

Cognitive Reasoning Framework: Possibilities, Problems, Prospects

Oleg Anshakov and Tamás Gergely

Abstract The paper discusses a general scheme of constructing different systems of artificial intelligence and data mining. This scheme interprets various intelligent technologies as kinds of reasoning. All of these kinds of reasoning aim to cognition and formation of domain models. We assume that reasoning has a referential character, i.e. reasoning can use semantic arguments as well as syntactic rules of deduction. We use some non-classical logics to formalize cognitive reasoning.

Keywords Reasoning · Cognition · Data mining · Domain model · Non-classical logics · Cognitive reasoning framework

1 Introduction

In book [1], we introduce cognitive reasoning framework (CRF) as three-leveled structure, which includes conceptual, formal, and implementational levels.

- (i) At the conceptual level we present an informal model of a cognizing agent (CA). This model consists of different units that implement various functions of cognizing agent. The model also depicts how information circulates between the agent's units and how information transforms due to some unit's activity. At this level we also describe the behavior of a cognizing agent (reasoning, perceiving, memorizing and so on).
- (ii) At the formal level we define some classes of non-classical logics and a general scheme of creating formal theories. We consider two kinds of deduction technique: traditional and original (non-monotonic). The second kind leads to

O. Anshakov (✉)

Russian State University for the Humanities, Moscow, Russia
e-mail: oansh@yandex.ru

T. Gergely

Applied Logic Laboratory, Budapest, Hungary
e-mail: gergely@all.hu

© Springer International Publishing Switzerland 2016

A. Abraham et al. (eds.), *Proceedings of the First International Scientific Conference "Intelligent Information Technologies for Industry" (IITI'16)*, Advances in Intelligent Systems and Computing 450, DOI 10.1007/978-3-319-33609-1_1

constructing the so-called modification calculi and modification theories. Here ‘modification’ means that one can revise truth valuation of sentences within an inference of a specific type.

- (iii) At the implementation level we propose an object model of a cognitive reasoning system (CRS) that includes objects and collections for representing the cognizing agent’s units and its activity. At this level, one can express:
- Data representation for the results of perceiving
 - Knowledge representation and manipulation
 - Formalization of cognitive reasoning

We expect that the above object model is quite adequate representation of the so-called “cognitive engine” which can be consider as the “brain” of an artificial cognizing agent.

In this paper, we revise substantially the logical language for CRF. In [1], we use logical languages with special connectives, which represent characteristic functions for different sets of truth values. These connectives are called J operators. They was introduce by Rosser and Turquett [2].

In [1], we introduce original inference technique, which we call *modification calculi*. Namely modification calculi allow us to represent dynamic and non-monotonic inference where we can use semantic consideration that was interpreted syntactically. However, the logical languages from [1] permit us to define modification calculi only for first order logics. These languages have no expressive tools for defining this technique for propositional logics.

In this paper, we introduce special connectives for cause-effect relation, which allow us to define a logical language, that permit us to consider modification calculi for propositional logics. This lead us to a noticeable simplification of the example of modification inference with temporal contradiction (compare inference from Sect. 3 in this paper and inference from example 14.2.28 in [1]).

Notice that in [1], we have to use different logical languages for different logics with J operators. The logical language from this paper *can be the same* for different logics with different sets of truth values. This allows us to consider interesting problems for suitable class of logics. However, in this paper, we regard only the simplest logic from this class.

2 Cognizing Agent: General Scheme

Now we consider a general scheme of a cognizing agent. We expect that this scheme represents the structure and the behavior of the various kinds of these agents. We suppose that cognizing agent can be (for example) a living entity (particularly a human being), or a group of them or a technical system, which can adapt to changing environmental conditions.

Note that the structure we discuss below includes units, which are contained in different models of cognitive architecture, that are defined by different researchers. Our contribution consists in detailed description of a cognizing agent's behavior in various situations. The set of constituents of a cognizing agent is quite usual.

