

Expert System with Web Interface Based on Logic of Plausible Reasoning

Grzegorz Legień¹(✉), Bartłomiej Śnieżyński¹, Dorota Wilk-Kołodziejczyk^{1,2},
Stanisława Kluska-Nawarecka², Edward Nawarecki¹, and Krzysztof Jaśkowiec²

¹ AGH University of Science and Technology, Al. Mickiewicza 30,
30-059 Krakow, Poland

{glegien, bartlomiej.sniezynski}@agh.edu.pl

² Foundry Research Institute in Krakow, Zakopianska Street 73, Krakow, Poland

Abstract. The paper presents an expert system based on Logic of Plausible Reasoning (LPR). This formalism reflects human ways of knowledge representation and reasoning. The knowledge is modeled using several kinds of formulas representing statements, hierarchies, similarities, dependencies and implications. Several types of inference patterns are defined. Knowledge uncertainty can be modeled. The paper is structured as follows. Research related to LPR is presented. Next, the formalism is introduced and a Web-based application, which was developed for this research, is described. Finally, a case study is presented – a prototype expert system which recommends a material and a technology for a casting process.

Keywords: Web-based expert system · Logic of plausible reasoning · Knowledge representation and processing

1 Introduction

There are many knowledge modeling and reasoning techniques used in Expert Systems [12]. Some of them allow to represent certain knowledge only (like very popular rule-based systems based on classical logic [13, 17]). There are also many approaches that take into account uncertainty of knowledge (e.g. Bayesian networks [15], fuzzy logic [26], certainty factors [19], Dempster-Shafer theory [18], and rough sets [16]). These techniques are based mostly on logic and probability theory. The common feature is that all of them are strong simplifications of human reasoning methods. As a result, application of these techniques may be unnatural to users. For example, statement generalization or similarity between entities can not be represented (e.g. in Bayesian networks or fuzzy logic) or should be defined using rules (implications) instead of relations (in rule-based systems).

Logic of plausible reasoning (LPR) applied in this research has a different origin. Collins analyzed scripts describing human ways of answering common life questions [7] and extracted frequently repeating patterns, which were next formalized as LPR in cooperation with Michalski [8]. As a consequence, LPR

provides several inference patterns and many parameters to represent uncertainty. Application of this method allows to model and process the knowledge in a natural way.

In the following sections a related research is discussed and the LPR language is presented. Next, web-based software developed is described and a case study is discussed.

2 Related Research

The experimental results confirming that the methods of reasoning used by humans can be represented in the LPR are presented in subsequent papers [4,5]. The objective set by the creators has caused that LPR is significantly different from other known knowledge representation methods mentioned in the Introduction. Firstly, there are many inference rules in LPR, which are not present in the formalisms mentioned above. Secondly, many parameters are specified for representing the uncertainty of knowledge.

Studies described in [25] present RESCUER, a UNIX shell support system. By tracing changes in the file system and knowledge of the interpreter commands, the system is able to recognize the wrong commands and suggest appropriate substitutes. In this work only small part of LPR is used. In [24] LPR-based tutoring tool is presented. Knowledge for teaching is represented in LPR and the system is able to infer what the student should know after learning.

Another field of formalism application is presented in the work done by Cawsey [6]. It discloses a system generating a description of the concepts based on the recipient's model, taking into account his/her current knowledge.

Hierarchies' concept is also core element of ScubAA system [1], recommending best services in analyzed context, e.g. most accurate Internet search engines. It stores system knowledge in a tree, automatically updated during reasoning process and according to users' feedback. Moreover, in comparison to LIIS, solution uses only three transformations (generalization, specialization and similarity) and limits statements to hierarchy-related.

Research on LPR applications has been also performed at the AGH University. It concerned, in particular, diagnostics, knowledge representation and machine learning [11,22].

Important factor during design and implementation of expert system with web interface is creation of intuitive and user-friendly GUI. Verification of this assumption was important part of works on eXtraSpec [2]. When problem complexity is affecting user interface, like query specification in mentioned system, application should provide supporting tools. For example, when filling a form in eXtraSpec, system suggests correct values in the current edit box.

