated by means of very simple examples, one in Prolog for appending/concatenation of two lists, the second with the application of rules typical for the expert system in its simplest, general form (with one ‘a’ attribute).

Prolog:

```
append([],L,L).
append([X|L1],L2,[X|L3]) :- append(L1,L2,L3).
```

Expert system:

a **if** a; or
if a **then** a;

While in case of prolog recursion, it will cause the proper adding of the content of two lists and the application of recursion is here practically necessary, in the second case we deal with the harmful recursion, which not detected on time may lead to e.g. hang-up of the system, due to the unlimited inference process.

A direct recursion, meant here as the one, in which condition relates to the same rule, may be built according to one of the following schemes:

$p \leftarrow p$
$q \leftarrow q \wedge p_1 \wedge .. \wedge p_n$
$q \leftarrow p_1 \wedge .. \square\ p_n \wedge q$
$q \leftarrow p_1 \wedge .. \square\ q \wedge .. \wedge p_n$

The feature of this kind of recursion is a relative ease of detection by the knowledge engineer, on the condition that he reviews the whole set of rules of a given knowledge base. Much more difficult situation appears in case of indirect recursion, that is such, in which reference to a given rule is not happening from within this rule, but through a set of references to other rules. This can be illustrated in the following way:

$q_1 \leftarrow p_{11} \wedge .. \wedge p_{1j} \wedge .. \wedge p_{1x}$
$q_i \leftarrow p_{i1} \wedge .. \wedge p_{il} \wedge .. \wedge p_{iy}$
$q_n \leftarrow p_{n1} \wedge .. \wedge p_{nm} \wedge .. \wedge p_{nz}$
when : $p_{1j} = q_i$, $p_{il} = q_n$, $p_{nm} = q_1$.

It is worth stressing, that the equal sign in expressions $p_{1j} = q_i$, $p_{il} = q_n$ and $p_{nm} = q_1$ is a specific case**.** More generally speaking one should look at it as the ability to match two expressions in the process of inference. In practice detection of this type of recursion by the knowledge engineer may be very difficult, especially in case of a bigger knowledge base and the scattering of rules creating the indirect recursion in its different places. Then the application of tools supporting the knowledge engineer, in form of automatic detection of such anomalies, may be indispensable. It should then be stressed, that the examples presented have a very clear structure, as they are limited to single symbols, in the same way as it is customary to use in logic expressions. In the real knowledge base these are most frequently no single symbols, but triple symbol sets <O,A,V> or double symbol sets <A, V>, where O – object, A – attribute, V – value. Moreover, as a rule the single-character symbols used here for clarity are texts, so as to give the rules the feature of self-commenting. As a result detection of the discussed recursion relations by the knowledge engineer is additionally made much more difficult.

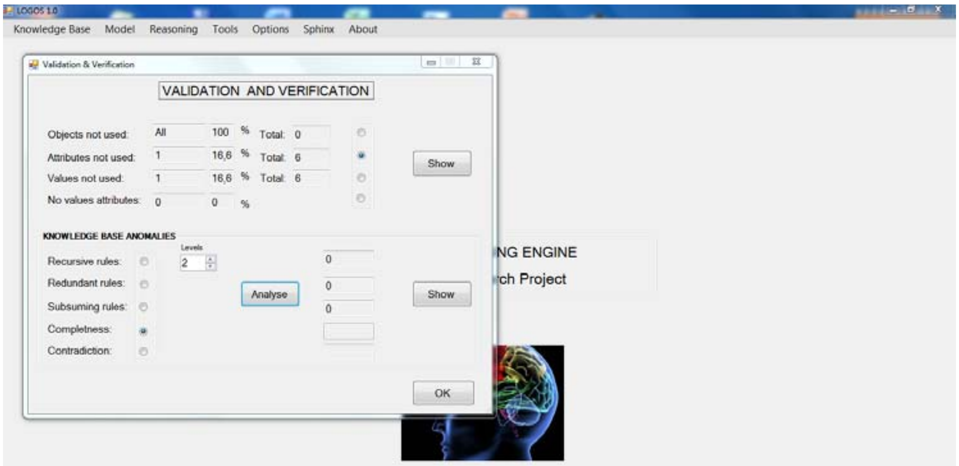

Fig. 1. General window for V&V of Logos 1.0 system

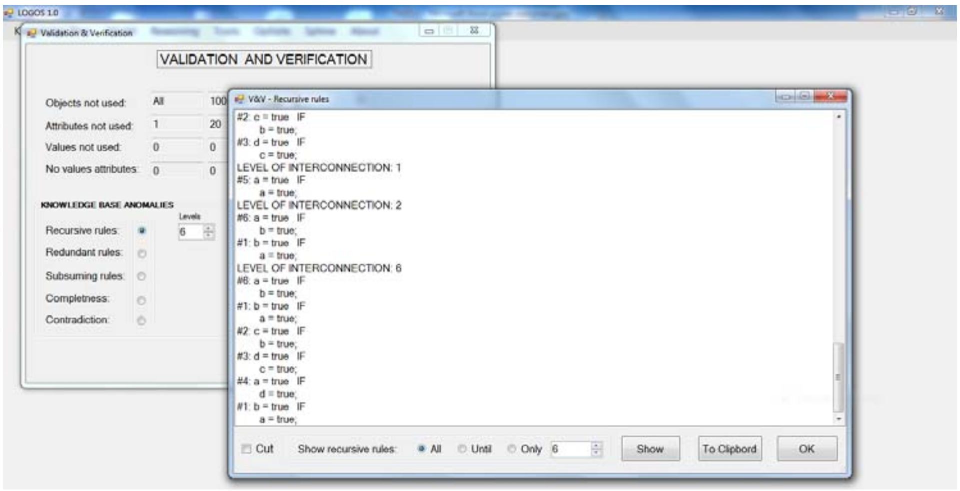

Fig. 2. Example of recursive rules detection to level 6 of interconnections in Logos 1.0 system