The most common architecture of cognitive systems include the following subsystems:

- A sensing subsystem, which can receive signals from the environment
- An information-processing subsystem, which creates a “world model” for the agent; this model can be used by the agent for making decisions
- An effecting subsystem, which performs actions, that are conclusions of the agent's decisions: we suppose that the actions can modify environment

We suppose that one can use different methods and techniques for constructing the world model of a cognizing agent. For instance, the world model can be formed by methods of machine learning or soft computation.

The basic constraints of the proposed model are as follows:

- A cognizing agent extracts knowledge by the use of *reasoning in a broad sense*. The agent can infer deductively, argue plausibly, and use computational procedures.
- A cognizing agent acts *discretely*; its activity includes alternating phases of perceiving and reasoning;
- A cognizing agent forms a *history* of its activity

We require the agent's activity to be discrete because this makes formation of algorithms for the agent's behavior more easy. However, we think that even natural cognizing agents cannot perceive and analyze information simultaneously.

Note that we mean a cognizing agent as a *researcher*, but not as a hunter or a soldier. Therefore, we are interested in its abilities such as capacity of extracting knowledge, accuracy of reasoning, etc.

We consider the history of activity of a cognizing agent as a theory of a special type. This theory will be called *open cognitive* or *quasi-axiomatic theory*.

A cognizing agent consist of the following subsystems:

- A sensing subsystem
- An information-processing subsystem
- An effecting subsystem
- The long-term memory for storing the history of the cognizing agent's activity

The *sensing subsystem* receives signals from the environment where we can distinguish the physical and the information part. The signals from the physical environment are used for forming data. The information environment is understood as a set of other cognizing agents or their components which have the possibility to transfer information. One can interpret signals from the information environment as additional information, advice or instructions (Fig. 1).

Long-term memory stores history of the discrete activity of the cognizing agent, which we consider here as a theory of a special kind. This history can be interpreted

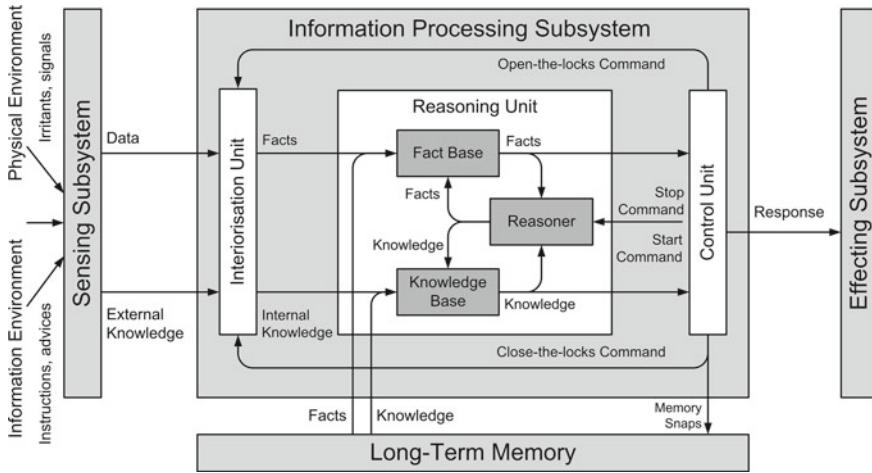


Fig. 1 General scheme of cognizing agent

as a sequence of *knowledge states*. The functioning of the long-term memory will be described in detail in the next chapter.

Consider the information-processing subsystem more detailed than the other subsystems. *The information-processing subsystem* contains the following units:

- The interiorization unit
- The reasoning unit
- The control unit

The agent uses *interiorization unit* for the accumulation of the information arriving from the environment and for its transformation into an internal representation (an internal format) of the cognizing agent. The data and knowledge are collected in the corresponding buffers (the *data buffer* and the *knowledge buffer*). This accumulation can occur in parallel with other actions of the cognizing agent, for example cognitive reasoning.

When it becomes necessary, the data and knowledge from the buffers will be transformed by corresponding converters (the *data converter* and the *knowledge converter*) into an internal format of the cognizing agent. The data in the internal format of the cognizing agent are called *facts*. The transformation of the data and external knowledge into facts and internal knowledge is called *interiorization* of data and knowledge.