In CoMES system [3] authors attempted to join many popular techniques from Artificial Intelligence and Software Engineering. Machine learning is used for updating the knowledge base, which can be accessed by few algorithms in parallel. The system use agent architecture to integrate knowledge from human experts and other expert systems.

3 Outline of the Logic of Plausible Reasoning

LPR may be formalized as a labeled deductive system (LDS) [10]. The *language* consists of a finite set of constant symbols C , variables X (represented by capital letters), seven relational symbols and logical connectives: \rightarrow, \wedge . The relational symbols are: V, H, B, S, E, P, N . They are used to represent: statements (V), hierarchy (H, B), similarity (S), dependency (E), precedence (P) and negation (N).

Statements are represented as object-attribute-value triples: $V(o, a, v)$, where $o, v \in C \cup X, c \in C$. It is a representation of the fact that object o has an attribute a that equals v . If object o has several values of a , there should be several appropriate statements in a knowledge base. To represent vagueness of knowledge it is possible to extend this definition and allow to use composite value $[v_1, v_2, \dots, v_n]$, list of elements of $C \cup X$. It can be interpreted that object o has an attribute a equals v_1 or v_2, \dots , or v_n . If $n = 1$ instead of $V(o, a, [v_1])$ notation $V(o, a, v_1)$ is used. Relation $H(o_1, o, c)$, where $o_1, o \in C \cup X, c \in C$, means that o_1 is o in a context c . Context is used for specification of the range of inheritance. o_1 and o have the same value for all attributes which depend on attribute c of object o . To show that one object is below the other in any hierarchy, relation $B(o_1, o)$, where $o_1, o \in C$, should be used. Relation $S(o_1, o_2, c)$ represents a fact that o_1 is similar to o_2 ; $o_1, o_2, c \in C \cup X, c \in C$. Context, as above, specifies the range of similarity. Only these attributes of o_1 and o_2 have the same values which depend on c . Dependency relation $E(o_1, a_1, o_2, a_2)$, where $o_1, o_2 \in C \cup X, a_1, a_2 \in C$, means that values of attribute a_1 of object o_1 depend on attribute a_2 of the second object (o_2). Precedence relation $P(o_1, o_2)$, where $o_1, o_2 \in C \cup X$, says that object o_2 follows o_1 in some order. Negation relation is represented with $N(o_1, o_2)$, where object $o_1, o_2 \in C \cup X$. It represents situation where o_1 and o_2 can never have equal value. This relation do not exist independently, always as a permise of some implication.

In object-attribute-value triples, value should be placed below an attribute in a hierarchy: if $V(o, a, [v_1, v_2, \dots, v_n])$ is in a knowledge base, there should be also $H(v_i, a, c)$ for any $1 \leq i \leq n, c \in C$.

Using relational symbols, *formulas* of LPR can be defined. If $o, o_1, \dots, o_n, a, a_1, \dots, a_n, v, c \in C, v_1, \dots, v_n$ are lists of elements of C , then $V(o, a, v), H(o_1, o, c), B(o_1, o), S(o_1, o_2, o, a), E(o_1, a_1, o_2, a_2), P(o_1, o_2)$ and $\alpha_1 \wedge \dots \wedge \alpha_n \rightarrow V(o, a, v)$, where α_i is $V(o_i^\alpha, a_i^\alpha, v_i^\alpha), P(v_i^\alpha, w_i^\alpha)$ or $N(v_i^\alpha, w_i^\alpha)$, are LPR formulas (variables can be used in implications).

To manage uncertainty, the following *label algebra* is used: $\mathcal{A} = (A, \{f_{r_i}\})$. A is a set of labels which estimate uncertainty of formulas. *Labeled formula* is a pair $f : l$ where f is a formula and $l \in A$ is a label. A set of labeled formulas can be considered as a knowledge base.