Conclusions and future works

The process of verification and validation of knowledge should be treated as an important process in the context of knowledge management. The knowledge acquired by the knowledge engineer from the specialist is the basis for solving specified problems in the organization. And so it should be free from anomalies previously considered, to give correct solutions. Anomalies, which may appear in the rule-based systems may be very difficult to detect by the knowledge engineer, both due to the degree of complexity in some cases, and their variety. This requires the application of special tools supporting this process. One example of this is CAKE system, being probably the first Polish system of this kind and at present also Logos system, being currently created by the author and based on different philosophy. The future research in this subject will aim develop V&V procedures to such an extent, that the system would automatically detect all possible to recognize anomalies, or at least give a word of warning to the knowledge engineer in doubtful cases. Implementation of such an advanced V&V functionality requires deeper theoretical research in this field, prior to construction of effective algorithms. A transparent presentation of detected anomalies is an additional very important objective, which should favor the usage of Logos system at universities, both for research and didactic purposes. As regards education it could aid the teaching processes in such subjects as artificial intelligence, expert systems, automated reasoning or intelligent decision support systems. This is why after the research on version 1.0 is completed the author plans to give free access to this system to all scientists and students, who are interested.

Among the currently planned applications in the scientific research there is, among other things, the project of the system supporting cardiac surgery[10], as well as an trial of construction of a temporal knowledge base and a proper module of temporal reasoning, in the latter case in cooperation with Prof. Maria Mach-Król[11].

Literature

Hrapkowicz T., Maruszewski M., Kempa A., Michalik K., Zacny B., *Decision Making Process in Cardiac Surgery – Concept of Building an Expert System*, „Management Systems in Production Engineering", Scientific and Technical Quarterly 2013, 1(9).

---

[10] T. Hrapkowicz, M. Maruszewski, A. Kempa, K. Michalik, B. Zacny, *Decision Making Process in Cardiac Surgery – Concept of Building an Expert System*, „Management Systems in Production Engineering", Scientific and Technical Querterly 2013, 1(9).

[11] M.A. Mach, *Temporalna analiza otoczenia przedsiebiorstwa – Techniki i narzędzia inteligentne*, Wydawnictwo Uniwersytetu Ekonomicznego, Wrocław 2007.

Ligęza A., *Logical Foundations for Rule-Based Systems*, Springer, Berlin 2006.

Mach M.A., *Temporalna analiza otoczenia przedsiębiorstwa. Techniki i narzędzie inteligentne*, Wydawnictwo Uniwersytetu Ekonomicznego, Wrocław 2007.

Michalik K., *Business Decision Support Using Hybrid Expert System*, Business Informatics 21(2011), Publishing House of Wrocław University of Economics, Wrocław 2011.

Michalik K., Kwiatkowska, Kielan K., *Application of Knowledge Engineering Methods in Medical Knowledge Management*, [in:] *Fuzziness and Medicine: Philosophical Reflections and Application Systems in Health Care*, eds. R. Seizing, M. Tabacchi, Springer, Berlin, Heidelberg 2013.

Michalik K., PC-Shell/SPHINX jako narzędzie tworzenia systemów ekspertowych [w:] *Systemy ekspertowe wczoraj, dziś i jutro – Wiedza i komunikacja w innowacyjnych gospodarkach*, red. J.Gołuchowski, B. Filipczyk, Wydawnictwo Uniwersytetu Ekonomicznego, Katowice 2010.

Michalik K., *Wspomaganie wybranych procesów zarządzania wiedzą przez regułowe systemy ekspertowe*, „Organizacja i zarządzanie", Zeszyty naukowe nr 53, Politechnika Łódzka, Łódź 2013.

Niederliński A., *Regułowo-modelowe systemy ekspertowe rmse*, Wydawnictwo Pracowni Komputerowej Jacka Skalmierskiego, Gliwice 2013.

Owoc M.L., *Wartościowanie wiedzy w inteligentnych systemach wspomagających zarządzanie*, Wydawnictwo Akademii Ekonomicznej, Wrocław 2004.

Vermesan A.I., *Foundation and Application of Expert System Verification and Validation* [in:] The Handbook of Applied Expert Systems, ed. J. Liebowitz, CRC Press, New York 1998.

VALIDATION AND VERIFICATION OF KNOWLEDGE BASES IN THE CONTEXT OF KNOWLEDGE MANAGEMENT. LOGOS REASONING SYSTEM CASE STUDY

Summary

The aim of this article is to turn attention to the fact, that validation and verification of knowledge, before it is applied for solving real life problems occurring in the organization, should be treated as one of important processes of knowledge management. The paper presents some anomalies that may appear in the knowledge bases of the rule-based systems. The readers have also been informed about research on the new project and the system called Logos, in which those problems play an important role, showing the examples of chosen aspects of anomalies in recursion rules.