The *reasoning unit* is the kernel of the considered subsystem. It contains the *fact base*, the *knowledge base* and the *reasoner* (the mechanism of reasoning). The fact base and the knowledge base together form the short-term (working) memory. The reasoner introduces changes in the working memory during its operation (both in the fact base and in the knowledge base). Changes in the knowledge base can be interpreted as acquisition of new knowledge, and changes in the fact base can be interpreted as generation of predictive hypotheses.

3 Modification Calculi and Modification Theories

Modification theories locate at the formal level of CRF. These are non-monotonic and based on specific non-classical logics. In [1], we use the so-called finite pure J (FPJ) logic to formalize modification theories. Here J means J operator, which was introduced by Rosser and Turquett in [2]. The more general class of logics with J operators than the class of FPJ logics was considered in [3]. The algebraic models of such logics was studied in [4]. The class of FPJ logics is interesting. However, now we can apply more convenient logics for formalization of modification theory. Particularly, we can represent modification rules by means of some propositional logics.

Consider an example of such logics and rules.

3.1 Syntax

Definition 1 The *alphabet* of the language of logic **M** contains the following groups of symbols:

- (i) A countable set of internal propositional variables
- (ii) A countable set of external propositional variables
- (iii) The classical propositional connectives $\neg, \wedge, \vee, \rightarrow, \leftrightarrow, \perp, \top$
- (iv) The special connectives $?, !, \overset{?}{\rightarrow}, \overset{!}{\rightarrow}, : \triangleright$.

As for the arities of the above connectives, we say:

- \perp and \top are nullary connectives;
- $\neg, ?,$ and $!$ are unary;
- $\wedge, \vee, \rightarrow, \leftrightarrow, \overset{?}{\rightarrow},$ and $\overset{!}{\rightarrow}$ are binary.
- $: \triangleright$, these two symbols are used for representing a ternary connective.

Definition 2 Let us define some kinds of *atomic formulas*:

- (i) The internal propositional variables are called *internal atomic formulas*.
- (ii) The expressions $?p, !p, p \overset{?}{\rightarrow} q, p \overset{!}{\rightarrow} q,$ and $p : q \triangleright s,$ where p and q are internal propositional variables, are called *J-atomic formulas*.
- (iii) Each external propositional variable and each J-atomic formula are *external atomic formulas*.

Remark 1 The informal interpretation of $?p, !p, p \overset{?}{\rightarrow} q,$ and $p \overset{!}{\rightarrow} q$ are the following:

- $?p$ is “ p is unknown”
- $!p$ is “ p is known”
- $p \overset{?}{\rightarrow} q$ is “it is unknown that p cause q ”

- $p \overset{!}{\rightarrow} q$ is “it is known that p cause q ”
- $p : q \triangleright s$ is “it is known that in instance p of phenomenon q there exist a circumstance s ”

Definition 3 The internal atomic formulas are called *internal formulas*. Let us inductively define external formulas as follows:

Basis: Each external atomic formula and each J-atomic formula are *external formulas*. \perp and \top are *external formulas*.

Induction step: Let φ and ψ be *external formulas*. Then $(\neg\varphi)$, $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \rightarrow \psi)$, and $(\varphi \leftrightarrow \psi)$ are *external formulas*.

Agreement 4 We have to agree on omission of parentheses for simplifying the representation of external formulas. We set the priority of the connectives \neg , \wedge , \vee , \rightarrow , and \leftrightarrow as follows:

- \neg , $!$, and $?$ have the same priorities, their priority is the highest;
- \wedge and \vee have the same priority and their priority is lower than priority of \neg ;
- \rightarrow , \leftrightarrow , $\overset{!}{\rightarrow}$, $\overset{?}{\rightarrow}$, and $: \triangleright$ have the same priority and their priority is lowest.

We agree to omit the external parentheses of an external formula and to omit parentheses if it is possible due to priority of its connectives.

3.2 Semantics

Definition 5 The *algebra of formulas of \mathbf{M}* is a two-sorted algebra

$$\mathcal{F} = \left\langle \mathcal{E}, \mathcal{I}; \perp, \top, \neg, \wedge, \vee, \rightarrow, \leftrightarrow, ?, !, \overset{?}{\rightarrow}, \overset{!}{\rightarrow}, : \triangleright \right\rangle$$

where \mathcal{E} is the set of external formulas, \mathcal{I} is the set of internal formulas, the operations of this algebra are define as usual. This algebra is an absolutely free algebra of its own type.