LPR inference patterns are defined as *proof rules*. Every proof rule r_i has a sequence of premises α_i (of length n_i) and a conclusion α :

$$\frac{\alpha_1 : l_1, \alpha_2 : l_2, \dots, \alpha_n : l_n}{\alpha : l} \quad (1)$$

$\{f_{r_i}\}$ is a set of functions which are used in proof rules to generate a label of a conclusion: for every proof rule r_i an appropriate function $f_{r_i} : A^{n_i} \rightarrow A$ should be defined. For rule r_i the plausible label of its conclusion is calculated using $f_{r_i}(l_1, \dots, l_n)$. Examples of definitions of plausible algebras can be found in [20].

There are five main types of proof rules: *GEN*, *SPEC*, *SIM*, *TRAN* and *MP*. They correspond to the following inference patterns: generalization, specialization, similarity transformation, transitivity of relations and modus ponens. Some transformations can be applied to different types of formulas, therefore indexes are used to distinguish different versions of rules. Formal definitions of all these rules can be found in [8, 21]. *GEN_o* and *SPEC_o* change the scope of objects in statements. *GEN_v* and *SPEC_v* change the value in statements decreasing or increasing the description detail level. *SIM_o* allows to reason by analogy by changing the object, while *SIM_v* changes the value. *MP* is a classical Modus Ponens inference rule representing deductive reasoning pattern.

Proof can be defined as a tree. A tree P is a proof of labeled formula $\varphi : l$ from a set of labeled formulas KB if a root node of P is equal $\varphi : l$ and for every node $\psi : l_\psi$:

- if $\psi : l_\psi$ is a leaf, then $\psi : l_\psi \in KB$,
- else, there are nodes $(\psi_1 : l_{\psi_1}, \dots, \psi_k : l_{\psi_k})$, connected to $\psi : l_\psi$ and a proof rule r_i such, that $\psi : l_\psi$ is a consequence of r_i and $(\psi_1 : l_{\psi_1}, \dots, \psi_k : l_{\psi_k})$ are its premises (label of ψ is calculated using f_{r_i}).

We say, that a labeled formula ψ is a syntactic consequence of a set of labeled formulas KB ($KB \vdash \psi : l$) if there exist a proof of $\psi : l$ from KB .

To represent reasoning complexity, every proof rule has its cost assigned. Cost of a proof is equal to sum of costs of its proof rules.

4 LIIS System

LPR Intelligent Information System (abbr. LIIS) is a web-based application written in Java. It is created with Google Web Toolkit, supporting browser-based application development. The toolkit provides Java API and widgets, which can be compiled to JavaScript frontend. LIIS uses MySQL database for storing knowledge bases (formulas) and system data (e.g. user profiles, privileges). Object-relational mapping between Java objects and database records is provided by Hibernate framework. System builds with Maven, project dependency management tool. LIIS architecture is presented in Fig. 1.

A core element of LIIS is its reasoning engine, called LPR-Library. It implements the proof searching algorithm discussed above. In the current version of library E-formulas are omitted. Therefore context in hierarchies and similarities specifies a single attribute that may be inferred.

Reasoning process is performed with the LPA algorithm [21] based on the *AUTOLOGIC* system developed by Morgan [14]. Around hypothesis, taken as input, it constructs a reasoning tree, which nodes are lists of LPR formulas.

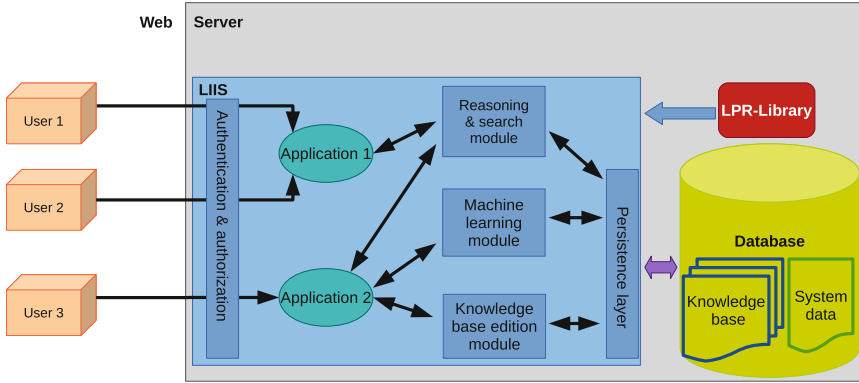


Fig. 1. LIIS architecture

Label algebra is defined as follows. Formulas are labeled by real numbers from range $(0, 1]$. Label of the consequence is equal to the product of premises' certainty parameters. The only exception is generalization of value, that copies statement's label from the first premise.