Definition 6 The *algebra of logic \mathbf{M}* is a two-sorted algebra

$$\mathcal{M} = \left\langle \{\mathbf{0}, \mathbf{1}\}, \{?, !\}; \perp, \top, \neg, \wedge, \vee, \rightarrow, \leftrightarrow, ?, !, \overset{?}{\rightarrow}, \overset{!}{\rightarrow}, : \triangleright \right\rangle$$

where $\mathbf{0}$ and $\mathbf{1}$ are classical **False** and **True**, respectively, the informal interpretation of $?$, $!$ are **Unknown** and **Known**, respectively, operations \perp , \top , \neg , \wedge , \vee , \rightarrow , and \leftrightarrow are the usual classic logical operation, operations $?$, $!$, $\overset{?}{\rightarrow}$, $\overset{!}{\rightarrow}$, and $: \triangleright$ are defined as follows:

$$?a = \begin{cases} \mathbf{1} & a = ? \\ \mathbf{0} & \text{otherwise} \end{cases} \quad !a = \begin{cases} \mathbf{1} & a = ! \\ \mathbf{0} & \text{otherwise} \end{cases}$$

$$a \overset{?}{\rightarrow} b = \begin{cases} \mathbf{1} & a = ! \text{ and } b = ? \\ \mathbf{0} & \text{otherwise} \end{cases} \quad a \overset{!}{\rightarrow} b = \begin{cases} \mathbf{1} & a = ? \text{ or } b = ! \\ \mathbf{0} & \text{otherwise} \end{cases}$$

$$a : b \triangleright c = \begin{cases} \mathbf{1} & a = ? \text{ or } c = ? \text{ or } b = ! \\ \mathbf{0} & \text{otherwise} \end{cases}$$

Definition 7 We can define a *tautology* by the usual way. We also can define *semantic entailment* as usual. We write $\models \varphi$ if φ is a tautology (of course φ is an external formula). We write $\Gamma \models \varphi$ if φ is *semantic conclusion* of Γ where Γ is a set of external formulas.

Remark 2 Let φ be an external formula. Then $\models \varphi$ if and only if $\emptyset \models \varphi$.

3.3 Calculus

Definition 8 (*Axioms*) Let φ, ψ , and θ be external formulas, c, p , and q be internal propositional variables; then the following external formulas are *axioms*:

- (1) $\varphi \rightarrow (\psi \rightarrow \varphi)$
- (2) $(\varphi \rightarrow (\psi \rightarrow \theta)) \rightarrow ((\varphi \rightarrow \psi) \rightarrow (\varphi \rightarrow \theta))$
- (3) $\varphi \wedge \psi \rightarrow \varphi$
- (4) $\varphi \wedge \psi \rightarrow \psi$
- (5) $\varphi \rightarrow (\psi \rightarrow \varphi \wedge \psi)$
- (6) $\varphi \rightarrow \varphi \vee \psi$
- (7) $\psi \rightarrow \varphi \vee \psi$
- (8) $(\varphi \rightarrow \theta) \rightarrow ((\psi \rightarrow \theta) \rightarrow (\varphi \vee \psi \rightarrow \theta))$
- (9) $(\varphi \rightarrow \psi) \rightarrow ((\varphi \rightarrow \neg\psi) \rightarrow \neg\varphi)$
- (10) $\neg\neg\varphi \rightarrow \varphi$
- (11) $(\phi \leftrightarrow \psi) \rightarrow (\varphi \rightarrow \psi)$
- (12) $(\phi \leftrightarrow \psi) \rightarrow (\psi \rightarrow \varphi)$
- (13) $(\psi \rightarrow \varphi) \rightarrow ((\psi \rightarrow \varphi) \rightarrow (\phi \leftrightarrow \psi))$
- (14) \top
- (15) $\neg\perp$
- (16) $?p \leftrightarrow \neg!p$
- (17) $\left(p \overset{?}{\rightarrow} q\right) \leftrightarrow \neg\left(p \overset{!}{\rightarrow} q\right)$
- (18) $(c : p \triangleright q) \leftrightarrow \left(!c \rightarrow \left(q \overset{!}{\rightarrow} p\right)\right)$

Definition 9 (*Modus ponens*) The rule

$$\frac{\varphi, \quad \varphi \rightarrow \psi}{\psi}$$

is called *modus ponens*.

Definition 10 We can define an *inference*, a *proof*, and the relations of *derivability* and *provability* as usual. We write $\Gamma \vdash \varphi$ if φ is derivable from Γ where Γ is set of external formulas (the set of hypotheses). We write $\vdash \varphi$ if φ is provable.

Remark 3 Let φ be an external formula. Then $\vdash \varphi$ if and only if $\emptyset \vdash \varphi$.

Theorem 11 (Soundness Theorem) *Let Γ be a set of external formulas, φ be an external formula. Then*

$$\Gamma \vdash \varphi \text{ implies } \Gamma \models \varphi$$

Theorem 12 (Completeness Theorem) *Let Γ be a set of external formulas, φ be an external formula. Then*

$$\Gamma \models \varphi \text{ implies } \Gamma \vdash \varphi$$

3.4 Modification Calculus

Definition 13 Let us define some non-reliable rules, which are the additional rules for **M**:

(i) The rule

$$\frac{p \overset{?}{\rightarrow} q, \quad c_1 : q \triangleright p, \quad c_2 : q \triangleright p}{p \overset{!}{\rightarrow} q}$$

where $c_1 \neq c_2$ (i.e. c_1 and c_2 are different internal variables) is the *induction rule*.

(ii) The rule

$$\frac{?q, \quad !p, \quad p \overset{!}{\rightarrow} q}{!q}$$

is the *prediction rule*.

(iii) The rule

$$\frac{?p, \quad !q, \quad p \overset{!}{\rightarrow} q}{!p}$$

is the *abduction rule*.

Remark 4 The informal explanation of modification rules is the following:

- (i) Suppose we do not know that p causes q . Assume that two different instances (c_1 and c_2) of one phenomenon q have the same circumstance p . Then we can suppose that we know p to cause q . This is informal comprehension of the *induction rule*. Note that this rule is a rather superficial formalization of John Stuart Mill's first canon (method of agreement): *If two or more instances of the*

phenomenon under investigation have only one circumstance in common, the circumstance in which alone all the instances agree, is the cause (or effect) of the given phenomenon. (See [5].)

- (ii) Suppose we do not know q and we know p . Assume that we know p causes q . Therefore we can suppose that we know q . This is informal comprehension of the *prediction* rule. This rule is similar to *modus ponens*. Such a rule is often called *deduction*. But the term “deduction” is usually applied to reliable reasoning. However, this rule is actually non-reliable because the assertion “we know p causes q ” is not necessary true but only known.
- (iii) Suppose we do not know p and we know q . Assume that we know p causes q . Therefore we can suppose that we know p . This is informal comprehension of the *abduction* rule. This rule is used to form a possible explanation for the occurrence of a phenomenon: q is, why? because p is, so p causes q .

Remark 5 Note that we cannot use the rules from Definition 13 in the same manner as we use reliable rules such as modus ponens. We obtain a contradiction immediately. Therefore we have to define a special dynamic inference for to represent updating of estimation of internal atomic formulas. In this inference, we can hide rows that lost their relevance. We also hide all rows that depend on hidden rows.

Notice that formal definition of dynamic inference in our sense is too long and it demands to define some auxiliary constructs. Therefore we have to introduce some constructs informally.

The table below represents an example of modification inference. The last column contains the number of the inference step (i.e. row number) where the current row has been hidden. The number 0 (in the last column) means that the current row is not hidden.