5 Experimental Results

In this section a case study is discussed. LPR is used for material and technology recommendation in a casting process. The problem is that for the same purpose many materials may be applied. The goal of the system is to choose the material for fitting the requirements provided by the user (application of the product, maximal acceptable production cost, mass of the product etc.). Below we present fragments of a prototype knowledge base, which was built in cooperation with casting technicians, and three scenarios showing how this knowledge is used for recommendation.

The knowledge base starts with statements describing possible applications and properties of the materials that can be used. Hierarchies present facts that ADI is a type of cast iron (in the context of attributes related to cost and production volume) and there are five subtypes of ADI (ADLGJS-1000-5, ADLGJS-1200-2, ADLGJS-1400-1, ADLGJS-800-8). Similarity formula represents the fact that ADI is similar to carburized steel in the context of application.

There are two groups of implications (rules). Implications with conclusion $V(\text{casting, material_alternative, X})$ allow to find the recommended material X. The rule with more tests in premise part (e.g. rule no. 30) is more certain than one with less tests (e.g. no. 32). Implications with conclusion $V(M, \text{cost}, C)$ define the production cost C of the material M, which depends on the weight of the product and the production volume.

1. $V(\text{adi, application, rake}):0.9$
2. $V(\text{steel_carburized, application, rack}):0.9$
3. $V(\text{adi_gjs-1000-5, minimum_elongation_A, 5}):1.0$

```

11. H(adi,cast iron,cost):1.0:0.0
24. H(adi_gjs-800-8,adi,minimum_elongation_A):1.0:0.0
25. H(adi_gjs-1000-5,adi,cost):1.0:1.0
29. S(adi,steel.carburized,application):0.8
30. [V(casting,application_required,A), V(X,application,A),
V(casting,cost_required,COST_MAX), V(X,cost,COST_CALCULATED),
P(COST_CALCULATED,COST_MAX),
V(casting, strength_tensile_Rm_required,STRENGTH_MIN),
V(X, strength_tensile_Rm,C), P(STRENGTH_MIN,C),
V(casting,minimum_elongation_A_required,ELONG_MIN),
V(X,minimum_elongation_A,E), P(ELONG_MIN,E)]
-> V(casting,material_alternative,X):1.0
32. [V(casting, application_required,A), V(X, application,A),
V(casting,cost_required,COST_MAX), V(X,cost,COST_CALCULATED),
P(COST_CALCULATED,COST_MAX)] -> V(casting ,material_alternative,X):0.5
39. [V(casting, weight, medium), V(casting, volume_production,large)]
-> V(adi,cost,14):1.0
40. [V(casting , weight,heavy), V(casting, volume_production,small)]
-> V(adi,cost,16):0.8
42. [V(casting, weight, heavy), V(casting, volume_production,large)]
-> V(adi,cost,12):1.0

```

Scenario 1. The first scenario is the simplest. Application of the material is a rake¹, the maximum cost required is equal to 15, product weight is heavy, and the batch size is large. System recommends ADI with the confidence 0.45. The recommendation is general (no specific subtype is chosen) because there is not enough information. To infer the conclusion, Modus Ponens (MP) rule is applied twice to implications no. 32 and 42.

Scenario 2. In this scenario more requirements are provided by the user: application, cost, weight and batch size as above, minimal tensile strength Rm is equal to 1100, and minimal elongation A is equal to 2. As a result, the system recommends ADI_GJS-1200-2 with confidence 0.9. The result is more specific and the confidence is larger. The proof was obtained by double application of the MP rule (to implications no. 30 and 40) and double object specialization (SPECo) rule. Certainty is higher because more information is provided by the user and implication 30 instead of 32 may be used.

Scenario 3. In this scenario other application is required by a user – a rack. The maximum cost is set to 15, casting weight is medium, batch size is large. The answer of the system is again ADI with confidence 0.45 This proof will be discussed in a more detailed way. It was obtained by double application of the MP rule and object similarity (SIMo) rule. In the first step, the MP rule was applied to implication no. 32, which means that if the required application of casting under consideration is equal to A (premise 1) and is the same as the application allowed for an alternative material in the rule marked by variable L (premise 2), the required maximum cost is equal to COST_MAX (premise 3), and the cost calculated for an alternative material is equal to

¹ A rake is a tool used in sewage-treatment plants. Its main task is to mix organic materials such as straw, grass, hay, etc. with semi-liquid material.

COST_CALCULATED (premise 4), and is lower than the maximum cost (premise 5), then the alternative material (L) should be used with confidence 0.5. Premises 1, 3 and 5 can be unified with the knowledge base elements or answers to questions provided by the user. Premise 2 (application acceptable for ADI) was inferred using object similarity rule (SIMo) for a similarity between ADI and carburized steel, the premises of which are included in the knowledge base. Premise 4 (ADI cost at a given weight and size of the batch) was inferred using MP rule and implication no. 39 representing the production cost. Thus the value of COST_CALCULATED equal to 14 was obtained.

6 Conclusions and Further Works

The presented LIIS system based on LPR allowed to create a web-based expert system for material and technology recommendation in a casting process. The system was tested in many scenarios, three examples of which are described above. Technologists confirmed that the answers are right and the proofs are valid and easy to follow.

The knowledge base created consists of various types of formulas: statements, hierarchies, similarities and implications. Therefore, in the knowledge processing various types of inference patterns are applied (deductive reasoning, generalization, and similarity). As a result, the knowledge and reasoning reflect human way of thinking, what makes the creation of the knowledge base more natural.

Further works will concern adding learning capabilities to the system. Learning module is already implemented (see Sect. 4). However, appropriate knowledge base and use cases are still under construction. Other works concern application in a system in other domains. Knowledge-based systems for telemetry-oriented applications [23] and money laundering detection [9] are under investigation. Also consistency check of the Knowledge Base should be added.

Acknowledgments. The research reported in the paper was supported by the grant of The National Centre for Research and Development (LIDER/028/593/L-4/12/NCBR/2013) and by the Polish Ministry of Science and Higher Education under AGH University of Science and Technology Grant 11.11.230.124.

References

1. Abedinzadeh, S., Sadaoui, S.: A trust-based service suggestion system using human plausible reasoning. *Appl. Intell.* **41**(1), 55–75 (2014)
2. Abramowicz, W., Bukowska, E., Dzikowski, J., Filipowska, A., Kaczmarek, M.: Semantically enabled experts finding system ontologies, reasoning approach and web interface design. In: 15th ADBIS Conference (2011)
3. Althoff, K., et al.: Collaborative multi-expert-systems realizing knowledge-lines with case factories and distributed learning systems. In: *Proceedings of the 3rd KESE Workshop* (2007)
4. Boehm-Davis, D., Dontas, K., Michalski, R.S.: Plausible reasoning: an outline of theory and the validation of its structural properties. In: *Intelligent Systems: State of the Art and Future Directions*, North Holland (1990)
5. Boehm-Davis, D., Dontas, K., Michalski, R.S.: A validation and exploration of the Collins-Michalski theory of plausible reasoning. Technical report, George Mason University (1990)