#	Formula	Reason	Hidden at
1	$!p$	Hypothesis	0
2	$?q$	Hypothesis	13
3	$p \overset{?}{\rightarrow} q$	Hypothesis	7
4	$d_1 : q \triangleright p$	Hypothesis	0
5	$d_2 : q \triangleright p$	Hypothesis	0
6	$\left(p \overset{!}{\rightarrow} q \right) \rightarrow (!p \rightarrow !q)$	Additional hypothesis	0
7	$p \overset{!}{\rightarrow} q$	Induction (3, 4, 5)	0
8	$!p \rightarrow !q$	Modus ponens (7, 6)	0
9	$!q$	Modus ponens (1, 8)	0
10	$!q \rightarrow (?q \rightarrow \perp)$	Tautology	0
11	$?q \rightarrow \perp$	Modus ponens (9, 10)	0
12	\perp (Contradiction)	Modus ponens (2, 11)	13
13	$!q$	Prediction (2, 1, 7)	0

This example displays the so-called temporal contradiction. The contradiction appears at the step (row) number 12 and disappears at the step (row) number 13.

4 Open Problems and Promising Areas of Research

Here we discuss some open problems that connected with cognitive reasoning framework. We group these problems by themes.

Theme 1. Modification calculi: further development

Modification calculi have many restriction, which reduces their ability for expressing the natural reasoning process. Relaxing these restrictions allows us to solve different important problems.

Some problems:

- Problem 1.1: to define various logical languages (propositional and first-order) for modification calculi
- Problem 1.2: to define logics without J operators that can be used in modification calculi
- Problem 1.3: to define generalized modification inference that can use more complicated algorithms of formalized reasoning

Theme 2. Kinds of similarity: formalization and investigation

Recognizing the similarity is the main part of induction. We can regard similarity as the common part (meet) of objects under consideration. This kind of similarity is used in the JSM method proposed by V.K. Finn (see [6, 7]), more informal description of the JSM method is in [8]. Another approach to similarity is to measure the degree of similarity of different objects. We can find technologies that based on this approach in many work on machine learning and data mining. It is interesting that calculating the degree of similarity can be combined with general scheme of JSM method (see [9]).

Some problems:

- Problem 2.1: to find regularities that are common for similarity as meet (intersection) and similarity as degree of likeness
- Problem 2.2: to develop algorithms for calculating or constructing similarity and minimize the complexity of these algorithms;

Theme 3. Modification calculi with generalised quantifiers

We increase the abilities of modification calculi to express informal reasoning if we use generalized quantifiers. For example we can define various statistical quantifiers that are similar to quantifiers from [10].

One problem:

- Problem 3.1: to define and to study modification calculi with generalized quantifiers

References

1. Anshakov, O., Gergely, T.: *Cognitive Reasoning: A Formal Approach*. Springer (2010)
2. Rosser, J.B., Turquette, A.R.: Many-valued logics. *Br. J. Philos. Sci.* **5**(17), 80–83 (1954)
3. Anshakov, O.M., Finn, V.K., Skvortsov, D.P.: On axiomatization of many-valued logics associated with formalization of plausible reasonings. *Stud. Logica.* **48**(4), 423–447 (1989)
4. Anshakov, O., Rychkov, S.: On finite-valued propositional logical calculi. *Notre Dame J. Formal Logic* **36**(4), 606–629 (1995)
5. Mill, J.S.: *A System of Logic*. University Press of the Pacific, Honolulu (2002)
6. Finn, V.K.: Plausible inferences and plausible reasoning. *Itogi Nauki i Tekhniki. Ser. Teor. Veroyatn. Mat. Stat. Teor. Kibern.* **28**, 3–84 (1988). [English version in: *Journal of Soviet Mathematics*, **56**:1 (1991), 2201–2248.]
7. Finn, V.K.: Timely notes about the JSM method for automatic hypothesis generation. *Autom. Documentation Math. Linguist.* **43**, 5, 257–269 (2009)
8. Anshakov, O.M.: The JSM method: a set-theoretical explanation. *Autom. Documentation Math. Linguist.* **46**(5), 202–220 (2012)
9. Lyfenko, N.D.: An approach to text data categorization based on the ideas of J.S. Mill. *Autom. Documentation Math. Linguist.* **49**(6), 202–212 (2015)
10. Hajek, P., Havranek, T.: *Mechanizing Hypothesis Formation*. Springer (1978)