6. Cawsey, A.: Using plausible inference rules in description planning. In: Proceedings of the 5th EACL Conference, Congress Hall, Berlin (1991)
7. Collins, A.: Human plausible reasoning. Technical Report 3810, Bolt Beranek and Newman Inc. (1978)
8. Collins, A., Michalski, R.S.: The logic of plausible reasoning: a core theory. *Cogn. Sci.* **13**, 1–49 (1989)
9. Drezewski, R., Sepielak, J., Filipkowski, W.: The application of social network analysis algorithms in a system supporting money laundering detection. *Inf. Sci.* **295**, 18–32 (2015)
10. Gabbay, D.M.: *LDS - Labeled Deductive Systems*. Oxford University Press, Oxford (1991)
11. Kluska-Nawarecka, S., Nawarecki, E., Śnieżyński, B., Wilk-Kołodziejczyk, D.: The recommendation system knowledge representation and reasoning procedures under uncertainty for metal casting. *Metalurgija* **54**(1), 263–266 (2015)
12. Liao, S.H.: Expert system methodologies and applications a decade review from 1995 to 2004. *Expert Syst. Appl.* **28**(1), 93–103 (2005)
13. Ligeza, A.: *Logical Foundations for Rule-Based Systems*. Springer, Heidelberg (2006)
14. Morgan, C.G.: Autologic. *Logique et Analyse* **28**(110–111), 257–282 (1985)
15. Neapolitan, R.E.: *Probabilistic Reasoning In Expert Systems: Theory and Algorithms*. CreateSpace Independent Publishing Platform, USA (2012)
16. Pawlak, Z.: Rough sets. *Int. J. Comp. Inf. Sci.* **11**, 344–356 (1982)
17. Riley, G.: Clips - an expert system building tool. In: *Proceedings of Technology 2001 Conference*, San Jose, CA (1991)
18. Shafer, G.: *A Mathematical Theory of Evidence*. Princeton University Press, Princeton (1976)
19. Shortliffe, H., Buchanan, B.G.: A model of inexact reasoning in medicine. *Math. Biosci.* **23**, 351–379 (1975)
20. Śnieżyński, B.: Probabilistic label algebra for the logic of plausible reasoning. In: Kłopotek, M., et al. (eds.) *Intelligent Information Systems 2002. Advances in Soft Computing*, vol. 17, pp. 267–277. Springer, Heidelberg (2002)
21. Śnieżyński, B.: Proof searching algorithm for the logic of plausible reasoning. In: Kłopotek, M., et al. (eds.) *Intelligent Information Processing and Web Mining. Advances in Soft Computing*, vol. 22, pp. 393–398. Springer, Heidelberg (2003)
22. Śnieżyński, B., Kluska-Nawarecka, S., Nawarecki, E., Wilk-Kołodziejczyk, D.: Intelligent information system based on logic of plausible reasoning. In: Hippe, Z.S., Kulikowski, J.L., Mroczek, T., Wtorek, J. (eds.) *Issues and Challenges in Artificial Intelligence. Studies in Computational Intelligence*, vol. 559, pp. 57–74. Springer, Switzerland (2014)
23. Szydło, T., Nawrocki, P., Brzoza-Woch, R., Zielinski, K.: Power aware MOM for telemetry-oriented applications using gprs-enabled embedded devices - levee monitoring use case. In: *Proceedings of the FedCSIS 2014*, pp. 1059–1064, IEEE, Sept 2014
24. Virvou, M.: A cognitive theory in an authoring tool for intelligent tutoring systems. In: Kamel, A.E., Mellouli, K., Borne, P. (eds.) *Proceedings of the 2002 IEEE International Conference on Systems, Man and Cybernetics* (2002)
25. Virvou, M., Boulay, B.D.: Human plausible reasoning for intelligent help. *User Model. User-Adap. Inter.* **9**, 321–375 (1999)
26. Zadeh, L.A.: Fuzzy sets. *Inf. Control* **8**, 338–353 (1965)

Database and Expert Systems Applications
26th International Conference, DEXA 2015, Valencia,
Spain, September 1-4, 2015, Proceedings, Part II
Chen, Q.; Hameurlain, A.; Toumani, F.; Wagner, R.;
Decker, H. (Eds.)
2015, XXVI, 538 p. 174 illus., Softcover
ISBN: 978-3-319-22851-8